SJSU CMPE 138 Term Project Spring 2016

You and your teammates will demonstrate your mastery of the material taught in this course with a multi-phased term project. The project consists of a non-trivial relational database to support a particular activity, event, or organization, and a database application as front-end to access the database. The initial project phases will concentrate on the design and construction of the relational database; the final phases will add an application with sufficient power to update, retrieve, and display information from the database, and to allow new data to be added to the database.

Objectives

- To reinforce and practice database queries and design concepts learned in class
- To follow good software engineering practices in design, implementation, testing and documentation
- To learn how to write good technical report and do good presentation
- To collaborate effectively in a team environment

Team

Each team consists of **?** to **?** students – to be announced after the drop-class deadline. There is no restriction on who can team up with whom; graduate students can team up with undergraduates.

Each team should select a team lead to coordinate various team activities and submit team assignments and reports. All assignments should be submitted to Canvas on time.

Each team member should collaborate with each other and be a good team player to complete the term project.

Schedule

Just like any homework assignment, you must be submitted each of the followings **on time**.

1. Team Formation: 1:30pm on 2/15
    - List of names, IDs, and email address. Who is the team lead.
    - Email the following info to kong.li@sjsu.edu with subject "CMPE138_TEAM"

      L3SID lastName firstName
      … … …
      Team lead: …

    - L3SID is the last three digits of your student ID.
    - A unique team number n will then be assigned.
    - Each team becomes a "user group" in Canvas. Each team would need to complete several Term Project assignments, each of which is a group assignment in Canvas.
    - Set up Canvas notification if you haven't done so.
2. Project Proposal: 1:30pm on 2/29
    - Submit the following to the Term Project assignment "Proposal" in Canvas.
        - CMPE138_TEAMn_PROPOSAL (.pdf, .doc, or .docx)
    - You have to re-submit the proposal until it is approved by the instructor, and must get the final approval no later than one week from the deadline. *Each resubmission must high-light the portions you modified.*
    - The proposal needs to be approved by the instructor in order to proceed.
3. ER or EER Diagram: 1:30pm on 3/9

- Submit the ER diagram to the Term Project assignment "ER Diagram" in Canvas.
    - CMPE138_TEAMn_ERD (.pdf, .doc, or .docx)
- Upon rejection, you must re-submit the ER Diagram within one week until it is approved by the instructor. *Each resubmission must high-light and indicate the portions you modified either within ERD or by using comment on Canvas.*
- ER/EER Diagram needs to be approved by the instructor in order to proceed.

4. Final Report, source code, and presentation slides: 1:30pm on 5/9
    - Submit the following to the Term Project assignment "Final Report" in Canvas.
        - CMPE138_TEAMn_FINAL_REPORT (.pdf, .doc, or .docx)
    - Zip the entire source code (all DB/app codes, SQL scripts, sample data, etc) and submit the following to the Term Project assignment "Source Code" in Canvas.
        - CMPE138_TEAMn_SOURCES.zip
    - Submit the following to the Term Project assignment "Presentation Slides" in Canvas.
        - CMPE138_TEAMn_SLIDES" (.pdf, .ppt, or .pptx)

Note

Each project must be original – *the project should be created by the team from scratch specifically for this course **ONLY***. Any copying/cheating activity is not allowed. Please refer to the section "POLICY ON CHEATING" in syllabus.

Failing to turn in project related assignments (report, slides, etc) would result in penalty for the entire team.

If a team member does not participate in any team activities, other team members should notify the instructor immediately without delay, and this team member will receive 0 point for the entire project.

Each project-related assignment is a group assignment, though project grade for each member is individual.

Project Details

The focus should be on the relational database side. After all, this is a database course.

Database

Design the database based on your choice of relational database engine. You should include E/R diagram, tables, views, queries, DML, stored procedures, sample data, etc.

You should describe the purpose of each table/view and the meaning of each column with each table. You also should describe purposes of each query, stored procedure, etc.

The operational side of the database, done by queries/DML/stored procedures, should include

- administrative portion: by the database application on behalf of the site owner to add/update/delete records, load sample data, etc.
- end-user portion: invoked by the database application on behalf of end-user to interact with the database server.

You also need to load sample data into your database. The loading can be done by INSERT or stored procedure.

The database script (source code) should include creating various database objects, as well as loading sample data. (Creating database itself is optional.)

Database application

The database application provides user interface to access the database for administrative side and end user side. The application can be as simple as a command line application or a GUI interface (Web-based or not). It can be based on any languages (e.g., Java, C#, C++, Python, Perl, PHP, etc) and/or any technologies (JDBC, ODBC, etc) and/or any frameworks.

Based on the user of the database application, the application (or database) should selectively enable certain functionalities including

- administrative portion: for the site owner to add/update/delete records, load sample data, etc.
- end-user portion: for the end-user to interact with the database server.

If the application is multi-threaded, explain why, what, and how.

The DB application should have logging facility to record various operations (request, response, error, etc) into files.

Project Proposal

In less than 2 pages, describe the problem you try to resolve and propose a unique project topic – no duplicates. The proposal must include (but not limited to) the following sections:

- Team members
- Project title
- Miniworld (high level) description, the purpose of this DB, and the intended users
- Actors within the miniworld, and roles played by each actor (i.e., which actors can do what)
- Include a few scenarios to explain exactly how each actor interacts with one another
- *Specify the planned functionalities and operations for each actor*. Please be as specific as possible.

The instructor may reject the idea if it is too broad or too narrow. Upon rejection, you have to re-submit your proposal ASAP until it is approved, and you must get the final approval within one week from the deadline.
Need to be approved by the instructor in order to proceed.

ER or EER Diagram

Design the preliminary ER or EER diagram of your project by utilizing any existing tools. Hand-drawing is not acceptable.
Need to be approved by the instructor in order to proceed.

Final Report

The final report should include

- Team #
- Names and IDs of team members
- The choice of database project
- The choice of database engine, DB application technologies, frameworks, languages, etc.
- Final list of functionalities/operations
    - If different from the planned list, status of each planned one
    - Any missing functionalities and the status of each
- Final major areas/components/tasks (including completion date) *for each team member*
- Final design of database portion
    - ER (or EER) diagram
    - Table or view design

    - Functional dependencies and normalization
    - Denormalization, if any (which ones and why)
    - The normal form (3NF, BCNF, etc) of tables/relationship. Justify the reasons if any of them is below 3NF
    - Specification of each DB object (table, column, view, stored procedure, trigger, etc) and its meaning/purpose
    - Any specific functionality involving modification of more than one table and if any transaction is used to implement such functionality
    - Any implicit/explicit database transactions initiated from DB server side
    - Any additional DB objects/concepts utilized (view, stored procedure, trigger, index, isolation, CC, etc.)
    - Source script (create DB objects, sample data, etc) – separate file
    - Sample execution of query/sp (screen shots, etc.) for important operations
- Final design of DB apps portion
    - Technology, framework, language, etc
    - DB access technology
    - Any specific functionality involving modification of more than one table and if any transaction is used to implement such functionality
    - Any implicit or explicit database transactions initiated from DB apps side
    - Source code – separate files
    - Sample execution (screen shots, etc.)
    - Sample application log file (not database transaction log file) – separate file
- Final architecture, components, sequence or flow of data
- Any major design decisions, trade-offs (and why)
- Any major modifications from the proposal, ERD, EERD and why
- Any unique designs you are proud of
- Final major milestones and task lists: by which members, when each was done, etc (please be specific)
- Test plan execution
- Any potential improvements
- Project postmortem (such as issues uncovered, how to improve XYZ, etc.)

Final Presentation and Demo

You should have slides (e.g., Power Point, PDF, etc) to highlight your final report. Each team member should present his/her own areas/tasks. Keep in mind the focus should be on database server side.

After presentation, you should have *live demo* of your database and applications. Show your DB objects. Use DB application to run through important functionalities/operations (both administrative side and end user side).

The length of presentation/demo per team will be announced later.

RDBMS

You can choose any relational database system, such as

- MySql, GPL version
    - Server: http://dev.mysql.com/downloads/
    - Connectors (e.g., JDBC, etc): http://dev.mysql.com/downloads/connector/
    - Workbench: http://dev.mysql.com/downloads/workbench/

- Microsoft SQL Server Express with tools: https://www.microsoft.com/en-us/server-cloud/Products/sql-server-editions/sql-server-express.aspx
- Oracle Database Express: http://www.oracle.com/technetwork/products/express-edition/downloads/index.html
- PostgreSQL: http://www.postgresql.org/download/
- Etc.

(All of the above are free.)

Project Ideas

These are just some examples. You are highly encouraged to create your own topic.

- Ride share (Uber-like)
- Real Estate Business
    - Search properties
    - Renovate the properties (items to be fixed, contractors used, estimation, etc)
    - List the properties
        - Recent/similar sales in nearby areas
        - Features of the property (# of bedrooms, # of bathrooms, fireplace, pool, etc)
    - Open House, submit offers, accept offers
- Online Education
- Stock Exchange
    - Stock quote, event, news, e.g., http://www.nasdaq.com/
- Stock Brokerage Account
    - Market update, customers, place order/option/short, market update, e.g., www.schwab.com

The followings are a few sample projects from the past -- You cannot choose any of these topics nor anything close/related to these topics:

- Library system
- Online shopping/mall/clothing/car/etc
- Movie/book/video/music/DVD/CD/tool/etc rental/selling/exchange/store
- University/school system - student/grade/class/department/professor/etc
- Hospital system – doctor/patient/appointment/room/nurse/etc
- Restaurant order/management
- Job search
- Flight routing and reservation
- Event management (ticketmaster-like, calendar, etc)
- Professional Network (LinkedIn-like)
- Social network (Facebook-like)
- Apartment/house leasing/rental (landlords/tenants/etc)