ICS STUDENTS' ACADEMIC INFORMATION SYSTEM (ICS SAIS)

General Description:

The ICS SAIS is offered as a secured web-based application that can be viewed using any of the popular web browsers (Chrome, IE, Firefox, Netscape, Opera, etc). SAIS handles all the academic information of students and presents them in a presentable manner that is understood and utilized by the users of the system according to their needs. SAIS focuses on serving UPLB BSCS students only.

There are four types of users of the system:
Students
Instructors
Encoders
Admin

Students are the users whose records are put to the system for management. The data they supply includes, but is not limited to the following:

> Basic user information (Name, Home Address, College Address, Home Address Contact No, College Address Contact No, Student Mobile No, etc)
>
> Academic Information (Registration Adviser, SP Adviser, Previous SP Advisers if any, Curriculum followed, GE chosen, Electives, Extra Subjects Taken, Grades, etc.)

Instructors are the ICS faculty members who can view student basic information, view students who have satisfied the prerequisite of a course (example: prerequisite of CMSC 127 is CMSC 123 or COI), view list of advisees if any and others. Information needed from instructors are as follows:

> Basic Information (Name, Mobile No, Current Schedule)
>
> Academic Information (Designation, Rank, Faculty Room, SP Advisees if any, Registration Advisees if any, Previous Subjects Taught, Current Subjects Taught, Previous Schedules, etc.)

Encoders are users that input data for students. The said data are not readily available for them (ex. Grades, for a subject, etc.). The information needed for an encoder is as follows:

> Basic Information (Encoder Name Role [Faculty, SA, Admin Staff])
>
> Note: Encoders are users created by the Admin

Admin or administrator is (take note: is) the user that monitors the behavior of other users as well as the behavior of the system against user actions and inputs. He / She approves the requests for activation of accounts and other powerful tasks. The following information is needed from the Admin:

> Basic Information (Name, Position/Rank, Email Address)

Functionalities

The following functionalities are expected to be delivered:

Student Users
> Add, edit, delete, view basic information
> Change Password
> View Subject Grades
> View Running GWA
> View Subjects Taken
> View List of Electives Allowed to be taken
> View List of Approved Electives
> View List of Approved GE
> View List of Adviser History (SP and Registration Advisers)
> View List of University Cases (if involved in any serious cases)
> View List of Academic History (Good Standing, Probation, etc.)
> Send/Receive Private Message to/from Admin

Instructor Users
> Add, edit, delete, view basic information
> Change Password
> View student list who have satisfied the prerequisite of a given course
> View Previous Students associated with the instructor
> View Current Advisees (Registration and SP, in reverse chronological order)
> View Advisees who graduated from BS Program (SP, in reverse chronological order)
> View Previous Advisees (those who transferred to another Adviser, applicable to current/registered students only, in reverse chronological order)

Encoder Users
> Add, edit, delete, view student academic information (academic info only. The functionalities are based on approved documents submitted by the student and are received at the institute level such as but not limited to Change of Adviser, Plan of Study, GE Plan, etc.)
>
> It would be beneficial if data can be updated/uploaded in bulk.

Admin
> Add/Delete a user account
> Approve/Disapprove registered user
> Deactivate User Account
> Monitor user log activities
> Send/Receive private message to/from student
>
> Note: The system has the option to let the user register using the registration module or let the admin add him/her. In the case of the latter, there is no need to approve/disapprove the user (user can be student, instructor or encoder).

Security Features

The system must be secured whenever a user logs in. Other security features aside from what is described in this document can be included by the developers.

HTTPS
Secure connections via https.

Encrypt Data over the Internet
Encrypt overly sensitive data whenever they are transmitted to the server.

Encrypt Passwords and other sensitive data in the Database
Security features must persist in the different layers of the system and that includes the database!

Clean user data and Error Catching Mechanism
Ensure that user inputs are valid. Provide error catching mechanisms for forms or for "unexpected" user or system behavior.

Privacy Issues

Ensure that the right to privacy of students and other stockholders of the system are observed. Encoders as much as possible should only be provided of the student number of the student in order to update the student's grade. Similar approach must be used whenever applicable.

User friendliness

Ensure rich user experience by making the system user friendly. Add descriptions and messages whenever necessary.

Programming Language and Database

You are free to use any web-based scripting language (and if possible, an additional programming language for backend use). You are encouraged to use frameworks.

Likewise, you are allowed to use any DBMS provided that it is free for academic use (example: PostgreSQL and MySQL).