

1 Introdução

Uma das principais preocupações de área de engenharia de software diz respeito à reutilização [1]. Isso porque a reutilização no contexto de desenvolvimeto de software pode contribuir significativamente para aumentar a qualidade do software assim como reduzir o esforço no seu desenvolvimento. Uma das técnicas de maior impacto de reuso de projeto e código em larga escala que foi proposta nos últimos anos é o desenvolvimeto e codificação de Frameworks Orientados a Objetos (OO) [3, 4]. Frameworks OO possibilitam o desenvolvimento com reuso, de família de aplicações [8], criadas a partir de uma estrutura que captura os conceitos mais gerais das aplicações. Eles permitem gerar famílias de aplicações de um mesmo domínio, através da definição da implementação de um núcleo que lida com os requisitos comuns dessas aplicações, e definem um conjunto de pontos de extensão na forma de classes abstratas ou interfaces, que são estendidas para concretizar a funcionalidade específica de aplicações concretas da família.

Apesar dos benefícios trazidos pela tecnologia de frameworks OO, diversos problemas foram identificados a partir da experiência do uso de tal tecnologia na construção de sistemas OO, tais como: (i) dificuldade para configuração e customização do framework para atender diferentes cenários de reuso; e (ii) dificuldade para compor o framework de forma transversal a outros frameworks existentes. A partir da identificação de tais problemas, pesquisadores têm proposto a implementação de frameworks usando de forma complementar técnicas OO e técnicas de Desenvolvimento de Software Orientado a Aspectos.

Desenvolvimento de software orientado a aspectos (DSOA) [5, 6] é uma abordagem que busca a modularização dos chamados interesses transversais, os quais os paradigmas atuais (ex: orientação a objetos) não são capazes de capturar explicitamente. Interesses transversais são aqueles que entrecortam¹ diversos módulos em um dado sistema de software. DSOA propõe o uso de técnicas e

linguagens de programação que permitam a especificação de software complexo com uma separação clara entre a funcionalidade básica do sistema e seus interesses transversais. A abstração usada para modularizar os interesses transversais é chamada de Aspecto.

A utilização de forma complementar, dos paradigmas de orientação a aspectos (OA) e orientação a objetos (OO), tem demonstrado benefícios para o desenvolvimento de softwares. A modularização de determinadas propriedades sistêmicas ou requisitos não funcionais transversais, tais como, rastreamento, segurança, demarcação de transações, e auditoria, são notadamente beneficiados com o uso de técnicas de programação orientada a aspectos. De forma geral, a utilização do paradigma de orientação a aspectos, visa melhorar a diminuição do acoplamento e aumentar a modularidade entre os componentes de um sistema de software.

1.1. Definição do Problema

Com o crescimento dos sistemas de informação baseados na internet, empresas que mantêm seus negócios eletrônicos baseados em *sites* web necessitam acompanhar o andamento das atividades dos seus negócios, através do monitoramento das requisições web que chegam aos seus sistemas.

Atualmente, a maioria do monitoramento que ocorre em sistemas de informação web, objetiva a implementação ou suporte a um ou mais de seus requisitos não-funcionais, tais como:

- Disponibilidade: realiza o monitoramento de requisições feitas para o servidor web e verifica se ocorre uma resposta do servidor. Esse tipo de serviço é usado tipicamente para verificar se um site web está fora do ar;
- Desempenho/Carga: tem como objetivo monitorar a capacidade e carga do sistema web, para analisar o quão escalável é o mesmo para aceitar outros usuários simultaneamente, além de questões como tempo de resposta de uma requisição do usuário;

¹ Do inglês: *Crosscutting*

- **Segurança:** visa monitorar ataques ou acessos indevidos ao sistema web.

Outra forma de monitoramento de sistemas web consiste em utilizar a mineração de dados em arquivos de log de dados² dos servidores web, tais como, o Servidor Web Apache³. Dessa forma, é possível verificar e obter informações de acesso ao sistema, tais como: número de acessos ao site web, quantidade de dados transferidos, etc.

Porém, as abordagens utilizadas para o monitoramento não tratam questões de mais alto nível, referentes ao monitoramento dos processos de negócio que ocorrem em sistemas web. O monitoramento das atividades executadas por usuários de um *site* web é de grande utilidade para auxiliar as organizações em suas tomadas de decisões, possibilitando o contínuo aumento na qualidade dos serviços oferecidos para seus usuários.

A funcionalidade de monitoramento de processos de negócio de sistemas web não faz, em geral, parte dos seus requisitos essenciais. Além disso, este monitoramento pode ser caracterizado como sendo transversal às demais funcionalidades definidas para o sistema web. Já que para ser implementado, necessita acompanhar e coletar informações a respeito das requisições web que são feitas ao sistema. Devido a tais características, a implementação do serviço de monitoramento de processos de negócio em um sistema web acaba inevitavelmente sendo invasivo, demandando a inserção de linhas de código em diversos trechos das classes principais do sistema.

1.2. Solução Proposta

Neste contexto, essa dissertação propõe a definição de um framework para monitorar os processos de negócio que ocorrem em sistemas de informação web. Os objetivos principais de tal framework são: (i) permitir a implementação da funcionalidade de monitoramento de processos de negócio, de forma independente do código que implementa a funcionalidade principal do sistema

² Arquivo que descreve o processo de registro de eventos relevantes em um sistema computacional

³ <http://www.apache.org>

web; e (ii) possibilitar a customização do framework para ser integrado com diferentes implementações de sistemas web.

O projeto e implementação do framework de monitoramento de processos de negócios explora a integração de técnicas de orientação a objetos (OO) e orientação a aspectos (OA) [2]. Padrões de projeto OO [33] são utilizados na implementação do núcleo do framework que oferece a funcionalidade básica para armazenamento e consulta aos processos de negócio capturados para um sistema web, assim como dos pontos flexíveis que permitem definir as estratégias de monitoramento, de processamento concorrente e de comunicação entre os monitores de sistemas web e o núcleo do framework. Técnicas de programação orientada a aspectos [2] são usadas com o intuito de monitorar classes específicas do sistema, para capturar informações relevantes sobre os processos de negócio em execução no sistema. O framework foi implementado usando as linguagens de programação Java [24] e AspectJ [12]. O uso de técnicas OA permite uma composição transversal entre: (i) a aplicação web cujos processos de negócio serão analisados; e (ii) o núcleo de nosso framework OA que implementará as funcionalidades de análise e caracterização dos processos de negócio ocorrendo na aplicação web. Para validar o projeto e implementação de nosso framework, ele foi instanciado para dois diferentes estudos de caso: estudos de caso: (1) uma aplicação web tradicional em camadas; e (2) uma aplicação web corporativa implementada com J2EE.

1.3. Organização do Texto

Esta dissertação está organizada da seguinte forma. No capítulo 2, serão apresentados alguns conceitos básicos relevantes para o entendimento deste trabalho, assim como alguns trabalhos de pesquisa da comunidade que combinam o uso de programação orientada a aspectos e frameworks. O capítulo 3 apresenta a contribuição principal dessa dissertação, um framework orientado a aspectos para monitoramento e análise de processos de negócio em sistemas de informação web, apresentando uma visão geral de sua arquitetura. No capítulo 4 detalhamos o framework proposto neste trabalho, abordando o projeto detalhado de cada um dos componentes envolvidos na sua implementação. Dois estudos de caso

instanciados do framework são apresentados no capítulo 5. Por fim, o capítulo 6 apresenta as conclusões e contribuições da dissertação, assim como uma lista de trabalhos futuros.