Capstone Project – spring 2011

Sponsored by Tietronix Software Inc. (http://www.tietronix.com**).**

Title: Tietronix Metrics Collection and Analytics Tool

Background: Software Metrics are measurements of different aspects of a software project that help us determine whether or not we are progressing toward the goal of that project. They are used extensively as management tools to provide some calculated, observable basis for making decisions. Some common metrics for projects include schedule deviation, remaining budget and expenditure rate, presence or absence of specific types of problems, and milestones achieved. An important part of software engineering is to provide a metrics program that supports the development of quality software. Capturing and storing metrics in a repository allows an organization to improve its planning estimates, it's benchmarking, and its processes based on objectives measurements.

Description: The intent of this project is to develop a web based tool that allows the end users to define, collect, store and report on a set of custom metrics.

Requirements:

- ***User Requirements***
  - See appendix A
- ***System Requirements***
  - The system will be JEE (Java enterprise Edition) web based application
  - The system will be designed to be compatible with leading database management systems (MS SQL, Oracle, and MySQL). For purpose of this capstone the team will use MySQL 5.x as the backend database.
  - The web application will be compliant to at least XHTML Transitional DTD

Project Deliverables:

The project team will provide the following deliverable in three phases:

- ***Phase 1: Design***
  The project team will modify and extend an existing design document to elaborate on how the system will be built
- ***Phase 2: Build Beta***
  The project team will implement the design
- ***Phase 3: Test and Deliver GA***
  The project team will test the beta version, fix any issues and deliver the system along with source code, test scripts, deployment instructions and final report.

Required Skills

A good understanding of following technologies/concepts

- Java
- Web technologies (XHTML, CSS)
- Object Oriented Design and Programming
- Databases

Mentor Information

Abbasi Dhilawala
Sr. Software Engineer, Tietronix Software Inc
abbasi.dhilawala@tietronix.com
281-404-7243

Appendix A: User Requirements

1. System shall support multiple users
    a. Each user shall have a userid, user name, first name, last name and email address
    b. Userid shall be system generated unique identifier
    c. User name shall be used for login and shall be user specifiable
    d. User name shall be unique

2. System shall support multiple projects
    a. Each project shall have a unique name
    b. Each project shall have one or more project member
    c. Each project shall have one or more project administrator
        i. Project administrator shall be a project member
        ii. Project administrator shall be able to assign users as members for their project
        iii. Project administrator shall be able to assign other members as project administrator of their project

3. System shall support one or more system administrator
    a. System administrator shall be able to create projects
    b. System administrator shall be able to create users
    c. System administrator shall be able to edit users
    d. System administrator shall be able to delete users
    e. System administrator shall be able to assign project members and administrator
    f. System administrator shall be able to define global metrics that are applicable to each project
    g. System administrator shall be able to make other user system administrator

4. Project administrator shall be able to define metrics for their project at any time
    a. Metrics shall have a name
    b. Metrics name shall be unique for that scope (project or global)
    c. Metrics shall have a type (primitive or derived)
    d. Primitive metrics shall have a data type (number, date, text)
    e. Primitive metrics shall have a collection method (automatic or manual)
    f. For derived metrics the system shall support the following functions
        i. Add: Allowing for addition of 2 or more primitive or derived metrics of data type number or date
        ii. Subtract: Allow subtracting 2 or more primitive or derived metrics with data type of number or date (for date data type the result is number of days)
        iii. Product: Allow multiplication of 2 or more primitive or derived metrics with the data type of number
        iv. Divide: Allow dividing 2 or more primitive or derived metrics with data type of number
    g. The data type of derived metric shall be number
    h. Derived metrics shall be chainable e.g. (M1-(M2+M3+((M4*M5)/M6)))

5. Project administrator shall be able to define one or more metric collection instances
    a. Metric collection instance shall have a name

    b. Metric collection instance shall have a date

6. For each metric collection instance the project administrator shall be able to specify values for the global and project defined primitive metrics.
    a. The system shall validate the metrics value to make sure it matches the data type of the metrics
    b. Metric value shall not be required

7. The system shall support metric reports
    a. Project members shall be able to run reports on projects they are part of
    b. System administrators shall be able to run reports on all projects in the system
    c. System shall support reporting raw values for all metrics for a project across many collection instances (see example 1)
    d. System shall support comparison reports for metrics with data type number and date for a project across many collection instances (see example 2)
    e. System shall support both textual and graphical formats for comparison reports
    f. All reports shall be print friendly

Example 1: Raw value report for a single project

My Project has 3 metrics. Two primitive: Lines of Code (number) and Defects (number). One derived: Defects / Line of Code (number). Has 5 collection cycles

Project Name: My Project

	Lines of Code	Defects	Defects / Line of Code
Collection 1	1000	10	0.01
Collection 2	300	2	0.01
Collection 3	700	5	0.01
Collection 4	200	3	0.02
Collection 5	1500	15	0.01

Second Project has 3 metrics. Two primitive: Planned Milestone date (date) and Actual Milestone date (date). One derived: Delay in days (number). Has 4 collection cycles

Project Name: Second Project

	Planned Milestone Date	Actual Milestone Date	Delay in days
First Phase	1/1/2009	1/6/2009	-5
Second Phase	2/1/2009	2/1/2009	0
Third Phase	3/1/2009	3/3/2009	-2
Fourth Phase	4/1/2009	4/21/2009	-20

Example 2: Textual and Graphical Comparison Report
Comparison of two metrics Lines of Code (number) and Defects (number) in textual format

Project Name: My Project

	Lines of Code	Defects
Collection 1	1000	10
Collection 2	300	2
Collection 3	700	5
Collection 4	200	3
Collection 5	1500	15

Comparison of two metrics Lines of Code (number) and Defects (number) in graphical format