Creation of a Risk Data Application

Client: Anonymous
Business Size: Corporation
Industry: Insurance and Risk Management
Country: UK
Technology: SQL Server, MySQL, Pentaho Spoon

Objective: Reporting Dashboard Enhancement

The Brief
This assignment was to create and update reports for the client's internal reporting dashboard.

Background
The client is a global insurance brokerage, risk management, and HR & benefits consulting company with an extensive range of reports for both internal and broker use.

Methodology
The project utilized the Agile framework with two-week sprints, daily stand-up meetings, sprint planning, backlog refinement, sprint review, and retrospective sessions. The reporting tool used was Pentaho Report Designer, which included Spoon, the Data Integration Tool. This allowed us to write Kettles (ETL processes) to transfer data from the live system to the reporting database, and also manipulate data for use in the report designer. The back-end database was MySQL.

We worked as a team of 4 (2 from OCS and 2 internal report developers), along with business analysts to provide requirements from the actuaries, introducing peer code reviews and standardized style for SQL to aid in long-term development and maintenance.

Challenges
Lack of experience in the Pentaho tools made some development aspects difficult as nobody in the team had any prior knowledge of this tool suite. With little documentation available the team needed to spend a large amount of time investigating how to solve problems.

Also, the Pentaho tools were somewhat unreliable, crashing regularly but for seemingly different reasons for each developer.

All the development team had worked in recent versions of SQL Server, so working with a relatively old version of MySQL proved challenging as a lot of recent SQL functionality was not implemented. As this was a new team, and lacked any prior business knowledge, we had regular discussions to clarify where to pull data from or where complicated business logic had to be applied to obtain the correct values that could be seen in the front-end screens.

Consultant Contribution

The role involved extracting the relevant data from the main system database into the reporting data warehouse and then displaying this on a report front-end.

Determining where the data was held in the main database involved locating the field in the front-end, and then going through xml tags in the html pages to locate the table and field names. Sometimes these could not be found, or were wrongly tagged, so we needed to contact the main system developers to get the correct tags for the field.

Once all the data was located a flat table structure was created as the reports were invariably lists for the actuaries to export to Excel, with the relevant ETL process created in Pentaho Spoon to create the correct data structure.

Git was used for source control, with a branch for each ticket with changes merged before compiling and testing. The code review process was triggered by configuration of the pull request mechanism. Azure DevOps build and release pipeline facilities were utilized to manage releases.

Lessons learned

As a team of developers, we collaborated extremely well on this project. Without this we would have struggled to deliver good results. It reinforced just how important it is to be able to talk through problems with peers. Another lesson learned, or reinforced, is that it is so important not to use software that is out of date, or is not

supported, as this can cause many issues. This should be considered during project planning stages.