
2019 U.S. Information Technology Collegiate Conference

Systems Analysis and Design Competition

Sponsored by Global Shop Solutions



**Do not put your name(s) or your school's name on anything that you submit.
Doing so will result in disqualification of your team.
The only identifying information you should use is your team number.**

Student Membership Tracking System

US Information Technology University (ITU) has a student organization whose mission is to facilitate student collaborations beyond the classroom. Over the years, the organization has grown in size from just a few members to more than 100 registered members. They regularly host guest speakers, student networking events, and career events across the campus throughout the year. With the growth in members and the demand for a variety of different events, they feel their traditional pen and paper plus Excel approach is no longer sufficient for managing the organization's membership roster and various activities. They need a new system to help facilitate their ongoing operations and have reached out to you for a solution.

Here is some background information to help you get started. The student organization is open to all students in the university with all majors. But, for better management and tracking, students are asked to provide the following information when they sign up: name, major, classification (freshman to senior or graduate student), and their university email address. The university email address that a student provides will be the primary communication channel. They have annual dues of \$85 for first time members and continuing dues of \$40 for returning members. Memberships are good for a whole calendar year after their payments have cleared in the third-party payment system. They want to utilize an information system that can help with the following:

- The system should be able to track memberships and should generate email notifications when a member's membership is due for renewal.
- The system will provide a portal where all current members can log in and view events, sign up as volunteers, and post discussions.
- The system will allow members to self-register for competitions and send them confirmations once their requests have been processed.

- Every year in the spring, the organization will have an election for its executive council. Members who are in good standing and not graduating in the spring are eligible to be elected. Once the executive council is elected, the system will grant them administrator access.
- The administrators of the system will be able to create events, send mass communication emails, check membership information, and also manage the discussion board.
- As described in the bylaws of the organization, a member can be terminated if the requirements for attendance at meetings and events are not met. In general, a member should attend at least half of the scheduled meetings.
- After each meeting, the system would update the attendance information and indicate if a member has been absent.
- Each year, the executive council will determine if a member is still in good standing by reviewing an attendance report generated by the system.
- The system also allows members to form interest groups and manage their own activities within the organization.
- When a student wants to become a member, they go to the portal page and signup. The executive council will be notified to review information submitted and invoice the students for dues. If the student has paid the invoice within 7 business days, they become a regular member; if not, the system will send one reminder email to the student. If they have not paid after 7 business days, their information will be automatically deleted by the system.
- The system will be housed inside the university network and should provide adequate levels of information security because student member data will be kept inside the system.
- In addition to the students, a faculty advisor will also have access to the system to help ensure proper operation of the organization.
- The faculty advisor will have access to all management functionalities of the system. He/She can also edit the members' role(s) inside the organization.
- When members graduate, their information will be kept in the system if they continue to be enrolled in the university. For example, if they finished undergraduate studies and then became graduate students. Otherwise their information will be removed from the system.
- Every semester the system will perform a status check to make sure all of the member information is correct by contacting the member and asking them to review their membership details so that the member information can be updated as needed.

Your team has been asked to help the ITU student organization design a membership tracking system that can provide the above functionality and that is easy to use by students. In your solution, the system specification must include elements to assure no violation of the requirements. Multiple users may log in the system at the same time; therefore, it is important to protect the working sessions so that there is no inconsistent data regarding events, members, etc.

YOUR ASIGNMENT IS to use only **one** technique (either Object Oriented **OR** Structured/Traditional Technique) to specify how the system should operate. If you use a structured technique you must specify the flow of data inside the system. If you use an OO technique, then you must specify the classes inside the system and how they are used in order to achieve the system's objectives.

WHAT TO TURN IN:

If you are using the **structured/traditional approach**, then you are expected to turn in the following:

1. A Context Diagram.
2. A level 0 (zero) Dataflow Diagram.
3. A Level 1 DFD for each one of the processes that you identified in your Level 0 System DFD.
4. Process descriptions for the processes contained in your DFD's.
5. An Entity Relationship Diagram (ERD) showing the 3rd Normal Form Database that will support the system you designed.
6. Prototype with Windows Forms and/or Web Pages.

If you are using **an Object-Oriented approach**, then you are expected to turn-in the following:

1. Use-case Diagrams.
2. Use-case Descriptions.
3. Sequence and/or Activity Diagrams.
4. A Class Diagram (for objects in persistence storage).
5. State machine diagrams.
6. Prototype with Windows Forms and/or Web Pages.

For creating models, you may use your own business modeling software. This could include any CASE, I-CASE or other model-based development product. **ALL SUBMISSIONS FOR JUDGING MUST BE SAVED IN PDF FORM so that the judges can view your work.**

The prototype must be developed based on your models. It does not have to be fully implemented; however, a system design that provides mocked-up screens with window form/web page interaction will be considered in the over-all grading. The screens can be created using any graphical drawing software (such as Microsoft Paint or Photoshop) or you can take screen shots from development tools (such as Microsoft Visual Studio, Access or Eclipse). Given the time limit of the contest, handwritten mock-ups are allowed; however, the screens created by computer software will be given better grades. AGAIN, ANY IMAGES OR DOCUMENTS YOU PREPARE MUST BE SAVED IN PDF FORM FOR JUDGING.

When submit your work, make sure you submit **only one pdf file through the contest LMS**. Do not include any identifying information about yourself or your college. **ONLY** type your **team number** on your submission. If your submission contains any personal or college information, your team will be disqualified.

Contest Evaluation

The judges will use the following categories as they evaluate your team's recommended solution. The models that your team will be required to develop depend on which methodology is selected.

NOTE: Competitors are expected to utilize ONE and ONLY ONE Analysis and Design approach. Using a combination of components from both the Structured/Information Engineering approach and the Object-Oriented approach should be avoided.

	%	Structured/Info. Engineering	Object-Oriented Approach
Information Flow	40	Decomposition, DFD's, Dependency and Process Action Diagrams.	Use Cases, Sequence and/or Activity Diagrams
Information Structure	40	Entity Relationship Diagrams (ERD's) and Data Constraints	Class Diagrams (for objects in persistence storage) and State Charts (State Machine Diagrams)
Prototyping	20	Windows, Screens and/or Web Pages	Windows, Screens and/or Web Pages
Overall	100		