

# Illinois Institute of Technology

## CS425: Database Design and Applications

Spring 2004

### Group Project

Date: April 1<sup>st</sup>

Due Date: Demo in the last 2 weeks of semester (will be scheduled per group!)

---

You are volunteered to create a database application for a Court. The assumption is that this database is a centralized and a single user database. You are to design and implement this database application. The application needs to be able to support the storage, update, and delete, and query of court information, as described below. Thus, you are to build an application that provides a friendly user interface (this means that the user does not write any sql query for insert, update, delete and query) for the user to insert, manipulate and query the data. The database management system used for this application is Oracle DBMS. The system must support the business requirements described below:

#### Case information:

Each case of the court gets a case number that uniquely identifies that case. Each case belongs to a category that is decided by the court clerk to be assigned to. These case categories are to be maintained by the court. There are different case statuses that court needs to maintain. Each case at a given time has only one status. Status can be such as: New, Open, Closed, etc. The system, however, should be able to keep track of different statuses of a case and knowing the date of each status of the case. A case cannot be at the same status more than once. Each case might have jury assigned to it. Each case has plaintiff(s) and defendant(s). Obviously one person cannot be both plaintiff and defendant of a case. However, the same person can be plaintiff or defendant of different cases. Each case has only one prosecutor but can have several defendant attorneys. The defendant attorneys of a case are related to the defendant(s). If a case has several defendants, each of the defendants can have its own attorney or attorneys or all can have the same attorney(s). Each attorney can be the attorney of many cases at the same time. This is also valid for a prosecutor, who can work as the prosecutor of many cases at the same time. To each court case a fee as associated.

#### Attorney/Prosecutor information:

For each attorney and prosecutor the ssn, name, and phone are stored.

#### Plaintiff/defendant Information:

For each plaintiff and defendant the ssn, name, address, and one phone number, current employer name and position are stored. As soon as a person is a plaintiff for 3<sup>rd</sup> time, then the personal information and the case numbers and case date of that plaintiff for all previous 2 cases must be printed on the screen.

#### Court Data:

The court contains the data on its court employees. This includes the personal information of the employees and their type of role and responsibility in the court. Court maintains list of pre-defined of jobs/roles, along with the description of each role and salary range, minimum salary and maximum salary. For each court clerk, one of the roles/jobs in the court, all the cases that the court clerk has handled should be stored. Same the cases that each judge has handled are kept.

**Besides the above requirements of the system, which must be supported by your implementation, the following queries are to be answered:**

Given a person's ssn, find all cases that the person has been a plaintiff or a defendant. Show some of the case data, such as case number, case subject, and date.

Show all cases that are opened for 1 year or more.

Which attorney has handled most number of cases? Which attorney the least?

Show case categories that have had more than a specified number (user input) cases in a user given period of time. (example: if user wants to see in last 2 years how many cases have been in civil category – note: year and category are user input).

For a given case number, show the date of all statuses of that case along with date of the status.

Data should be easily inserted in all database tables, updated in all tables and deleted, considering referential integrity. All the data in the database has to be easily queried.

**Your database application MUST be in Third Normal Form (3NF). Note that any database transaction should support data integrity and consistency in your database.**

**Project Demo & Deliverables:**

❖ Each group will give a demo of a working project for the description given above. In the course of your demo it is your responsibility to have necessary data available in your database to be able in the allocated time to give a perfect demo to show the functionality of your application. Make sure that you have this data in your system and an organized and systematic test plan that you can demonstrate that your database application is functional and supports all the requirements (REQUIRED). You will also be asked to enter new data and perform new transactions. The first thing you have to provide before your actual demo is your documentation. You need to hand in your documentation (hard copy) at the beginning of the demo to be able to start your demo.

❖ Database Design Document (hard copy): Document your database design and all assumptions made in your design. Your database tables MUST be at least in 3<sup>rd</sup> normal Form (3NF). List all the schemas and underline the primary keys on each schema. Underneath of each schema specify all functional dependencies, and all candidate keys. All schemas must be fitted in one page for better readability.

❖ A Test document: to show how you have tested all the requirements of the system (hard copy). Via your test document you need to show that you have tested successfully each requirement and query. This is a list of all requirements your application has to satisfy (first column), your test path/steps (second column), the expected result (third column), the generated result (fourth column), and the last column specifies if the test was successful or not. Make sure that the result is reported correctly.

Each group is required to give ONE project demo. All the members of the group must attend in that demo. – The date and time will be scheduled for each group in the last two weeks of the semester. All Demos are held in the Main Campus.

**Late projects will not be accepted and will receive a grade of zero.**