Idea

Task1

Utalk

Open Forum Software for university (Android app)

University Information Sharing Software?

Brief description:

We want to create a user-specific Software that can provide students with a platform to discuss course content, course evaluation and program assessments.

1. User：

(also has a user authorization page and different users have different levels of authority)

client-oriented, management control

Program officer:

(methods for example)

course.add(), course.delete(), comments.control().gf

program.add(), program.delete(), program.comments(), course.assessment()

Student: different categories based on year of study, etc.,

program of study + course enrolled.

2. Outline different course information; inside comment
3. function realization

course analysis

4. some other possible functions:

display the relationship between different courses,

basics for different programs

5. Basic structure:

register -> program -> different forums.

diagram

Specification

Our program is dedicated to provide a service for students to discuss course content, for faculty to post about course and program information, and for professors to post and delete comments. All three types of users should go through the registration process to access this program. Students will need their student id, and if already known, names of program enrolled and names of courses enrolled. Faculty and professors will need their staff id.

Students who enrolled in the same courses can post to the current course discussion board, reply to others' comments and delete their own comments. They also have access to the past post pages to learn from past students who previously enrolled in the course. They can also see other courses taught by their professors to learn about them.

Inside the course page, students can see course name, course information, professor information, current discussion board and the discussion boards in the past three semesters. Professor and faculty can see the same course page, with faculty being able to update course information. Faculty can create and update course pages. For professors, they have the right to delete comments and reply to the comments.

entities classes: User, CoursePage, AcademicSemester

case classes: Student, Office,

controller:

interface: UI interface

CRC model

User	
Responsibility ● int student_id ● String utorid ● String user’s name ● Arraylist course list ● Void function create new comments (addcomment) ● String function DeleteOwnComment (Return a notice if the user didn’t have access or delete otherwise) ● Void function register (Act as constructor, override in each subclasses) ● Void function enroll_course (All users can voluntarily enroll courses)	**Collaborators** Subclasses: Student Office Interact: Comment

AllCourse	
Responsibility ● store all course that can be enrolled: Dictionary[Str name, CoursePage course]	**Collaborators** CoursePage Faculty Professor

AllPrograms	
Responsibility ● Store all programs that can be enrolled Dictionary[Str name, ArrayList[CoursePage]]	Collaborators Faculty

CoursePage	
Responsibility ● string course_info (can be edited by professor) ● Arraylist[Professor] professor_list ● Arraylist[Student] student_List ● Arraylist[Postpage] (Store all relevant postpages)	**Collaborators** Comment Professor Faculty PostPage

ProgramPage	
Responsibility ● string course_info (can be edited by	**Collaborators**

professor) ● Arraylist[id CoursePage] (The list of course that contains in program arranged by faculty through edit_program_course)	Faculty

PostPage

Responsibility	**collaborators**
● Store Comments: Dict[int:Comment] ● store String current_time ● ●	Comment CoursePage

Comment	
Responsibility ● Store instance attributes: User user; String content; Arraylist[Comment] replies int poster_ID int Comment_ID	Collaborators User Comment

Student	**User**
Responsibility Arraylist [course] Courses_enrolled Arraylist[program] Programs_enrolled ● instantiate instances of this class ● enroll in courses through EnrollCourse ● Void function add_program from data in program page ● Void DeleteOwnComment	Collaborators User Comment Course

Faculty	**User**
Responsibility instance variable: ● ArrayList: course_management ● Void function Change_info ● Void function add_course ● Void function edit_program	Collaborators User Comment PostPage ProgramPage

<table>
<tr><td colspan="2"><b>User</b>
<b>Professor</b></td></tr>
<tr><td>Responsibility
● Void Function delete_comment
● Void Function add_pro_comment
● Void Function edit_comment
● Void function edit_course_info</td><td>Comment
AddComment
PostPage</td></tr>
</table>

Interface

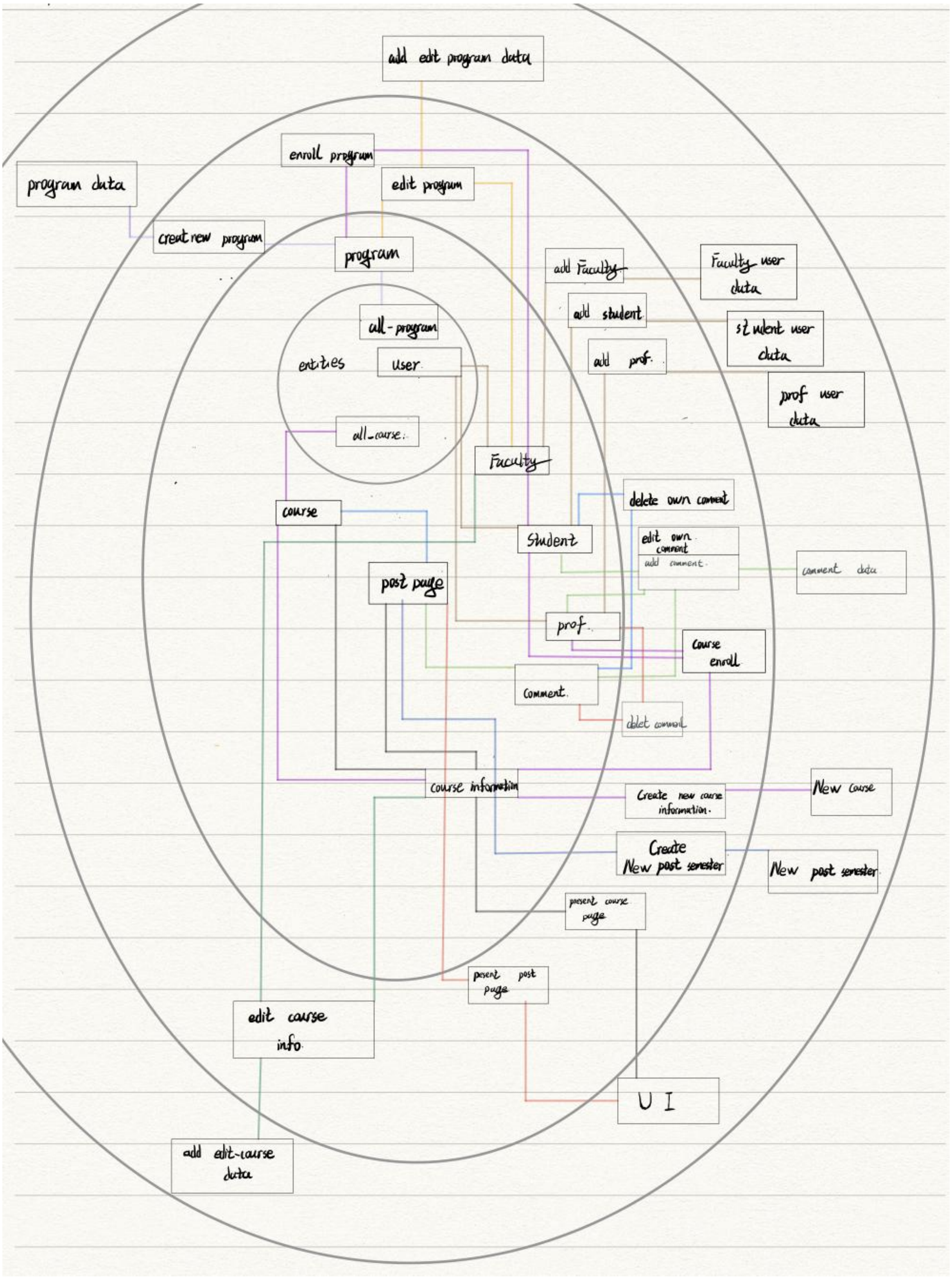
AddComment

Responsibility	Collaborators
● Void function add_comment	User CoursePage

Interface **Registeration** **Student, Office, Professor**	
responsibility ● Void constructor register()	**Collaborators** **User**

Interface **EnrollCourse** **Student, Professor**	
responsibility ● Void function enroll()	**Collaborators** Student Professor

contoller: pass the data and change it type to the one java can deal with
course; parent class of comment
Storage: in the database


add edit program data
enroll program
edit program
program data
creat new program
program
all-program
entities
user
all_course.
add Faculty
Faculty user data
add student
student user data
add prof.
prof user data
Faculty
delete own comment
course
Student
edit own comment
add comment.
comment data
post page
prof.
course enroll
Comment.
delet comment
course information
Create new course information.
New course
Create New post semester
New post semester
present course page
present post page
edit course info
UI
add edit-course data