

TestView

Estudo multidisciplinar sobre um projeto de automação industrial

Rodolfo Boffo de Souza
Orientador: Prof. Dr. Marco Dimas Gubitoso
Instituto de Matemática e Estatística
Universidade de São Paulo

Introdução

Durante a graduação, com frequência desenvolvemos pequenos projetos com o objetivo de conhecer novas tecnologias, estruturas de dados e conceitos computacionais. Porém, a maioria desses projetos possui um domínio de aplicação reduzido, já que deve abranger não mais que um ou dois semestres do ano letivo.

Como consequência, temos pouco contato com um projeto de abrangência multidisciplinar, que seja capaz de aproximar o estudante a um ambiente de desenvolvimento mais próximo do encontrado no mercado de trabalho. Neste trabalho abordaremos o desenvolvimento de um sistema de automação industrial, projeto originado em um estágio supervisionado durante a graduação, e que foi escolhido como tema desta monografia por abranger diversos tópicos de computação.

Motivação

Este projeto teve início com o desafio de se criar e comercializar um software capaz de obter dados de uma máquina universal de ensaios, analisá-los e gerar relatórios para ensaios de tração em corpos de prova de materiais metálicos.

Com o sucesso do desenvolvimento do primeiro produto, houve a posterior demanda de desenvolvimento de um novo software com o objetivo de coletar dados para ensaios de compressão em corpos de prova de concreto.

Os dois produtos teriam características idênticas com relação à coleta de dados dos equipamentos; por outro lado, as rotinas de ensaios e análise de dados seriam diferentes entre os dois produtos.

Por este motivo, decidimos reaproveitar o código pré-existente, tomando a decisão de criar uma bifurcação no projeto. Essa decisão diminuiu o tempo de desenvolvimento do novo produto, porém aumentou o custo de manutenção do código comum, já que qualquer alteração nos métodos de coleta de dados deveria ser realizada no código dos dois produtos distintos.

Isso trouxe a necessidade de refatoração do código, lançando mão de melhores práticas de desenvolvimento, reutilização de código e padrões de projetos.

Objetivos

O objetivo deste trabalho consiste em desenvolver um arcabouço (framework) de ensaios, capaz de centralizar e disponibilizar funções relacionadas à coleta de dados dos equipamentos de ensaios, calibração de instrumentos de aferição, criação e impressão de relatórios, e funções de armazenamento e recuperação de ensaios no banco de dados ou sistema de arquivos.

Framework

Segundo [1], podemos definir um arcabouço, também conhecido como framework, como um conjunto de classes que cooperam entre si com a finalidade de formar um design reutilizável para um escopo específico de software.

Arcabouços são usados com o objetivo de facilitar o desenvolvimento de aplicações, pois seu design predefine a estrutura global da aplicação, bem como as responsabilidades das classes, interação entre objetos e classes, e o fluxo de controle, deixando para o desenvolvedor apenas o trabalho relacionado às especificidades de sua aplicação. Dessa forma, pode-se dizer que arcabouços enfatizam a reutilização de design em comparação com a reutilização de código.

O arcabouço TestView é organizado em classes como mostrado na Figura 1. A classe GerenciadorDeEnsaio centraliza as funções necessárias para a realização dos ensaios, como por exemplo o acesso aos condicionadores de sinal, responsáveis pela coleta de dados dos equipamentos.

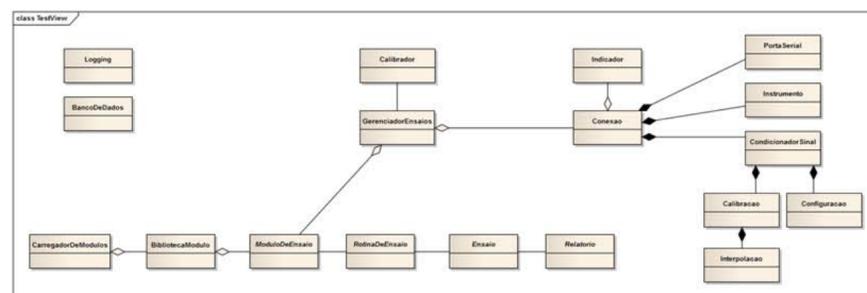


Figura 1: Diagrama de classes do framework TestView

Extensões

Também conhecidas como *plug-in* ou *add-in*, as extensões são componentes de software que adicionam habilidades específicas à uma aplicação maior. Esses componentes possuem uma interface comum que possibilita à aplicação detectar e registrar a extensão em questão. Por sua vez, extensões usualmente não podem funcionar por si mesmas e portanto necessitam de uma interface de serviços disponibilizada pela aplicação.

Em nosso arcabouço, uma nova extensão pode ser facilmente desenvolvida, bastando para isso apenas a criação de um novo projeto onde as classes *ModuloDeEnsaio*, *RotinaDeEnsaio*, *Ensaio* e *Relatorio* devem ser extendidas. A compilação desse novo projeto resultará em uma biblioteca ou *plug-in*, capaz de ser detectado pelo arcabouço, trazendo para a aplicação todos os recursos necessários à realização, armazenamento e recuperação de um novo tipo de ensaio.

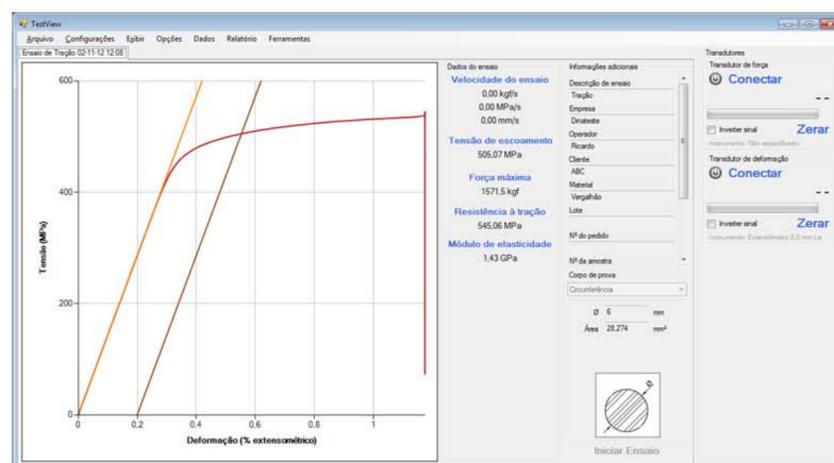


Figura 2: Captura de tela do framework em funcionamento, exibindo um ensaio de tração em vergalhão de aço.

Resultados

De acordo com os objetivos almejados, temos como resultado deste trabalho um arcabouço de ensaios, o qual disponibiliza funções de coleta de dados, calibração de instrumentos, entre outras características que viabilizam o desenvolvimento de extensões, chamadas aqui de módulos de ensaio, capazes de acrescentar recursos ao arcabouço, tornando possível a realização de ensaios específicos, como por exemplo um ensaio de tração em materiais metálicos, exibido na Figura 2.

Referências

[1] E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional; 1 edition, November 10th, 1994