

Modelagem do Subsistema de Administração Imobiliário Baseado em Linhas de Produto

Silvia Ribeiro Mantuani
UTFPR – PG
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa, Brasil
silviamantuani@gmail.com

Fernando Henrique Campos
UTFPR – SH
Rua Cerejeira, s/n – São Luiz
Santa Helena, Brasil
fhcampos@utfpr.edu.br

Vinícius Camargo Andrade
UTFPR – PG
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa, Brasil
vcandrade@utfpr.edu.br

RESUMO

O desenvolvimento baseado em linha de produto permite a identificação das similaridades e variabilidades entre determinados sistemas de maneira que se torne possível construir aplicações com uma arquitetura. Este trabalho propõe a modelagem de um subsistema de administração imobiliário baseado em Linha de Produto de Software (LPS). Neste contexto são apresentadas as vantagens e desvantagens de utilizar tal abordagem. Para modelar o sistema, realizou-se uma análise dos métodos baseados em LPS a fim de utilizar àquele que melhor se adequa ao contexto do presente trabalho. Os métodos analisados foram: *Product Line UML-Based Software Engineering*, *Family-Oriented Abstraction*, *Specification and Translation*, *Feature Oriented Domain Analysis* e *Delazeri e Wolf*, sendo este último o escolhido para a realização da modelagem. Com o modelo proposto, o desenvolvedor possui uma arquitetura flexível a qual pode ser reutilizada e/ou estendida para um novo sistema de mesmo domínio.

Palavras-chave

Linhas de Produto de Software; *UML Components*; Sistema Imobiliário.

ABSTRACT

The development based on product line allows the identification of similarities and variabilities among stated systems so that it becomes possible to build applications with an architecture. This paper proposes the modeling of a real estate management subsystem based on Software Product Line (SPL). In this context, are presented the advantages and disadvantages of the use this approach. To model the system, was performed an analysis of the methods based on SPL in order to use the most suitable to the context of the present paper. The methods analyzed were: *Product Line UML-Based Software Engineering*, *Family-Oriented Abstraction*, *Specification and Translation*, *Feature Oriented Domain Analysis* and *Delazeri e Wolf*, being the last one chosen for the accomplishment of the modeling. With the proposed model,

the developer has a flexible architecture that can be reused and/or extended to a new system of the same domain.

Keywords

Software Product Line; *UML Components*; Real Estate System.

1. INTRODUÇÃO

Para alcançar a excelência no desenvolvimento de seus produtos, ou seja, fornecer produtos de alta qualidade aliado a baixo custo em menor tempo de produção, empresas de software adotam abordagens baseadas em Linhas de Produto de Software (LPS), pois esta permite o desenvolvimento e manutenção de produtos de mesmo domínio, isto possibilita uma reutilização de artefatos, os quais atendem um número maior de clientes, o que, a longo prazo, significa um produto entregue em curto período e de maior confiabilidade devido aos inúmeros testes realizados no mesmo [1].

Este trabalho propõe modelar o subsistema de Administração do sistema imobiliário proposto por Canteri e Dvulatka [2]. Para isto, estendeu-se a análise do domínio dos sistemas Imobilis [3] e CIPRO [4] realizada na pesquisa de Canteri e Dvulatka [2] e realizou-se uma entrevista com o gerente de uma Imobiliária da cidade de Ponta Grossa - Paraná. Além disso, alguns métodos de desenvolvimento baseados em LPS foram analisados a fim de optar pelo qual mais se adequa ao escopo deste trabalho: *FAST (Family-Oriented Abstraction, Specification and Translation)* [5], *PLUS (Product Line UML-Based Software Engineering)* [6] e *FODA (Feature Oriented Domain Analysis)* [7] e *Delazeri e Wolf* [8].

Para a realização deste trabalho optou-se pela utilização do método de *Delazeri e Wolf* [8], pois este contempla características dos outros métodos analisados, de forma a produzir artefatos de entrada e saída para cada fase/subfase da Engenharia de Domínio e Aplicação.

2. LINHA DE PRODUTO DE SOFTWARE

Segundo *Clements e Northrop* [9], LPS é o desenvolvimento de famílias de produtos de software, que atendem um determinado segmento de mercado, considerando requisitos particulares de cada cliente de acordo com a análise de domínio, o reúso de código e características (*features*) comuns entre LPS de forma mais inteligente, reduzindo o tempo e o custo com o desenvolvimento.

Para determinar se os softwares estão ou não na mesma linha de produto, leva-se em consideração uma coleção de *features* [10]. Esta pode ser entendida como parte do sistema de suma importância para o cliente e que é utilizada para definir pontos comuns ou para diferenciar produtos em uma linha de produção.

Segundo Linden et al. [11], há três atividades principais da LPS e que juntas proporcionam práticas de negócios e tecnologia. São elas:

- Engenharia do Domínio: Consiste em coletar, organizar e armazenar experiências anteriores na construção de aplicações em um domínio específico na forma de artefatos empregáveis que possam ser utilizados na construção de novas aplicações [12].
- Engenharia da Aplicação: É atividade de construção de sistemas a partir de resultados obtidos por meio da engenharia de domínio, ou seja, a construção nesta atividade se dá por meio do *core*.
- Gerenciamento: abrange gestão técnica e organizacional da linha de produto [13].

A Figura 1 ilustra o relacionamento entre as três atividades, sendo cada atividade representada por um círculo. Nota-se a iteratividade entre as mesmas, a saída de uma atividade favorece a entrada da outra e assim sucessivamente.



Figura 1. Atividades essenciais no processo de linhas de produtos [9]

3. MÉTODOS DE DESENVOLVIMENTO BASEADO EM LPS

Há várias abordagens que podem ser utilizadas para o desenvolvimento baseado em LPS. Dentre estes, destacam-se: FAST [5], PLUS [6], FODA [7] e Delazeri e Wolf [8].

Após o estudo das abordagens, realizou-se uma análise qualitativa entre elas considerando os seguintes critérios:

- baseados em features: tipo de abordagem do método.
- baseado em família: tipo de abordagem do método.
- fases: são as divisões e subdivisões de cada método, que são produzidos os artefatos de entrada e saída.
- iterativo: processo de desenvolvimento cíclico.

A Tabela 1 detalha as abordagens citadas e suas respectivas características. As linhas contendo o “X” indicam a presença da característica em um determinado método, caso contrário, este método não contempla tal particularidade.

Observou-se que o método que Delazeri e Wolf possui as mesmas características dos métodos FODA, PLUS, FAST, pois o mesmo é uma adaptação dos três citados. Isto resulta em uma obtenção das melhores qualidades de cada um e ainda um acréscimo de novos artefatos, proporcionando que este método resulte em fases, documentações e diagramas distintos, sendo este adotado para a realização deste trabalho.

Tabela 1. Características dos Métodos baseados em LPS

Métodos	Características			
	Baseado em <i>feature</i>	Baseado em família	Possui fases e subfases	Iterativo
FAST [5]		X	X	
PLUS [6]	X		X	X
FODA [7]	X		X	
Delazeri e Wolf [8]	X	X	X	X

4. MODELAGEM

Esta seção descreve o uso do método de Delazeri e Wolf [8] para a identificação das similaridades e variabilidades dos subsistemas contemplados em um Sistema Imobiliário, estes identificados por Canteri e Dvulatka [2]. A Figura 2 ilustra o modelo de contexto obtido pela identificação dos subsistemas.

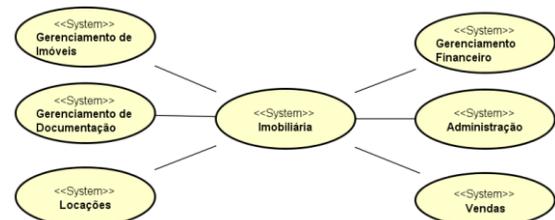


Figura 2. Diagrama de Modelo de Contexto [2]

Ao todo, identificou-se seis subsistemas que fazem parte de um sistema imobiliário. São eles: Gerenciamento de Imóveis, Gerenciamento Financeiro, Gerenciamento de Documentação, Administração, Locações e Vendas. Dentre os subsistemas identificados, optou-se por demonstrar a modelagem do subsistema de Administração. Demais subsistemas não serão abordados neste trabalho.

4.1 Requisitos e Modelagem do Domínio

Considerando a análise de domínio definiram-se os seguintes requisitos comuns para o subsistema Administração: Manter Clientes, Detalhar Descrição, Validar Dados, Armazenar e Manter Pessoas. Da mesma maneira, identificou-se os requisitos específicos, são eles: Terceiros, Construtoras, Funcionários e Verificar Contratos.

A Figura 3 ilustra o diagrama de caso de uso com as similaridades do subsistema de Administração.

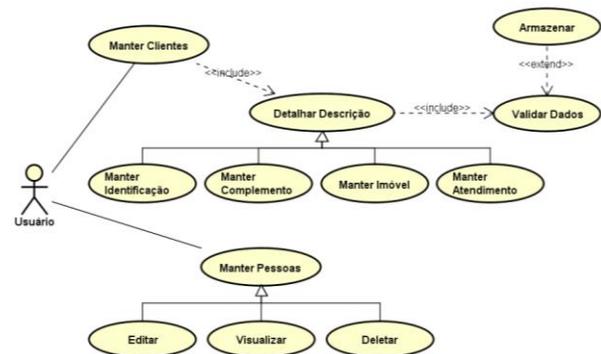


Figura 3. Diagrama de Caso de Uso com as similaridades do Subsistema de Administração

4.2 Modelagem da Arquitetura

Nesta subfase cada Caso de Uso já analisado no domínio torna-se uma interface da camada do sistema, tal como especifica o modelo UML *Components*. A Figura 4 ilustra as interfaces geradas com seus respectivos métodos para o Subsistema de Administração.

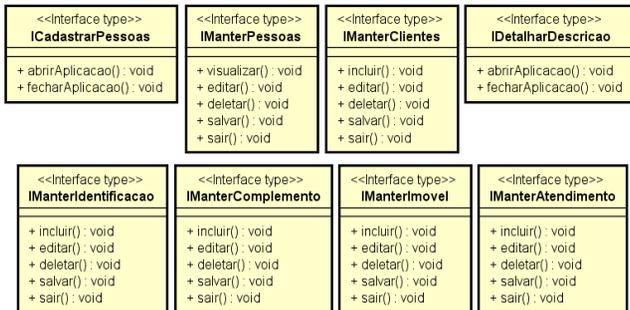


Figura 4. Interfaces com os Métodos do Software de Administração

4.3 Identificação dos Componentes

Com a identificação dos componentes que implementam as interfaces especificadas nas subfases anteriores, e de acordo com o processo UML *Components*, é associado um novo componente a cada interface, tanto de sistema quanto de negócio.

Assim, para a modelagem da camada Sistema, utilizam-se os Casos de Uso com as similaridades dos subsistemas. Neste trabalho foi apresentado apenas o Caso de Uso com as similaridades do subsistema de Administração (Figura 3) por questão de limitação de páginas. Com estes, identificou-se as seguintes interfaces: ICadastrarContas, IDetalharDescricao, ICadastrarImoveis, ICadastrarPessoas e IManterEndereco.

Considerando o conceito de linha de produto, ISoftwareImoveis controla as operações contidas no sistema, desta maneira, esta interface deve estar associada a todas as outras interfaces que fazem operações dentro do sistema.

Assim com as interfaces identificadas, com componentes e seus respectivos estereótipos relacionados com os casos de uso incluídos em cada componente é possível definir a arquitetura do Software de Imóveis com as camadas de Sistema e Negócio combinadas. Esta ilustrada na Figura 5.

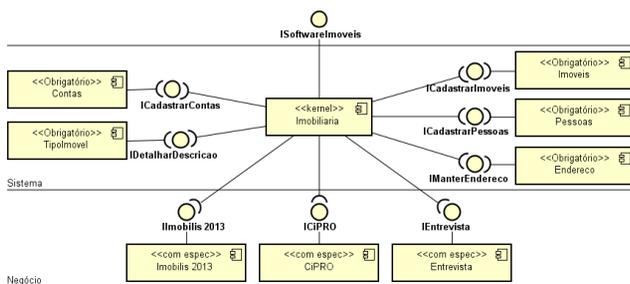


Figura 5. Arquitetura do Software de Imóveis

5. ENGENHARIA DA APLICAÇÃO

A fase de Engenharia de Aplicação, proposta por Delazeri e Wolf [8], é responsável pela identificação e modelagem dos requisitos da aplicação em diagrama de caso de uso e criação o

diagrama de *feature* variáveis inclusas no domínio. Além disso, é nesta fase que ocorre a implementação, validação com testes, funcionais e de integração, e a análise da integridade dos requisitos para, só então, ocorrer a entrega do produto ao cliente bem como seu posterior suporte.

5.1 Requisitos de Aplicação

Ao realizar análises no Subsistema de Administração de acordo com o Sistema Imobilis [3], foram encontradas as seguintes variedades:

- Funcionários: cadastro das pessoas que trabalham na imobiliária.
- Verificar contratos: possibilidade do usuário modificar o contrato de acordo com o estabelecido entre empresa e cliente.

Assim com a análise detalhada de cada requisito, construiu-se os Diagramas de *Features* que exibem as variabilidades dos sistemas analisados. Neste trabalho apenas o Diagrama de *Features* do subsistema de Administração é apresentado (Figura 6).



Figura 6. Variabilidades Subsistema de Administração

De acordo com os resultados obtidos a partir da análise das variabilidades dos subsistemas é possível ilustrá-los com Diagrama de Casos de Uso. A Figura 7 exibe as variabilidades do subsistema de Administração. O caso de uso “funcionários”, possui a funcionalidade de cadastrar as pessoas que trabalham na imobiliária, enquanto “verificar contratos” é responsável por possibilitar ao usuário modificar o contrato de acordo com o que ficou estabelecido entre a empresa e cliente.

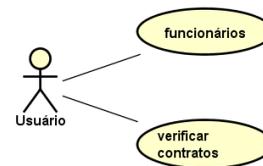


Figura 7. Caso de Uso das variabilidades do Subsistema de Administração

Por fim, com as interfaces definidas na fase de Projeto de Domínio, que contém os atributos de domínio e os de variabilidade, é possível modelar o produto. As classes com os estereótipos *Application* pertencem a fase de aplicação (variabilidades), *type* identificam as classes definidas na fase de domínio (similaridades) e *core* é o Kernel (núcleo) do sistema. A Figura 8 ilustra o diagrama de classe para a instanciação da aplicação dos subsistemas de Gerenciamento de Documentação, Gerenciamento Financeiro e Administração para o Sistema de Imobilis [3].

A implementação, teste e entrega da aplicação Sistema de Imóveis é realizada por meio da instanciação das interfaces comuns entre eles, incluídas de suas classes específicas.

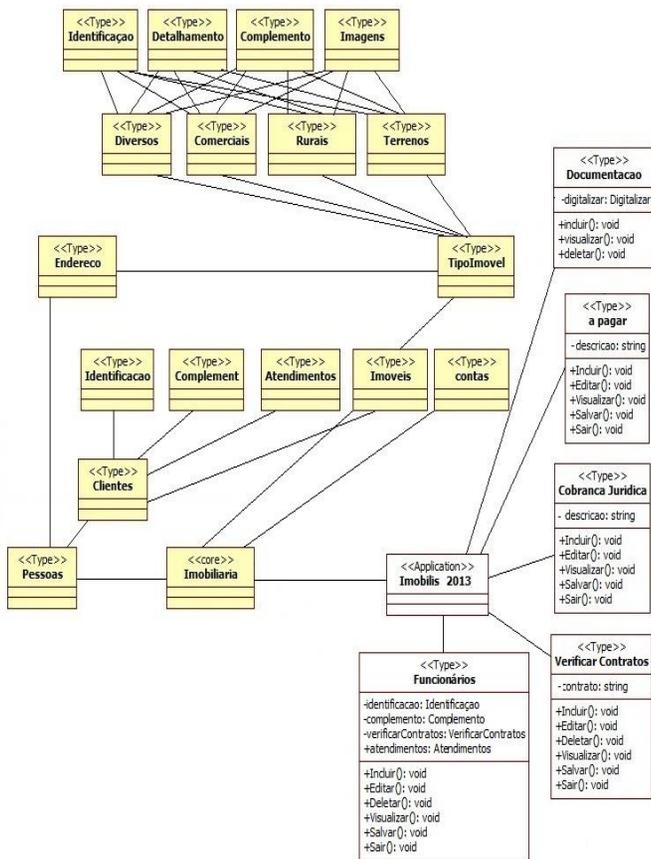


Figura 8. Modelo de Classe para Instanciação do Sistema de Imóveis

6. CONCLUSÕES

Este trabalho apresentou a modelagem do subsistema de Administração de um sistema imobiliário baseada em Linhas de Produto de Software.

Para a escolha da metodologia utilizada no desenvolvimento, realizou-se uma análise nos métodos FAST [5], PLUS [6], FODA [7] e Delazeri e Wolf [8], sendo este último o selecionado.

O desenvolvimento se deu em duas fases, são elas: Engenharia de Domínio e Engenharia de Aplicação. Na primeira foram realizadas análises de requisitos do domínio com o intuito de definir a relação de similaridades e variedades entre os sistemas. Ainda nesta fase, modelou-se o domínio e desenvolveu-se a arquitetura do domínio dos produtos estudados. Na segunda fase analisaram-se os requisitos variáveis. A partir dos mesmos criaram-se os Diagramas de Casos de Uso para definir o Diagrama de Classes como forma de instanciação de cada produto analisado.

Por meio deste trabalho pode-se concluir que o desenvolvimento baseado em linhas de produto de software possui vantagens tanto para os desenvolvedores, que terão um produto de forma rápida e

com maior qualidade, quanto para os clientes, que desfrutarão de um software desenvolvido de acordo com suas necessidades.

7. AGRADECIMENTOS

À Universidade Tecnológica Federal do Paraná, campus Ponta Grossa e campus Santa Helena, representado por suas respectivas diretorias, pelo apoio ao trabalho.

8. REFERÊNCIAS

- [1] A. E. de Campos Lobo and C. M. F. Rubira. Um estudo para implantação de linha de produto de software baseada em componentes.
- [2] C. Canteri and S. D. Dvulatka. Modelagem e implementação de um sistema de venda e de locação de imóveis baseado em linhas de produto. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2013.
- [3] Imobilis. 2017. *Imobilis 2013*. Retrieved August 7, 2017. DOI=<http://www.pjsoft.com.br>.
- [4] Ci-Pro. 2017. *CI-PRO - Consultoria Imobiliária Profissional*. Retrieved August 7, 2017. DOI=<http://www.baixaki.com.br/site/dwnld13464.htm>.
- [5] M. Harsu. *A survey on domain engineering*. Tampere University of Technology, 2002.
- [6] H. Gomaa. *Designing software product lines with UML*. IEEE, 2005.
- [7] L. Northrop, P. Clements, F. Bachmann, J. Bergey, G. Chastek, S. Cohen, P. Donohoe, L. Jones, R. Krut, R. Little, et al. A framework for software product line practice, version 5.0. *SEI*. {2007}{<http://www.sei.cmu.edu/productlines/index.html>, 2007.
- [8] B. R. Delazeri and E. C. Wolf. Modelagem de um sistema organizador baseado em linhas de produto. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2012.
- [9] P. Clements and L. Northrop. *Software product lines*. Addison-Wesley. 2002.
- [10] D. R. Nunes. C. e Reis, R. (2010). "Educação em engenharia de software". *A carreira do pesquisador em Engenharia de Software: princípios, conceitos e direções. 1ª edição*. Salvador, Brasil: UFBA, cap, 5:132{181.
- [11] F. Linden. vd, Schmid, K., and Rommes, E. (2007). Software product lines in action: The best industrial practice in product line engineering. *Springer-Verlag New York, Inc., Secaucus, NJ, USA*, 10:11{13.
- [12] A. P. da Silva, E. de Barros Costa, and I. I. Bittencourt. Uma linha de produto de software baseada na web semântica para sistemas tutores inteligentes. 2012.
- [13] D. Ferreira Santana Neiva. Riple-re: A requirements engineering process for software product lines. 2009.