Clinical Information Systems (BIOM9450)
Major Project: Web-based Falls Management System

Due: Monday 29th October 11:59 pm – Submitted online

Tasks

In this project you are required to design web-based software for managing research data related to falls detection and falls prevention. The underlying database, as well as all web pages in your application should be ***your own*** design. Essential user information must be entered and validated and the referential integrity of the data maintained. You will not be given in-depth details on how the tasks are to be achieved: that is up to your ingenuity, creativity and perseverance. Make sure you ***read*** the criteria given below ***very carefully*** as marks are allocated for each one of them.

As a starting point, a database 'BIOM9450Falls.mdb' has been placed on Moodle. This database contains two tables. The first table 'Subject_and_FallsRiskData' contains records from 27 unique subjects. It includes a mix of demographic subject data (name, birthdate, sex) as well as falls risk data. For the falls risk data there is a test date 'TestDate', and a 'TrueFallsRisk' that is a number from -5 to 5 with -5 indicating a very high chance of falling and 5 indicating a very low risk of falling. There is also a 'Description' that associates the test data with a particular clinical trial. It is possible for a subject to have multiple falls risk assessments on different dates within one clinical trial (as well as being recruited into more than one clinical trial).

The second table 'TriaxData' has a link to the first table by way of a 'Subject_ID'. It contains a 'TestDate' field as well as 'Sig_Type' and 'Data' fields. It is possible for a subject to have multiple tests on either the same or on different dates. The 'Data' field has two different formats depending on the 'Sig_Type'. If the type is that of 'Triax??' i.e. 'TriaxMP', 'TriaxAP' or 'TriaxVT', then it relates to data from a triaxial accelerometer that was placed on the waist of a subject and used to record body accelerations as a subject performed a Sit-to-Stand 5 (STS5) test, whereby they were asked to rise and sit in a chair five times as rapidly as possible. The accelerometer signals could be recorded in the Anterior-Posterior 'TriaxAP' axis, the Medio-Lateral 'TriaxML' axis, or the Vertical 'TriaxVT' axis. Typically the table will have all three signals as separate rows in the table but this is not always the case. The triax data is encoded as a string comprising a series of offset binary numbers. In offset binary, 0 corresponds to negative full scale (in this case an acceleration of -1.5 G (m/s/s)), 2048 to zero acceleration, and 4095 to positive full scale (in this case an acceleration of +1.5 G (m/s/s)). All the data are acquired with a sampling rate of 40 Hz.

If the signal type is that of 'Markers' then the 'Data' contains a string encoding a series of numbers (in milliseconds) representing where movement transitions were identified (i.e. the transition from sit-to-stand and vice-versa).

Note that the database ***is not normalised and you are expected to create your own normalised set of tables*** with appropriate primary and foreign keys, relationships and indexes as part of the project requirements. The database should be placed in your root directory and be named `project.mdb` in order for it to be connected to the ODBC alias that has been created for you in PHP.

You **must also include** a 'Practitioner' table. Each Subject is to be associated with one or more Practitioners. As part of this design there should be user name and a password for maintaining security. Confidentiality of data is to be maintained via a secure login feature whereby a username and password are entered. The login must be adequately secure such that if a person was to know the URL of a page (not necessarily the initial logon page) then they would be denied access unless they had gone through appropriate authentication. There should be two levels of access. A login entry should be possible for a 'researcher/practitioner' so that they can view data and generate reports on ***their assigned*** subjects in a particular trial. Note that it is possible for one subject to have multiple Practitioners associated with them. An administrator login should also be possible to have

universal access to data as well as to assign Subjects to a particular Practitioner and perform other database administration functions.

More specifically, practitioners should be able to insert new Subject data (not triax /marker data though) into the database, as well as search, browse and edit data for only those Subjects that are part of their research trials. When viewing data, Practitioners should be able to view a list of Subjects and be able to 'drill down' using hyperlinks to a ***detailed display of that subject's falls data if it exists***. These data should be graphed using TeeChart and all Triax signals for a particular test on a particular day should be plotted along with annotation of the plot with the time Markers (if they exist).

In contrast, Administrators should be able to search and browse the data for all Subjects in the database. In addition, they must be able to add new Practitioners into the database or update/change details for existing Practitioners, as well as be able to assign an existing Subject to a new Practitioner.

One additional report that the Administrator (only) can link to is a data trend graph. This graph should plot the 'TrueFallsRisk' against Subject age (for all subjects). If more than one falls risk data record exists for a particular Subject ***then an average*** of the subject's falls risk should be used.

Server side validation of data should be performed using PHP and database integrity enforced using appropriate database rules. You may choose to also perform some client side data validation using JavaScript in order to improve the user experience.

Report

You are expected to hand in a complete report detailing the tasks undertaken and the approaches adopted, including database design and user interface descriptions. This should include a brief "user's manual". It is not necessary to print your PHP/Javascript code (unless there are particular code snippets that you want to make mention of) as we will be accessing this from the server.

Ideally, the system should be complete and bug free. If sections of your program do not work as expected, you should explain where and why the bug(s) occurred, and what attempts you have made to track the source of the error.

Please make sure as part of the user manual you provide sample login usernames and passwords so we can easily check you design. Also, if you are coding the project offsite ensure it works fully on the computers in the laboratory!

Marking Schema

The broad marking schema will comprise the following sections (marks per section given in parentheses):

- Database design (20): check that required data is stored, appropriate normalisation, keys, relationships and indexes are in place, design is justified in report, adherence to project requirements.
- Web functionality (30): login/logout functionality, patient, practitioner and administrator portals are fully functional, charts and statistical analysis are functional, appropriate navigation, security.
- Error checking and validation (15): applied to all data entry, SQL errors handled for incorrect data entry, appropriate error messages, combination of Javascript, PHP and database rules.
- Web interface, design (aesthetics, ease of use, inventiveness), presentation (20)
- Report (15)

NB: A non-plagiarism declaration form must be attached to your hand-in report.