**E-business Systems for Online Collaboration**

dai-hoa

Instructor: Dr. Hanh Pham

**We’ll add more details to this as we go. Please check daily!!!**

**A) Project General Description:**

Together we’ll build a website called Collaborative Online System (COS) for users (customers, staffs, managers) to collaborate online. This kind of e-business systems can be used to monitor and conduct many business processes/jobs. For examples: ordering and shipping a product to customers, new hiring, requesting IT services, or requesting and approving a travel.

Each collaborative JOB has steps/tasks. Each TASK has users/participants and data items involved. Data items can be FORMS or FILES. Users can communicate via communication channels (mailboxes) and emails.

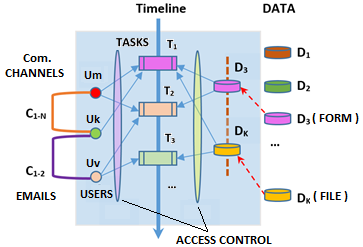


Figure 1. Inside a JOB

Students are assigned into groups which work on different units/components of the system, all must be developed and run on the CS shared folders. Besides developing codes and database tables students will also need to write and submit technical reports including developer and user guides.

The system consists of the following units:

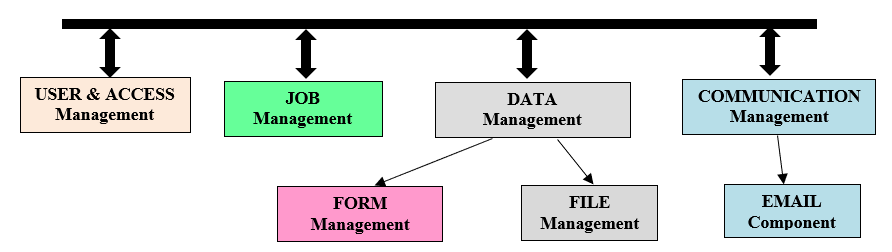


Figure 2. Units/Components of the Collaborative Online System (COS)

**USER & ACCESS Management (Unit A)**

This unit lets users register, edit profile, login, logout; view, join and quit jobs; check access control. Staff, manager, and admin users may manage other users and their roles and access policies… Every time when a user does/clicks-on something the system should check (via the access/policy control matrix) who the user is, does the given user has permission to do (view,edit,create,delete,…) that(form,file,…)?

**JOB Management (Unit B)**

This unit lets users view, create, modify, and delete jobs. It also lets users view, create, modify, and delete TEMPLATES which are similar to Java classes. A job, which is similar to a Java instance/object, can be created based on a template. A template or a job consists of tasks/steps. Each task/step has participants/users and data items(forms or files). A new template can be created based on a prebuilt/existing template.

**DATA Management (Unit C)**

This unit lets users view, create, modify, and delete data items in tasks and jobs. Data items can be forms or files. This unit should let users upload new, view, … and remove files (for a task/step). It should check beforehand if the given user has permissions to do that with the given data item.

**FORM Management (Unit D)**

This unit lets users view, create, modify, and delete forms. A form consists of components which can be label, input textbox, radio button, checkbox, and button. Users can add/modify/delete components of a form.

**COMMUNICATION Management (Unit E)**

This unit lets users send messages to each other. Each user has a mailbox to store in-coming and going-out messages. Email alerts are sent for emergency messages. Users should be able to view their own messages.

EXAMPLE 1: Three workers have to collaborate to create, edit, and approve a company report of three parts (each part includes a form and a file), a JOB can be created with three users A1, A2, and A3 and the three tasks T1, T2, and T3 which represent the report three parts as shown below.

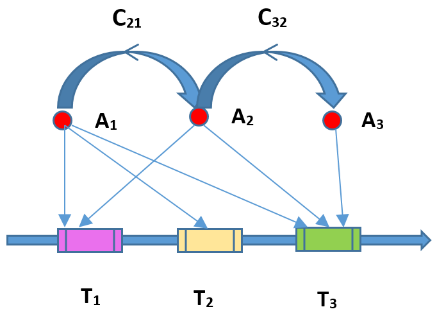


Figure 3. Collaboration in a Document Co-edit JOB

The first worker A1 can be a manager who writes the second part and also oversees and approves the contents of all parts of the given document while worker A3 is in charge of the third part and worker A2 can provide statistics information for the first and the last parts. As workers A1 and A2 co-write the first part and worker A2 and A3 co-write the last part significant communications between them are required and therefore we see the two communication channels C21 and C23 are created. C13 is not needed as A1 only needs to approve or reject the contents of the third part represented by task T3 without participating in the creation of this part contents.

EXAMPLE 2:

In another example, when a salesman needs to make a trip to see a client he needs to ask the secretary for a travel request. Together with the client, the salesman become the two users who need to work together on defining the schedule of the meetings with all of the details including the objectives of the meetings. Then, it’d need to be forwarded to his supervisor who may like to change certain parts of the meeting schedule. After the supervisor approves the request it would go to the finance/accounting department head who will check to see if there is available fund for the whole trip. Once the finance department head verifies the needed fund the CEO can approve it. Then, the travel plan will progress further by the Purchase office worker who will use the company Bank account to buy the transportation tickets, making hotel reservations, etc.

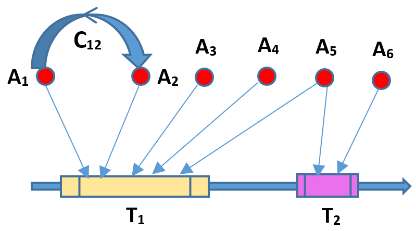


Figure 4. Collaboration in a Travel Request and Approval

With COS these collaboration logistics can be represented by a JOB in a simple way. As this kind of travel requests is popular, the secretary may have already built a template TR, called Travel Request, for that. When the salesmen asks the secretary creates a workflow based on that that TR template. TR should have the following users: salesman A1, client A2, supervisor A3, finance head A4, A5 for the CEO, and purchase worker A6. There should also be a communication channel C12 between the salesman and the client so that they can discuss details of their meeting schedule. These two actors must also work together to define and complete the trip objectives in T1. Once the salesman actor A1 and the client user A2 finalize and approve the trip plan in T1 it should be viewed and approved by A3, A4, and A5. After that, the purchase is generated via task T2 and should be implemented by A6.

**B) Unit & Group/Team Assignments**

Please login to Blackboard and go to MyGrade then GroupID to see your group ID and look up in this table to see your assignment.

|  |  |  |
| --- | --- | --- |
| **Group ID** | **Unit ID** | **Leader/Coordinator** (b-f20-XX) |
| A1 | A | Charulata Yadu (48) |
| A2 | A | Akshay Baba Koka(12) |
| B1 | B | Conor O’Riordan (27) |
| B2 | B | Akhila Reddy Ganta (07) |
| C1 | C | Azer Khan (11) |
| C2 | C | Venugopal Sai Sankar Naredla (26) |
| D1 | D | Madhu Rakshith Narasimharaju (25) |
| D2 | D | Purna Chandra Rao Vellalacheruvu (46) |
| E1 | E | Punith Srinivas (38) |
| E2 | E | Venkata Sripadh Kausthub (01) |

For example, if your groupID is B1 then it means you are in the group/team B1 and need to work on Unit B. You can go to Blackborad, DISCUSSION, your team discussion thread (in this example: "Discussion for Group B1:") To see all member of your group.

**IMPORTANT:** the assignment for EACH student in a group is the same which is to do the WHOLE unit. At the end each student gets individual grade for the project and is responsible for delivering the whole unit and related documents. However, students within the same group are allowed and encouraged to exchange their ideas and work. Ideally, students can split the work between themselves so that every and all students contribute their shares. For example: student #1 does part X, student #2 does part Y, they exchange their work so together both of them have X and Y done. If a student fails to contribute his/her part then the student may need to be on his/her own which is to do the whole unit without the group: this decision can be made by the group leader based on consultations with other group members.

**Question:** at the end of the project, can a student claim that because other students didn’t do their work therefore he/she couldn’t do his/hers and it is not this student’s fault that he/she couldn’t complete the project?

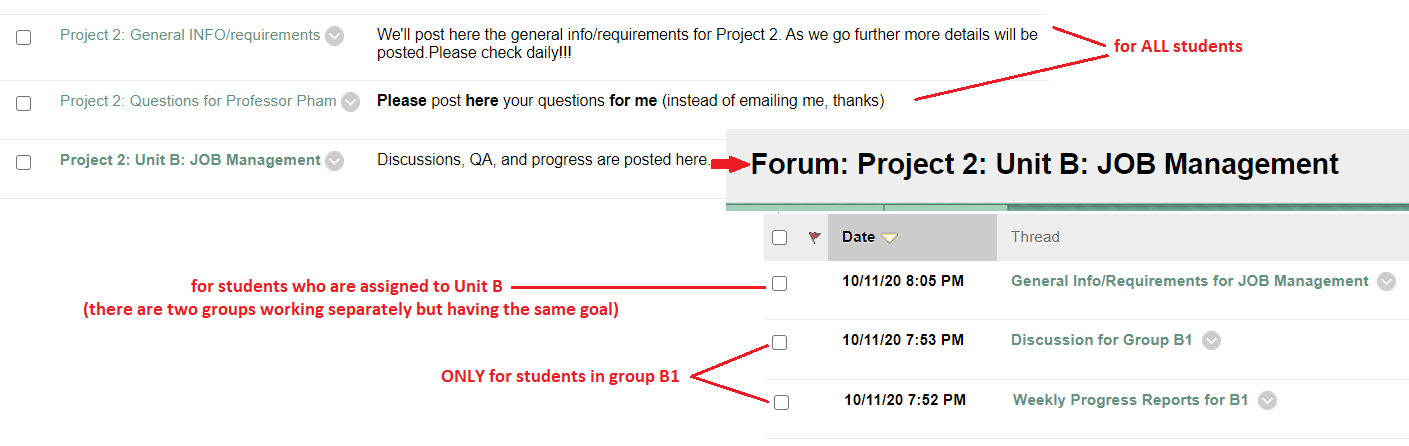
**Answer:** NO, as stated above, the assignment for each student is to do the whole unit, being in the same group means students can collaborate and share work (if agreed) it doesn’t mean that others do work for you.

**C) Working PLACE:**

Since the project must run and be stored on CS server students must use our MySQL database and use only programming languages and supporting frameworks already there without additional installation by the admin. I strongly recommend and prefer that everyone stays with PHP which is what we used in Project 1.

During development time a copy of the group current website and database must be at the CS shared folder of the group coordinator/leader. The group should meet at least once per week to assess the current progress, set goal, and divide the work for next week. Group leader should set up the meetings, post the link to the group website and also the group weekly plan/assignment at the “Discussion for Group …” thread.

COMMUNICATION and REPORTING YOUR PROGRESS: please go to **DISCUSSION at Blackboard and click on the forums with your unit and group/team ID**. You’ll also have access to the general info/requirement and QA forums.



**D) Project Requirements/Steps & Deadlines:**

|  |  |
| --- | --- |
| **To do** | **Due** |
| Make plans and schedules for the project.  Design:   * Types of Users * User Actions => Transactions * Database Tables * Architecture * Protocols (for each transaction) | 10/18 (draft) => feedback |
| Implementation:   * Write codes * Debug & Test | 10/18 – 11/07 |
| Demo/Presentation | 11/10 => Grading Software |
| Write & Submit Documentations:   * User Guide * Developer Guide * Report | 11/24 => Grading Report |

More details may be added later. Please check daily.