March 31, 2013

Spring 2013
Final Project

Develop a NetBeans project comprised of Graphical User Interface (GUI) components from the *Swing* and *AWT* APIs using **MP2** as the back-end code base.

Project Requirements

Application data model:

- Use a *MySQL* database called *itm411DB* for all of the record data.
- Create the single database table with fields from the *CSV* file. The DB does not have to be created dynamically, but can be using *JDBC* if you wish.
- Provide the capability of re-initializing the table data if the user wishes to start over.

Application functional model:

- Manipulate *DayLightRecord* objects from the GUI using database CRUD (Create, Retrieve, Update and Delete) JDBC operations.
- Provide components for user interactions for all of the functions required in **MP2** (e.g. data analytic queries, etc.).
- Provide GUI components to trigger serialization and deserialization.
- Display GUI results in either tabular, text field or text area formats based on the user-specified output type. For example, if the user wishes to view all records, then a tabular format would be rendered. Or, if the user wished to view or manipulate a single record, then a *TextField* format would be used to render the data of a single record.
- Display GUI status of application state and user interactions (e.g. successfully loaded DB). Hint: Use a *JLabel* to update GUI status by rewriting the contents.
- Consider usability in your GUI components layout (i.e. the easier it is to use, the higher the grade). Ensure that the UI is intuitive to use as a 'newbie' user. Use context areas (functional decomposition) for similar functionality. So if the user wishes to execute any of the data analytics, the GUI components to execute this functionality should be grouped together and easy to understand at a common location on the GUI. Consider using tabbed frames, internal frames, pop-up sub-windows, etc.
- At a high-level, the project does serialization-deserialization and data analytics with the addition of typical *JDBC* database functions.

Application documentation:

- Project cover page, abstract, diagrams, code structure, ***README*** details.
- Project specifications for each requirement (data and application models).
- Project summary, project insights and complete screen captures.

Submit a compressed file called ***<name>_FP.zip*** (zip only) of all project code and documentation by 05/07/13, 23:59/CDT. Late mini-projects will lose points. (**100 points**)