

**FELIPE RODRIGUES DO PRADO
JOÃO PAULO NAKAJIMA PEREIRA**

MODULARIZAÇÃO DE SOFTWARES

**UNIVERSIDADE DO VALE DO SAPUCAÍ
POUSO ALEGRE
2015**

**FELIPE RODRIGUES DO PRADO
JOÃO PAULO NAKAJIMA PEREIRA**

MODULARIZAÇÃO DE SOFTWARES

Pesquisa para desenvolvimento do projeto
apresentado à disciplina de TCC 1 do curso de
Sistemas de Informação como requisito parcial
para obtenção de créditos.

Orientador: Ms. Márcio Emílio Cruz Vono de
Azevedo.

**UNIVERSIDADE DO VALE DO SAPUCAÍ
POUSO ALEGRE
2015**

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
BSD	<i>Berkeley Software Distribution</i>
CSS	<i>Cascading Style Sheet</i>
EJB	<i>Enterprise JavaBeans</i>
HTML	<i>HiperText Markup Language</i>
ICC	<i>Inatel Competence Center</i>
IDE	<i>Integrated Development Environment</i>
IoT	<i>Internet of Things</i>
JDBC	<i>Java Database Connectivity</i>
JEE	<i>Java Platform, Enterprise Edition.</i>
JMS	<i>Java Message Service</i>
JPA	<i>Java Persistence API</i>
JVM	<i>Java Virtual Machine</i>
NTT	<i>Nippon Telegraph and Telephone</i>
OSGi	<i>Open Services Gateway initiative</i>
SENAC	Serviço Nacional de Aprendizagem Comercial
SQL	<i>Structured Query Language</i>
SRA	<i>Systems Research and Applications Corporation</i>
UNIFEI	Universidade Federal de Itajubá
UNIVAS	Universidade do Vale do Sapucaí
XHTML	<i>Extensible HyperText Markup Language</i>
XML	<i>Extensible Markup Language</i>
W3C	<i>World Wide Web Consortium</i>

SUMÁRIO

INTRODUÇÃO.....	7
2 OBJETIVOS.....	9
2.2 Objetivo geral.....	9
2.3 Objetivos específicos.....	9
REFERÊNCIAS.....	10

INTRODUÇÃO

O desenvolvimento de um *software* não envolve apenas métodos de programação. Para desenvolvê-lo é necessário preocupar-se com planejamento, engenharia, metodologia e tecnologia a ser utilizada. Esses fatores influenciam na manutenção, atualização e expansão do mesmo. Dessa forma, definir tais fatores é fundamental para que o *software* seja flexível a mudanças. Pensando dessa maneira, será demonstrado um modelo de desenvolvimento modular, utilizando a especificação OSGi.

OSGi, abreviação para *Open Services Gateway Initiative*, possibilita o desenvolvimento de *softwares* em módulos, disponibilizando o gerenciamento dos mesmos e facilidades na manutenção e expansão da aplicação.

Segundo Fernandes (2009), um sistema modular possui algumas propriedades. Deve ser autocontido, os módulos podem ser incluídos, retirados, instalados ou desinstalados. Outra propriedade é ter alta coesão. Um módulo deve cumprir apenas sua finalidade, ou seja, deve fazer somente as funções que lhe foram atribuídas. O baixo acoplamento é outra propriedade muito importante. Um módulo não precisa se preocupar com implementações de outros módulos que interagem com ele, além de permitir alterá-lo sem a necessidade de atualizar os outros.

Muitos *softwares* são desenvolvidos de maneira semelhante à modularização, divididos em partes, tendo cada parte uma responsabilidade. Porém não são realmente modularizados, ou seja, não atendem ao conceito de modularização como descrito acima. Dois exemplos de projetos de *softwares* que foram desenvolvidos utilizando a especificação OSGi e que atendem a definição de modularização, são, IDE¹ Eclipse, ferramenta de desenvolvimento de softwares e o GlassFish, servidor de aplicações JEE².

O uso da modularização, com certeza, traz grandes benefícios para o desenvolvimento e manutenção de um software. Poder parar parte de uma aplicação para fazer uma manutenção ou poder instalar novas funcionalidades, garantindo que todas as outras partes restantes continuem funcionando normalmente, seria uma característica notável da aplicação.

Fernandes (2009) forneceu uma visão geral sobre OSGI, mostrando seus benefícios e

1 IDE – Abreviação para *Integrated Development Environment*.

2 JEE – Abreviação para *Java Platform, Enterprise Edition*.

salientando a importância da plataforma Java possuir um melhor suporte à modularidade, até demonstrando com um exemplo simples as premissas e vantagens do OSGi³.

Mayworm (2010) demonstra a tecnologia OSGi no contexto de aplicações distribuídas, permitindo a disponibilização de seus serviços remotamente, integrando com diferentes *frameworks* de *middleware* para o desenvolvimento de aplicações empresariais.

Malcher (2008) apresenta um modelo de componentes adaptável para se usar em ambientes distribuídos. Também analisa alguns aspectos como modelo de distribuição, transparência, descoberta de novos módulos disponíveis, desempenho e *performance* dos mesmos. Afirma ainda que para se escolher entre um modelo de distribuição – *deployment* local ou execução remota – é necessário analisar o contexto e o objetivo da aplicação, pois cada um se adapta melhor a determinadas situações e ambientes de execução.

Portanto, após despertar um grande interesse pelo modelo de desenvolvimento modular, será realizado estudos sobre a especificação OSGi, e ainda, o desenvolvimento de uma aplicação modular que demonstre tal modelo de desenvolvimento.

3 OSGi – Abreviação para *Open Services Gateway initiative*.

2 OBJETIVOS

Os objetivos desta pesquisa serão demonstrados a seguir.

2.2 Objetivo geral

- Demonstrar o modelo de desenvolvimento modular utilizando a especificação OSGi para aplicações empresariais.

2.3 Objetivos específicos

Para atingir o objetivo geral os seguintes objetivos específicos é apresentado:

- Pesquisar as melhores práticas e *frameworks* que contribuem para a produtividade no desenvolvimento modularizado;
- Desenvolver uma aplicação que exemplifique o modelo de desenvolvimento modularizado através da especificação OSGi;
- Obter através dos resultados uma conclusão sobre as vantagens do modelo de desenvolvimento modular.

REFERÊNCIAS