Term Project

Due Dates

5PM November 24 (Monday), 2014	1st Preliminary Design Document
2PM November 26 (Wednesday), 2014	1st Prototype Demo
5PM December 15 (Monday), 2014	2nd Preliminary Design Document
2PM December 17 (Wednesday), 2014	2nd Prototype Demo
5PM January 5 (Monday), 2014	Complete Design Document
January 5 (Monday) – January 6 (Tuesday), 2015	Complete System Demo
2PM January 7 (Wednesday), 2015	Final Report and Oral Presentation

Project Description

Every university desires to keep in contact with as many of its alumni as possible, for good reasons. A university would certainly be interested in knowing, among other things, the career paths of a particular group of alumni so as to measure the success of the education it provides. However, maintaining current contact information of each alumnus is already hard enough, let alone information about her/his employment history. Given the popularity of social networks, it is not unrealistic to envision a Web-based forum for the alumni of a university, via which such information may be more easily collected, maintained, and put into good use.

The objective of this term project is to develop a Web-based system, called "AlumniBook" perhaps, for facilitating the interactions between a university (or an academic department within a university) and her alumni and also interactions among the alumni. Given the time and person-hour constraints, we shall focus on the following three groups of functions (besides authentication/authorization and access control).

- Maintenance of User Profiles
  - The user of the system may be an alumnus, student, staff, or faculty member of the university (or department).
  - A user profile should include the user's name, phone number, email address, current employer, position, and a short remark (perhaps a very brief autobiography, special skills, or a list of hobbies).
  - For alumni, such profile information may be derived from the Alumni Database, to be described later.
  - A user may update his profile, for example, changing his email address.
  - A user may look up the profiles of other users when permitted.
  - It should be a good idea to allow the user to post a recent picture as part of his profile.
- Forum for Exchanging Information/Ideas
  - A user may raise an issue (including asking a question, announcing an event or job opening, looking for a job, etc.), hoping others to respond.
  - A user may follow up an issue by posting his comments.

  - A user may browse all issues that have been raised. To make browsing easier, the issues may be sorted, for example, by date or category.
  - A user may view all comments addressing a particular issue, which may be sorted or filtered in some way.
  - A user may subscribe to a category of issues and be informed by email when new comments are added or a new issue in the category is raised.

- The Alumni Database (with Contact/Education/Experience Information)
  - The contact information of an alumnus should include postal address, email address, mobile phone number, and phone number at work or home.
  - Regarding education, each degree should include the starting date (year and month), the ending date, the degree, the granting university/instution, the department/program, student ID, possibly subject(s) of concentration, and a completion check indicating whether the degree has been successfully obtained.
  - As for experience, each employment should include the starting date, the ending date (if the employment has terminated), company/organization, department (if applicable), position, and a short description of the job.
  - Interesting statistics may be taken from this database (perhaps with auxiliary information), for example, which company/organization hires the most alumni and what kind of job is most popular among the alumni with a same degree.

To anticipate the use of the system by alumni of different nationalities or even the deployment of this system in universities around the world, the user interfaces of the system should be multilingual, including at least Chinese and English. Additionally, as a demonstration of the capability of accommodating users with mobile devices, the system should provide APIs for mobile applications to query the phone number of a user.

We shall try to decide together (including discussions in class) a common database scheme for the alumni database and the exact details of the mobile APIs. *We assume that the user of this system has a computer/email account with the university for authentication purposes.* A simulated single sign-on server, that maintains an agreed set of users, will be provided for the demos.

Though the exact set of functions/features is up to each team, the ultimate goal is the following:

> Ultimate goal: *make AlumniBook so fun and effortless to use that the alumni will want to visit the website often (or at least once in a few months), and if sufficiently incentivized, will keep updating their contact/education/experience information.*

APIs for Mobile Applications

The description below may be incomplete and we shall try to resolve any remaining issues in class.

For querying phone numbers, the fields and their types are as follows:

field name	type	comment
name	string	name of the user

For simplicity, we do not require a parameter that carries authentication credentials. The result of a query is an array, each entry of which includes the following fields:

field name	type	comment
name	string	name of the user
id	string	account ID
phone	string	phone number

All data should be encoded in the JSON format.

Non-functional requirements

There are also non-functional requirements, including security (secrecy, privacy, access control, software security, etc.), concurrency control, and system robustness.

- **Secrecy**: Transmission and storage of sensitive data should be protected.
- **Privacy**: Privacy of all users should be respected. A policy of privacy should be in place and enforced.
- **Access Control**: An adequate access control policy should be in place. Every piece of data can be accessed only by a person with the access right. Note again that a single sign-on server will be provided for demos.
- **Concurrency Control**: Several users may access the website at the same time, without interfering with each other or causing inconsistency in the data.
- **System Robustness**: The system should be robust and gracefully handle any illegal inputs by the user.

General Instructions

- You must use the Git version control system, set up for this course, to manage your development work.
- The AlumniBook website and the alumni database should be on different and remote servers (rather than the local host).
- The design documents and the final report should be in printed form. Please use A4 paper and *double-sided* printing. Simply staple on the upper left corner; NO plastic or cardboard covers and NO binders, either. Drop each design document, by its deadline, in the physical mailbox of Yih-Kuen Tsay (the instructor); put the final report on the instructor's desk before the final presentations start. Late submissions will be penalized 20% for each working day overdue.
- If you are willing to make your design and implementation available to future participants of the course, we would appreciate very much a copy of CD-ROM that contains all relevant sources to accompany your final report. Please include in the CD-ROM compilation and installation instructions.
- DO NOT plagiarize (i.e., do not use material without crediting the source).

Design Documents

The term project is expected to be implemented in three stages:

1. Stage 1: functions related to user profiles maintenance and the forum; multilingual interfaces ready already at this point
2. Stage 2: functions related to the Alumni Database (and improved functions from Stage 1)
3. Stage 3 (completion of the project): access control, the mobile APIs (a mobile client is optional), and other enhancements (including improved functions from previous stages)

Accordingly, there will be three required design documents: two preliminary design documents and one complete design document, at most 12, 16, and 20 pages long respectively. The preliminary design documents constitute an evolution to the complete design document, which gives a complete and thorough description of your system design.

A design document should include at least the following items:

- an overview of the entire system,
- design of the components in the covered scope, including the various UML diagrams and their accompanying specifications,
- any other verbal or diagrammatic descriptions that would help clarify the design (e.g., the graphical interfaces), and
- discussion on how knowledge learned from the course has been applied.

Demonstrations

- Preliminary prototype demos
  - A prototype demo should be short, about 10 minutes, showing sufficient evidence that the current implementation meets the goals of its preliminary design.
  - They will be scheduled during the breaks of the class meeting on their due date.
- Complete system demo
  - The complete system demonstration should be about 30 minutes long.
  - To allow time for discussions, one hour will be allotted to each team.
  - Please schedule well in advance (at least one week before the due dates) a date and time with the instructor.

Oral Presentation

Each team should give a 30-minute oral presentation with an appropriate set of slides; the presentation is to be followed by a Q&A session. The slides should be designed in such a way that they can be made publicly available on the course website. The presentation must include a demo.

Final Report

The final report should be at most 15 pages long and include the following two parts:

Part One

- an overview of the system from the users' perspectives
- simple (but self-contained) manuals for the user or application developer

Part Two

- a summary of the final design (including possible changes and the reasons for these changes, since the final design document)
- the lessons (not necessarily technical) you have learned
- the task allocation, identifying what each team member has contributed to the project

Grading

Item	Percentage
1st Preliminary Design Document	10%
1st Prototype Demo	10%
2nd Preliminary Design Document	10%
2nd Prototype Demo	10%
Complete Design Document	10%
Complete System Demo	10%
Final Report	10%
Oral Presentation (with slides)	20%
Source Code (style, documentation, etc.)	5%
Usage of Tools (IDE, Git, etc.)	5%

All members of a team basically will receive the same score for the term project. However, a peer evaluation will be conducted within each team following the final oral presentations. The evaluation result will be used to adjust the score of each team member, up to 20% more or 50% less than the original score.