COMS 561 - PRINCIPLES OF DATABASE SYSTEMS (FALL 2013)
PROJECT REPORT

GRADUATE ADMISSION EVALUATION SYSTEM

TEAM:

FENG YUNFEI (yunfei@iastate.edu)

KIM NI WANG (kimni@iastate.edu)

PRIYANGIKA PIYASINGHE (rumesh@iastate.edu)

ABSTRACT

This project is composed of two parts: student application part and committee members evaluation part. A new coming student can submit his/her application after filling out application form. The committee chair, department chair and staffs can upload students data from Excel file and assign two committee members for each graduate applicant. The committee chair and those committee members have their own assigned applicants to evaluate, and they can review the completed evaluations. The committee chair, department chair and staffs can keep track of admission progress and application round statistics, import and export applicant's data as well. Also the committee chair and department chair can make admit/deny decision for applications, then the staffs can send admission decision emails to each applicant.

TABLE OF CONTENTS

RESEARCH FOR THE PROJECT

In the graduate admission review system, the user interface for the committee member plays an important role.

A research was done to select the set of functionalities that needs to be included to the interface during the requirement gathering phases of the project. A part of the research included communicating with some of the previous committee members to get an idea about new functionalities needs to be included and the modifications needs to be done to the existing functionalities.

We also noticed the absence automated ways of assigning the applicants to committee members in a fair manner. We have designed several assigning algorithms to make sure that each committee member should receive approximately the same number of MS and PHD applications and each application must be reviewed by 2 committee members.

TARGET AUDIENCE OF THE PROJECT AND THEIR NEEDS

This project focuses on new coming students and four groups of faculties and staffs, including a committee chair, several committee members, a department chair and staffs.

New coming student:

Submitting an application form.

Committee member, committee chair:

Reviewing assigned MS/PHD applicants and give evaluation scores and comments.

Committee chair, department chair:

Reviewing assigned MS/PHD applicants and give evaluation scores and comments.

Keeping track of evaluation progress, which shows the scores committee member gave.

Making admission decision for each applicant.

Committee chair, department chair and staffs:

Reviewing the email list of all applicants.

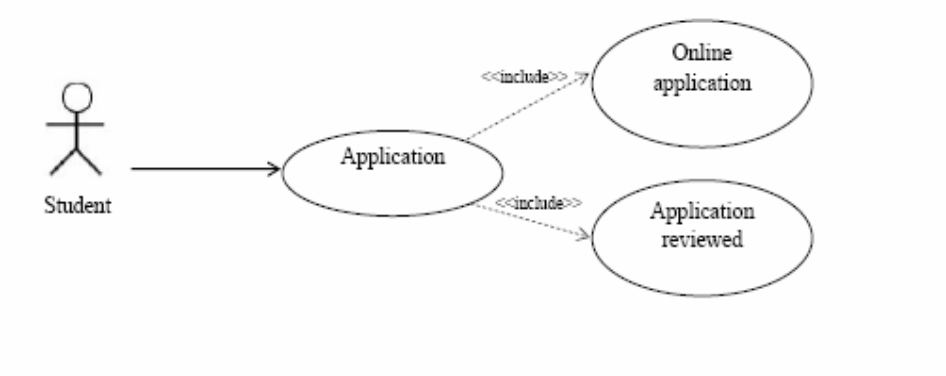
Reviewing round statistics

Having the access to Import/Export students' data to/from the database.

SYSTEM IMPLEMENTATION

Use Case Diagram

Mainly there are five types of roles involves with this part of the system: students, committee members, committee chair, department chair and staffs. The following Use Case diagram shows the interactions between the role and the system.



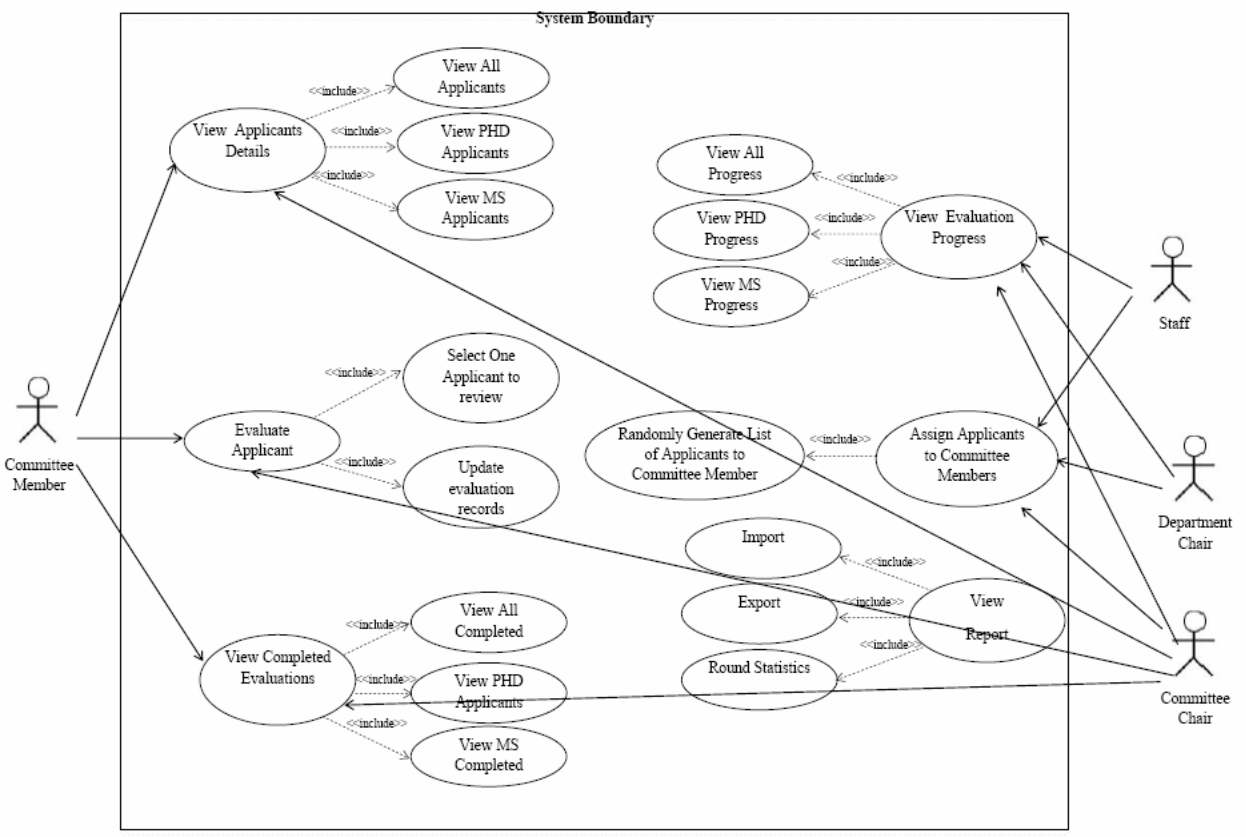


Figure 1: Use Case Diagram

CLASS DIAGRAM

The structure of this part with regarding classes, their attributes, methods, and the relationships among the classes are shown in the following Class diagram.

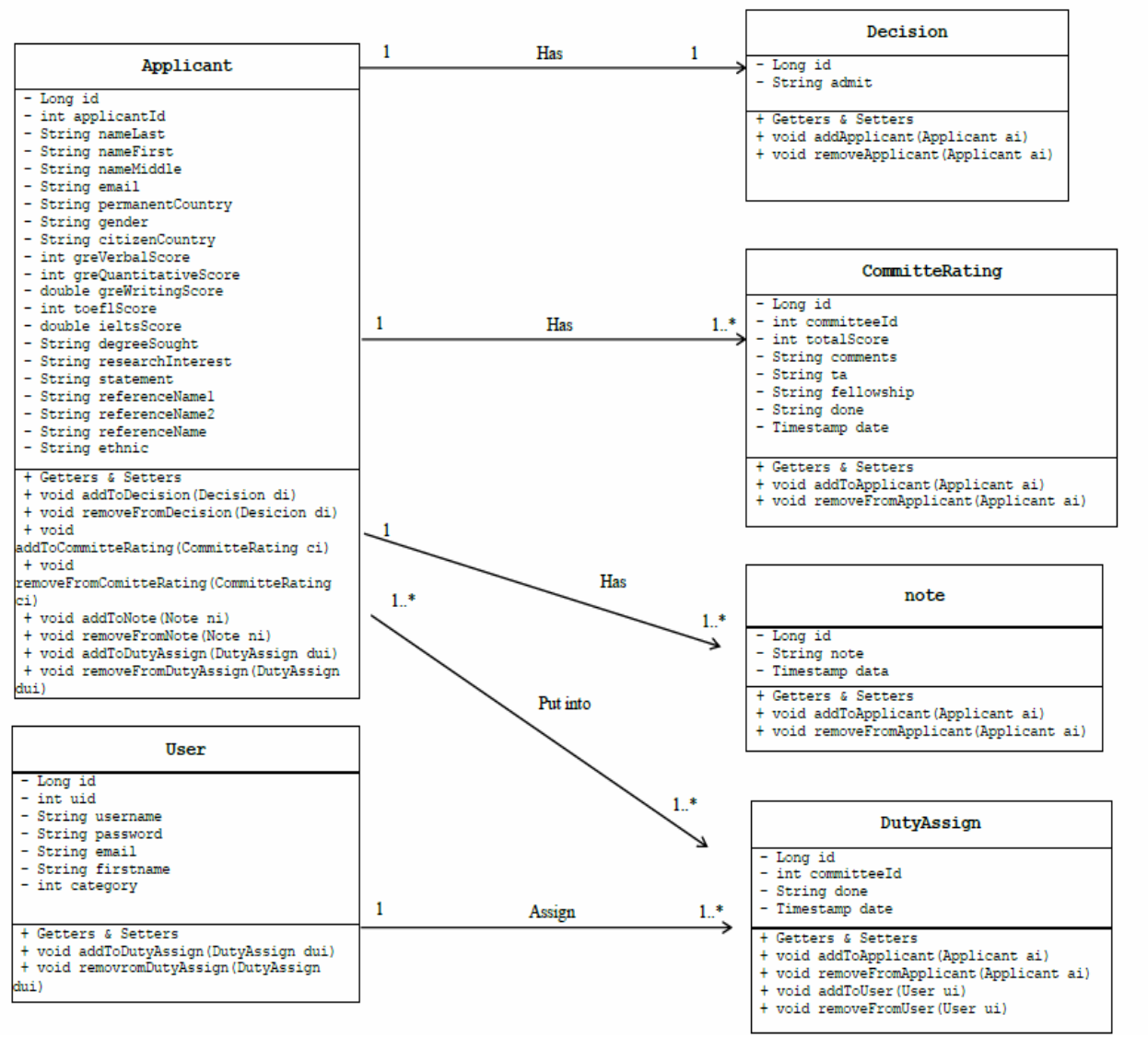


Figure 2: Class Diagram

DATA FLOW DIAGRAM

The following diagram shows who data flow through the system from front end to the backend.

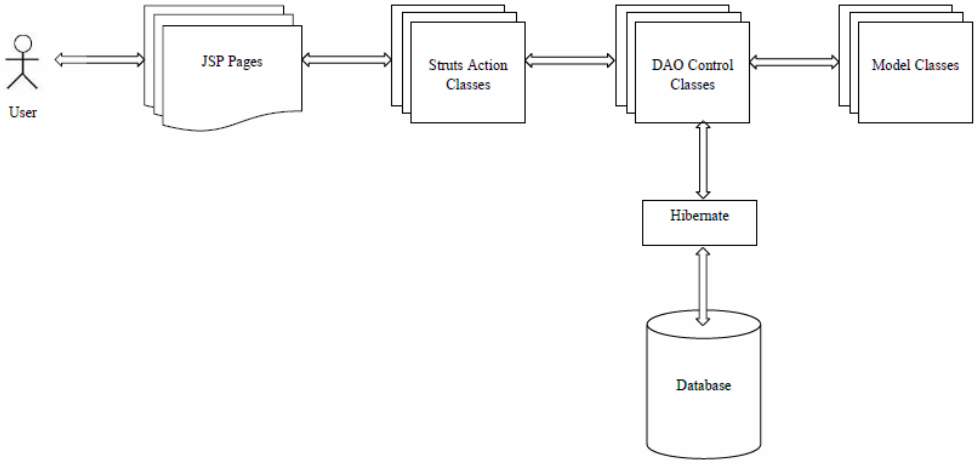


Figure 3: Data Flow Diagram

DATABASE LAYOUT

The next page shows the ER diagram used to design the database for the graduate admission review system.

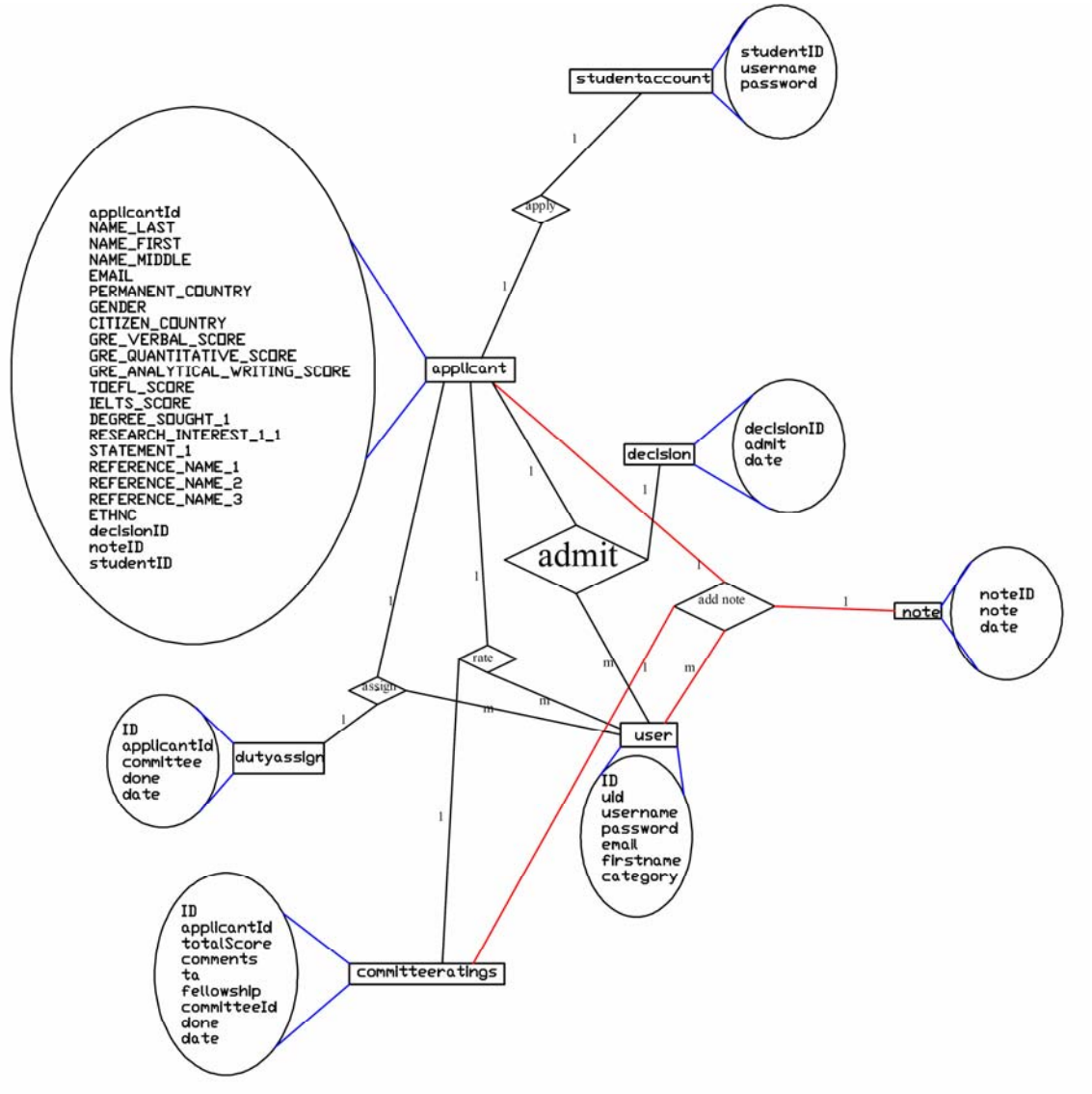


Figure 4: ER Diagram

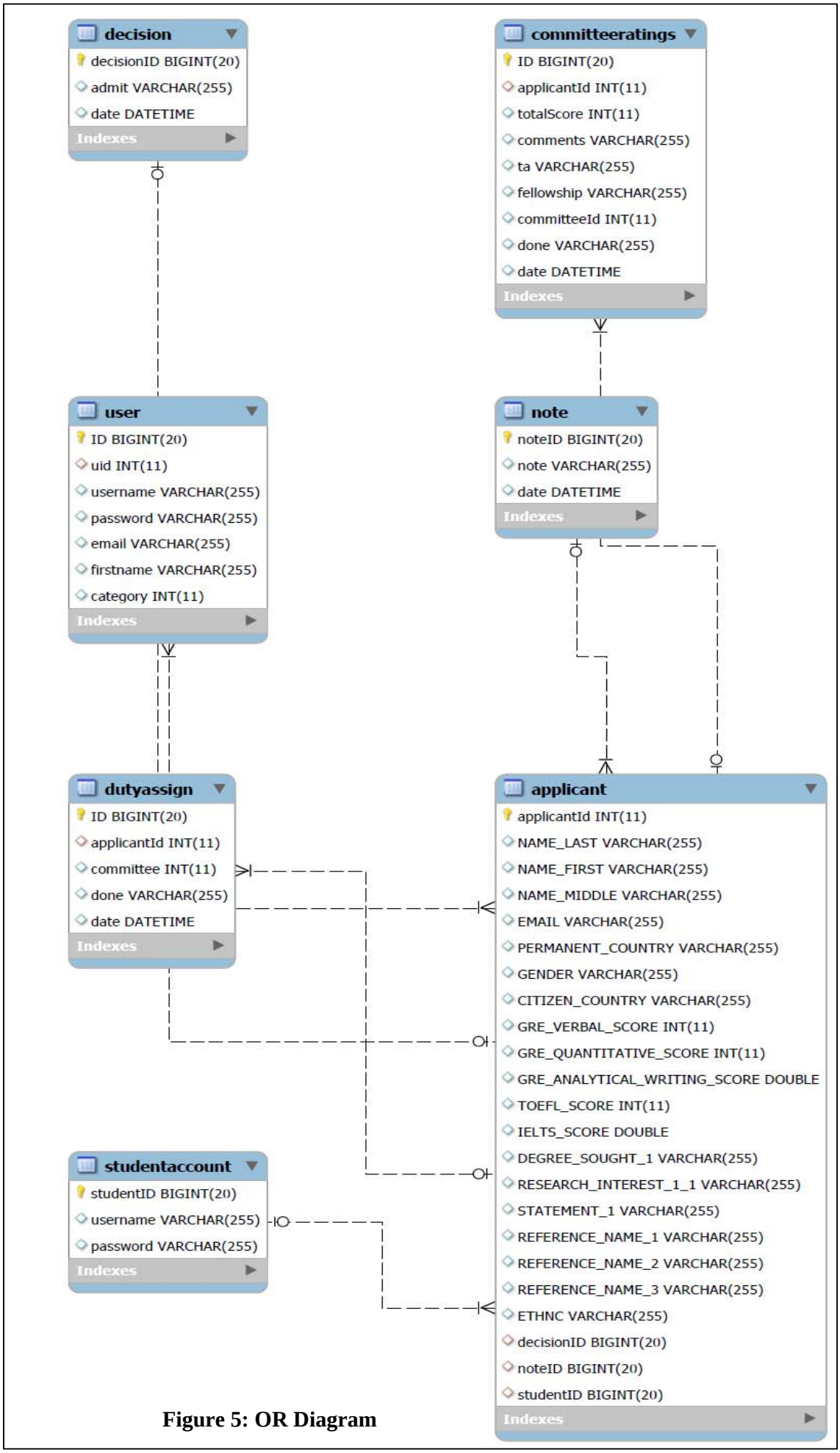


Figure 5: OR Diagram

Database Schema

Hibernate/MySQL was used was used to develop database schema.

The description of the tables in the system database “Admission” is as follows.

Table1: Applicant

Field	Type	Null	Key	Default	Extra
applicantId	int(11)	NO	PRI	NULL	auto_increment
NAME_LAST	varchar(255)	YES		NULL	
NAME_FIRST	varchar(255)	YES		NULL	
NAME_MIDDLE	varchar(255)	YES		NULL	
EMAIL	varchar(255)	YES		NULL	
PERMANENT_COUNTRY	varchar(255)	YES		NULL	
GENDER	varchar(255)	YES		NULL	
CITIZEN_COUNTRY	varchar(255)	YES		NULL	
GRE_VERBAL_SCORE	int(11)	YES		NULL	
GRE_QUANTITATIVE_SCORE	int(11)	YES		NULL	
GRE_ANALYTICAL_WRITING_SCORE	double	YES		NULL	
TOEFL_SCORE	int(11)	YES		NULL	
IELTS_SCORE	double	YES		NULL	
DEGREE_SOUGHT_1	varchar(255)	YES		NULL	
RESEARCH_INTEREST_1_1	varchar(255)	YES		NULL	
STATEMENT_1	varchar(255)	YES		NULL	
REFERENCE_NAME_1	varchar(255)	YES		NULL	
REFERENCE_NAME_2	varchar(255)	YES		NULL	
REFERENCE_NAME_3	varchar(255)	YES		NULL	
ETHNC	varchar(255)	YES		NULL	
decisionID	bigint(20)	YES	UNI	NULL	
noteID	bigint(20)	YES	UNI	NULL	
studentID	bigint(20)	YES	UNI	NULL	

Table2: CommitteeRatings

Field	Type	Null	Key	Default	Extra
ID	bigint(20)	NO	PRI	NULL	auto_increment
applicantId	int(11)	YES	MUL	NULL	
totalScore	int(11)	YES		NULL	
comments	varchar(255)	YES		NULL	
ta	varchar(255)	YES		NULL	
fellowship	varchar(255)	YES		NULL	
committeeId	int(11)	YES		NULL	
done	varchar(255)	YES		NULL	
date	datetime	YES		NULL	

Table3: Decision

Field	Type	Null	Key	Default	Extra
decisionID	bigint(20)	NO	PRI	NULL	auto_increment
admit	varchar(255)	YES		NULL	
date	datetime	YES		NULL	

Table4: DutyAssign

Field	Type	Null	Key	Default	Extra
ID	bigint(20)	NO	PRI	NULL	auto_increment
applicantId	int(11)	YES	MUL	NULL	
committee	int(11)	YES		NULL	
done	varchar(255)	YES		NULL	
date	datetime	YES		NULL	

Table5: Note

Field	Type	Null	Key	Default	Extra
noteID	bigint(20)	NO	PRI	NULL	auto_increment
note	varchar(255)	YES		NULL	
date	datetime	YES		NULL	

Table6: User

Field	Type	Null	Key	Default	Extra
ID	bigint(20)	NO	PRI	NULL	auto_increment
uid	int(11)	YES	MUL	NULL	
username	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	
email	varchar(255)	YES		NULL	
firstname	varchar(255)	YES		NULL	
category	int(11)	YES		NULL	

Table7: StudentAccount

Field	Type	Null	Key	Default	Extra
studentID	bigint(20)	NO	PRI	NULL	auto_increment
username	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	

REFERENCES

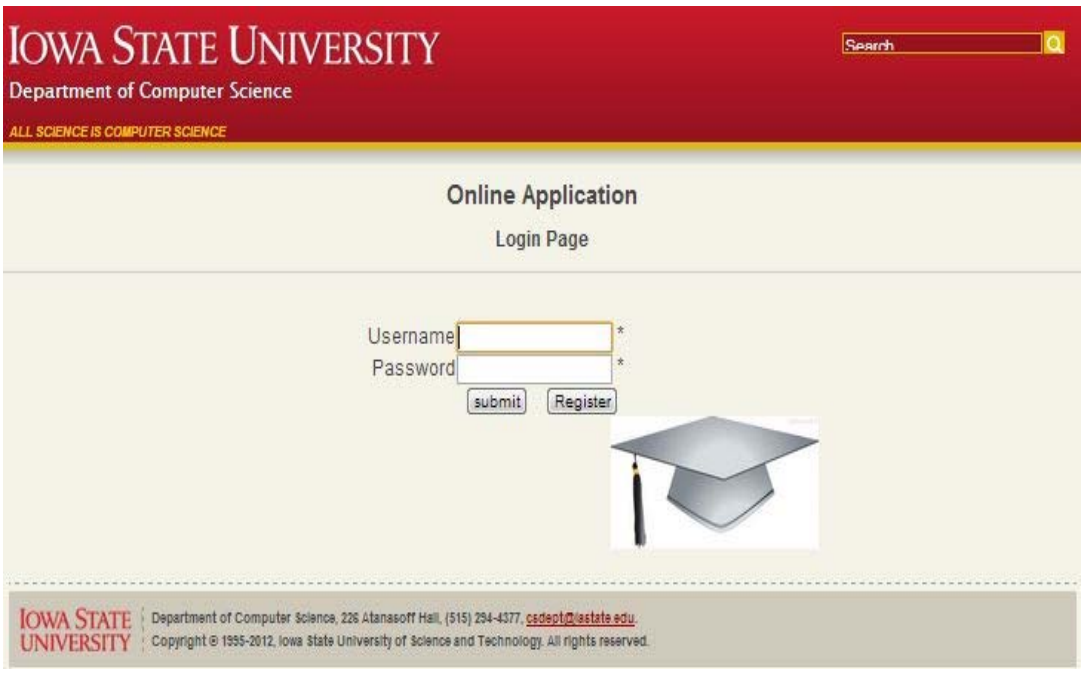
1. Hibernate ORM documentation http://hibernate.org/orm/documentation/
2. Hibernate: A J2EE Developer's Guide, Will Iverson
3. Java persistence with Hibernate, Christian Bauer and Gavin King.
4. Beginning Hibernate: From Novice to Professional, Dave Minter Jeff Linwood
5. Agile Java Development with Spring, Hibernate and Eclipse, Anil Hemrajani
6. Struts Design and Programming: A Tutorial, Budi Kurniawan, Brainy Software Corp. 2005
7. Tutorial: Create Struts 2 Application in Eclipse By Viral Patel (http://viralpatel.net/blogs/tutorial-create-struts-2-application-eclipse-example/)

APPENDIX: USER'S GUIDE

I. Student Application Module

1) Every applicant should log in using his/her username and password.

2) New applicant should register before applying for admission.



Online Application Form

1)Applicants need to fill up all the mandatory fields.

2)Should give a valid email id.

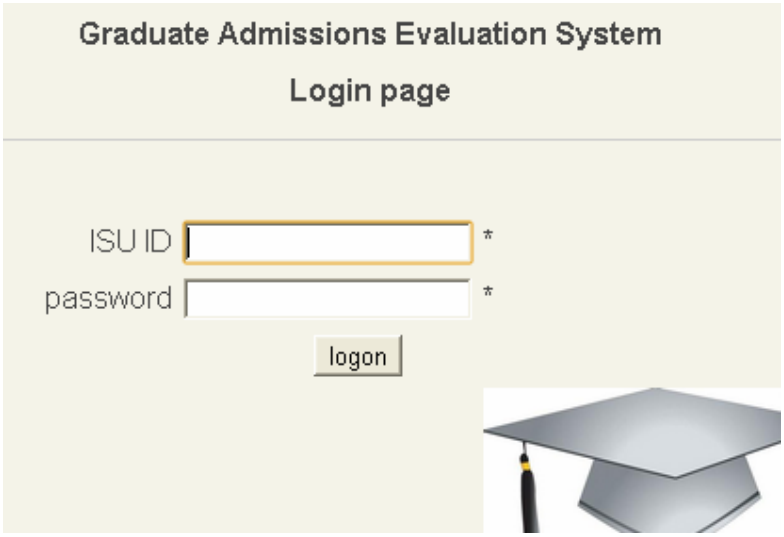
3)Should give a valid scores for GRE and TOEFL/IELTS.

Application Submitted

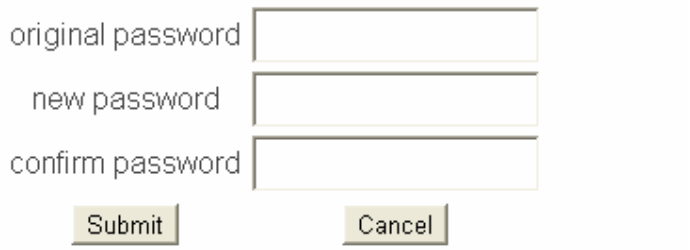
Display the information of the submitted application to the applicant.

II. Evaluation system Component

1. Every user should log in this system using ISU ID and assigned password.

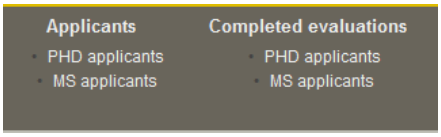


2. New users could register. However, this job done by an administrator is recommended.
3. Users could change password.

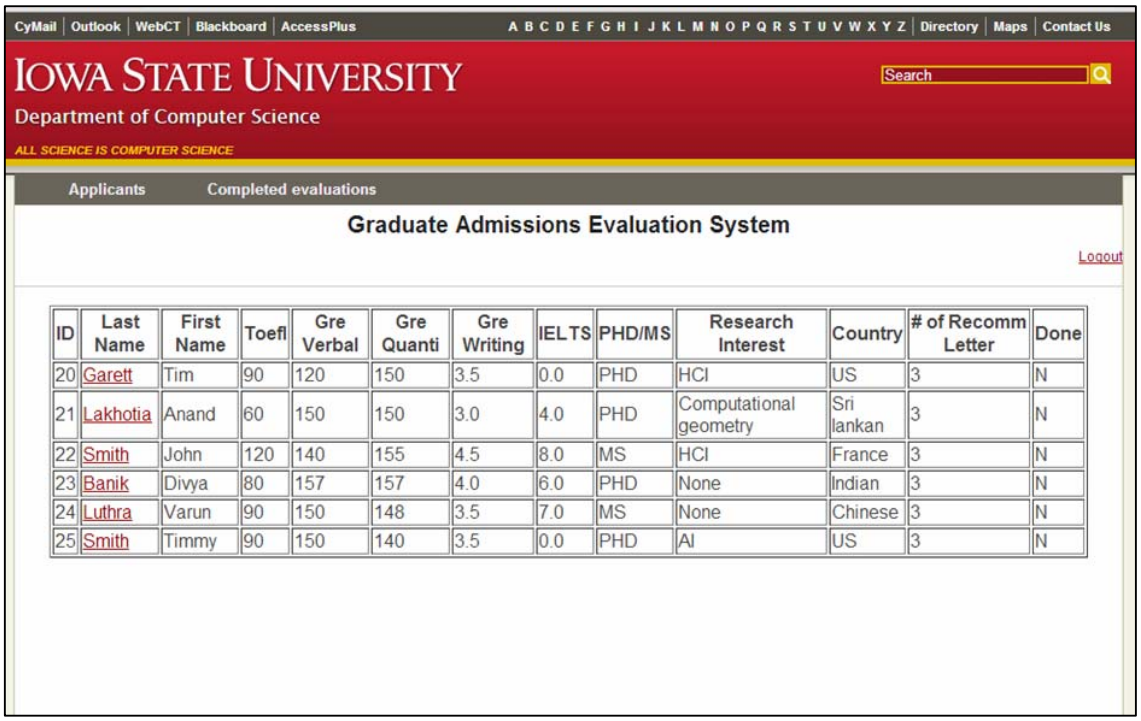


Committee Member

Committee member's menubar:

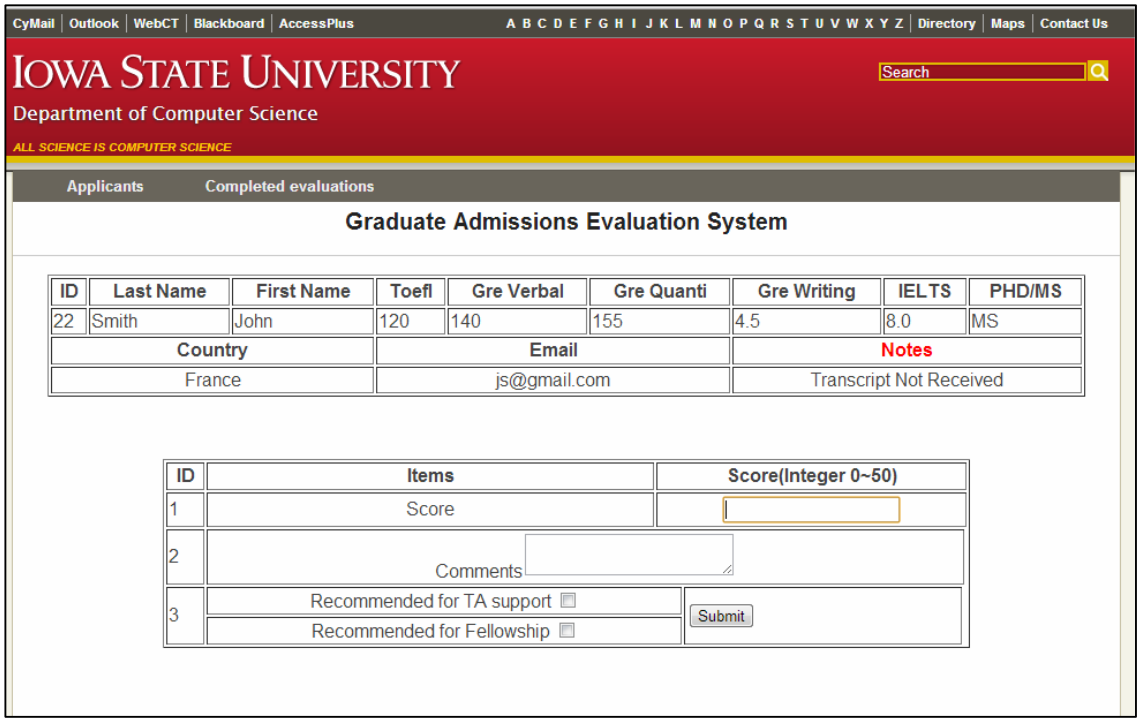


Committee member can get their own assigned applications list. They can select either All applicants, only PHD Applicants or MS Applicants. Once evaluating is done, the "Done" cell will change to "Y".



ID	Last Name	First Name	Toefl	Gre Verbal	Gre Quanti	Gre Writing	IELTS	PHD/MS	Research Interest	Country	# of Recomm Letter	Done
20	Garett	Tim	90	120	150	3.5	0.0	PHD	HCI	US	3	N
21	Lakhotia	Anand	60	150	150	3.0	4.0	PHD	Computational geometry	Sri lankan	3	N
22	Smith	John	120	140	155	4.5	8.0	MS	HCI	France	3	N
23	Banik	Divya	80	157	157	4.0	6.0	PHD	None	Indian	3	N
24	Luthra	Varun	90	150	148	3.5	7.0	MS	None	Chinese	3	N
25	Smith	Timmy	90	150	140	3.5	0.0	PHD	AI	US	3	N

By clicking the LastName and go to evaluating page.



ID	Last Name	First Name	Toefl	Gre Verbal	Gre Quanti	Gre Writing	IELTS	PHD/MS
22	Smith	John	120	140	155	4.5	8.0	MS
Country			Email			Notes		
France			js@gmail.com			Transcript Not Received		

ID	Items	Score(Integer 0~50)
1	Score	
2	Comments	
3	Recommended for TA support ☐	Submit
	Recommended for Fellowship ☐	

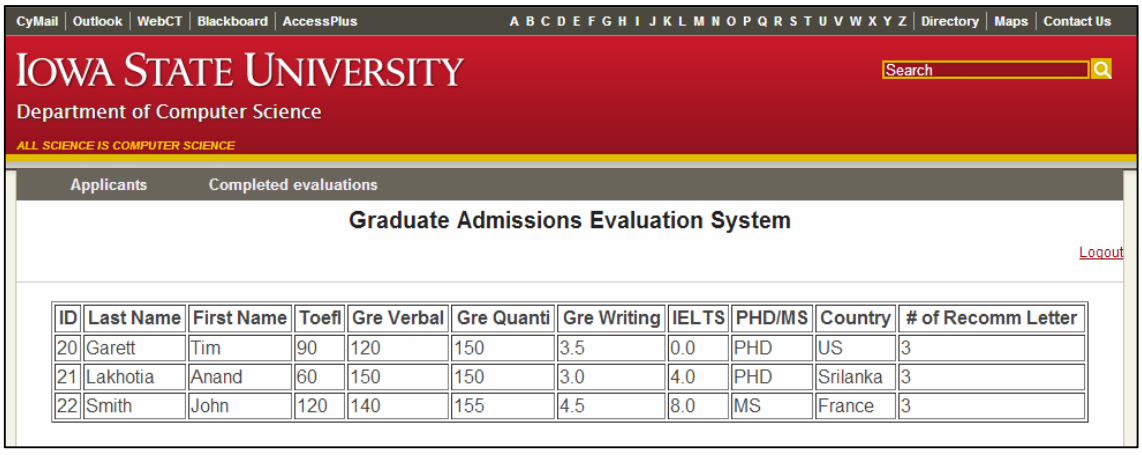
Type the score into Score textbox.

- Leave comment in Comments textbox.
- Click recommendation for TA / fellowship in checkboxes.
- Then submit.

If the applicant is not evaluated yet, then this will display the success page as follow:

If the applicant is already evaluated, this will display already exist page.

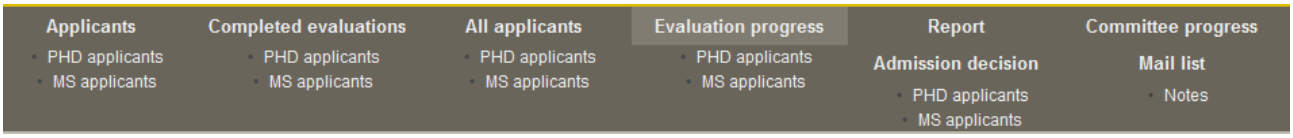
Committee member can check the evaluation for applicants in Completed evaluations.



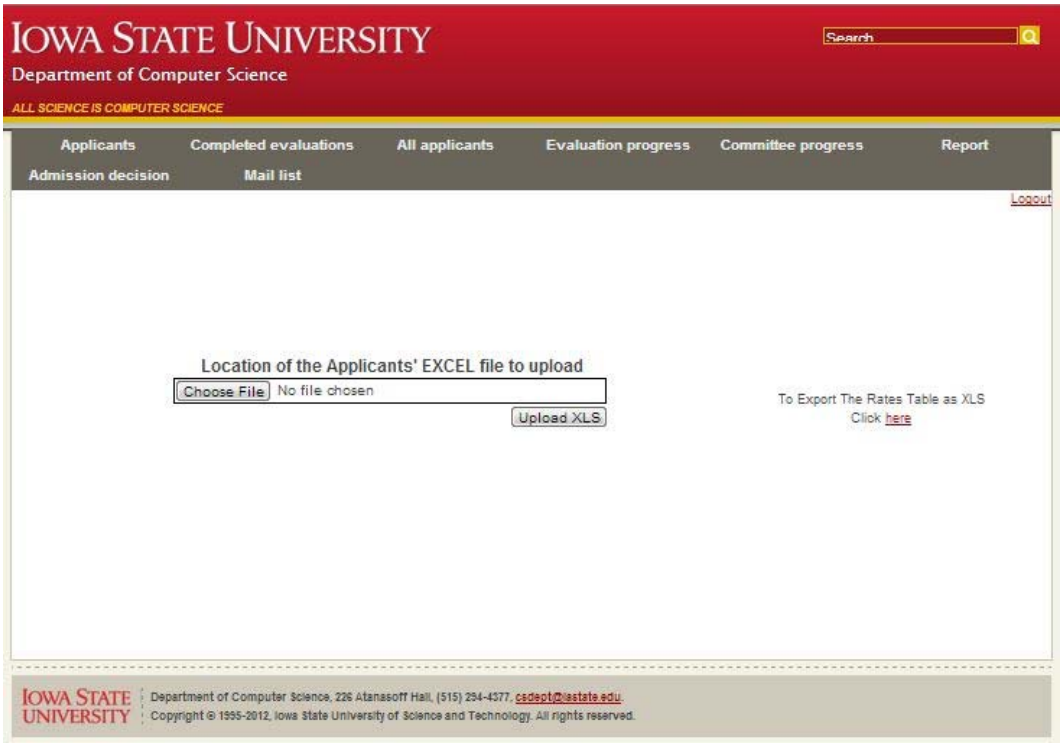
ID	Last Name	First Name	Toefl	Gre Verbal	Gre Quanti	Gre Writing	IELTS	PHD/MS	Country	# of Recomm Letter
20	Garett	Tim	90	120	150	3.5	0.0	PHD	US	3
21	Lakhotia	Anand	60	150	150	3.0	4.0	PHD	Srilanka	3
22	Smith	John	120	140	155	4.5	8.0	MS	France	3

Committee chair

Committee chair's menubar:



1. Committee chair has all the same evaluating work as that of committee member.
2. Committee chair is able to retrieve the list of all applicants.
3. Only Committee chair, Department chair and Staffs have the access to Import/Export data to/from the database in Excel format.



4. In report, it shows round statistics.

Application Round Statistics

1) Number of applicants
2) Number of US Citizen applicants
3) Number of Minority applicants
4) Number of women applicants
5) Average GRE QUANT score
6) Average GRE VERBAL score

7) Distribution of average evaluation scores (0-10, 10-20, 20-30, 30-40, 40-50)

Applicant Round Statistics

Item	Report	
Number Of Applicants	25	view
Number of US citizens Applicants	0	view
Number of Minority Applicants	25	view
Number of Women Applicants	14	view
Average GRE QUANT SCORE	119.36	
Average GRE VERBAL SCORE	123.4	

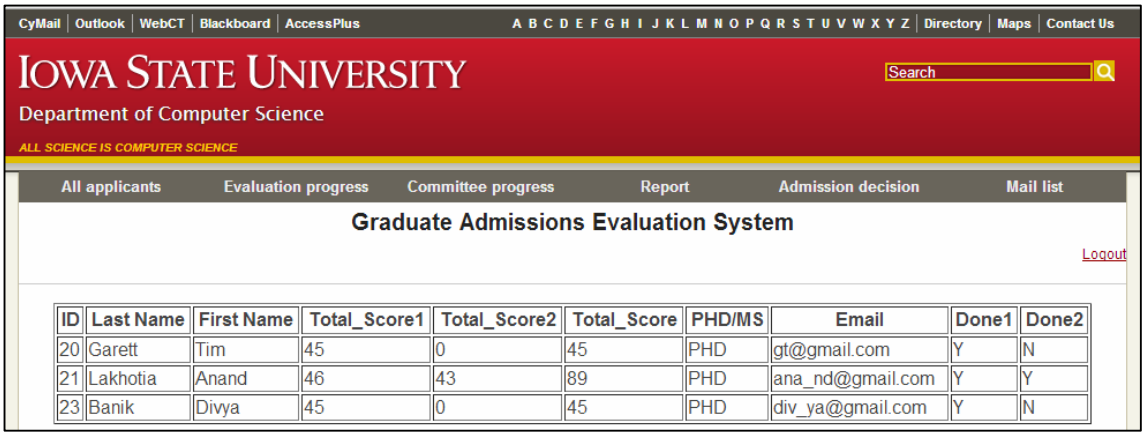
Distribution of Average Evaluation Score									
0-9	10-19	20-29	30-39	40-49	50-59	60-69	70-79	80-89	90-99
0	0	2	0	0	0	0	0	0	0

5. Committee chair is able to keep track of committee progress, which shows the progress of all committee members.

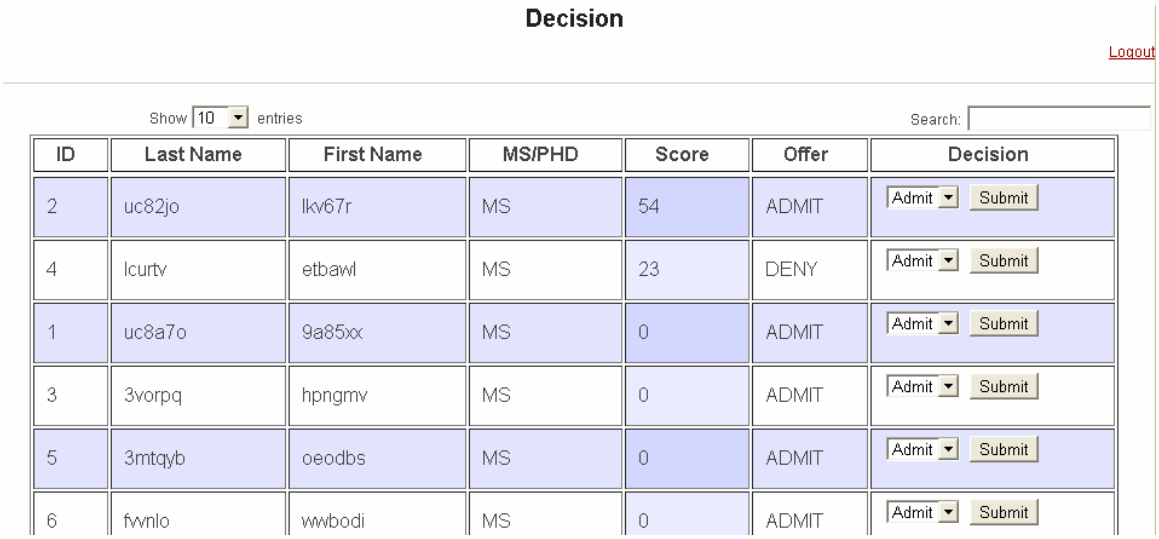
No Of reviews required to complete admission review process :	1
No Of reviews completed by Committee :	0

	MS	PHD
chaudhur	0/0	0/0
gsong	0/0	0/0
jil	0/1	0/0
jtian	0/0	0/0
pavan	0/0	0/0
tavanapo	0/0	0/0
wzhang	0/0	0/0
yingcai	0/0	0/0

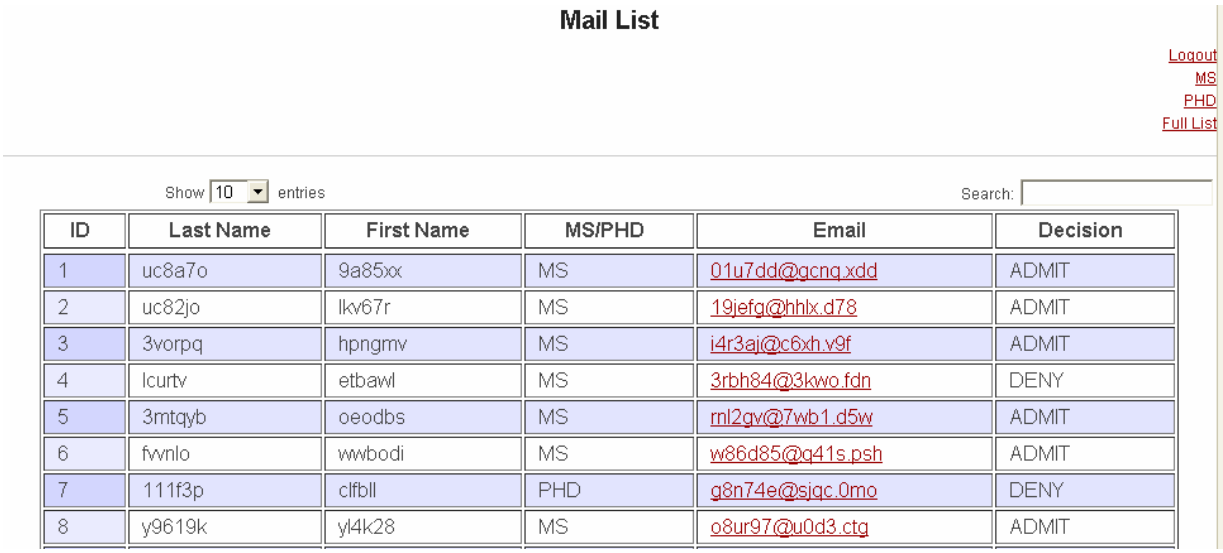
6. In evaluation progress, it shows the evaluation progress for each applicant.



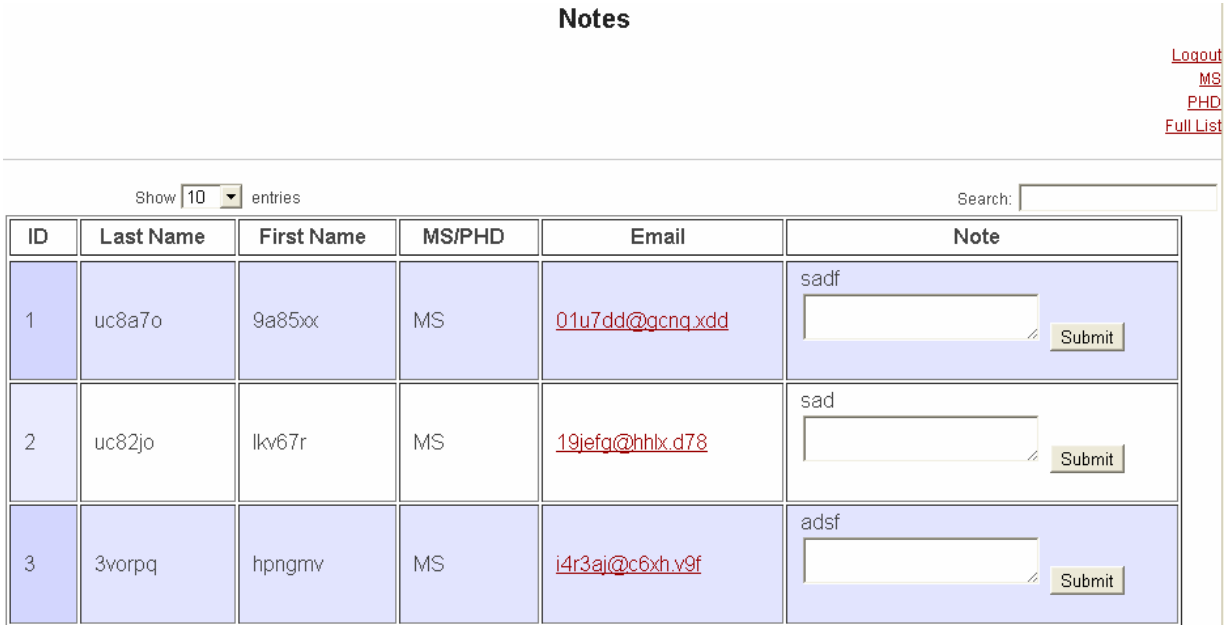
7. In admission decision, it lists all the evaluation results including comments ordering by total scores by clicking on “Score” in the table header. Committee chair could give admit/deny decision here.



8. Evaluation Mail list shows the final decisions of applicants and make easy for staff to sending email by clicking email address.



9. You can add notes for some application, for example," Somebody updated his ibt score to 115." and submit. Then this applicant should be evaluated again.

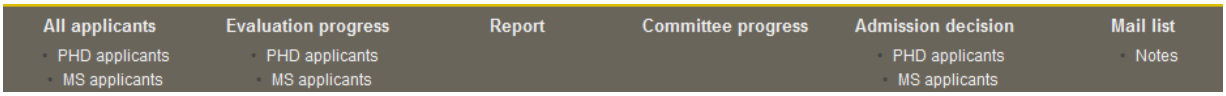


Department chair

Department chair's menubar:

Staffs

Staff's menubar:



1. Department chair and staffs are able to retrieve the list of all applicants.
2. Department chair and staffs are able to keep track of evaluation progress, which shows the scores committee member gave.
3. In report, it shows round statistics.
4. In admission decision, it lists all the evaluation results including comments ordering by total scores. Committee chair could give admit/deny decision here.
5. Mail list gives the final decisions of applicants and make easy for staff to sending email by clicking an applicant's email address.
6. Users can add notes for some application, for example," Somebody updated his ibt score to 115." and submit. Then this applicant should be evaluated again.

Members' tasks:

FENG YUNFEI 's part:

1. Contacting users, including Laurel, Dr. Yingcai, and some other committee members. Making clear of requirement, business logic and user inferfaces.

2. Integrating the Struts 1.3 and Hibernate 4.2.6.

3. Starting this project and developing the underlying structure of this project.

3.1 Design the Object/Relational Mapping in Hibernate (POJO files and xml config files.)

3.2 Develop DAO layer, including:

ApplicantDAO.java

ApplicantCommitteeRatingDAO.java

DecisionDAO.java

DutyAssignDAO.java

HibernateDAO.java

HibernateUtil.java

HibernateFilter.java

NoteDAO.java

ReportDAO.java

UserDAO.java

3.3 Develop utils files, including:

RandomString.java

Randomx.java

FinalConstant.java

GreScoreConvert.java

RanAver.java

Utils.java

3.4 Develop codes for automatical importing testing data into database, including:

DaoTest.java

GenerateDataInApplicantTable.java

GenerateDataInDecisionTable.java

GenerateDataInNoteTable.java

GenerateDataInUserTable.java

StringUtils.java

3.5 Define Exception code:

AuthorityException.java

DAOException.java

Exception.java

PasswordErrorException.java

3.6 Develop the servletsession part

AuthorityUtil.java

UserInfo.java

3.7 Develop the strutsaction part

AddDecisionAction.java

AddNoteAction.java

BaseAction.java

DispBaseAction.java

LikeStringSearchAction.java

ListAllApplicantsAction.java

ListAllApplicantsDispAction.java

ListAllApplicantsMapAction.java

LogonAction.java

LogoutAction.java

MappBaseAction.java

RegistAction.java

ResetPasswordAction.java

3.8 Develop the actionform part

AddDecisionForm.java

AddNoteForm.java

IdForm.java

LikeStringSearchForm.java

LogonForm.java

RegistForm.java

ResetPasswordForm.java

3.9 Develop page files (Modules)

- List Module

listAllApplicants.jsp

- Decision Module

listDecisions.jsp

- Mail Module

listMails.jsp

- Note Module

listNotes.jsp

- Log in and Change password Module

login_failure.jsp

login_success.jsp

logout.jsp

regist.jsp

resetPassword.jsp

3.10 Define the validator-rules.xml

3.11 Importing and adjusting to computer science department CSS layout including footer.jsp, header.jsp, designing menubar.jsp according to different category of users.

3.12 Paging format, sorting and searching in pagination functionality.

3.13. Configuration settings

hibernate.cfg.xml

struts-config.xml

web.xml

4.1 Integrating and testing the final version of project.

4.2 Composing this final project report.

KIM NI WANG‘s part

1. Develop the servletsession part

StudentInfo.java

ApplicantInfo.java

2. Develop the strutsaction part

ApplicationFormAction.java

CommitteeProgressAction.java

ImportApplicantXLSAction.java

ImportExportAction.java

LogonStudentAction.java

ReportAction.java

StudentLogoutAction.java

StudentRegisterAction.java

StudentResetPasswordAction.java

3. Develop the dao part

ApplicantDAO.java

CommitteeProgressDAO.java

DutyAssignDAO.java

HibernatDAO.java

StudentAccountDAO.java

ReportDAO.java

4. Develop the actionform part

ApplicationForm.java

ImportApplicantXLSForm.java

LogonStudentForm.java

StudentRegisterForm.java

StudentResetPasswordForm.java

4. Develop jsp page

- Student application Module

OnlineApplicationLogin.jsp

applicationForm.jsp

applicationSubmitted.jsp

studentLogout.jsp

studentResetPassword.jsp

studentRegister.jsp

- Committee Progress Module

committeeProgress.jsp

- Report Module

importExport.jsp

6. Develop codes for importing testing data into database, including:

ReportTest.java

ExportXLS.java

7. Configuration settings

hibernate.cfg.xml

struts-config.xml

web.xml

PRIYANGIKA PIYASINGHE's part:

1. Develop the servletsession part

Adding modification needed for CommiteeMember to existing servlets

2. Develop the strutsaction part

CMApplicantEvaluationAction.java

CMApplicantEvaluationActionSubmit.java

CMApplicantListAction.java

CMApplicantListDispAction.java

CMAppRandomAssignAction.java

CMCompletedListAction.java

CMCompletedListDispAction.java

EvaluationProgressAction.java

3. Develop the dao part

CMApplicantListDAO.java

ApplicantCommitteeRatingDAO.java

CommitteeRatingDAO.java

4. Develop the actionform part

CMEvalApplicantForm.java

5. Develop page files (Modules)

- Evaluation Module

cm_appEvalSuccess.jsp

cm_applicantEvaluation.jsp

evaluationProgress.jsp

- Evaluation Progress Module

list_applicantsForCommitteeMember.jsp

list_completedApplicants.jsp

6. Develop codes for assigning algorithm:

DutyAssignDAO.java

GenerateDataInDutyAssignTabla.java

7. Configuration settings

hibernate.cfg.xml

struts-config.xml

web.xml

8.1 Integrating and testing the final version of project.

8.2 Composing this final project report.