

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
FLUMINENSE**
Campus Campos-Centro

Secretaria de Educação
Profissional e Tecnológica

Ministério
da Educação



CURSO DE TECNOLOGIA EM DESENVOLVIMENTO DE SOFTWARE

**ELISABELLY PESSANHA ARÊAS
HELVIO JERONIMO JUNIOR
RENATA APARECIDA ESPANO NALIM**

MODELAGEM DE UM FRAMEWORK PARA O DOMÍNIO DE HORÁRIO ESCOLAR

**CAMPOS DOS GOYTACAZES/RJ
2009**

**ELISABELLY PESSANHA ARÊAS
HELVIO JERONIMO JUNIOR
RENATA APARECIDA ESPANO NALIM**

**MODELAGEM DE UM FRAMEWORK PARA O DOMÍNIO DE
HORÁRIO ESCOLAR**

Monografia apresentada ao Instituto Federal de Educação, Ciência e Tecnologia Fluminense Campus Campos-Centro como requisito parcial para conclusão do Curso de Tecnologia em Desenvolvimento de Software.

Orientadora: Prof^a. Aline Pires Vieira de Vasconcelos, D.Sc.

**CAMPOS DOS GOYTACAZES/RJ
2009**

**ELISABELLY PESSANHA ARÊAS
HELVIO JERONIMO JUNIOR
RENATA APARECIDA ESPANO NALIM**

**MODELAGEM DE UM FRAMEWORK PARA O DOMÍNIO DE
HORÁRIO ESCOLAR**

Monografia apresentada ao Instituto Federal de Educação, Ciência e Tecnologia Fluminense Campus Campos-Centro como requisito parcial para conclusão do Curso de Tecnologia em Desenvolvimento de Software.

Aprovada em 27 de Julho de 2009.

Banca Avaliadora:

Prof^ª. Aline Pires Vieira de Vasconcelos – D.Sc. (Orientadora)

Prof. Jonivan Coutinho Lisboa – M.Sc.

Prof^ª. Simone Vasconcelos Silva – M.Sc.

AGRADECIMENTOS

Elisabelly Pessanha Áreas:

Em primeiro lugar a Deus, meu amigo, por me permitir estar aqui e me ajudar a concretizar um dos meus sonhos. “... sem Ele nada do que se foi feito se fez.” Jo1: 3;

A nossa Orientadora Profª. Aline Pires V. de Vasconcelos pelo incentivo e auxílio às atividades, discussões e várias correções desta monografia.

Ao Helvio e a Renata por me aturarem ao longo do curso. Brincadeiras a parte, mas em vocês eu descobri grandes amigos que estiveram ao meu lado nas notas altas e baixas desses três anos.

A minha mãe Eliana, por acreditar em mim e ao meu pai Sivaldo, pelos conselhos no momento em que pensei em desistir. A minha vovó amada Maria Isabel, que foi um exemplo de luta pela vida (*in memorian*). Ao meu avô Oracy, que me ajudou com algumas despesas durante o Curso. Aos demais familiares que contribuíram de alguma forma para eu concluir a minha Graduação. Aos professores participantes da banca avaliadora.

Helvio Jerônimo Junior:

As palavras ganham um novo sentido quando se tornam imprevisíveis, ganham uma nova vida quando inquietam o pensamento, mas há momentos, como este, em que nos sentimos desassossegados com as palavras porque se tornam insuficientes para dizer com elas o que queremos. Assim, ciente da limitação das palavras, aproveito para expressar os meus sinceros agradecimentos a todos aqueles que me acompanharam e que, de alguma forma, contribuíram com o seu saber científico e/ou dedicação pessoal para a concretização deste trabalho: Primeiramente agradeço a Deus, pois sem Ele nada do que foi feito seria possível. Obrigado Senhor por estar comigo a todo instante.

A nossa Orientadora e agora amiga, Profª. Aline Pires V. de Vasconcelos pelos ensinamentos transmitidos, paciência e palavras de críticas e incentivo quando necessário.

As minhas amigas, Elisabelly e Renata pelos incentivos, brincadeiras e toda a amizade que foi construída e que jamais será destruída. E a todos os amigos guerreiros deste curso.

Aos meus pais, Helvio e Sebastiana que me ajudaram a chegar até aqui. Não posso deixar de destacar a minha mãe, pelas palavras de sabedoria, amor e apoio constantes. Aos meus irmãos, Natalia e Igor, pelo o amor e amizade infinita.

Aos meus familiares, tios, tias, padrinhos e primos que sempre me ajudaram de alguma forma em particular.

A todas as amizades construídas nesta caminhada. Aos professores Jonnathan Carvalho e Davi pela colaboração. E os professores participantes da banca avaliadora deste trabalho: Jonivan e Simone.

Renata Aparecida Espano Nalim:

Agradeço a Deus que é o responsável por tudo que tenho e que terei. Só através Dele posso alcançar meus objetivos. José Nalim e Maria Amélia, esses são meus pais, meus maiores professores. Meus pais me ensinaram o que chamo de algoritmo da vida, num loop infinito, onde devo estudar, batalhar, trabalhar e conseguir chegar onde quero com coragem e dedicação. Profª. Aline Pires V. de Vasconcelos por ser essa profissional dedicada, inteligente e humana. Saiba que para mim você se tornou um exemplo a seguir.

Ao Helvio e a Elisabelly obrigada por me fazerem sentir essa coisa tão boa e abençoada que se chama amizade. Obrigada pela força e também pelas brincadeiras para descontrair ao longo de nosso estudo.

Aos meus irmãos, amigos, colegas de trabalho, são tantas as pessoas especiais que contribuíram para a conclusão dessa etapa. E não posso esquecer dos Professores Jonnathan Carvalho que sempre esteve disposto a ajudar e a Professora Michelle Freitas por seus conselhos e palavras de otimismo. Agradeço também aos Professores da banca avaliadora.

“Nem olhos viram, nem ouvidos ouviram, nem jamais chegou ao entendimento humano, o que Deus tem preparado para aqueles que o amam”.

(I Coríntios 2:7-9)

*Aos meus pais, Eliana e Sivaldo.
Aos meus Avôs, Oracy e Francisco.*

Elisabelly. P. Arêas

*Aos meus pais, Sebastiana e Helvio.
Aos meus irmãos, Natália e Igor.*

Hélvio J. Júnior.

*Aos meus pais, José Nalim e Maria Amélia.
Aos meus irmãos José Carlos e Marcos.*

Renata A. E. Nalim.

RESUMO

Este projeto visa à aplicação de técnicas de Reutilização de Software para a construção de um *framework* para o Domínio de Horário Escolar. A Reutilização de software visa o desenvolvimento a partir de artefatos pré-existentes, levando à melhora da qualidade e produtividade no desenvolvimento. As técnicas empregadas englobam a Engenharia de Domínio (ED), envolvendo principalmente a Análise de Domínio (AD), e princípios de *frameworks*, como determinação de seus pontos configuráveis (i.e. hot spots). A ED envolve o desenvolvimento de software para um domínio ou família de aplicações, onde a AD representa a sua primeira etapa e os *frameworks* a arquitetura do domínio. A montagem de Horário Escolar representa uma atividade que é realizada por todas as instituições de ensino, onde cada instituição realiza esta atividade de forma particular, porém com muitos aspectos em comum. Daí a escolha desse domínio para aplicar a Reutilização de Software. O domínio de Horário Escolar deve conter as informações necessárias para o gerenciamento de horários de turmas, professores e ambientes de aprendizagem. Através da AD, visa-se determinar aspectos comuns, opcionais e variáveis das atividades que são realizadas neste domínio. Para esta análise, três coordenações de curso foram entrevistadas, permitindo, dessa forma, o desenvolvimento de artefatos reutilizáveis de software. É utilizado o ambiente Odyssey de reutilização de software, desenvolvido pela COPPE/UFRJ, para apoiar o processo proposto, pois o desenvolvimento voltado à reutilização envolve maior esforço do que aquele voltado a uma aplicação específica.

Palavras-chaves: Engenharia de Domínio (ED), Horário Escolar, Análise de Domínio (AD), *Frameworks*, Variabilidades e Opcionalidades.

ABSTRACT

This project involves the application of Software Reuse techniques in order to build a *framework* for the School Hour Schedules Domain. The reuse of software aims the development from pre-existing artifacts, leading to improved quality and productivity in software development. The employed techniques include Domain Engineering (DE), mainly involving Domain Analysis (DA), and principles of *frameworks*, such as the determination of their configurable points (*i.e. hot spots*). The DE involves the development of software for a domain or family of applications, where the DA is the first step and the *frameworks* define the domain architecture. The elaboration of School Hour Schedules is an activity that is performed by all educational institutions, where each institution performs this activity in a particular way, but with many points in common. Therefore this domain was chosen to implement the software reuse. The School Hour Schedules domain must contain the necessary information for managing schedules of classes, teachers and learning environments. Through the DA, it is possible to determine common, optional and variable aspects of activities that are performed in this domain. For this analysis, three course coordinations were interviewed, allowing thus the development of reusable software artifacts. The software reuse environment Odyssey developed at COPPE / UFRJ, is used to support the proposed process, since the development for reuse requires a greater effort than the one involving a specific application.

Keywords: Domain Engineering (DE), School Hour Schedules, Domain Analysis (DA), Frameworks, Variabilities and Optionalities.

LISTA DE SIGLAS

AD	Análise de Domínio
CBD-Arch-DE	Component Basead Development - Architecture – Domain Engineering
DSSA	Domain Specific Software Architectures
EA	Engenharia de Aplicação
ED	Engenharia de Domínio
FeatureRSEB	Reuse – Driven Software Engineering Business
FEX	Feature Extend
FODA	Fetaure Oriented Domain Analysis
FORM	Feature Oriented Reuse Method
IFF	Instituto Federal de Educação, Ciência e Tecnologia Fluminense
LP	Linha de Produto
ODM	Organization Domain Modeling
PLP	Product Line Practice
RF	Requisitos Funcionais
SEI	Software Engineering Institute
UC's	Use Cases
UML	Unified Modeling Language
VP	Variant Point

LISTA DE FIGURAS

FIGURA 2.1 ATIVIDADES ESSENCIAIS EM LINHAS DE PRODUTOS	19
FIGURA 2.2 CICLO DE VIDA NA ENGENHARIA DE DOMÍNIO (OLIVEIRA, 2006)	22
FIGURA 2.3 FASES DA ENGENHARIA DE DOMÍNIO (UFPE, 2007)	23
FIGURA 2.4 UM MODELO GENÉRICO DE DESENVOLVIMENTO PARA / COM REUSO (GUIZZARDI,2007)	25
FIGURA 2.5 CARACTERÍSTICAS CONCEITUAIS E FUNCIONAIS NA NOTAÇÃO ODYSSEY-FEX	33
FIGURA 2.6 PROPRIEDADES ORTOGONAIS DA NOTAÇÃO ODYSSEY-FEX	34
FIGURA 3.1 DIAGRAMA DE CONTEXTOS DO DOMÍNIO DE HORÁRIO ESCOLAR NO AMBIENTE ODYSSEY	45
FIGURA 3.2 DIAGRAMA DE CARACTERÍSTICAS CONCEITUAIS DE CURSO	47
FIGURA 3.3 DIAGRAMA DE CARACTERÍSTICAS DA MATRIZ CURRICULAR DOS CURSOS SUPERIORES E DO CURSO DE PÓS-GRADUAÇÃO	48
FIGURA 3.4 CARACTERÍSTICAS DE CURSOS, MATRIZ CURRICULAR, PERÍODO E MÓDULO ...	49
FIGURA 3.5 DIAGRAMA DE CARACTERÍSTICA DO SUBDOMÍNIO DE DISPONIBILIDADES DE PROFESSOR E COMPETÊNCIA	51
FIGURA 3.6 DIAGRAMA DE CARACTERÍSTICAS DE AMBIENTES DE APRENDIZADO	52
FIGURA 3.7 DIAGRAMA DE CARACTERÍSTICAS DO SUBDOMÍNIO DE CONTROLE DE HORÁRIO ESCOLAR	54
FIGURA 3.8 CARACTERÍSTICA CURSO E SUAS VARIANTES	55
FIGURA 3.9 DIAGRAMA DE CARACTERÍSTICAS DE PARTE DO DOMÍNIO DE HORÁRIO	56
FIGURA 3.10 OPÇÕES DE MAPEAMENTOS NO ODYSSEY	57
FIGURA 3.11 DIAGRAMA DE CLASSES DE PARTE DO DOMÍNIO DE HORÁRIO ESCOLAR GERADO A PARTIR DAS CARACTERÍSTICAS CONCEITUAIS	57
FIGURA 3.12 DIAGRAMAS DE CLASSES DO FRAMEWORK	59
FIGURA 3.13 DIAGRAMA DE CARACTERÍSTICAS FUNCIONAIS E DE ENTIDADES	60
FIGURA 3.14 WIZARD DE MAPEAMENTO DE CARACTERÍSTICAS FUNCIONAIS PARA CASOS DE USO NO AMBIENTE ODYSSEY	61
FIGURA 3.15 DIAGRAMA DE CASOS DE USO	61
FIGURA 3.16 CRIANDO UMA APLICAÇÃO BASEADA EM UM MODELO DE DOMÍNIO	65
FIGURA 3.17 SELECIONANDO CONTEXTOS	65
FIGURA 3.18 SELECIONANDO AS CARACTERÍSTICAS QUE IRÃO COMPOR A APLICAÇÃO	66
FIGURA 3.19 MODELO DE DOMÍNIO PARA A ÁREA DE CONSTRUÇÃO CIVIL OBTIDO ATRAVÉS DA ENGENHARIA DE APLICAÇÃO	67

LISTA DE TABELAS

TABELA 2.1 TIPOS DE CARACTERÍSTICAS NA NOTAÇÃO ODYSSEY-FEX (OLIVEIRA, 2006)	31
TABELA 2.2 RELACIONAMENTOS DA NOTAÇÃO ODYSSEY-FEX.....	32
TABELA 2.3 TIPOS DE CARACTERÍSTICAS NO AMBIENTE ODYSSEY	34
TABELA 3.1 CARACTERÍSTICAS DO DOMÍNIO DE HORÁRIO ESCOLAR E SUAS CLASSIFICAÇÕES	69

SUMÁRIO

LISTA DE SIGLAS	9
LISTA DE FIGURAS	10
LISTA DE TABELAS	11
CAPÍTULO 1: INTRODUÇÃO.....	14
1.1 MOTIVAÇÕES	14
1.2 OBJETIVOS	15
1.3 ORGANIZAÇÃO DO TRABALHO	17
CAPÍTULO 2: REFERENCIAL TEÓRICO.....	18
2.1 INTRODUÇÃO.....	18
2.2 ENGENHARIA DE DOMÍNIO.....	21
2.2.1 Análise de Domínio	26
2.3 AMBIENTE DE REUTILIZAÇÃO DE SOFTWARE ODYSSEY	28
2.3.1 A notação Odyssey-Fex e a Variabilidade na Engenharia de Domínio	30
2.4 FRAMEWORKS	35
2.4.1 Tipos de Frameworks	36
2.5 CONSIDERAÇÕES FINAIS	37
CAPÍTULO 3: MODELAGEM DE UM FRAMEWORK PARA O DOMÍNIO DE HORÁRIO ESCOLAR.....	39
3.1 INTRODUÇÃO.....	39
3.2 ANÁLISE DE DOMÍNIO	40
3.2.1 Descrição do Domínio	42
3.2.2 Contextos	44
3.2.3 Subdomínio de Controle de Curso e Sua Matriz Curricular.....	46
3. 2.3.1 Características de Curso e sua Matriz Curricular	46
3.2.4 Subdomínio de disponibilidade e Competência de Professores	50
3.2.4.1 Características de Professores e suas Disponibilidades e Competências	50
3.2.5 Subdomínio de Controle de Ambientes de Aprendizagem	52
3.2.5.1 Características de Ambientes de Aprendizagem	52
3.2.6 Subdomínio de Controle de Horário.....	53
3.2.6.1 Características de Horário de Turmas, Professores e Ambientes de Aprendizagem.....	53
3.3 PROJETO DO DOMÍNIO	55
3.4 MAPEANDO O MODELO DE CARACTERÍSTICAS FUNCIONAIS PARA O MODELO DE CASOS DE USO	60
3.5 ENGENHARIA DE APLICAÇÃO NO CONTEXTO DA REUTILIZAÇÃO	63
3.5.1 Instanciando O Framework para uma nova Aplicação Específica do Domínio.	64
3.6 CONSIDERAÇÕES FINAIS	70

CAPITULO 4: CONCLUSÕES E TRABALHOS FUTUROS	71
4.1 CONCLUSÕES	71
4.2 CONTRIBUIÇÕES E BENEFÍCIOS DO TRABALHO	72
4.3 DIFICULDADES ENCONTRADAS	73
4.4 TRABALHOS FUTUROS	73
REFERÊNCIAS BIBLIOGRÁFICAS.....	75
ANEXO A: DESCRIÇÃO DOS CASOS DE USO DO FRAMEWORK IMPLEMENTADO	80
ANEXO B: ALGUMAS TELAS DO SISTEMA IMPLEMENTADO PARA O FRAMEWORK	92
ANEXO C: FORMULÁRIOS PREENCHIDOS COM BASE NAS ENTREVISTAS REALIZADAS NAS COORDENAÇÕES DE CURSO.....	95
ANEXO D: HEURÍSTICAS DE MAPEAMENTO DE CARACTERÍSTICAS DA NOTAÇÃO ODYSSEY-FEX	104

CAPÍTULO 1: INTRODUÇÃO

1.1 MOTIVAÇÕES

Atualmente, a sociedade está diante dos avanços tecnológicos impostos pelo mundo e não dispõe de muito tempo. Nesse contexto, o software é extremamente necessário ao atendimento desses avanços. Com o passar dos anos, o desenvolvimento de software vem lidando com software cada vez mais complexo e, ao mesmo tempo, com uma demanda cada vez maior por software de qualidade e entregue em menor tempo. A busca pelo aumento da qualidade e redução do esforço no desenvolvimento de software é crescente na comunidade de desenvolvedores. Percebe-se que a sociedade precisa de sistemas que utilizem técnicas sistemáticas com tempo de desenvolvimento menor possível, mas com toda qualidade que possa apresentar.

Para isso, é apresentado o fator de reutilização de software que consiste basicamente no processo de criação de sistemas de software a partir de outros existentes. Reduz-se, assim, o custo do software em si e em pesquisas extensas. Para boa reutilização, a idéia é que se desenvolva um projeto que utilize métodos e ferramentas de análises e modelagem de software. Tais como: Engenharia de Domínio (ED), Análise de Domínio (AD) e Engenharia de Aplicação (EA) etc., possibilitando que projetos futuros reutilizem a estrutura atual e artefatos do sistema, como requisitos, projeto, programas etc. Possibilitando ainda a criação ou extensão para sistemas mais complexos em menos tempo.

Aplicando a reutilização de software, uma das técnicas possíveis de ser adotada é a Engenharia de Domínio, que oferece um apoio sistemático à reutilização desde o início do ciclo de vida do software, apoiando a reutilização também de artefatos de análise e de projeto, ao invés de somente a reutilização de código. O termo Domínio é utilizado para referenciar ou agrupar uma coleção de aplicações, existentes ou futuras, que compartilhem características comuns, i.e., uma classe de sistemas que apresentam funcionalidades similares (WERNER & BRAGA, 2005). A Engenharia de Domínio pode ser definida como: o processo de identificação e organização do conhecimento sobre uma classe de problemas, i.e. o domínio do problema, para apoiar a sua descrição e solução (PRIETO-DIAZ e ARANGO, 1991). A ED trabalha na criação de soluções genéricas que venham a ser reutilizadas pelas aplicações de domínio, ressaltando as suas variabilidades, que, por sua vez, se mostram bastante importantes, pois deixam claros os pontos onde os sistemas

se assemelham para serem reutilizados e os pontos onde se diferem, permitindo a configuração de diferentes aplicações do domínio.

A ED incorpora a etapa de Análise de Domínio (AD), que oferece suporte para o reuso sistemático em larga escala, através da captura das características comuns e das variabilidades entre os sistemas do domínio. A AD é inicializada logo após a conclusão da fase de planejamento na ED, ou seja, a AD, uma das etapas da ED, cria um arcabouço propício de reuso para a implementação dentro de um escopo bem definido.

Aplicando-se os conceitos citados em reutilização de software focada na ED e em uma de suas etapas, a AD, busca-se desenvolver um projeto voltado para a implementação de um domínio através de um *framework*. Um *framework* é um conjunto de classes que cooperam para construir um sistema reutilizável para uma classe específica de software ou domínio (GAMMA, HELM, JOHNSON e VLISSIDES, 2000). Com os *frameworks*, uma estrutura genérica pode ser construída para um domínio específico, com divisão de classes e objetos, ou seja, na verdade os *frameworks* definem a arquitetura de um determinado domínio, contemplando as suas variabilidades e flexibilidade. Além disso, capturam funcionalidades comuns em várias aplicações e as disponibilizam em uma estrutura que tende a ser de fácil manuseio e entendimento.

Um domínio possível para explorar os conceitos citados, é o domínio de Horário Escolar, esse domínio é tema de pesquisas em várias universidades do mundo, conhecido como o “o problema de quadros de horários”. Trata-se, assim, de um problema clássico, que historicamente vem sendo solucionado ou de forma artesanal ou através de soluções específicas para a própria instituição que as desenvolveram. Segundo LOBO (2005) este domínio pode ser declarado como um problema que consiste em fixar, num determinado período de tempo conhecido, um conjunto de aulas, de forma a atender às exigências acadêmicas estabelecidas por um certo currículo de estudos para um certo grupo de disciplinas. Onde cada aula visa o atendimento de um único grupo de estudantes e exige pelo menos o comparecimento de um único professor à mesma.

A necessidade de cada Instituição de Ensino faz com que este problema seja tratado de forma particular. As aplicações existentes no mercado voltados para este domínio possuem variações quanto às suas funcionalidades, motivando assim, este trabalho a propor uma arquitetura genérica para o referido domínio.

A Reutilização de Software e suas abordagens tais como: Engenharia de Domínio (ED), Linha de Produtos e *Frameworks*, além da Engenharia de Aplicação (EA) são conceitos poucos explorados nesta Graduação e de grande valia para a Engenharia de Software. Daí a oportunidade deste tema para pesquisa e aprofundamento neste trabalho focado na Engenharia de Domínio baseado no contexto de reutilização do Projeto Odyssey explorando técnicas de Reutilização de Software baseada na Engenharia de Domínio (ED), Engenharia de Aplicações (EA) e *Frameworks*.

1.2 OBJETIVOS

No ano de 2008, o CEFET Campos passou a ser chamar Instituto Federal de Educação, Ciência e Tecnologia Fluminense (IFF), com todas as prerrogativas que lhe são pertinentes. Atualmente, o IFF atua nos três níveis da formação profissional, trabalhando na qualificação básica e continuada de trabalhadores, oferecendo Cursos Técnicos e Superiores de Tecnologia, Licenciatura e Bacharelados. Além de Ensino e Educação de Jovens e Adultos, Cursos de Pós-Graduação, incluindo o Mestrado.

Ensino, Pesquisa e Extensão compõem hoje o universo de ação do IFF, considerando o tripé que caracteriza um Centro de Educação, daí a necessidade das tecnologias acompanharem o desenvolvimento e o crescimento da Instituição.

O objetivo deste trabalho é explorar o desenvolvimento de software voltado à Análise de Domínio (AD), explorando algumas técnicas baseadas em reutilização, como a Engenharia de Domínio (ED), Engenharia de Aplicação (EA) e *Frameworks*, tendo sido escolhido como estudo de caso o domínio de Horário Escolar do Instituto Federal Fluminense (IFF) Campus Campos-Centro. Para o desenvolvimento deste trabalho, três Coordenações de Cursos foram escolhidas para embasar o desenvolvimento deste. As três coordenações contempladas apresentam boa representatividade das funcionalidades envolvidas neste domínio. Uma vez que o *framework* esteja definido, outras coordenações podem ir sendo incorporadas, através da instanciação e extensão deste *framework*.

O domínio de Horário Escolar do Instituto Federal Fluminense (IFF) Campus Campos-Centro foi escolhido para representar o domínio de Horário Escolar por apresentar características que demonstram claramente a necessidade deste *framework*, pois ele é fundamental dentro de uma Instituição de Ensino, principalmente para o IFF por trabalhar

com diferentes níveis de ensino, onde estes possuem características em comum e aspectos distintos, porém uma vez o modelo de domínio construído com base na análise realizada nesta instituição ele poderá ser adaptado para outras instituições futuramente a partir do *framework* apresentado neste trabalho.

Como já mencionado o domínio de Horário Escolar vem conquistando várias pesquisas, a maioria destas pesquisas é voltada para solucionar o famoso problema de quadro de Horário Escolar através de propostas baseadas em algoritmos genéticos. Com base nestas pesquisas, várias aplicações vêm sendo desenvolvidas para este domínio, porém essas aplicações são desenvolvidas de forma particular para atender um determinado problema ou uma Instituição de Ensino. A montagem de Horário Escolar que atenda aos interesses de professores e alunos e utilize da melhor forma possível os recursos disponíveis em qualquer nível de ensino representa uma atividade que é realizada por todas as Instituições de Ensino, onde cada instituição realiza esta atividade de forma particular, porém com muitos aspectos em comum. Dessa forma, justifica-se a necessidade de uma análise de domínio com a finalidade de identificar as semelhanças e variações existentes que possam ser representadas através de um *framework* e junto a conceitos e técnicas de reutilização, demonstrando a sua flexibilidade, pontos comuns e configuráveis.

Pretende-se também apresentar um ambiente de modelagem baseado em reutilização, aprimorando o conhecimento adquirido, e assim, sendo possível à construção de um *framework* para o referido domínio. O ambiente citado acima é o Odyssey (ODYSSEY, 2007), que apóia as atividades da ED e da EA, e se encontra disponível para uso no site da equipe de reutilização da COPPE/UFRJ. Esse ambiente foi desenvolvido e vem sendo mantido pelo grupo de reutilização da COPPE/UFRJ.

O principal objetivo aqui, portanto, é contribuir para o desenvolvimento de sistemas de software futuros de Horário Escolar, representando, dessa forma, a base para aplicações do mesmo domínio. Dessa forma, o *framework* pode ser instanciado por diferentes aplicações para diferentes Coordenações e Cursos. Para exemplificar a instanciação através da Engenharia de Aplicação, o *framework* construído será implementado através das suas características em comum, sendo instanciado para uma aplicação específica no domínio mencionado.

1.3 ORGANIZAÇÃO DO TRABALHO

Partindo deste capítulo, o restante do trabalho está organizado da seguinte forma:

O **Capítulo 2** aborda o referencial teórico necessário para o desenvolvimento deste trabalho, onde são exploradas abordagens de reutilização de software, tais como: Engenharia de Domínio (ED), Linhas de Produtos (LP), *Frameworks* e Engenharia de Aplicação (EA). É apresentado também neste capítulo o Ambiente de Reutilização, Odyssey, sendo explorada a sua notação.

O **Capítulo 3** apresenta a modelagem de um *framework* para o domínio de Horário Escolar. Esse modelo foi desenvolvido com base nas técnicas da Engenharia de Domínio descritas no Capítulo 2. Uma das etapas da Engenharia de Domínio (ED) mais exploradas neste Capítulo foi a Análise de Domínio (AD). A arquitetura do domínio é apresentada através do *framework* desenvolvido. Todo este processo foi desenvolvido no Ambiente de Reutilização Odyssey.

O **Capítulo 4** apresenta as considerações finais e as possíveis propostas para trabalhos futuros.

Os Anexos constam ao final deste trabalho, após as referências bibliográficas. O **Anexo A** apresenta a descrição dos casos de usos que foram implementados do domínio. O **Anexo B** ilustra algumas telas do Sistema de Apoio ao Gerenciamento de Horário Escolar (SAGHE) implementado a partir do *framework*. O **Anexo C** apresenta o estudo de variabilidade realizado em três Coordenações de Curso do IFF. E o **Anexo D** apresenta algumas restrições da notação Odyssey-FEX e as heurísticas de mapeamento de características funcionais para casos de uso e de características conceituais para classes que foram adotadas neste trabalho.

CAPÍTULO 2: REFERENCIAL TEÓRICO

2.1 INTRODUÇÃO

A maioria das organizações que desenvolve software costuma construir sistemas de software em um domínio de aplicação particular, repetidamente entregando variantes de produtos pela adição de novas características (SUGUMARAN *et al.*, 2006). Dessa forma, é comum que essas organizações estejam migrando para abordagens de Engenharia de Domino (ED) (PRIETO-DIAZ & ARANGO, 1991). A produção de software e seu nível de complexidade têm aumentado nas últimas décadas, mas não o suficiente para acompanhar a demanda da indústria. Com isso, a reutilização apareceu como uma das poucas abordagens reais para tratar o problema, aumentando a produtividade e a qualidade que a indústria necessita, através do aproveitamento de partes de software já existentes e testadas em outras situações. A reutilização é uma atividade que está sendo muito disseminada pela comunidade de desenvolvimento de software. Ela pode ser definida como o processo de criação de sistemas de software a partir de software preexistente (KRUEGER, 1992), trazendo consigo a promessa de aumentar a produtividade e diminuir custos do desenvolvimento, além de melhorar a qualidade do produto, através da reutilização de artefatos de software já testados e utilizados em outros contextos (FRAKES e KANG, 2005). Ela pode ser considerada como uma sub-área da Engenharia de Software que cresce cada vez mais. No contexto da Engenharia de Software, diferentes abordagens buscam melhorar a qualidade dos artefatos¹ de software, bem como diminuir o tempo e os esforços necessários para produzi-los, assim como abordagens de reutilização.

A reutilização não é uma técnica nova, ela já vem sendo realizada desde o início da Engenharia de Software, porém, anteriormente, se reutilizava apenas bibliotecas de rotinas ou parte de código em ambientes de programação. Mas, para que a reutilização seja mais efetiva, ela não deve ser aplicável somente a fragmentos de código fonte, mas a todo o trabalho gerado durante o processo de desenvolvimento de software, como requisitos, arquitetura e projeto, o que definimos aqui como o objetivo principal deste trabalho. Portanto, a informação suscetível de reutilização inclui a análise de requisitos, especificações do sistema, estruturas de projeto ou qualquer artefato que seja gerado no

¹ Artefato é o termo usado para qualquer produto do trabalho de desenvolvimento de software.

processo de desenvolvimento de software. Ou seja, para que a reutilização possa ser efetiva, deve-se considerá-la em todas as fases de desenvolvimento de aplicações, desde a fase de análise até a fase de implementação. Abordagens de reutilização de software, como Engenharia de Domínio (ED) e Linha de Produtos (LP) apresentam essa preocupação, estabelecendo o desenvolvimento de artefatos de software reutilizáveis desde o nível de abstração de análise até o nível de implementação (VASCONCELOS, 2007). Sendo este trabalho voltado à adoção e exemplificação de técnicas de reutilização através da ED, vale ressaltar aqui um paralelo entre estas duas abordagens, sendo as premissas apresentadas válidas também para a LP.

A maioria das empresas costuma desenvolver sistemas de software em um domínio de aplicação particular, motivando a adoção de abordagens de ED e LP. As abordagens de ED e LP proporcionam a reutilização em todas as fases do desenvolvimento de software, ou seja, desde o início do seu ciclo de vida.

Segundo VASCONCELOS (2007), a ED e a LP apresentam muitas semelhanças entre si. A ED possibilita que características comuns e variáveis de um domínio possam ser especificadas e modeladas tendo como base um processo previamente modelado, gerando artefatos que possam ser instanciados para uma aplicação específica do domínio. A LP pode ser considerada como uma vertente da ED, cujo foco foi transferido para o âmbito empresarial (VASCONCELOS *et al.*, 2007). Ela tem como objetivo principal promover a geração de produtos específicos com base na reutilização de uma infra-estrutura central. Essa infra-estrutura, também conhecida como núcleo de artefatos, é formada normalmente por uma arquitetura de software e seus componentes. A figura 2.1 apresenta as atividades essenciais em LP.



Figura 2.1 Atividades Essenciais em Linhas de Produtos

O SEI (*Software Engineering Institute*), através da iniciativa da PLP (*Product Line Practice*), estabeleceu as atividades essenciais de uma abordagem de LP (SEI, 2004). Essas atividades ilustradas na figura 2.1 são: o **Desenvolvimento do Núcleo de Artefatos**, também conhecida como Engenharia de Domínio, o **Desenvolvimento do Produto**, também conhecida como Engenharia de Aplicação, e o **Gerenciamento da Linha de Produto**.

Os três círculos da figura 2.1 indicam que as atividades de uma LP são altamente interligadas e interativas. As flechas rotativas indicam que, além dos artefatos gerados no desenvolvimento de produtos do núcleo, são também realizadas as revisões dos artefatos ou até mesmo inclusão de novos artefatos. Na medida em que os produtos são desenvolvidos, o núcleo dos artefatos pode evoluir.

Alguns dos objetivos das abordagens Engenharia de Domínio e Linha de Produtos, segundo a maioria dos autores pesquisados, são:

- A possibilidade para que as organizações possam explorar as semelhanças entre seus produtos, aumentando, assim, a reutilização de artefatos. Como consequência, tem-se uma diminuição dos custos de tempo no desenvolvimento (HEYMANS e TRIGAUX, 2003).
- Descobrir as opcionalidades e variabilidades dentro de um domínio ou dentro de uma família de produtos específicos desenvolvidos de maneira sistemática a partir de um conjunto comum de artefatos.
- Melhor compreensão do domínio e aumento da qualidade.
- Melhor análise e aumento da reutilização dos requisitos e dos artefatos e o estabelecimento de padrões e documentação reutilizável.
- Redução de Tempo de Entrega.

Uma outra importante abordagem de reutilização de software são os *frameworks*, onde estes visam construir um projeto reutilizável para uma aplicação específica. Quando essas estruturas são estendidas e adaptadas, elas permitem que aplicações específicas sejam produzidas.

Um *framework* visa que uma especificação adequada seja produzida a uma família de aplicações com características similares (i.e. um domínio) e que esta especificação seja reutilizada no desenvolvimento de uma aplicação específica. A sua vantagem não é apenas

o reuso de código, mas também de projeto. Com a finalidade de apoiar a especificação de um conjunto de aplicações, a Engenharia de Domínio (ED) pode ser adotada em conjunto com os *frameworks*, que vão representar a arquitetura do domínio, atendendo aos seus requisitos.

Partindo desta introdução, o restante do capítulo está organizado da seguinte forma: a Seção 2.2 aborda conceitos da Engenharia de Domínio; a Seção 2.3 trata do Ambiente de reutilização Odyssey; a Seção 2.4 aborda o conceito de *frameworks*; e a Seção 2.5 apresenta algumas considerações finais.

2.2 ENGENHARIA DE DOMÍNIO

O conhecimento sobre o domínio da aplicação é fundamental em qualquer processo de desenvolvimento de software (OLIVEIRA *et al.*, 2006). Por exemplo, uma das principais causas de falhas durante o desenvolvimento de software é a falta do conhecimento do domínio da aplicação. Visando resolver este problema existe na Engenharia de Software uma área de pesquisa que se preocupa com essa questão denominada Engenharia de Domínio (ED). O objetivo da ED é possibilitar que características comuns e variáveis de um domínio possam ser identificadas e modeladas com base em um processo previamente definido. Os artefatos gerados pela ED (i.e. modelos de domínio, arquiteturas de componentes, casos de uso de domínio, *frameworks*, dentre outros) podem ser instanciados para uma aplicação específica do domínio. A esta instanciação dá-se o nome de Engenharia de Aplicação (EA). A ED provê um conjunto de artefatos para reutilização enquanto que a EA constrói aplicações com base na reutilização de artefatos providos pela ED.

Identificando as semelhanças e as diferenças de um domínio, ou seja, as suas variabilidades e opcionalidades, podem-se tornar viável a reutilização de artefatos produzidos tanto para os sistemas em desenvolvimento como para novos sistemas do mesmo domínio. Os conceitos de modelagem de variabilidade compreendem: características mandatórias que devem estar presentes nas aplicações do domínio; características opcionais, que podem aparecer em apenas algumas aplicações do domínio; e pontos de variação com suas variantes, que definem os pontos onde as aplicações do

domínio podem ser configuradas. Estes conceitos serão definidos detalhadamente na Seção 2.3.1.

Segundo PRESSMAN (2006), pode-se definir como principal objetivo da ED: identificar, construir, catalogar e disseminar um conjunto de componentes de software que têm aplicabilidade existente e futura, em um particular domínio de aplicação. Na ED, todos os artefatos produzidos são catalogados e armazenados de forma que possam ser utilizados a qualquer momento por outra aplicação que esteja no mesmo domínio.

Enquanto a ED é voltada à construção de artefatos reutilizáveis para várias aplicações, ou seja, é voltada ao desenvolvimento para a reutilização, a Engenharia de Aplicação (EA) é responsável por implementar aplicações através da instanciação dos artefatos que foram produzidos pela ED, conforme mostra a figura 2.2. A EA se baseia na ED buscando componentes de software que estejam disponíveis para reutilização como em uma biblioteca. Ela tem o objetivo de estudar, desenvolver e aplicar as melhores técnicas, processos e métodos para a produção de aplicações, no contexto do desenvolvimento com reutilização. A EA pode produzir *feedback* para ED no sentido da descoberta de novos artefatos reutilizáveis durante o desenvolvimento de novas aplicações no domínio.

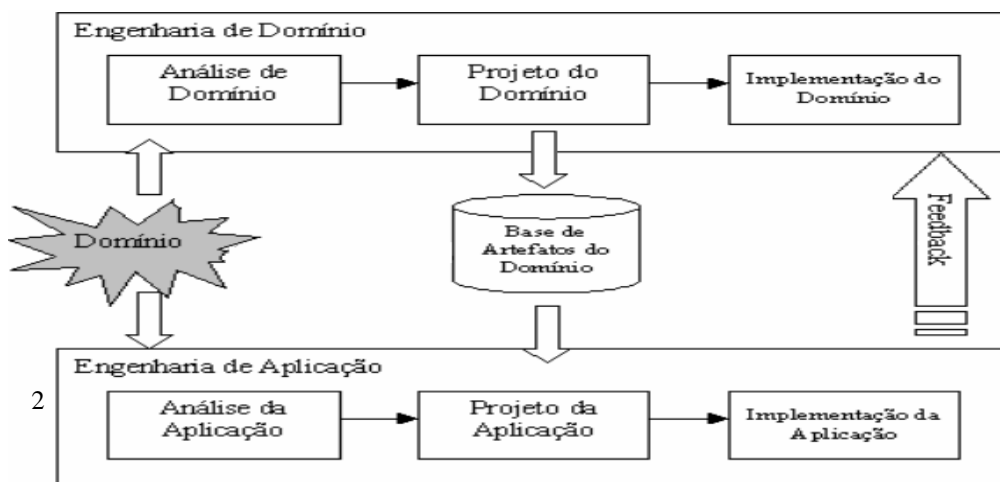


Figura 2.2 Ciclo de Vida na Engenharia de Domínio (OLIVEIRA, 2006)

Vários métodos de ED têm sido propostos na literatura tais como: FODA (KANG *et al.*, 1990), FORM (KANG *et al.*, 2002), FODACom (VICI e ARGENTIERI, 1998) dentre outros. Esses métodos apresentam sistemáticas para a modelagem de artefatos na ED, visando a reutilização na EA. Embora existam diferenças entre os métodos, eles são semelhantes quanto às fases envolvidas do processo (i.e. identificação do escopo do domínio, análise, projeto e implementação do domínio). Muitos deles prevêem atividades

em todas as fases, mas em geral apresentam maiores contribuições na fase de análise de domínio. No entanto, a etapa de projeto é muito importante, pois é a partir de seus artefatos e modelos que se geram aplicações no domínio modelado. Os artefatos gerados na Análise de Domínio servem para se fazer o recorte, i.e. selecionar características importantes do domínio para uma aplicação específica, mas são os artefatos de projeto, relacionados a essas características, que vão permitir a instanciação do domínio propriamente. Nesse contexto, este trabalho adota a abordagem de *frameworks* para suportar a etapa de projeto do domínio.

A ED defende que a reutilização de software deve ser aplicada desde o início do ciclo de vida para que o software alcance o sucesso, sendo que os artefatos produzidos são ao mesmo tempo específicos para o domínio em questão e genéricos o suficiente para atender toda a variedade de aplicações possíveis naquele domínio. Dessa forma, a ED propõe as seguintes fases ou macro-atividades que sistematizam a construção do modelo de domínio: (a) Planejamento, com a definição do escopo do domínio; (b) Análise de Domínio; (c) Projeto de Domínio; e (d) Implementação de Domínio. A Figura 2.2 mostra uma visão geral das fases da ED. A Análise de Domínio (AD) deve ter início assim que é concluído o seu planejamento, que envolve um estudo de viabilidade. Durante esta fase são originados alguns documentos onde estão documentados todos os conceitos encontrados no domínio, que servem como entrada para a fase de Projeto do Domínio, conforme apresenta a figura 2.3.

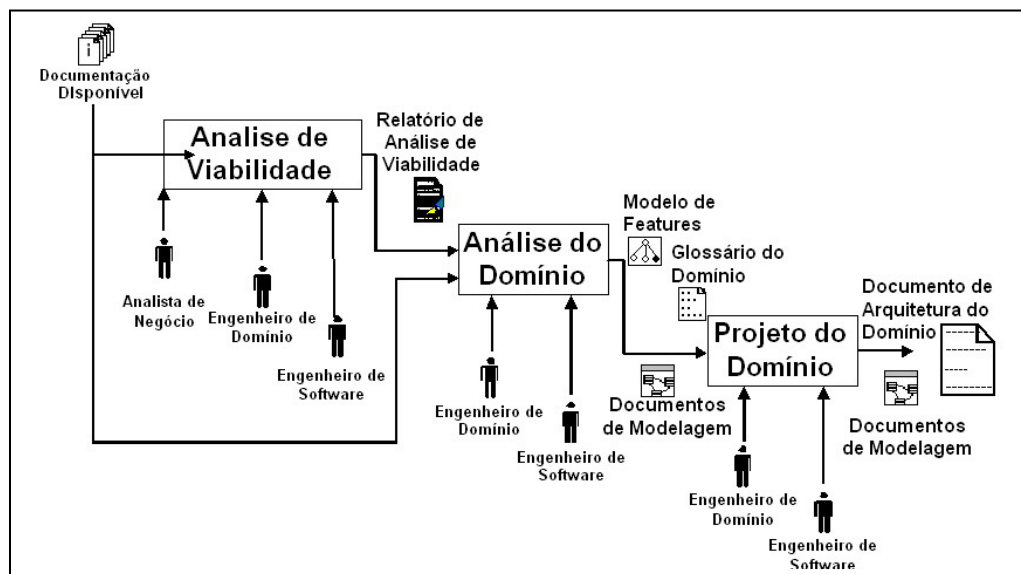


Figura 2.3 Fases da Engenharia de Domínio (UFPE, 2007)

Na ED, a fase de Planejamento é o início do processo que tem por objetivo fazer as estimativas de custos, cronogramas e, principalmente a Análise de Viabilidade do domínio. Quando o desenvolvimento de software é baseado em reutilização, antes que qualquer aplicação possa ser desenvolvida, um investimento inicial deve ser feito no desenvolvimento dos artefatos reutilizáveis. Neste contexto, para que não ocorram surpresas desagradáveis futuramente, é necessário que antes de tudo se faça uma Análise da Viabilidade do domínio.

Nesta etapa que se denomina Análise de Viabilidade do domínio, pode-se destacar, dentre várias atividades que são realizadas, o estudo de mercado do domínio, para buscar a existência de clientes e a presença de possíveis concorrentes, e a partir daí, se for constatado o potencial econômico, faz-se um estudo do domínio para identificar superficialmente suas variabilidades. A partir dessas informações, o próximo passo é verificar se a documentação existente sobre o domínio é suficiente para o seu estudo e para a extração de seus conceitos, ou seja, é verificada a maturidade do domínio. A maturidade do domínio (i.e. qualidade e estado de pleno desenvolvimento) indica o quanto de documentação, sistemas existentes, e informação disponível este possui, e além disso, indica se os requisitos em comum são estáveis para justificar o desenvolvimento para reutilização. Aqui também é definido o escopo do domínio, ou seja, a abrangência em relação aos requisitos no referido domínio. Partindo desde ponto, o próximo passo é estipular o esforço e os recursos necessários ao desenvolvimento e o tempo de retorno do investimento, sendo possível obter um conceito sobre a viabilidade do domínio.

Após o planejamento, constatada a viabilidade do domínio, inicia-se a Análise de Domínio (AD). O objetivo principal nesta fase é conhecer e entender os conceitos gerais do domínio e produzir modelos de requisitos que mostrem as semelhanças e diferenças do domínio, as quais podem ser representadas através de opcionalidades e variabilidades nos artefatos do domínio. A AD visa entre outras coisas, detectar e modelar variabilidades, isto é, um conjunto de características comuns e variáveis entre as aplicações do domínio (PRIETO-DIAZ *et al.*, 1991). O modelo comumente usado para modelagem de variabilidades e opcionalidades é o modelo de características, *features*. Segundo Kang *et al.* (1990), uma característica pode ser definida como uma qualidade, um aspecto ou uma característica visível ao usuário, proeminente ou distinta de um software.

A etapa de projeto de domínio é muito importante, pois, a partir de seus artefatos e modelos que se gera uma arquitetura de software que implementa soluções para os requisitos comuns do domínio. Ela visa, principalmente, a especificação de arquiteturas de software específicas de domínio (DSSAS-*Domain Specific Software Architectures*) com base nos requisitos, variabilidades e opcionalidades levantados na Análise de Domínio. Uma DSSA representa uma coleção de componentes especializados para um determinado tipo de tarefa (domínio), generalizados para que seja possível seu uso efetivo através do domínio e compostos em uma estrutura padronizada (topologia) efetiva para construção de aplicações (HAYES, 1994). Arquiteturas de referência são criadas para serem configuradas e estendidas durante o processo de instanciação de aplicações, representando um *framework* para o domínio.

Finalmente, a implementação do domínio transforma as oportunidades de reutilização e soluções do projeto em um modelo de implementação, que inclui serviços como: identificação, reengenharia e/ou construção, e manutenção de componentes reutilizáveis que suportem estes requisitos e soluções do projeto. (OLIVEIRA, ROCHA, GONÇALVES & SOUZA, 2007). Nesta etapa, os modelos de domínio e os artefatos reutilizáveis identificados nas fases de Análise e Projeto, são implementados na forma de componentes ou classes reutilizáveis. Pode-se realizar também a extração desses artefatos de sistemas legados existentes do domínio ou a busca desses componentes em bibliotecas de software, sendo então reaproveitados para construção de novas aplicações no mesmo domínio.

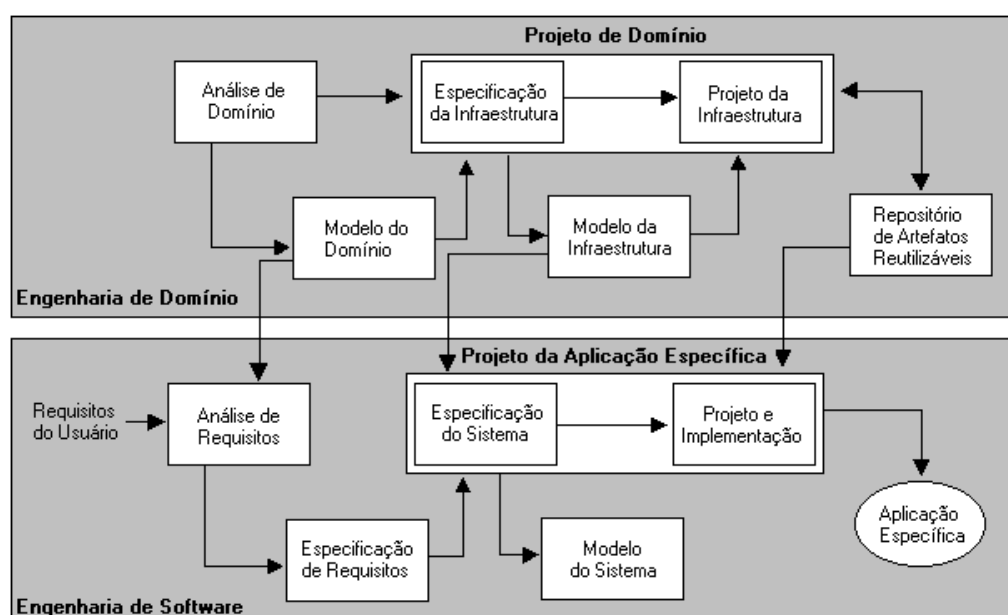


Figura 2.4 Um modelo genérico de desenvolvimento para / com reuso (GUIZZARDI, 2007)

Conforme mostra a figura 2.4, pode-se concluir que o objetivo da Engenharia de Domínio (ED) é, durante o processo de mudança do conhecimento de desenvolvedores acerca do domínio, criar um repositório de componentes, produzindo um conjunto de artefatos passíveis de reutilização na Engenharia de Aplicação, denominada Engenharia de Software na figura 2.4. Dessa forma, como mostrado na figura 2.4, um projeto de domínio da ED pode ser reutilizado para várias aplicações específicas na Engenharia de Software ou EA.

2.2.1 Análise de Domínio

A etapa de Análise de Domínio (AD) é a mais explorada deste trabalho, pois, ela é a base de uma Engenharia de Domínio (ED) de sucesso, nesta seção ela é mais detalhada. O estudo e o entendimento dos problemas e requisitos de um domínio é a principal atividade da AD. A Análise de Domínio (AD) envolve a especificação dos requisitos do domínio, assim como, a Análise de Requisitos de um software envolve a sua especificação. Segundo PRIETO-DIAZ e ARANGO (1991) a AD pode ser definida como processo de identificar e organizar o conhecimento a respeito de uma classe de problemas (i.e. o domínio do problema), de maneira a apoiar a descrição e solução de tais problemas. Certamente, isto é fundamental para que os artefatos próprios que serão gerados a partir desse estudo, possam de alguma forma ser reutilizados no desenvolvimento de outras aplicações que pertençam ao mesmo domínio.

Na AD, operações e objetos comuns, características, padrões e procedimentos de sistemas em um domínio são identificados e generalizados. A partir disso, um modelo é definido para servir como uma fonte unificada de definições para referência, um repositório de conhecimento compartilhado para a implementação de componentes reutilizáveis.

A Análise de Domínio (AD) deve considerar as especificações de uma família de sistemas (aplicações que compartilham características comuns) assim como antecipar as possíveis mudanças que possam ocorrer nestas especificações. Isto deve ser feito de tal forma que os modelos criados possam “gerar” sistemas alvos e ser capazes de evoluir suas especificações, além de acrescentar novas informações que possam surgir. A AD provê benefícios, tais como: entendimento dos conceitos envolvidos no projeto; criação de um

vocabulário comum entre os desenvolvedores de software, especialistas no assunto e usuários finais; melhor planejamento e conseqüentemente redução de custos; facilidade de modificações; identificação de características semelhantes entre produtos do mesmo domínio e, finalmente, reutilização.

Alguns métodos conhecidos e utilizados para especificar as variabilidades do domínio são: FODA (KANG *et al.*, 1990), FORM (KANG *et al.*, 2002), FODACom (VICI e ARGENTIERI, 1998), FeatuRSEB (GRISS *et al.*, 1998), ODM (SIMOS e ANTHONY, 1998), Odyssey-DE (BRAGA, 2000) e CBD-Arch-DE (BLOIS, 2006). O método CBD-Arch-DE é suportado pelo ambiente de reutilização Odyssey da COPPE/UFRJ (ODYSSEY, 2007) e emprega a notação Odyssey-FEX, que é adotada neste trabalho e será detalhada na seção a seguir.

O escopo do domínio deve ser definido para facilitar a Análise de Domínio, pois, sabendo o que se está analisando evita-se que se torne uma atividade interminável e sem um objetivo real e específico. Por outro lado, deve-se ter cuidado para que esse domínio não seja muito restringido a ponto de, ao final, não se gerar nenhum artefato útil as aplicações do domínio.

Com o escopo do domínio definido, devem-se extrair as suas características (*features*). A partir do modelo de características, devem ser derivados os demais modelos do domínio, com o objetivo de refletir os conceitos, funcionalidades e aspectos tecnológicos do domínio. Tanto os requisitos funcionais quanto os não-funcionais são analisados como *features*, as quais representam as partes comuns e variáveis do sistema. O modelo de características é o modelo de mais alto nível de abstração na Engenharia de Domínio (ED) e identifica o que é opcional ou mandatório em um domínio, e também o que pode variar (OLIVEIRA, 2006). Em termos gerais a AD pode ser vista como um processo que se concentra nos seguintes passos:

- **Definir o domínio a ser investigado:** para que seja possível aplicar a ED a um projeto de software, é necessário que seja definido o domínio onde esse software atuará assim como seu escopo.
- **Categorizar os itens extraídos do domínio:** após definir o domínio, deve-se extrair suas características e agrupá-las para que possam ser entendidas.
- **Coletar uma amostra representativa das aplicações do domínio:** verificar se existem aplicações referentes ao domínio, para que elas possam

ser analisadas, fazendo com que sejam descobertas mais características do domínio.

- **Analisar cada aplicação da amostra:** analisar as aplicações, visando o domínio como um todo, possibilitando identificar os pontos de variação existentes, i.e. as características específicas a cada aplicação e as características em comum.
- **Desenvolver um modelo de análise para as características extraídas:** após coletar todas as informações necessárias, deve-se gerar um modelo de domínio que seja coerente.

A seção a seguir descreve a importância dos ambientes de reutilização para apoiar o desenvolvimento baseado em reutilização, destacando o Odyssey.

2.3 AMBIENTE DE REUTILIZAÇÃO DE SOFTWARE ODYSSEY

A qualidade dos artefatos produzidos durante as fases iniciais de um processo de desenvolvimento de software é fundamental para o sucesso no decorrer do projeto e para manutenção de sua viabilidade. Muitos problemas e falhas tais como: a falta de conhecimento e requisitos inconsistentes surgem em um projeto de software em virtude das dificuldades das fases iniciais do desenvolvimento. Incertezas e erros podem ocorrer em função da má comunicação ou falta de especialista nas informações do domínio do problema sendo analisado.

A reutilização de software se aplicada de forma efetiva pode-se atingir melhor produtividade no desenvolvimento, destacando a construção de novos sistemas a partir do uso de qualquer artefato já produzido e utilizado em soluções de problemas similares. A fim de que a reutilização de software possa se tornar efetiva, uma vez que envolve grande esforço, ela deve ser realizada com o apoio de um ambiente de desenvolvimento de software baseado em reutilização (MOURA, CARVALHO e SILVA, 2006). Não apenas um ambiente que permita acompanhar a evolução dos modelos desenvolvidos, mas também que apóie a própria especificação dos modelos de análise e projeto do domínio e geração do código, além da reutilização dos artefatos desenvolvidos através da engenharia de aplicação. Uma infra-estrutura de suporte à reutilização baseada em modelo de domínio pode ajudar na utilização efetiva da reutilização durante o desenvolvimento de software,

fornecendo métodos, ferramentas e procedimentos adequados para a especificação de modelos e aplicações do domínio.

Um ambiente de desenvolvimento baseado em reutilização, disponível para utilização e desenvolvimento em nível nacional é o ODYSSEY (ODYSSEY, 2007). Trata-se de um protótipo acadêmico desenvolvido pela equipe de reutilização da COPPE/UFRJ. O Odyssey é um ambiente de suporte ao desenvolvimento baseado em reutilização a partir de modelos de domínio que pode auxiliar na utilização efetiva de uma estratégia de reutilização durante todo o processo de desenvolvimento. O principal objetivo do Odyssey é prover mecanismos baseados em reutilização para o desenvolvimento de software, servindo como um arcabouço onde modelos conceituais, arquiteturas de software, padrões de projeto e modelos de implementação são especificados para domínios de aplicação previamente selecionados. A distribuição Odyssey é um ambiente de reutilização de software centrado na Engenharia de Domínio e no Desenvolvimento Baseado em Componentes. Oferece ao Engenheiro de Domínio e ao Engenheiro de Aplicação um conjunto de serviços para criação e instanciação de modelos de domínio.

O Odyssey oferece as seguintes funcionalidades para apoio à reutilização:

- Engenharia Reversa dinâmica e estática (VASCONCELOS, 2007).
- Desenvolvimento de aplicações voltados para a reutilização, através de processos de Engenharia de Domínio (ED), permitindo o desenvolvimento de modelos reutilizáveis do domínio (BERNADETE, GONÇALVES e CHAGAS, 2007).
- Desenvolvimento de aplicações com reutilização através da Engenharia de Aplicação (EA), permitindo a instanciação de aplicações a partir dos modelos de domínio, onde é feito o recorte do domínio, determinando-se as características que devem compor as aplicações. Deve-se fazer uma ressalva para as características obrigatórias do domínio, que devem ser sempre selecionadas.
- Mapeamento de ligações de rastreabilidade entre modelos em diferentes níveis de abstração, como mostra a figura 2.5, permitindo a reutilização dos diferentes modelos de domínio que devem compor as aplicações (BLOIS, 2006).

- Instanciação de padrões de projetos e arquiteturas em modelos de domínio e de aplicações.
- Geração de Componentes do domínio no processo CDB-Arch-DE (BLOIS, 2006).

O ambiente Odyssey encontra-se disponível para *download* em sua versão *ligh*, na página <http://reuse.cos.ufrj.br/odyssey>. As funcionalidades essenciais que compõem o núcleo (*kernel*) do ambiente, sendo baixadas com ele, e as funcionalidades secundárias estão disponíveis na forma de *plugins*, que podem ser baixados de dentro do próprio Odyssey sob demanda.

Em função das suas funcionalidades, o ambiente Odyssey é adotado neste trabalho para apoiar o processo de Engenharia de Domínio.

O Odyssey implementa a notação Odyssey-FEX (OLIVEIRA, 2006) para a modelagem de características do domínio e suas variabilidades, a qual é apresentada a seguir.

2.3.1 A notação Odyssey-Fex e a Variabilidade na Engenharia de Domínio

A notação Odyssey-FEX foi elaborada a partir do refinamento da notação proposta por MILLER (2000), que vem a ser uma extensão do método FODA de (KANG *et al.*, 1990). Ela permite a representação de variabilidades no modelo de características do domínio. O modelo de características da Odyssey-FEX, incorpora além dos relacionamentos triviais entre características, relacionamentos da UML.

Para expressar variabilidades na ED, é necessário modelar explicitamente todos os conceitos inerentes, permitindo o entendimento entre todos os envolvidos no processo (i.e. usuário, especialistas de domínio) (MASSEN *et al.*, 2002). Para tanto, o modelo de características costuma ser o ponto de partida, direcionando o recorte de aplicações na Engenharia de Aplicação (VASCONCELOS, 2005). Diversas definições são atribuídas ao termo características (*features*). No presente trabalho, será adotada a definição de Kang *et al.* (1990), uma das principais referências em modelos de características para a ED. Segundo Kang *et al.* (1990), uma característica pode ser definida como “um aspecto, uma qualidade, ou uma característica visível ao usuário, proeminente ou distinta, de um sistema

(ou sistemas) de software”. Ainda de acordo com Kang *et al.* (1990), um modelo de características tem como propósito capturar o entendimento de clientes e usuários acerca das capacidades gerais de uma aplicação em um domínio. Representa as características-padrão de uma família de sistemas em um domínio e as relações entre elas.

Conceitos de modelagem de variabilidades são encontrados na literatura (KANG *et al.*, 1990) (MASSEN *et al.*, 2002) (KANG *et al.*, 2002) e compreendem: características mandatórias, que devem estar presentes nas aplicações do domínio; características opcionais, que podem aparecer em algumas aplicações do domínio; e pontos de variação com suas variantes, que definem os pontos onde as aplicações do domínio podem ser configuradas. Além desses conceitos, as regras de composição entre características também devem ser representadas, indicando características dependentes e mutuamente exclusivas.

Algumas categorias de características da notação Odyssey-FEX são representadas na tabela 2.1.


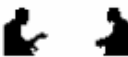



Ícone	Tipo de Característica	
	Características de Domínio – Características intimamente ligadas à essência do domínio. Representam as funcionalidades e/ou os conceitos do modelo e correspondem a casos de uso e componentes estruturais concretos.	Características de Análise
	Características de Entidade – São os atores do modelo. Entidades do mundo real que atuam sobre o domínio. Podem, por exemplo, expor a necessidade de uma interface com o usuário ou de procedimentos de controle.	
	Características de Ambiente Operacional – Características que representam atributos de um ambiente que uma aplicação do domínio pode usar e operar. Ex: tipo de terminal, sistemas operacionais, bibliotecas etc.	Características de Projeto (Tecnológicas)
	Características de Tecnologia de Domínio – Características que representam tecnologias utilizadas para modelar ou implementar questões específicas de um domínio. Ex: métodos de navegação em um domínio de aviões.	
	Características de Técnica de Implementação – Características que representam tecnologias utilizadas para implementar outras características, podendo ser compartilhadas por diversos domínios. Ex: técnicas de sincronização.	

Tabela 2.1 Tipos de Características na notação Odyssey-FEX (Oliveira, 2006)

Em relação aos relacionamentos possíveis entre características, segundo OLIVEIRA (2006), na notação Odyssey-FEX relacionamentos são utilizados para ajudar a

expressar o domínio, muitos deles importados da UML (OMG, 2005) como composição, agregação, generalização e associação. A tabela 2.2 demonstra os relacionamentos específicos na notação Odyssey-FEX.






<u>Representação</u>	<u>Descrição</u>	
	Composição – Relacionamento em que uma característica é composta de várias outras. Denota relação na qual uma característica é parte fundamental de outra, de forma que a primeira não existe sem a segunda.	Relacionamentos da UML
	Agregação – Relacionamento em que uma característica representa o todo, e as outras as partes. Similar à composição, porém as características envolvidas existem independentemente uma da outra.	
	Generalização – Relacionamento em que há uma generalização/especialização das características. Este tipo de relacionamento indica que as características mais especializadas (filhas) herdam as propriedades e atributos de características mais generalizadas (antecessores).	
	Associação – Relacionamento simples entre duas características. Denota algum tipo de ligação entre seus membros. Pode ser nomeada, indicando um tipo específico de ligação.	
	Alternativo (<i>Alternative</i>) - Relacionamento entre um ponto de variação e suas variantes, denota a pertinência de uma variante a um determinado ponto de variação.	Relacionamentos Específicos na Odyssey-FEX
<u><<Implemented By>></u>	Implementado por (<i>Implemented By</i>) - Relacionamento entre Características de Domínio e Características Tecnológicas, ou entre Características Tecnológicas que se encontrem em camadas diferentes.	
<u><<Communication Link>></u>	Ligação de Comunicação (<i>Communication Link</i>) - Relacionamento existente entre Características de Entidade e Características de Domínio. Cumpre o mesmo papel do relacionamento de associação entre atores e casos de uso na UML (OMG, 2004)	

Tabela 2.2 Relacionamentos da notação Odyssey-FEX

As características devem ser mapeadas para os demais artefatos do domínio, como classes e casos de uso, e as características tecnológicas mapeadas para artefatos da fase de projeto de domínio, como classes ou componentes utilitários e de infra-estrutura. Ainda

podemos subdividir as características de domínio em funcionais e conceituais. As características funcionais estão ligadas aos casos de uso e as conceituais aos conceitos ou classes de um domínio. Como exemplo, é apresentado na figura 2.5, um exemplo desse modelo referente a um domínio de cadastro de turmas e consulta de disponibilidades de horários de um determinado professor no ambiente Odyssey.

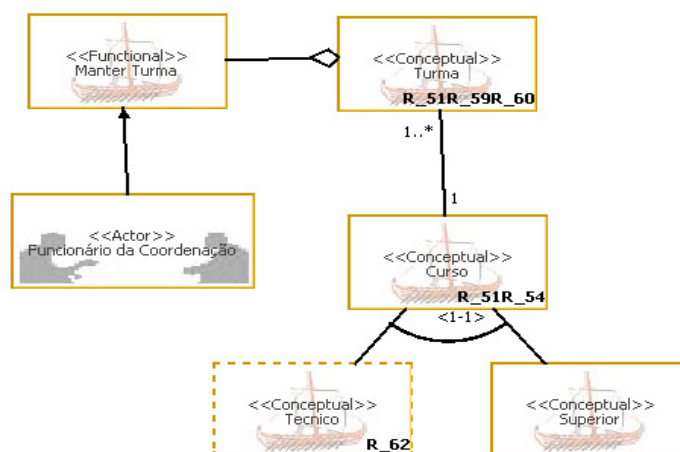


Figura 2.5 Características conceituais e funcionais na notação Odyssey-FEX

Na figura 2.5, **O Funcionário da Coordenação** representa uma Característica de Entidade, ou seja, um usuário do domínio. **Manter Turma** é uma Característica de Domínio funcional que dará origem a caso de uso. Por outro lado, **Turma**, **Curso**, **Técnico** e **Superior** são Características de Domínio conceituais que darão origem as classes ou atributos do domínio. Na notação Odyssey-FEX as características mandatórias têm contorno cheio e as características opcionais têm contorno tracejado. Na figura 2.5, as características **Turma**, **Manter Turma**, **Curso** e **Superior** são características mandatórias. Os **Elementos Mandatórios** descrevem elementos que devem obrigatoriamente estar presentes em produtos desenvolvidos em uma LP ou aplicações instanciadas a partir de um domínio. A característica **Técnico** é uma característica opcional. Os **Elementos Opcionais** descrevem elementos que podem ou não estar presentes em produtos desenvolvidos em uma LP ou aplicações instanciadas a partir de um domínio. Observa-se também neste exemplo que Curso é um **ponto de variação** tendo como **variantes** Técnico e Superior. Esses conceitos são explorados na tabela 2.3.

Características	Definição
Pontos de Variação	Determinados pontos em um sistema de software onde decisões são tomadas a respeito, por exemplo, de qual variante será utilizada. Em outras palavras, pontos de variação refletem a parametrização no domínio de uma maneira abstrata e são configuráveis através das variantes.
Variantes	Alternativas de implementação disponíveis para um ponto de variação são denominados Variantes. Em outras palavras, são elementos necessariamente ligados a um ponto de variação, que atuam como alternativas para se configurar aquele ponto de variação.
Invariantes	Os elementos “fixos”, que não são configuráveis no domínio.

Tabela 2.3 Tipos de Características no ambiente Odyssey

Através deste princípio pode-se verificar que essa classificação é de extrema importância para o desenvolvimento de um projeto baseado em reutilização, pois através dele se identifica quais pontos podem ser configuráveis na aplicação, nas especificações em seu domínio.

A combinação dessas classificações influencia não só o recorte no processo de Engenharia de Aplicação, como a própria construção dos artefatos no processo de ED, incluindo a criação de componentes (OLIVEIRA, BLOIS, VASCONCELOS e WERNER, 2005). Uma característica na notação Odyssey-FEX pode possuir propriedades ortogonais de variabilidade e opcionalidade como demonstra a figura 2.6.

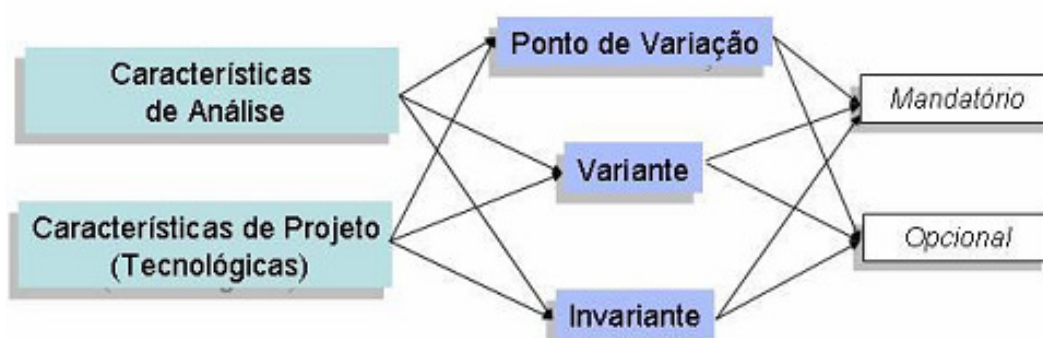


Figura 2.6 Propriedades ortogonais da notação Odyssey-FEX

As variabilidades podem ser rastreadas através de um modelo de rastreamento de variabilidades, onde este toma como base o modelo utilizado para rastrear *features* e casos de uso, em que se definem quais são os casos de uso que estão relacionados à quais *features* e vice-versa. O mesmo vale para as *features* conceituais relacionadas às classes. O Odyssey permite a realização de mapeamentos entre esses modelos.

2.4 FRAMEWORKS

A reutilização de software, conforme já mencionado, deve ser utilizada com o propósito de facilitar o trabalho do desenvolvedor de aplicações. Isso se dá, por exemplo, através de um projeto de software genérico, proporcionando uma melhoria na qualidade do sistema, através da reutilização constante do projeto e redução no tempo de desenvolvimento. Com isso, o desenvolvedor que cria aplicações se concentra nos aspectos específicos da sua aplicação. Os *frameworks* têm esse objetivo de fornecer soluções genéricas e reutilizáveis para variados problemas de projeto de software.

Um *framework* é um conjunto de classes cooperantes que constroem um projeto reutilizável para uma classe específica de software (GAMMA *et al.*, 2000). Essa especificação pode ser utilizada para a construção de diferentes domínios, definindo a estrutura geral, divisão de classes e objetos e suas responsabilidades. Os *frameworks* definem a arquitetura do domínio na fase de Projeto do Domínio.

De acordo com GAMMA *et al.*, (2000), os *frameworks* estão se tornando cada vez mais comuns e importantes. Eles representam a maneira pela qual os sistemas orientados a objetos conseguem maior reutilização. Com *frameworks*, é possível criar um software mais rapidamente. Porém, isso reduz a possibilidade de decisões estruturais, pois quase toda a infra-estrutura já foi definida e validada. Além disso, é importante perceber que não se deve criar um *framework* muito específico para abordar somente um determinado problema de uma aplicação em particular, mas sim visando um domínio como um todo (MOURA, CARVALHO e SILVA, 2006). A arquitetura deve funcionar para todas as aplicações do domínio e conforme uma aplicação evolui, o *framework* também deve evoluir. Isso mostra que o processo para construir um *framework*, assim como todo processo de software voltado à reutilização, é muito complexo, pois ele não deve ser

encarado como uma aplicação tradicional, e sim permitir uma variação de suas características.

Dentro de qualquer domínio, existem as características (i.e. *features*) que permitem identificar suas funcionalidades e conceitos, assim como as suas variabilidades e opcionalidades. As características comuns a todas as aplicações do domínio e invariáveis dão origem aos *frozen-spots* (pontos fixos) dos *frameworks*. Esses pontos fixos são representados por características invariantes e mandatórias e estão definidos no *framework* de forma inalterável para todas as aplicações do domínio, por exemplo, através de classes concretas. Existem também os *hot-spots* (*pontos variáveis*) que representam as características variáveis nas diferentes aplicações de um domínio, ou seja, os pontos de variação, variantes e características opcionais. Essas características devem ser genéricas, pois devem ser adaptadas às necessidades da aplicação, sendo representadas, por exemplo, através de classes abstratas, métodos abstratos, métodos *template* e outros padrões de projeto do GAMMA *et al*, (2000).

2.4.1 Tipos de Frameworks

De acordo com a forma de reutilização, os *frameworks* podem ser classificados, como: caixa branca (*white box*); caixa preta (*black box*) ou caixa cinza (*gray box*). Nos *frameworks* caixa branca, o reuso acontece por herança ou extensão, sendo necessário conhecer os detalhes de como *framework* trabalha. É importante que o desenvolvedor domine o funcionamento, para, a partir disso, criar novas classes que deverão implementar ou estender as classes abstratas. O *framework* caixa preta não exige que o desenvolvedor conheça os detalhes da implementação do *framework*, exigindo apenas que a classe use a composição de objetos e delegação em vez de herança. Já nos *frameworks* caixa cinza, existe uma mistura dos outros dois tipos de *frameworks*, caixa branca e caixa preta, ou seja o reuso pode ocorrer por herança e composição simultaneamente.

Os *frameworks* podem ainda ser classificados de acordo com o seu escopo, da seguinte forma:

- **Frameworks de Infra-Estrutura:** apóiam a infra-estrutura de sistemas em diferentes domínios de aplicação, oferecendo serviços como persistência, segurança, busca, interfaces gráficas etc.

- **Frameworks de Integração:** são utilizados para integrar aplicações e componentes distribuídos (ex: *middleware*, como Corba).
- **Frameworks de Aplicação:** são dirigidos a um domínio específico de aplicações, ou seja, a uma família de aplicações em uma determinada área.

De acordo com as descrições encontradas em (SCHMID, 1997), que considera o uso de *frameworks* como técnica de reutilização de software, os pontos de variação dão origem aos *hot-spots* no Projeto do Domínio. Tais *hot-spots* podem ter diferentes alternativas de configuração, que distinguem as aplicações que serão construídas a partir do *framework*.

2.5 CONSIDERAÇÕES FINAIS

Este capítulo abordou as técnicas de reutilização de software, destacando a importância desta sub-área na Engenharia de Software. Ele enfatizou também a abordagem de Engenharia de Domínio, estabelecendo um paralelo desta com a abordagem de Linha de Produtos e Engenharia de Aplicação, onde ambas compartilham características comuns e gerenciáveis que satisfazem a uma determinada necessidade. A Engenharia de Domínio, uma das técnicas apresentadas, cria um modelo de domínio de aplicação, que é utilizado como base para a instanciação de aplicações específicas no domínio através da Engenharia de Aplicação. A Análise de Domínio é uma das principais atividades da ED, pois permite estabelecer o escopo e requisitos do domínio, bem como as suas variabilidades e opcionalidades. As aplicações do domínio são instanciadas com base nessas variabilidades e opcionalidades do domínio.

Durante o projeto de domínio, uma arquitetura de software genérica que atenda aos requisitos do domínio é desenvolvida. Ela fornece a entrada para o projeto das aplicações, podendo ser representado como um *framework* do domínio. As aplicações devem estender e adaptar os *hot-spots*, i.e. os pontos configuráveis do domínio, durante a instanciação do *framework*.

De modo geral, a principal motivação para a reutilização está relacionada ao aumento dos níveis de qualidade e produtividade no desenvolvimento de software. O

aumento da qualidade é uma consequência da reutilização de artefatos que foram previamente documentados, testados e aprovados em outra situação. O aumento da produtividade é resultado de uma redução de tempo de desenvolvimento, evitando a reconstrução de partes do sistema que já existem.

Aplicando as técnicas de Engenharia de Domínio (ED) apresentados neste capítulo a partir da Análise de Domínio (AD), utilizando a notação Odyssey-FEX, esse trabalho tem o objetivo de modelar o *framework* com base nas três Coordenações de Curso onde esse domínio foi analisado para a Elaboração de Horário Escolar do IFF, mas precisamente, para a Elaboração de Horário de Turmas. Tal *framework* será analisado no Capítulo 3, seguindo os princípios de reutilização de software apresentados neste capítulo.

CAPÍTULO 3: MODELAGEM DE UM FRAMEWORK PARA O DOMÍNIO DE HORÁRIO ESCOLAR

3.1 INTRODUÇÃO

Neste capítulo, as técnicas de reutilização apresentadas no Capítulo 2 são aplicadas para a especificação do domínio que envolve a Elaboração de Horários Escolares do IFF. A Elaboração de Horários Escolares, como já mencionado, é um trabalho exercido pelas Coordenações ou Gerências de Cursos, que tem por objetivo informar a Coordenação de Curso, Coordenação de Turno, Professores e Turmas seus respectivos horários. Para professores e turmas podemos considerar as seguintes informações: horários de aula, dia da semana e local (i.e. ambiente de aprendizagem). Para a Coordenação de Turno, essas informações podem ser consideradas como: os locais onde irão ocorrer aulas de uma determinada turma em um determinado dia da semana e horário. Para a Coordenação de Cursos e Gerências essas informações podem ser consideradas como: horários de professores, turmas e ambientes de aprendizagens em um dia da semana em um determinado horário.

O ambiente de reutilização Odyssey (ODYSSEY, 2007) é utilizado para apoiar o processo. Esse ambiente aplica a notação Odyssey-FEX (OLIVEIRA, 2006) para modelagem das opcionalidades e variabilidades do domínio, apresentada no Capítulo 2, que define a semântica dos elementos funcionais, conceituais e tecnológicos utilizados na modelagem de domínio, incluindo seus relacionamentos.

O capítulo está organizado da seguinte forma: a Seção 3.2 apresenta a Análise de Domínio realizada para a determinação dos conceitos e funcionalidades inerentes à Elaboração de Horário Escolar do IFF; na Seção 3.2.2 são apresentados os contextos ou subdomínios com seus respectivos atores; as características dos subdomínios Controle de Curso e sua Matriz Curricular; Disponibilidade e Competência de Professores, Controle de Ambientes de Aprendizagem, e Controle de Horário são apresentadas respectivamente nas Seções: 3.2.3; 3.2.4; 3.2.5 e 3.2.6; a Seção 3.3 apresenta o projeto do domínio, ou seja, a arquitetura completa ou *framework*, que é apresentado através do diagrama de classes; a Seção 3.4 apresenta o mapeamento do modelo de características para o modelo de casos de uso; a Seção 3.5 aborda os conceitos da Engenharia de Aplicação no contexto da

reutilização, tendo como exemplo uma possível instanciação do *framework* para uma outra aplicação específica neste domínio; por fim, a Seção 3.6 apresenta as considerações finais deste capítulo.

3.2 ANÁLISE DE DOMÍNIO

Nesse ponto do trabalho, é apresentado o domínio analisado através de descrição textual e modelos.

Um dos pontos inerentes à Elaboração de Horários Escolares é a distribuição adequada de quadros de horários. Esta distribuição representa frequentemente um desafio na vida escolar, envolvendo tarefas como: ajustar aulas, negociar disponibilidades de professores e negociar disponibilidade de ambientes de aprendizagem. A formulação de um quadro de horários que atenda aos interesses de professores e alunos e utiliza da melhor forma possível os recursos disponíveis, é uma questão primordial na administração escolar, em qualquer nível de ensino. Trata-se, assim de um problema clássico, que historicamente vem sendo solucionado ou de forma artesanal ou através de soluções impéricas, típicas apenas da própria instituição que as desenvolveu. A solução artesanal desse tipo de problema é restringida principalmente pelo crescimento das próprias instituições.

Segundo LOBO (2005) o problema de quadro de Horários Escolares, e suas várias formulações, despertou o interesse da comunidade acadêmica a partir do início da década de sessenta. Em trabalhos desta época os autores enunciam técnicas para construção de solução para o problema por meio do emprego de computadores e delineavam a comparação entre o problema de quadro de horários.

Em 1962, Gotlieb apresentou a primeira formulação completa para o problema do quadro de horários para cursos (*Timetabling Problem-Course Scheduling*), ao declarar que o problema consistia em fixar, num determinado período de tempo conhecido, um conjunto de aulas, de forma a atender as exigências acadêmicas estabelecidas por certo currículo de estudos para certo grupo de disciplinas. Cada aula visava o atendimento de um grupo de estudantes, ou seja, turma para certa disciplina e exigia o comparecimento de um único professor à mesma, sendo que o professor para ser escolhido deveria estar disponível para as aulas que fossem agendadas a partir de sua competência. A partir daí, seguiram-se outras propostas de solução para o problema, onde estas propostas propuseram métodos para a automatização da geração de quadro de horários utilizando computadores como

ferramenta para a proposição de soluções práticas, muitos destes são algoritmos genéticos para soluções deste problema.

Uma vez que o domínio que envolve a Elaboração de Horário Escolar é bastante amplo e complexo, este trabalho optou pela modelagem de um *framework* de horário escolar que atenda as atividades envolvidas neste domínio. Isso foi proporcionado pela Análise de Viabilidade realizada nesta etapa, onde se verificou a maturidade do domínio (i.e. qualidade e estado de pleno desenvolvimento de sistemas existentes e documentação disponível). Vários sistemas existentes no domínio encontrados preocupam-se com o desenvolvimento de aplicações deste domínio para casos particulares, daí a proposta de um modelo que possa ser reutilizado em diferentes aplicações para o domínio modelado.

O *framework* modelado para o domínio de Horário Escolar neste trabalho é um *framework* de aplicação, ou seja, ele é dirigido a um domínio específico de aplicações pertencentes a este domínio. E é também um *framework* caixa branca, onde o reuso acontece por herança ou extensão, sendo necessário conhecer os detalhes de como *framework* trabalha.

Para o desenvolvimento deste *framework*, como mencionado anteriormente, três coordenações foram contempladas, sendo estas: a Coordenação de Informática, Coordenação de Geografia e a Coordenação de Telecomunicações. Porém, uma vez o *framework* modelado, este pode ser estendido para atender a toda a Instituição. A metodologia utilizada para conhecer e identificar as atividades comuns e variáveis na Elaboração de Horário Escolar destas Coordenações de Curso foram as entrevistas com os funcionários responsáveis por realizar esta atividade em cada Coordenação. Através destas entrevistas foram obtidas todas as informações acerca de atividades e conceitos relacionados ao domínio de Elaboração de Horário Escolar. As entrevistas realizadas encontram-se no Anexo C, no final deste trabalho. Para cada entrevista realizada nas Coordenações de Curso foram elaborados modelos de características na notação Odyssey-FEX de acordo com os requisitos levantados em cada uma. Posteriormente, os modelos de características implementados no Odyssey foram explorados através de mapeamentos. O modelo de característica funcional foi mapeado para caso de uso e o modelo de característica conceitual foi mapeado para classe, esses mapeamentos são oferecidos pelo Odyssey.

Com base nas entrevistas realizadas, o *framework* foi construído a partir da consolidação dos modelos das três Coordenações entrevistadas. Este *framework* modelado

é apresentado nas seções a seguir de acordo com a notação Odyssey-FEX, onde é possível verificar as variabilidades e opcionalidades deste domínio. Estas variabilidades serão apresentadas em subdomínios onde para cada um destes, são apresentadas suas características, relacionamentos, opcionalidades e variabilidades. O diagrama de classes foi mapeado a partir das características conceituais, enquanto o modelo de caso de uso foi mapeado a partir das características funcionais, ambos na notação Odyssey-FEX . Esses mapeamentos serão mostrados nas seções a seguir.

O mapeamento é realizado pelo ambiente Odyssey, seguindo as regras propostas por OLIVEIRA (2006), que realiza a consistência inter-modelos para possibilitar uma representação coerente dos requisitos de domínio, suas opcionalidades e variabilidades, em diferentes níveis de abstração. Dessa forma, através da proposta da Odyssey-FEX, o domínio da elaboração de Horário Escolar é modelado no ambiente de reutilização Odyssey, onde toda a análise de requisitos é representada, mantendo a opcionalidade e variabilidade das características nos diferentes níveis de abstração do domínio, i.e. classes, casos de uso, características e, futuramente, componentes.

3.2.1 Descrição do Domínio

A gestão de Elaboração de Horário Escolar envolve alguns setores de uma Instituição de Ensino tais como: Gerência de Cursos, Coordenações de Cursos, e Coordenações de Turnos, onde estas têm como um de seus objetivos, administrar a Elaboração de Horário Escolar, ou seja, controlar os horários de turmas, professores e ambientes de aprendizagem. Para esse controle, outras atividades também devem ser gerenciadas tais como: controle de ambientes, controle de cursos e sua respectiva matriz curricular, controle de disponibilidades de professores e controle de horário geral (i.e. horário de ambientes, horário de professores e horário de turma). Essa gestão é essencial dentro de uma Instituição de Ensino.

O domínio de Elaboração de Horário Escolar requer adaptações gerenciadas para atender necessidades de cada Coordenação de Curso. Novos problemas organizacionais e gerenciáveis devem ser tratados para cada caso particular. Dessa forma, apresenta requisitos em comum e requisitos que variam de uma aplicação para a outra, justificando a realização de uma Análise de Domínio. Além disso, trata-se de um domínio abrangente,

pois envolve subdomínios, como: Controle de Horário, Disponibilidades e Competências de Professores, Controle de Cursos e sua Matriz Curricular e Controle de Ambientes de Aprendizagem disponíveis para aulas. Para cada subdomínio, devem ser observados os aspectos gerenciáveis que norteiam o seu funcionamento. Os subdomínios da Elaboração de Horário Escolar do IFF podem ser definidos da seguinte forma:

- **Controle de Horário:** envolve a administração e montagem de horário de turmas, professores e ambientes de aprendizagem. Este envolve as cargas horárias das disciplinas de uma turma e seus horários de aula envolvendo também a carga horária de professores e seus horários de aula e os horários de aula de cada ambiente de aprendizagem (i.e. laboratório, multimídia e sala). Trata-se de uma emissão de horários, controlando os horários de turmas, professores e ambientes de aprendizagem.
- **Disponibilidades e Competência de Professores:** envolve o gerenciamento dos professores pertencentes a um curso. Devem ser tratadas suas disponibilidades para aula em um determinado dia na semana e quais as disciplinas no qual ele está apto a lecionar (ou seja, suas competências).
- **Controle de Cursos e sua Matriz Curricular:** envolve o gerenciamento dos cursos e suas respectivas turmas. Para cada curso deve ser tratada a sua matriz curricular. As turmas dos Cursos Superiores são associadas a uma matriz Curricular que é organizada em período, as turmas dos Cursos Técnicos são organizados em módulos.
- **Controle de Ambientes de Aprendizagem:** envolve o gerenciamento dos ambientes de aprendizagem disponíveis para aulas e seus recursos. Cada coordenação possui ambientes de aprendizagem próprios para disponibilizar estes para aulas. Alguns ambientes são controlados pela Coordenação de Turno, onde este tem a função de gerenciar os horários destes ambientes. Normalmente os ambientes de aprendizagem possuem recursos disponíveis de acordo com a necessidade de cada aula. Um exemplo são os laboratórios, que necessita de softwares específicos que serão usados em uma determinada aula.

A Análise de Domínio e o desenvolvimento deste trabalho estão baseados apenas nas características dos subdomínios citados anteriormente necessários à elaboração de horários de turmas.

3.2.2 Contextos

Conforme mencionado, o domínio de Horário Escolar está dividido nos seguintes subdomínios: Controle de Horário, Disponibilidades e Competência de Professores, Controle de Cursos e sua Matriz Curricular e Controle de Ambiente de Aprendizagem. Na figura 3.1, são apresentados os subdomínios no ambiente Odyssey, que são apresentadas em um diagrama através de contextos (OLIVEIRA, 2006).

Através deste diagrama, observam-se as ligações dos atores, i.e. usuários do domínio, com seus respectivos subdomínios. Os atores deste domínio são: Funcionário da Coordenação, Coordenador de Curso, Coordenador de Turno e Professor. Na notação Odyssey-FEX, atores são denominados entidades. Abaixo segue uma descrição sucinta dos atores e suas respectivas atribuições.

- ***Funcionário da Coordenação:***

Com base nas entrevistas realizadas, nem sempre o Coordenador é o responsável pela elaboração dos horários. Das três coordenações entrevistadas a atividade de elaborar os horários é exercida por um Funcionário que pertence a essa Coordenação. Esse profissional tem a função de elaborar e organizar os horários, para isso ele precisa do conhecimento de alguns elementos que são essenciais para essa elaboração, como: os ambientes de aprendizagem, professores, turmas, cursos e sua matriz curricular.

- ***Coordenador de Curso:***

Responsável pela gestão acadêmica de um curso, sendo uma de suas atividades, o controle e organização da matriz curricular de um curso e de uma turma. Dentro deste contexto ele é o responsável por manter a grade curricular de um curso sempre atualizada, realizando alterações sempre que precisar para atender os requisitos deste curso.

- **Coordenador de Turno:**

Este profissional é responsável por controlar os ambientes de aprendizagem disponíveis para atender a demanda de aulas de diversas turmas. Ele é responsável por fornecer a cada Coordenação quais os ambientes de aprendizagem que esta poderá utilizar para dispor as suas aulas.

- **Professor:**

Este profissional é um docente pertencente a um determinado curso que está apto a lecionar em uma ou mais turmas. Suas aulas ministradas ocorrem em um dia em um determinado horário. Este horário de aula na qual ele ministra é resultado de uma negociação de suas disponibilidades para ministrar que é feita entre o responsável pela Elaboração de Horário Escolar ou com o Coordenador de Curso.

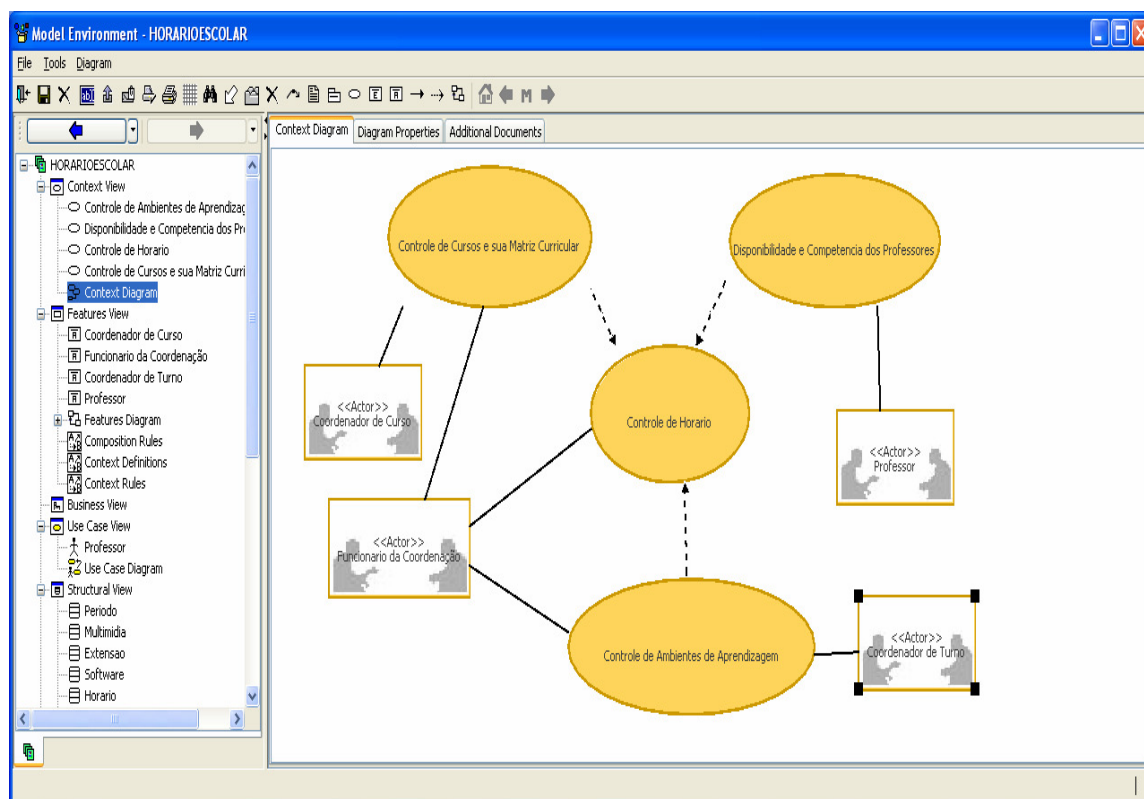


Figura 3.1 Diagrama de Contextos do Domínio de Horário Escolar no Ambiente Odyssey

É possível observar no diagrama na figura de contextos (figura 3.1) a dependência dos subdomínios de Controle de Curso e sua Matriz Curricular, Disponibilidades e Competência de Professores e Controle de Ambientes de Aprendizagem com relação ao subdomínio de Controle de Horário, que contém professores, disponibilidades, competências, cursos, turmas, matrizes curriculares e ambientes de aprendizagem e várias outras características que são utilizadas por todos os outros subdomínios.

Um diagrama de contexto situa o domínio em relação ao seu escopo, limites, relacionamentos com outros domínios e principais atores envolvidos. O diagrama de contexto evolui de acordo com as mudanças do domínio, sendo seu objetivo principal dar um panorama geral do domínio no contexto da organização.

Nas seções a seguir, são analisadas as características desses subdomínios necessários na elaboração do domínio de Horário de Escolar.

3.2.3 Subdomínio de Controle de Curso e Sua Matriz Curricular.

3. 2.3.1 Características de Curso e sua Matriz Curricular

Um curso dentro de uma Instituição de Ensino é considerado como um ensino especializado em uma área específica. Segundo o dicionário Aurélio podemos adaptar a sua definição como sendo um conjunto de matérias a serem ensinadas seguindo um programa de uma determinada área de conhecimento.

Na verdade cada curso representa um dos programas de estudos específicos e organizados, segundo a atividade ou profissão pretendida. A figura 3.2 apresenta o diagrama de características de cursos.

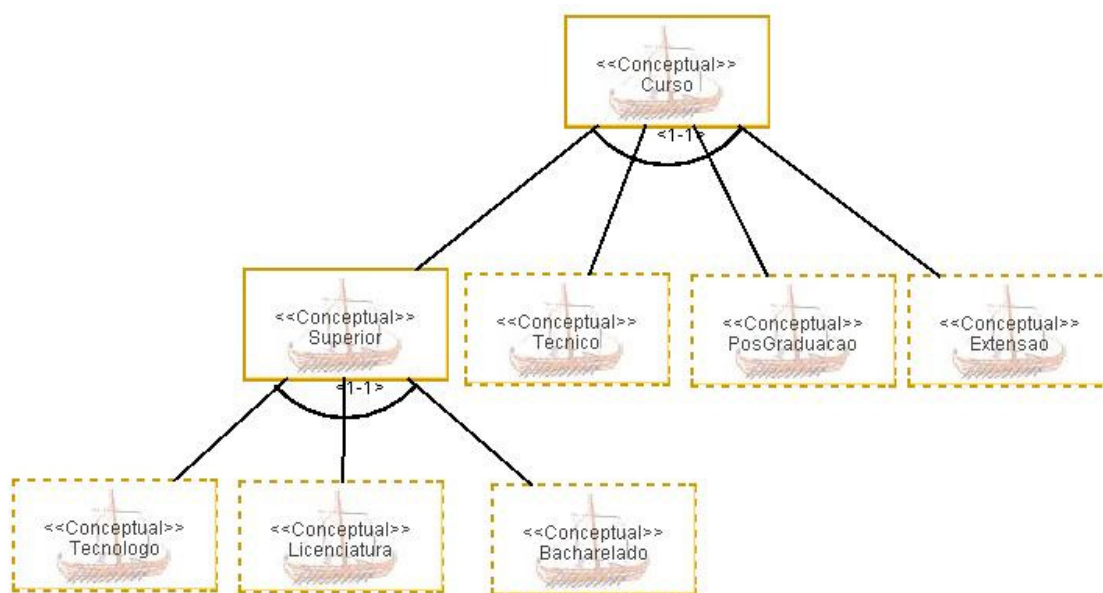


Figura 3.2 Diagrama de Características Conceituais de Curso

O diagrama de características apresentado na figura 3.2 apresenta a característica de Curso como mandatário, pois um curso é uma das principais características no domínio de Horário Escolar e é um ponto de variação tendo Técnico, Extensão e Pós-Graduação como variantes opcionais, devido ao fato de que nas três coordenações onde foram realizadas entrevistas para embasar este trabalho estes tipos de cursos citados anteriormente não existem nas três coordenações. Uma outra variante de Curso é Superior que também é um ponto de variação.

Nas três coordenações entrevistadas existe um tipo de Curso Superior. Logo, Superior é um ponto de variação obrigatório, tendo como variantes opcionais Bacharelado, Licenciatura e Tecnólogo.

Os tipos de cursos existentes em cada coordenação entrevista foram os seguintes:

- **Coordenação de Informática:** Bacharelado, Técnico e Tecnólogo.
- **Coordenação de Geografia:** Extensão, Licenciatura e Pós-Graduação.
- **Telecomunicações:** Técnico e Tecnólogo.

Observa-se que existe em cada uma das coordenações apresentadas acima um tipo de curso superior, porém os tipos de cursos superiores existentes em cada uma dessas

coordenações variam, fazendo assim, Superior ser considerado como um ponto de variação neste domínio.

Em relação aos dados de um curso, podemos considerar como um fator importante a sua matriz curricular, sobre a matriz curricular de um curso de acordo com a análise feita considera-se que:

- **Matriz Curricular:** compõe um conjunto de disciplinas de um curso. Normalmente todos os cursos possuem uma matriz curricular sendo associada a ele. No modelo do domínio analisado esse conceito foi empregado para os Cursos Superiores e de Pós-Graduação, pois obrigatoriamente estes são associados a uma determinada matriz curricular. Para os Cursos Técnicos e de Extensão o conceito de matriz curricular não é empregado. Um curso em determinado momento pode possuir mais de uma matriz curricular. Isso ocorre sempre que um curso modifica sua matriz curricular. Para os Cursos Técnicos o conceito de matriz curricular não foi empregado, os Cursos Técnicos são associados a um módulo enquanto os Cursos de Extensão são associados diretamente às disciplinas. Na figura 3.3 é ilustrado o digrama de características que demonstra os tipos de cursos existentes no *framework* modelado e sua associação com a matriz curricular.

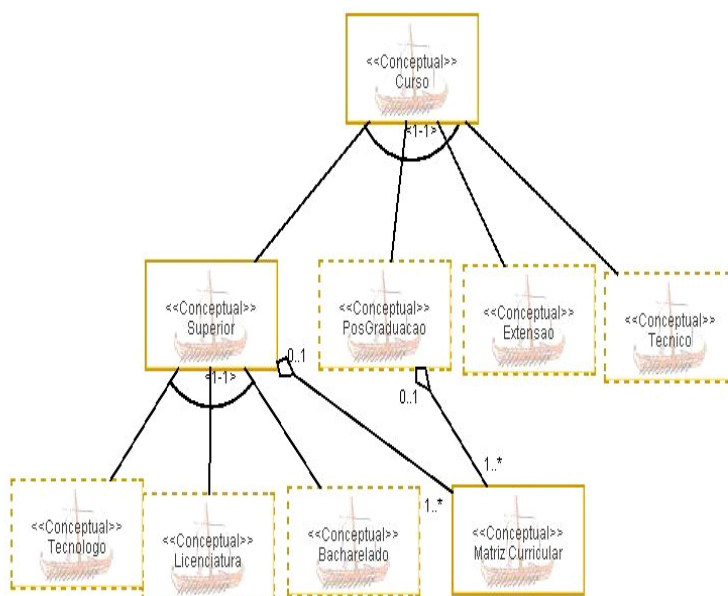


Figura 3.3 Diagrama de Características da Matriz Curricular dos Cursos Superiores e do Curso de Pós-Graduação

A figura 3.3 ilustra o diagrama de características que apresenta a organização da matriz curricular dos tipos de cursos: Superior e Pós-Graduação. Outras características do subdomínio de Controle de Cursos e sua Matriz curricular são: os Períodos e os Módulos. Um período é composto por conjunto de disciplinas, onde a disciplina pertence a uma ou mais matrizes curriculares de um Curso Superior. Já um módulo é organizado por um conjunto de disciplinas pertencentes à grade curricular de um determinado Curso Técnico. Somente os Cursos Superiores e a Pós-Graduação que estão relacionados com matriz curricular, para os Cursos Superiores a matriz está organizada em períodos. Já os Cursos Técnicos são organizados em módulos. A figura 3.4 ilustra as características Cursos, Matriz Curricular, Período e Módulo, onde observa-se a organização completa deste subdomínio.

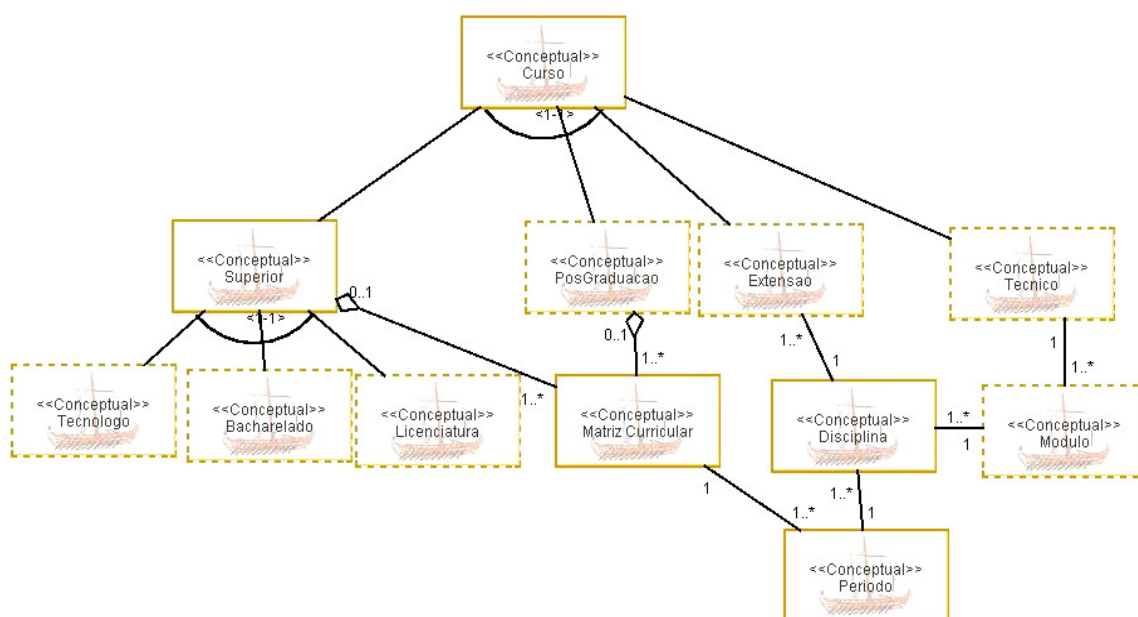


Figura 3.4 Características de Cursos, Matriz Curricular, Período e Módulo

Observa-se na figura 3.4 que a característica Extensão está associada diretamente à característica Disciplina, isso se deve ao fato que um curso de extensão, seguindo a análise que foi realizada na Coordenação de Geografia, é para uma determinada disciplina. Este curso é realizado quando os alunos estão com dificuldades em uma determinada disciplina, daí a coordenação vê a necessidade da abertura de um curso de extensão para dar reforço a esses alunos.

3.2.4 Subdomínio de disponibilidade e Competência de Professores

3.2.4.1 Características de Professores e suas Disponibilidades e Competências

Os professores ministram suas aulas de acordo com suas disponibilidades em determinados dias e horários na semana. Essas disponibilidades são negociadas normalmente entre o Professor e o Coordenador do Curso. Cada professor possui um tipo de dedicação. Um professor deve estar apto a lecionar várias disciplinas, porém ele sempre possui uma competência específica.

Como comentado anteriormente, cada professor possui um tipo de dedicação para ministrar suas aulas. Dentre as coordenações entrevistadas a maioria dos professores que fazem parte do corpo docente destas, possui uma carga horária de 40 horas semanais. Porém, essa carga horária total não é cumprida em sala de aula. Daí o fato de não restringir este subdomínio às cargas horárias do professor e sim às suas disponibilidades nos dias na semana.

Normalmente, os professores deixam suas disponibilidades para ministrar aulas na coordenação para que o funcionário responsável por realizar a elaboração do horário possa ter essas informações. A informação com suas disponibilidades normalmente é feita através de um formulário fornecido pelas coordenações. Neste formulário os professores listam quais são os seus horários e dias disponíveis na semana. Os horários são organizados tentando seguir essas disponibilidades dos professores.

Ao iniciar o período letivo, para cada disciplina existe um ou mais docentes responsáveis pelo andamento do mesmo para as turmas de um curso. Um período letivo é um tempo delimitado ao trabalho escolar efetivo. Portanto, a cada período letivo, as disciplinas, podem ser ofertadas por um ou mais docentes. Um docente é o professor, aquele que ensina uma ciência, uma arte, uma técnica ou uma disciplina.

A figura 3.5 ilustra o diagrama de característica do subdomínio de Disponibilidades e Competências dos professores.

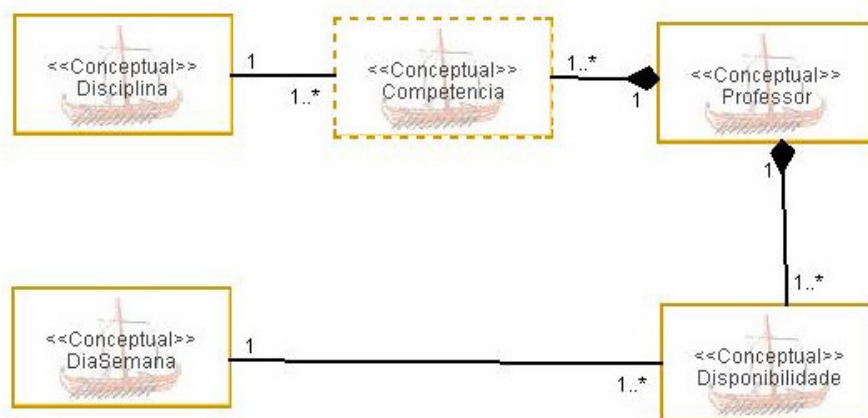


Figura 3.5 Diagrama de característica do Subdomínio de Disponibilidades de Professor e Competência

Observa-se na figura 3.5 que a característica Competência está pontilhada, ou seja, ela é uma característica opcional. A característica Competência é opcional devido ao fato que para as coordenações entrevistadas um professor deve possuir competências para lecionar todas as disciplinas específicas do curso a qual ele pertence. Porém, normalmente, para cada período letivo, um mesmo professor leciona a mesma disciplina que estava lecionando anteriormente. A competência de um professor está associada a ele e é uma determinada disciplina em ano e semestre letivo. De acordo com as coordenações, as competências não seriam uma característica de extrema importância já que normalmente as competências se repetiam para cada ano e semestre. Sendo que as competências variam de acordo com a situação de um professor. Um exemplo que pode ser dado é o seguinte: se o professor se afastou para cursar um mestrado e um novo professor vai assumir a disciplina na qual ele lecionava. Devido ao fato, onde se percebe que a maioria das coordenações entrevistadas não trabalha com o conceito de competência, a característica Competência foi considerada opcional neste modelo. A característica Competência é resultado da associação entre a característica Professor e a característica Disciplina, porém o Odyssey não possui a funcionalidade de criar classes associativas. Logo, competência foi associada a um professor e associada à disciplina diretamente no modelo implementado no Odyssey.

3.2.5 Subdomínio de Controle de Ambientes de Aprendizagem

3.2.5.1 Características de Ambientes de Aprendizagem

As aulas a serem ministradas são alocadas em um determinado local (i.e. ambiente de aprendizagem). Essa alocação é feita de acordo com os ambientes disponíveis que uma Coordenação de Curso ou a Coordenação de Turno tem a oferecer para as aulas. Para a alocação do ambiente a uma determinada aula, devem ser levados em consideração os recursos que cada ambiente possui.

Aqui no IFF existem diferentes tipos de ambientes de aprendizagens, sendo estes: salas, laboratórios, multimídia, auditórios e outros. Para o domínio de Horário Escolar ambientes de aprendizagem é uma característica importantíssima, pois uma aula ocorre em um ambiente em um determinado local.

Cada curso possui seus ambientes próprios para disponibilizar estas aulas. Sendo que nem todas as aulas de um curso são ministradas em seus ambientes próprios. Alguns destes ambientes são disponibilizados pela Coordenação de Turno. Esta é responsável por monitorar alguns ambientes de aprendizagem da instituição. Uma das atividades da Coordenação de Turno é disponibilizar os ambientes de aprendizagem para aulas e gerenciar estes horários. A figura 3.6 representa o digrama de características de Ambientes de Aprendizagem para o *framework* modelado no Odyssey.

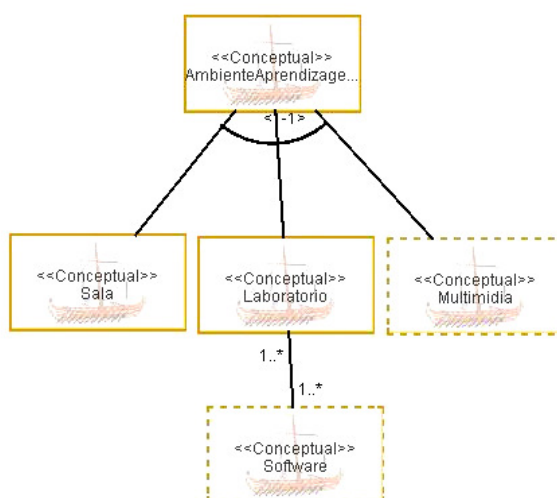


Figura 3.6 Diagrama de Características de Ambientes de Aprendizado

O diagrama de características apresentado na figura 3.6 apresenta a característica de Ambiente de Aprendizagem como obrigatória, pois, como comentado anteriormente, um Ambiente de Aprendizagem é uma importante característica do domínio de Horário Escolar.

A característica Ambiente de Aprendizagem é um ponto de variação, tendo como suas variantes: Sala, Multimídia e Laboratório.

Com a análise que foi feita nas três coordenações, foram obtidos esses três tipos de ambientes de aprendizagem. Sendo que nas Coordenações de Informática e de Telecomunicações, os tipos de ambientes onde são ministradas as aulas são os seguintes: **Salas e Laboratórios**. Já na Coordenação de Geografia, além desses dois tipos de ambientes, em alguns casos as aulas ocorrem na **Multimídia** devido aos seus recursos de vídeos e de espaço, daí a variante Multimídia é uma característica opcional no *framework* por não estar presente nas outras duas coordenações.

Observa-se também no diagrama da figura 3.6 que existe a característica Software associada à característica Laboratório. Um laboratório pode possuir um ou mais softwares instalados. Um software instalado em um laboratório visa atender às especificações necessárias para uma determinada aula. A característica Software é opcional, pois das três coordenações analisadas, em uma dessas, essa característica não se apresentou necessária. Os laboratórios pertencentes à Coordenação de Geografia não necessitam de softwares instalados. Estes são usadas para aulas de Geologia.

3.2.6 Subdomínio de Controle de Horário

3.2.6.1 Características de Horário de Turmas, Professores e Ambientes de Aprendizagem

No início de cada ano ou semestre letivo isso de acordo com o tipo de curso, os horários das turmas devem ser elaborados. Se houver a possibilidade para que os horários possam ser reaproveitados no início de um novo ano ou semestre letivo, ele é aproveitado, pois isso ajuda muito na tarefa dos responsáveis por realizar essa atividade. Porém, este reaproveitamento dependerá de vários fatores, tais como: se a matriz de um curso é ainda a mesma ou se os professores possuem as mesmas disponibilidades.

A partir dos horários das turmas são extraídos os horários dos professores e Ambientes de Aprendizagem.

O horário de uma turma é composto por uma disciplina, por um professor em um ambiente de aprendizagem em um dia e hora na semana. Este horário é organizado de acordo com as disciplinas na qual ela deverá cursar em determinado período da matriz ou módulo.

Os professores possuem seus horários onde contêm as disciplinas que irá lecionar para uma determinada turma em um local e em um determinado dia e hora na semana.

Na maioria dos casos, para cada ambiente de aprendizagem a coordenação de curso ou a coordenação de turno mantém um horário para cada um destes ambientes. A figura 3.7 representa o diagrama de características para o subdomínio de Controle de Horário. Vale ressaltar que este subdomínio contém os outros subdomínios apresentados neste capítulo. Pois é a partir dele que se obtém todos os outros tipos de horários, ou seja, todos os outros subdomínios possuem uma dependência em relação a este domínio.

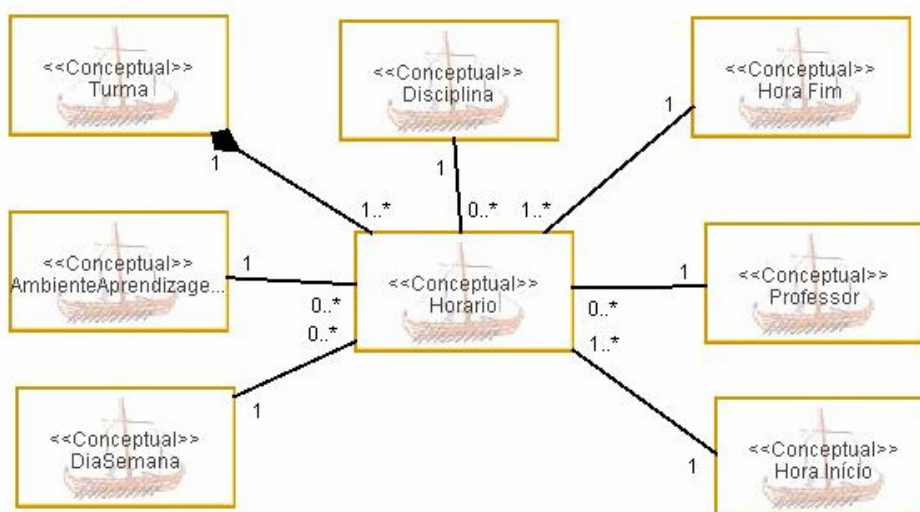


Figura 3.7 Diagrama de Características do subdomínio de Controle de Horário Escolar

O diagrama de características apresentado na figura 3.7 apresenta as características envolvidas no subdomínio de Controle de Horário para o *framework* modelado. Observa-se no diagrama deste subdomínio que a característica Turma contém a característica Horário, pois os horários são de uma turma, devendo ser criados e excluídos com ela.

Um horário está associado às características **Turma, Disciplina, Professor, Ambiente Aprendizagem, Hora Início, Hora Fim e Dia Semana**. Ou seja, um horário ocorre em um dia da semana, com um professor e uma turma para uma determinada disciplina em uma hora início e uma hora fim.

3.3 PROJETO DO DOMÍNIO

É apresentada nesta seção a arquitetura completa do *framework* modelado. Nas seções anteriores deste capítulo foram apresentados os subdomínios e as características pertencentes a este. Uma vez identificadas tais similaridades e diferenças, é necessário representá-las em um modelo genérico, denominado Modelo de Domínio (i.e. modelo de características), que posteriormente poderá ser refinado, originando artefatos que fazem parte de uma infra-estrutura de reutilização.

Nesta fase, os resultados da primeira etapa são utilizados para a construção de um projeto adaptável, fruto da identificação e generalização de soluções para as características comuns. As oportunidades de reutilização identificadas na Análise de Domínio são refinadas de forma a especificar as restrições do projeto. Os pontos de variação existentes no domínio viram os *hot-spots*, ou seja, pontos que podem ser configurados no domínio. Na figura 3.8 é apresentada um dos pontos de variação existentes no *framework*, a característica Curso com suas variantes Superior, Técnico, Pós-Graduação e Extensão, onde novas variantes podem ser adicionadas ou retiradas deste *hot-spot*.

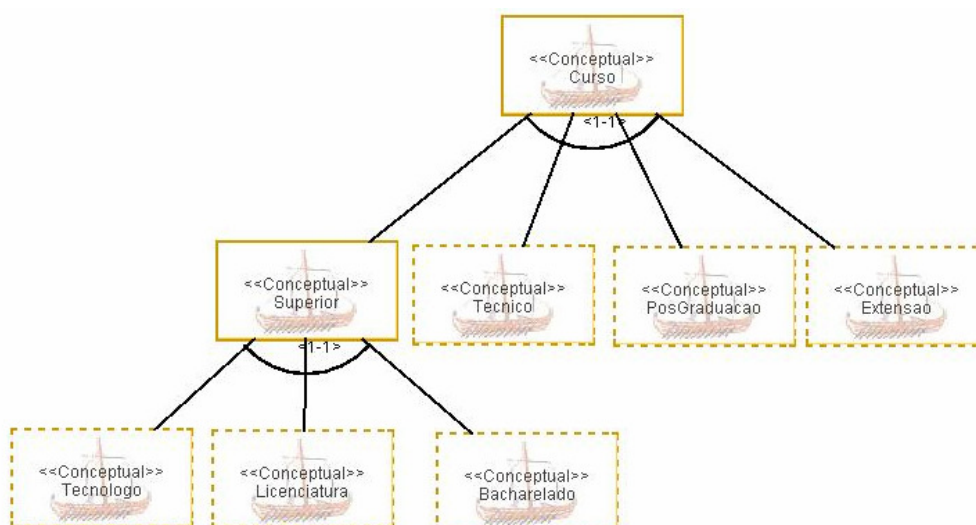


Figura 3.8 Característica Curso e suas variantes

Através da Análise de Domínio e da definição do escopo deste domínio chegou-se a uma arquitetura, que representa o *framework* do domínio, de acordo com os artefatos produzidos. Este *framework* foi mapeado a partir do modelo de características conceituais, este é um dos tipos de mapeamento existentes no ambiente Odyssey.

A seguir são apresentadas as figuras 3.9, 3.10 e 3.11 que representam uma parte domínio de Horário Escolar modelado. Onde a figura 3.9 representa parte do diagrama de características conceituais modelado no Odyssey, a figura 3.10 mostra as opções de mapeamentos de características e a figura 3.11 apresenta o diagrama de classes obtido através deste mapeamento.

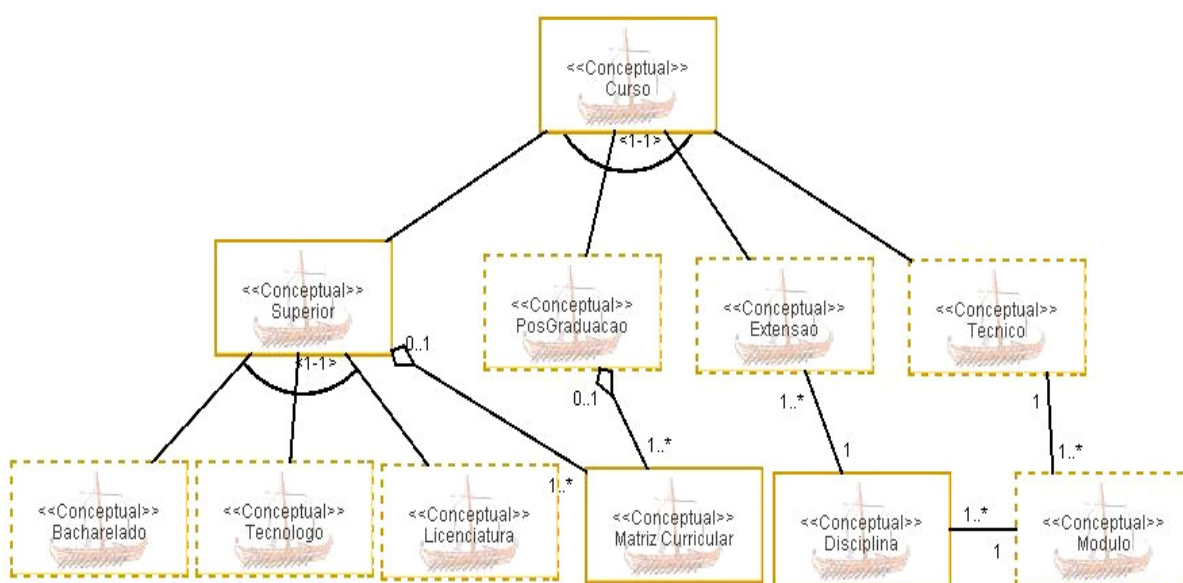


Figura 3.9 Diagrama de Características de parte do Domínio de Horário

Os modelos de características implementados no Odyssey foram mapeados para classes e casos de uso. Nesta seção é apresentado através do projeto de domínio o mapeamento de características conceituais para classes com a finalidade de obter-se uma arquitetura para este domínio. Como exemplo deste mapeamento realizado para chegar a uma arquitetura do referido domínio é apresentado o diagrama da figura 3.9 que apresenta apenas uma parte deste, onde as características conceituais foram mapeadas para classes seguindo as opções deste mapeamento no Odyssey como demonstra a figura 3.10.

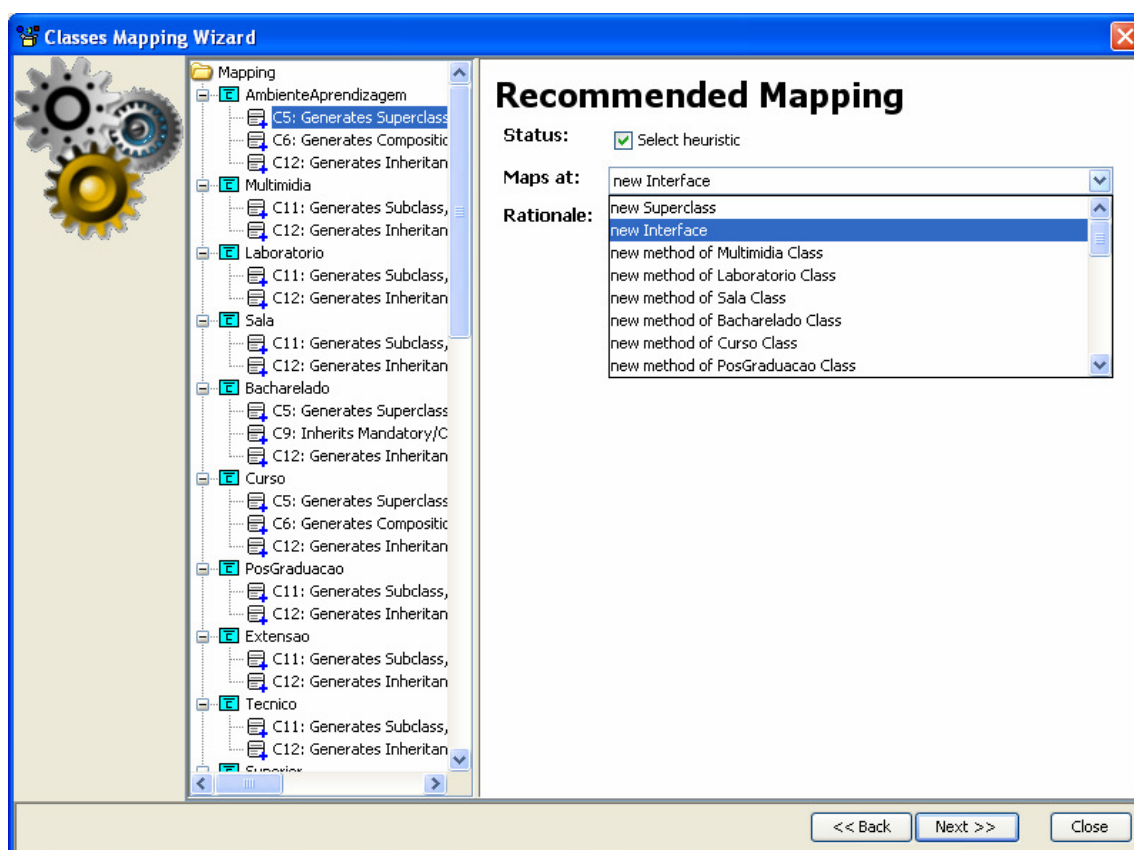


Figura 3.10 Opções de Mapeamentos no Odyssey

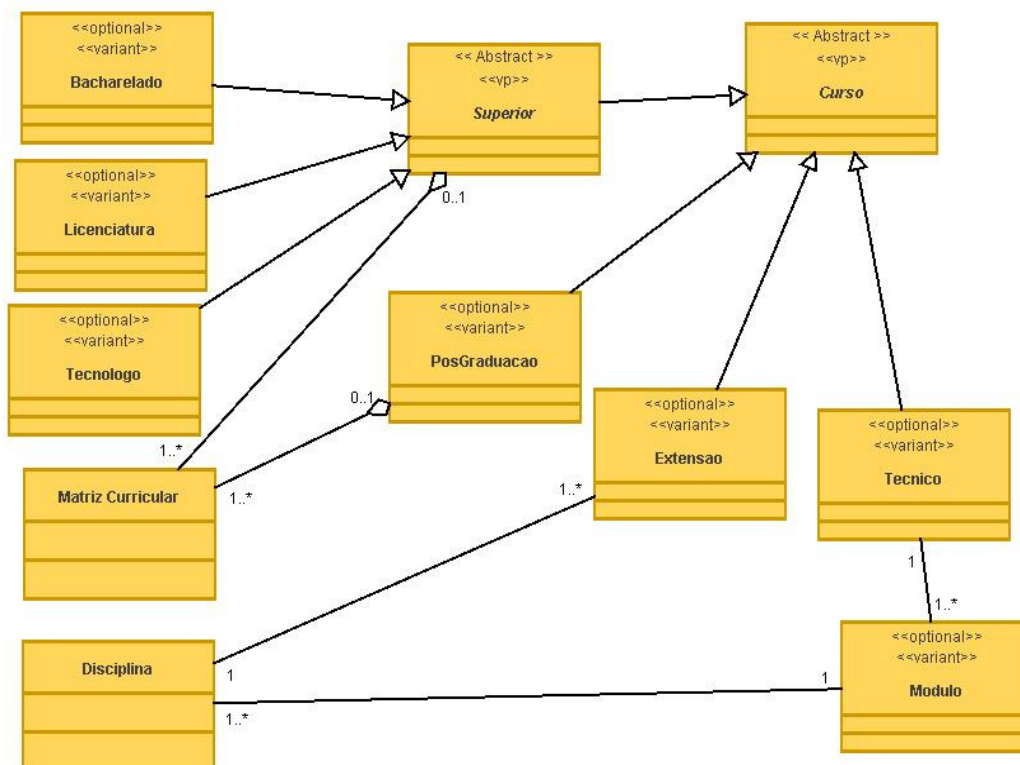


Figura 3.11 Diagrama de Classes de parte do Domínio de Horário Escolar gerado a partir das Características Conceituais

Como ilustrado nas figuras 3.9, 3.10 e 3.11 a partir do modelo de características conceituais implementado no Odyssey foi gerado o modelo de classes da *Unified Modeling Language* (UML). O ambiente Odyssey usa alguns estereótipos para diferenciar variabilidades e opcionalidades dos elementos do diagrama de classes, gerado a partir das características. Características conceituais podem ser mapeadas para uma classe ou atributo, já as características funcionais podem ser mapeadas para um caso de uso ou método, isso depende da heurística escolhida quando o mapeamento vai ser realizado. As opcionalidades devem ser apresentadas no diagrama de classes com estereótipo <<optional>>, pode-se observar isto nas classes Bacharelado, Licenciatura, Tecnólogo, Pós-Graduação, Extensão, Técnico e Módulo. Os pontos de variação devem ser expressos com a notação <<vp>>, o que significa que essa classe pode ser modificada de acordo com a necessidade da aplicação sendo esses pontos, como conceituado anteriormente, os *hot-spots*, pode-se observar isto nas classes Curso e Superior. Ainda temos o estereótipo <<variant>> que indica as alternativas de configuração de um determinado VP, que também são observadas nas classes Bacharelado, Licenciatura, Tecnólogo, Pós-Graduação, Extensão, Técnico e Módulo.

Como mencionado anteriormente, as características conceituais e funcionais podem ser mapeadas para classes e atributos. Observando a figura 3.7 apresentada na Seção 3.2.6, as características conceituais **Hora Início** e **Hora Fim** foram mapeadas como atributos para o diagrama de classes. Estas foram mapeadas como atributos pertencentes à classe Horário como é apresentado na figura 3.12 a seguir.

Na figura 3.12 é apresentado o diagrama de classes completo do *framework*, ou seja, a arquitetura do domínio modelado. Todas as características conceituais que foram apresentadas separadas nas seções anteriores deste capítulo para uma melhor apreciação, estão contidas nesta arquitetura. Este representa o mapeamento completo dos diagramas que foram implementados no Odyssey como foi demonstrado nas figuras 3.9, 3.10 e 3.11. Os métodos das classes foram ocultados por falta de espaço e também não é relevante nesta arquitetura.

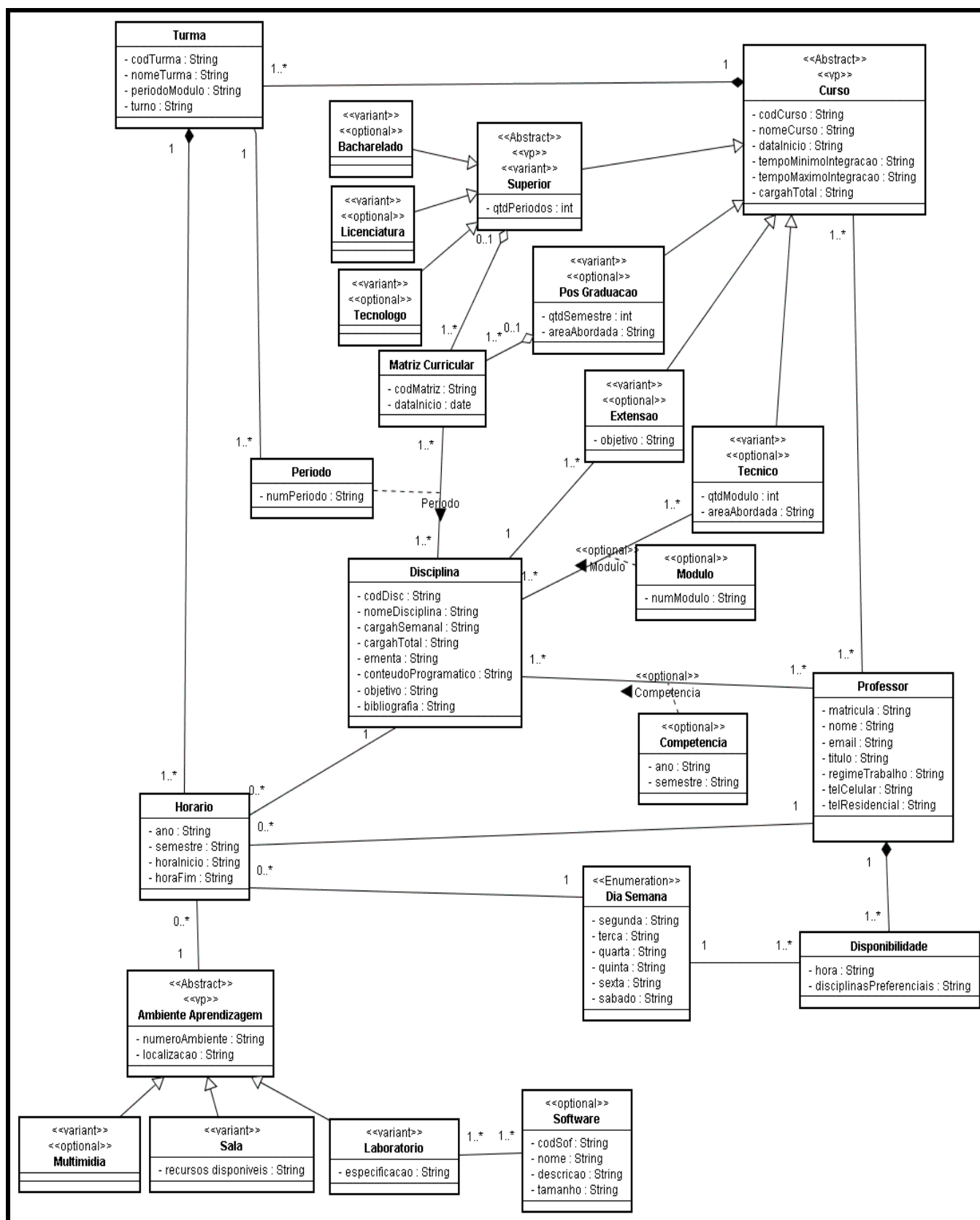


Figura 3.12 Diagramas de Classes do Framework

3.4 MAPEANDO O MODELO DE CARACTERÍSTICAS FUNCIONAIS PARA O MODELO DE CASOS DE USO

O modelo de *Uses Cases* (UC) foi obtido através de características funcionais. A Figura 3.13 apresenta o diagrama de características funcionais implementado no Odyssey e a Figura 3.14 demonstra o processo de mapeamento destas para casos de uso. A Figura 3.15 apresenta o resultado do mapeamento das características funcionais para o modelo de casos de uso realizado no ambiente Odyssey.

Vale ressaltar que as características funcionais do diagrama de características foram mapeadas para casos de uso, mas existem casos em que uma característica funcional é mapeada para um método de uma classe. Esses casos ocorrem quando a granularidade da característica funcional é pequena para representar um caso de uso. Quando isso acontece, seu mapeamento para o caso de uso deve ser desabilitado.

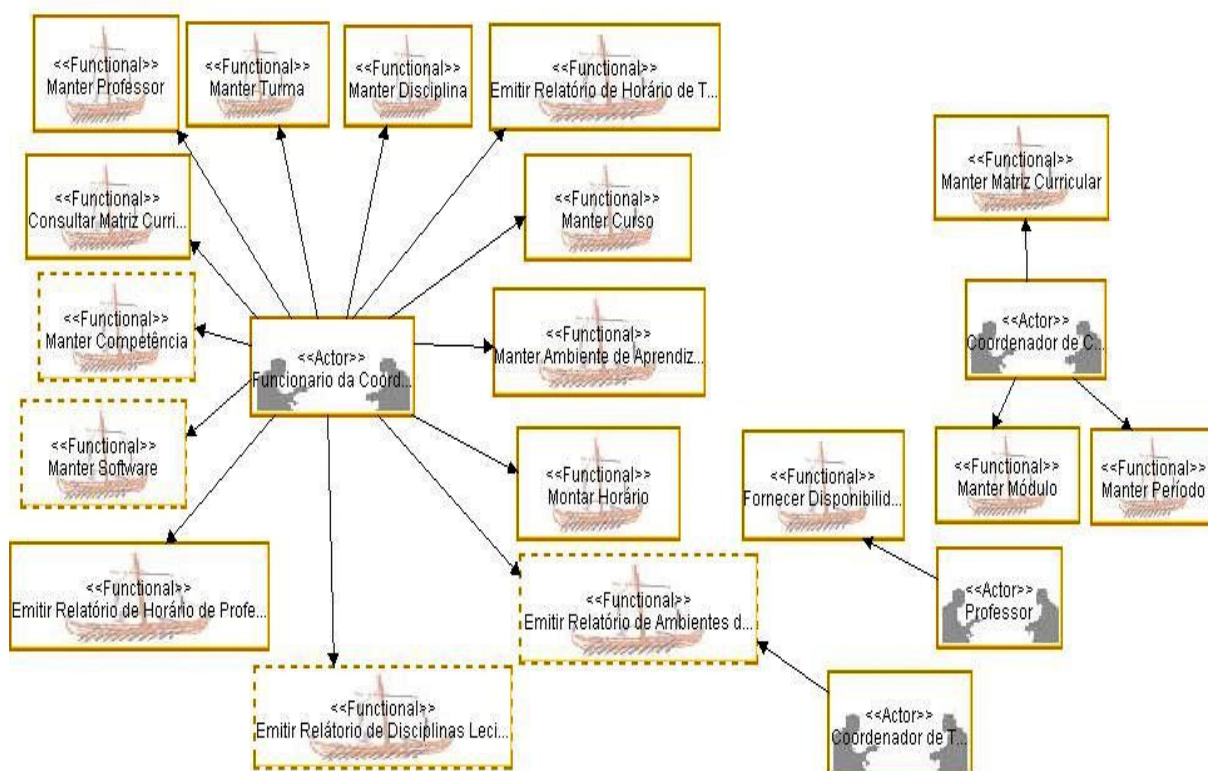


Figura 3.13 Diagrama de Características Funcionais e de Entidades

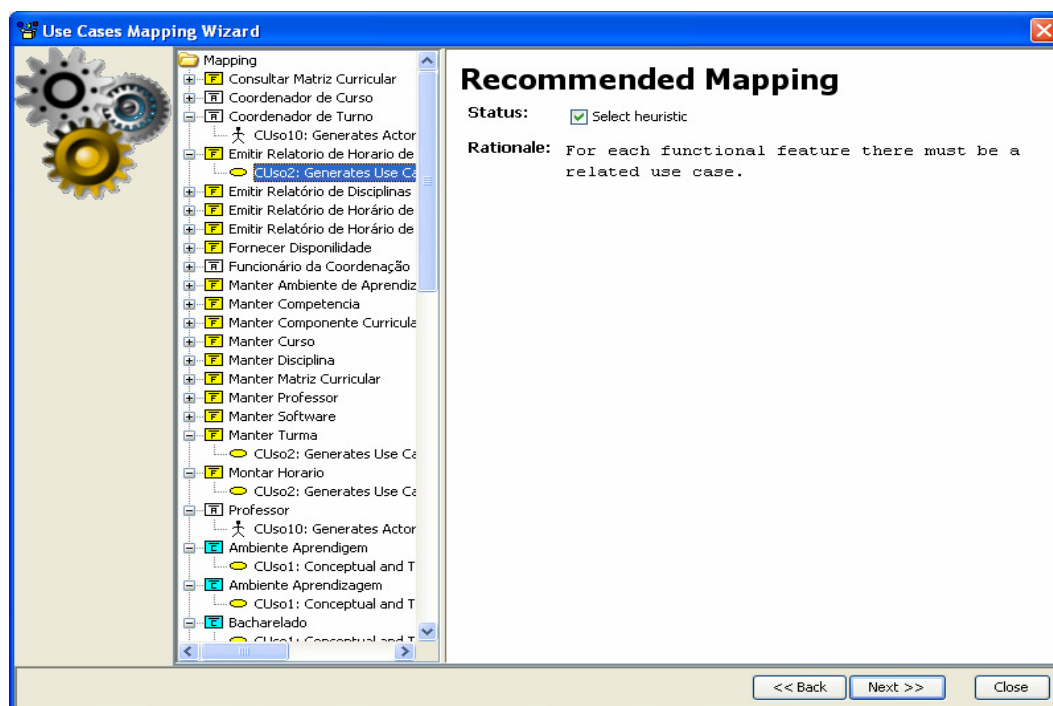


Figura 3.14 Wizard de mapeamento de características funcionais para casos de uso no ambiente Odyssey

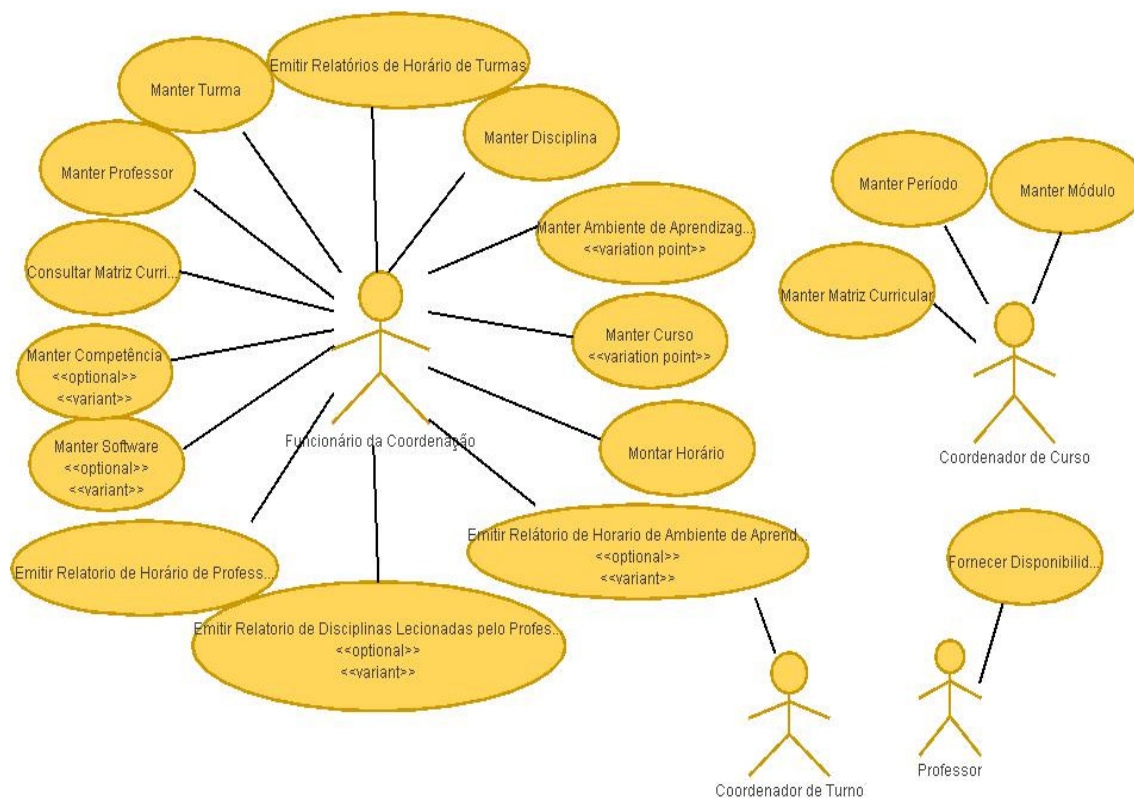


Figura 3.15 Diagrama de Casos de Uso

Observando o diagrama da figura 3.15 existem alguns casos de uso que possuem estereótipos. Os estereótipos existentes neste diagrama são: <<optional>>, <<variant>> e <<variant point>>. Os casos de uso que possuem os estereótipos <<optional>> e <<variant>> como: **Manter Competência**, **Manter Software**, **Emitir Relatório de Disciplinas Lecionadas pelo Professor** e **Emitir Relatório de Ambientes de Aprendizagem** são casos de uso que não foram obrigatórios no desenvolvimento do *framework*. Os casos de uso que possuem o estereótipo <<variant point>> tais como: **Manter Curso** e **Manter Ambiente de Aprendizagem** são aqueles que podem ser executados de formas diferentes. Os casos de uso que não possuem nenhum estereótipo são casos de uso obrigatórios no domínio.

Com relação aos mapeamentos do modelo de características para o modelo de casos de uso, alguns itens devem ser considerados.

- Seguindo as regras de mapeamento de implementação no Odyssey, os relacionamentos de agregação do diagrama de características são mapeadas com notação <<include>>, ou seja, uma inclusão de alguns casos de uso em um caso de uso base.
- Uma característica funcional opcional que é uma variante de um determinado VP é mapeado com o estereótipo <<extends>>, tornando-se então casos de uso de extensão do caso de uso base. (BERNADETE, GONÇALVES e CHAGAS, 2007).
- Um caso de uso mapeado que contém o estereótipo <<optional>> deve-se ao caso também de sua funcionalidade não ser aplicado a todo o domínio analisado.
- Um ponto de variação é explicitado com o estereótipo <<variant point>> e suas variantes como estereótipo <<variant>>.
- As características de entidades, ou seja, usuários do domínio que são modelados no diagrama de características funcionais são mapeados para atores no diagrama de casos de uso.

Para este domínio implementado no Odyssey os relacionamentos de agregação e os pontos de variação de acordo com a análise realizada não foi necessária.

A descrição dos casos de uso implementados para expressar o conceito do *framework* modelado se encontra no Anexo A. As heurísticas de mapeamento utilizadas

neste trabalho que foram o mapeamento de características funcionais para casos de uso e de características conceituais para classes se encontra no Anexo D.

3.5 ENGENHARIA DE APLICAÇÃO NO CONTEXTO DA REUTILIZAÇÃO

Mudanças no desenvolvimento de software nos últimos anos, causadas pela necessidade de produtos com funcionalidades mais complexas e de aplicação mais específica, desenvolvidos em curto espaço de tempo e com custo cada vez mais reduzido, têm criado um ambiente de contínuos desafios para a comunidade de Engenharia de Software. Para tratar estes desafios, a tendência tem sido a aceleração do processo de produção de software. Por outro lado, a complexidade do software construído não pára de aumentar, afetando adversamente sua confiabilidade e flexibilidade. Entre as estratégias para facilitar o alcance de objetivos tão diversos, se destacam algumas abordagens de Reutilização, como a combinação Engenharia de Domínio com Engenharia de Aplicação. Os conceitos destas abordagens foram explorados no Capítulo 2.

Para que uma técnica de reutilização de software possa ser efetiva, ela deve reduzir a distância cognitiva entre o conceito inicial de um sistema e sua implementação final. Sendo assim, no contexto de um processo de desenvolvimento de software baseado em reutilização, o reuso nas fases iniciais do desenvolvimento deve ter como resultado o reuso de componentes e artefatos na implementação da aplicação, ou seja, deve haver uma relação estreita entre os conceitos de um domínio nas fases iniciais do processo de desenvolvimento e os componentes codificados que serão utilizados na implementação da aplicação. Assim, para que o processo de reutilização seja efetivo, deve-se utilizar um processo de ED capaz de criar componentes de qualidade em todas as fases do desenvolvimento e um processo correspondente de EA que seja capaz de utilizá-los.

O processo de EA atua de forma paralela ao de ED, sendo que é na EA que os componentes e artefatos do domínio são efetivamente utilizados na construção de aplicações reais, ou seja, é onde ocorre a instanciação do domínio para uma aplicação específica. Num processo de EA, sistemas específicos podem ser gerados a partir do modelo de domínio, instanciando-o de acordo com os requisitos da aplicação. Com um modelo de domínio claro e bem definido, o Engenheiro da Aplicação pode desenvolver a especificação de sua aplicação a partir de componentes definidos previamente, não precisando realizá-la a partir do zero a cada vez que uma nova aplicação num dado

domínio precisar ser construída. Logo, o Engenheiro da Aplicação é o responsável pelo desenvolvimento da aplicação a partir dos componentes reutilizáveis construídos pelo processo de ED.

3.5.1 Instanciando O Framework para uma nova Aplicação Específica do Domínio

O *framework* apresentado neste capítulo foi modelado com base na Análise realizada com as Coordenações de Informática, Geografia e Telecomunicações. Com a finalidade de verificar o domínio modelado após o projeto de domínio, foi realizada mais uma entrevista com uma coordenação, desta vez, a coordenação entrevistada foi a Coordenação de Construção Civil e Arquitetura.

Com os resultados obtidos, verificou-se que o *framework* apresentado suporta a instanciamento para uma nova aplicação neste domínio, demonstrando a sua efetiva reusabilidade que é o objetivo principal deste trabalho. Os cursos existentes nesta área são: **Técnicos de Edificações e Estradas** e **Bacharelado em Arquitetura**. Os tipos de Ambientes de aprendizagem existentes nesta área são: **(a) Salas**: onde os tipos de salas existentes são **Salas Normais** e **Salas de Desenho**; **(b) Laboratórios**: onde os tipos de Laboratórios existentes são os **Laboratórios Normais** onde ocorrem aulas de informática e de algumas disciplinas que necessitem de um software específico para aula e os **Laboratórios de Solos** onde ocorrem as aulas de disciplinas relacionadas a construções.

Os conceitos de matriz curricular, período e módulo também foram os mesmos, pois os cursos citados acima são tipos de curso Técnico e Superior.

Logo, se uma aplicação neste domínio for desenvolvida para a Área de Construção Civil e Arquitetura, o *framework* modelado pode ser utilizado na Engenharia de Aplicação, onde uma seleção de características para compor este domínio deve ser feita, ou seja, um recorte para desenvolver uma aplicação específica, onde as características obrigatórias devem estar presentes na Engenharia de Aplicação. O ambiente Odyssey possui esse processo, onde pode ser feita a instanciamento do modelo gerado através da Engenharia de Domínio, podendo-se recortar partes deste modelo, através das características, e adicionar novas partes para uma aplicação específica a ser desenvolvida como mostra as figura 3.16, 3.17 e 3.18.

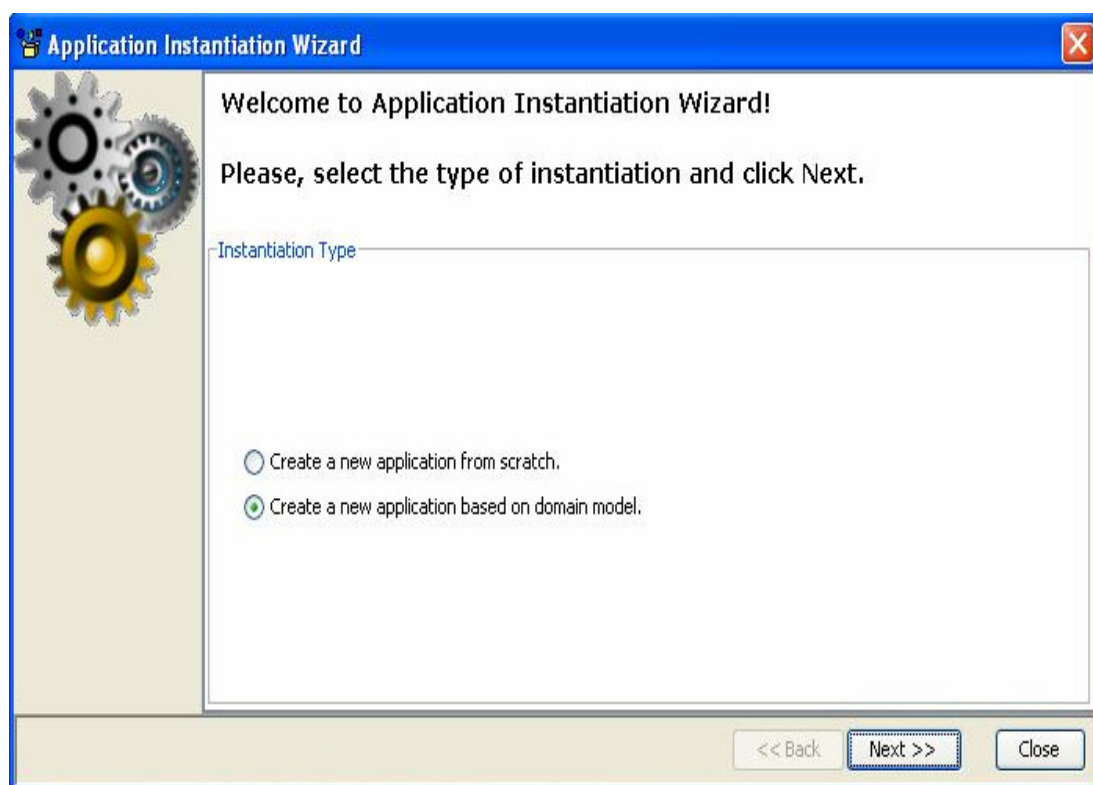


Figura 3.16 Criando uma Aplicação Baseada em um Modelo de Domínio

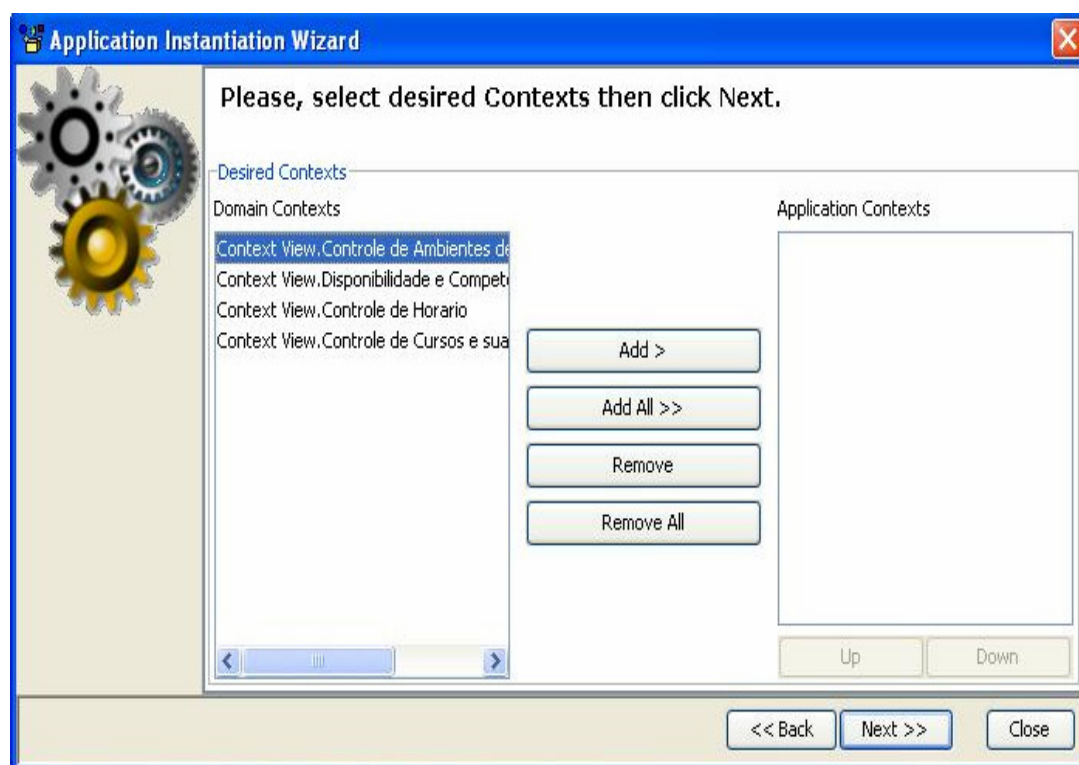


Figura 3.17 Selecionando Contextos

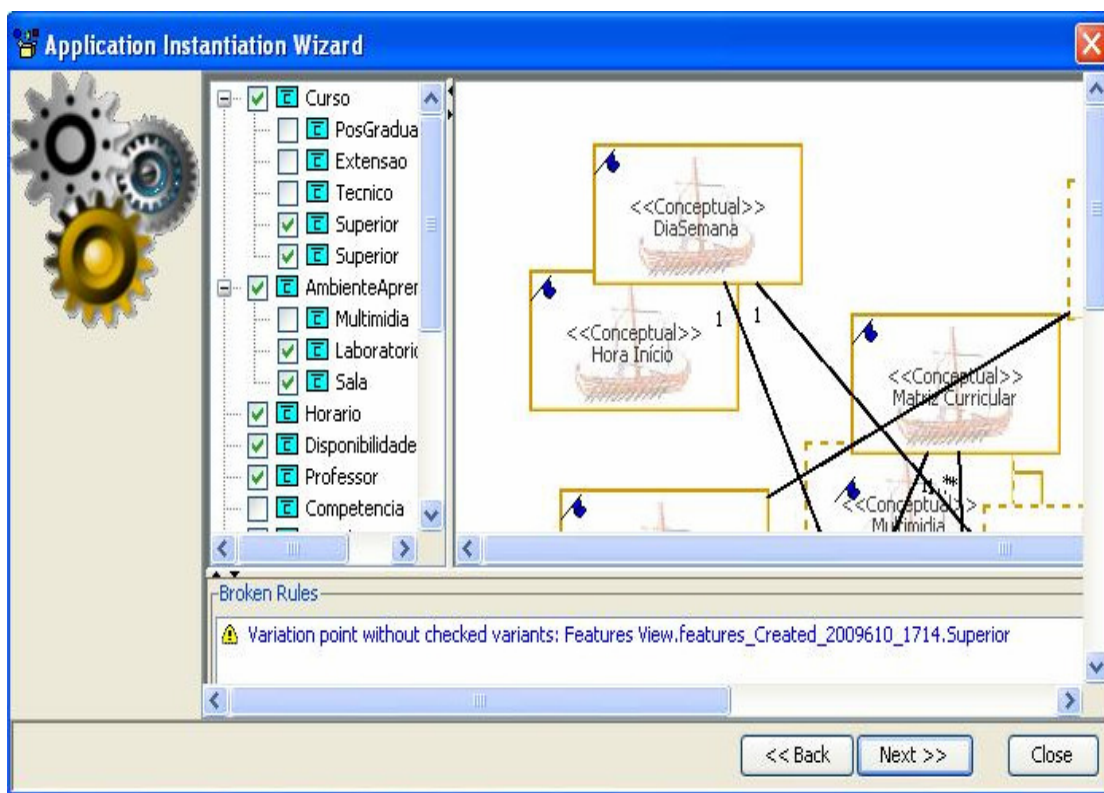


Figura 3.18 Selecionando as Características que irão compor a Aplicação

As figuras 3.16, 3.17 e 3.18 mostram como é realizado o processo de instanciação para uma aplicação específica a partir de um domínio existente, ou seja, como é realizado a Engenharia de Aplicação. A figura 3.16 mostra que uma nova aplicação será feita a partir de um modelo existente, na figura 3.17 os contextos são selecionados, e por fim, na figura 3.18 são escolhidos as características que irão fazer parte da aplicação. São nestas etapas que são feitos os recortes para desenvolver uma aplicação específica. A figura 3.18 mostra como é feito recorte através das características, após ter selecionado os contextos as características pertencentes a esses contextos também podem ser selecionadas. As características obrigatórias já vêm selecionadas, podendo ser escolhidas a partir daí as características opcionais que irão fazer parte da aplicação específica para este novo domínio.

Após ter selecionado as características, novas características podem ser adicionadas. A figura 3.19 apresenta o diagrama de classes de uma aplicação para o domínio de Horário Escolar para a Área de Construção Civil e Arquitetura.

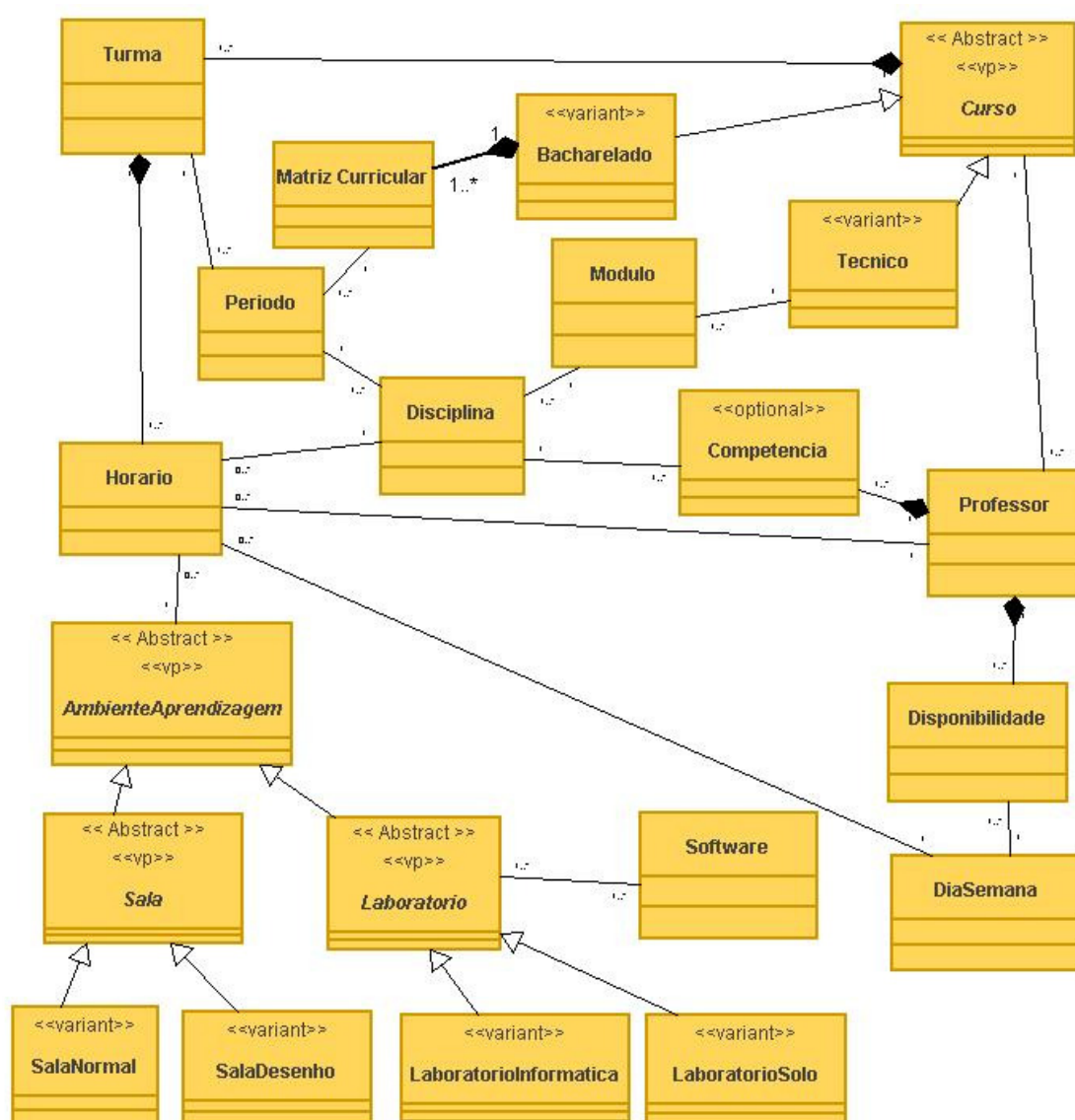


Figura 3.19 Modelo de Domínio para a Área de Construção Civil obtido através da Engenharia de Aplicação

A figura 3.19 apresenta o digrama de classes obtido através do diagrama de características da aplicação para a Área de Construção Civil e Arquitetura. Este diagrama é resultado da Engenharia de Aplicação realizada. Nesse processo de Engenharia de Aplicação foi realizado recortes, porém novas características foram adicionadas para atender essa aplicação. Os relacionamentos permaneceram os mesmos. Para esta aplicação foram retiradas as características: **Extensão**, **Pós-Graduação**, **Superior** e **Multimídia**. Foram acrescentadas as características **Sala Normal**, **Sala de Desenho**, **Laboratório Informática** e **Laboratório Solo**. As características **Sala** e **Laboratório** tornaram nesse modelo pontos de variação. A característica **Sala** é um ponto de variação para o modelo

desta aplicação específica tendo como suas variantes **Sala Normal** e **Sala de Desenho**. A característica **Laboratório** também se tornou um ponto de variação tendo como suas variantes **Laboratório de Informática** e **Laboratório de Solos**. Observa-se também que as características **Técnico** e **Módulo** tornaram-se obrigatórias. No *framework* que foi apresentado essas características eram opcionais, pois o curso Técnico não existia em todas as três coordenações que foram entrevistadas, porém para esta aplicação específica o curso Técnico é obrigatório, sendo assim a característica Módulo também passa ser obrigatório nesta aplicação, devido ao fato que Módulo é referente à Curso Técnico.

A finalidade da entrevista realizada com Área de Construção de Civil e Arquitetura foi verificar a consistência do *framework* modelado e como pode ser realizado o processo de Engenharia de Aplicação a partir de artefatos gerados na Engenharia de Domínio.

A tabela 3.1 apresenta todas as características que foram obtidas no domínio de Horário Escolar através das entrevistas realizadas nas Coordenações de Curso do IFF e sua classificação no desenvolvimento do *framework* que foi realizado neste trabalho seguindo a notação Odyssey-FEX. Este *framework* tem o objetivo de apoiar futuras aplicações que possam vir a ser desenvolvidas para o domínio de Horário Escolar através de sua arquitetura genérica de acordo com a Análise de Domínio realizada.

Das características obtidas e existentes no *framework*, 17,2 % representa os pontos de variação existentes e 37,9 % representam as variantes opcionais.

Analisando estatisticamente mais de 50 % das características são representadas pelos pontos de variação e suas variantes, justificando dessa forma o desenvolvimento de um *framework* para este domínio, o que foi realizado neste trabalho. Caso exista futuramente a necessidade de uma outra aplicação específica para este domínio, esta aplicação poderá ser desenvolvida através do processo de instanciação do *framework*. A flexibilidade está justamente nos pontos de variação (*hot-spots*) e suas variantes e nas opcionalidades. As características classificadas como mandatórias e invariantes são os locais em que todas as aplicações possuem as mesmas características e onde existe um ótimo aproveitamento da reutilização. Esse reaproveitamento também ocorre na configuração dos pontos de variação e de suas variantes.

NOME DA CARACTERÍSTICA	CLASSIFICAÇÃO			
	Ponto de Variação	Variante	Mandatário	Opcional
Professor			Sim	
Curso	Sim		Sim	
Superior	Sim	Sim	Sim	
Técnico		Sim		Sim
Tecnólogo		Sim		Sim
Bacharelado		Sim		Sim
Licenciatura		Sim		Sim
Pós-Graduação		Sim		Sim
Extensão		Sim		Sim
Matriz Curricular			Sim	
Disciplina			Sim	
Módulo				Sim
Período			Sim	
Competência				Sim
Disponibilidade			Sim	
Dia Semana			Sim	
Ambiente de Aprendizagem	Sim		Sim	
Sala	Sim	Sim	Sim	
Sala Normal*		Sim		Sim
Sala Desenho*		Sim		Sim
Laboratório	Sim	Sim	Sim	
Laboratório Informática*		Sim		Sim
Laboratório de Solos*		Sim		Sim
Multimídia		Sim		Sim
Software				Sim
Turma			Sim	
Hora Início			Sim	
Hora Fim			Sim	
Horário			Sim	
Total de Características	29			
Total de Pontos de Variação	5			
Total de Variantes Opcionais	11			

Tabela 3.1 Características do domínio de Horário Escolar e suas Classificações

(*) Características que foram obtidas somente no processo de instanciação para uma aplicação específica, ou seja, para a Área de Construção Civil e Arquitetura como foi apresentado neste capítulo.

3.6 CONSIDERAÇÕES FINAIS

Neste capítulo foi modelado um *framework* para o domínio de Horário Escolar utilizando técnicas de Engenharia de Domínio. Este *framework* foi modelado através de entrevistas realizadas com três coordenações de curso do Instituto Federal de Ciência e Tecnologia Fluminense Campus - Campos Centro. Tratar as diferenças entre os produtos ou aplicações é uma das questões essenciais que devem ser consideradas quando se constrói artefatos reutilizáveis. Tal atividade é conhecida por tratamento de variabilidade e se inicia no momento de reconhecimento de um ponto onde seja possível uma configuração no domínio. A partir disso, é possível obter o mapeamento deste ponto para o sistema específico, passando pela modelagem das variabilidades, o que constitui uma das mais importantes tarefas da ED.

O domínio de Horário Escolar é um essencial a qualquer Instituição de Ensino. O objetivo neste capítulo foi explorar as atividades envolvidas neste domínio e conhecer todos os artefatos existentes nele, projetando assim, uma arquitetura para este domínio. No desenvolvimento deste *framework*, foram explorados conceitos de variabilidades e opcionalidades que são conceitos empregados pela Engenharia de Domínio. O *framework* foi apresentado em subdomínios para uma melhor apreciação e desenvolvimento do trabalho. Este desenvolvimento foi voltado para a reutilização, onde aplicações futuras neste domínio possam ser desenvolvidas através da instanciação deste *framework*, ou seja, através da Engenharia de Aplicação. O modelo apresenta a base para que novas aplicações neste domínio possam reutilizar todos os artefatos gerados, ganhando assim resultados imediatos como: o aumento da produtividade e redução de tempo no desenvolvimento. Para verificar a efetividade do *framework* foi realizado a Engenharia de Aplicação do domínio para a Área de Construção Civil e Arquitetura a partir do modelo existente demonstrando a sua efetiva reutilização.

CAPITULO 4: CONCLUSÕES E TRABALHOS FUTUROS

4.1 CONCLUSÕES

Estamos vivendo em uma era onde tempo é dinheiro e a concorrência entre as empresas é cada dia mais acirrada. Com isso, a reutilização entra como uma poderosa ferramenta, pois ela traz benefícios que determinam a competitividade entre as organizações deixando na frente quem se utiliza dessa técnica. Porém, é preciso cautela para sua implantação, pois a reutilização, quando usada, deve ser adotada por todos os desenvolvedores e toda equipe tem que estar envolvida no processo para que tudo corra bem, pois os investimentos de implantação desta técnica não são poucos e os benefícios só serão vislumbrados em projetos futuros, ou seja, o retorno não é imediato. Por isso, é preciso comprometimento de toda a equipe e da gerência. Além disso, há ainda dificuldades como a compreensão dos artefatos recuperados, produzidos por outros desenvolvedores, a garantia da qualidade dos artefatos, a composição de aplicações a partir de componentes, e até mesmo barreiras psicológicas como a “síndrome do não foi inventado aqui”, entre outras dificuldades (MOURA, CARVALHO e SILVA, 2006).

Por outro lado, o investimento apesar de alto e trazer algumas dificuldades, traz também retornos positivos como: a redução dos custos e tempo de desenvolvimento de aplicações; provável aumento da qualidade do software, pois artefatos reutilizados são mais confiáveis e consistentes na medida em que já foram testados em outra aplicação; uma estrutura mais flexível que facilita a evolução e manutenção do software; e a conformidade com os padrões de desenvolvimento.

Visto isso, é preciso analisar muito bem para se tomar a decisão certa, pois existem os dois lados da moeda. Dessa forma, a reutilização tem seus prós e contras, mas sem dúvida nenhuma, a relação custo x benefício será favorável se a empresa souber administrar bem essas dificuldades e explorar melhor ainda os seus benefícios.

4.2 CONTRIBUIÇÕES E BENEFÍCIOS DO TRABALHO

Este trabalho apresentou mais uma contribuição para a reutilização de software, criando artefatos e analisando o domínio de Horário Escolar através da Engenharia de Domínio que incorpora em seu processo uma etapa de análise, que visa explicitar a variabilidade, i.e, aspectos comuns e específicos inerentes a uma família de sistemas. Apesar do domínio de Horário escolar vem conquistando várias pesquisas, a maioria destas pesquisas é baseada em algoritmos genéticos para solucionar o famoso problema de quadro de Horário Escolar que é estudado há anos em universidades de todo o mundo. Estas pesquisas vêm contribuindo com soluções para este problema. Porém, nas pesquisas encontradas na área observamos que nenhum dos trabalhos aborda este problema junto a um conceito e técnicas de reutilização. Daí, a oportunidade para que se pudesse desenvolver uma pesquisa para este domínio voltado para a reutilização. Neste trabalho, foram representados os termos capturados no domínio, através do modelo de características, e especificado um modelo de domínio para o referido domínio. Este modelo engloba termos e funcionalidades comumente usados que permite a reutilização e o vocabulário comum entre desenvolvedores e usuários de softwares das mais diversas origens, proporciona uma classificação mais precisa dos mais diversos tipos de software relacionados ao domínio e conseqüentemente de componentes de software a serem reutilizados em novas aplicações. Além disso, foi realizado a Engenharia de Aplicação demonstrando a efetividade de reutilização do modelo de domínio desenvolvido através do processo de instanciação, obtendo assim um ganho na produtividade quanto ao tempo e custo, pois todo o processo de busca, análise e formalização das informações do domínio relevantes para a implementação já haviam ocorrido. Para este domínio, sistemas específicos podem ser gerados a partir do modelo de domínio apresentado, instanciando-o de acordo com os requisitos da aplicação. Dessa forma, a aplicação não precisa ser realizada a partir do zero, a cada vez que um novo sistema de horário precise ser construído.

4.3 DIFICULDADES ENCONTRADAS

Foram encontradas muitas dificuldades na realização deste trabalho, uma delas foi quanto ao ambiente de reutilização Odyssey que foi usado na modelagem. Por ser uma ferramenta ainda desconhecida e voltada à reutilização, foi necessário antes aprender a usá-la para depois começar o desenvolvimento, e isto custou um tempo a mais que o previsto. Além disso, para a correta utilização do ambiente Odyssey, foi necessário aprender um novo paradigma, i.e. a Engenharia de Domínio.

Somadas a essas dificuldades, a delimitação do escopo foi mais uma barreira, por se tratar de um domínio muito amplo (Horário Escolar), havendo a necessidade de se limitar esse domínio, dividindo-o em subdomínios para que a análise ficasse focada no objetivo do trabalho “modelar um *framework* com base nas atividades envolvidas neste domínio”. O escopo de um domínio não deve ser tão extenso que não possa ser gerenciado e nem tão simples que não seja útil ao desenvolvimento de aplicações.

Enfim, como Engenharia de Domínio, *framework* e muitos outros temas eram conceitos novos, houve a necessidade de muito estudo e pesquisa para aprendê-los. No entanto, a experiência foi ótima e trouxe muitos novos conhecimentos ao grupo.

4.4 TRABALHOS FUTUROS

Como a ênfase do trabalho foi a reutilização de software, todo desenvolvimento foi voltado para a modelagem no ambiente de reutilização Odyssey e de acordo com a análise do domínio em questão. Como já foi dito anteriormente, as técnicas de reutilização foram aplicadas para a especificação do domínio que envolve o controle e gerenciamento de Horário Escolar, como ele é muito amplo e envolve uma série de atividades houve a necessidade de dividir esse domínio em subdomínios, analisando-se apenas algumas partes mais importantes. O modelo foi criado de maneira que novos conceitos e funcionalidades possam ser adicionados de acordo com o surgimento de novos artefatos em áreas não cobertas no modelo original, à medida que novas aplicações sejam instanciadas no domínio.

Como o foco do trabalho foi a análise deste domínio utilizando técnicas de reutilização, sendo enfatizada a Engenharia de Domínio, uma proposta de trabalho futuro seria a implementação do *framework* completo para representar este domínio, onde este

gerenciasse todas as atividades deste domínio analisado. Foi implementada uma pequena parte deste domínio para expressar apenas os pontos de variação, ou seja, os *hot-spots*. Esta parte implementada gerencia todos os cadastros possíveis para a elaboração de horário, porém ele não realiza as possíveis verificações inerentes a essa atividade de forma coerente para emitir uma grade horária. Daí uma das propostas para trabalhos futuros seria a implementação de um sistema web utilizando a Tecnologia Java EE, onde este sistema gerenciasse todo o processo de elaboração de Horário Escolar, informando e sugerindo disponibilidades de professores, turmas e ambientes de aprendizagem disponíveis para aula e fornecendo diferentes tipos de consultas para o seu uso.

Uma outra possibilidade de trabalho futuro seria a instanciação de aplicações no ambiente Odyssey a partir do *framework* gerado, através do processo de Engenharia de Aplicação, como foi exemplificado na Seção 3.5.1 do Capítulo 3, onde foi explorada uma possível instanciação para uma aplicação neste domínio para a Área de Construção Civil e Arquitetura. Neste processo, características específicas (*features*) do modelo de características do domínio podem ser selecionadas para aplicações, levando a um recorte do *framework*. Ainda com a evolução deste trabalho, sendo instanciadas novas aplicações a partir desse ponto, o modelo de características pode ser atualizado, assim como os demais modelos de domínio e, conseqüentemente, o próprio *framework*, através de novas características e funcionalidades do domínio que venham a ser descobertas. O desenvolvimento deste *framework* foi representado para as Coordenações de Curso do IFF. Um trabalho futuro seria adaptá-lo para atender outras Instituições de Ensino.

Uma outra proposta de trabalho futuro seria a possível geração de componentes para o domínio no Odyssey, permitindo também a implementação do *framework* em uma tecnologia de componentes.

REFERÊNCIAS BIBLIOGRÁFICAS

BERNADETE, D. H. C., GONÇALVES, E. D, CHAOAS, R. L. S., *Aplicando a Engenharia de Domínio na Construção de um Framework de Emissão do Perfil Profissiográfico Previdenciário*. Monografia de Conclusão de Curso, CEFET Campos, Setembro de 2007.

BLOIS, A. P. T. B., *Uma abordagem de Projeto Arquitetural Baseado em Componentes no Contexto de Engenharia de Domínio*. Tese de D.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2006.

BLOIS, A. P., BECKER, K., WERNER, C. M. L., 2004., “Um Processo de Engenharia de Domínio com foco no Projeto Arquitetural Baseado em Componentes”. In: IV Workshop de Desenvolvimento Baseado em Componentes, pp. 15-20, João Pessoa, Paraíba, Brasil.

BOSH, J., 2004, “Software Variability Management”. In: *Proceedings of the 26th International Conference on Software Engineering (ICSE'04)*, pp. 720-721, Scotland, UK.

BRAGA, Regina; *Busca e Recuperação de Componentes em Ambientes de Reutilização de Software*, Tese de D.Sc., COOPE/Sistemas, Dezembro 2000.

BRANQUINHO, L. D; CARVALHO, S. T., Project Framework Model – PMF: Para o desenvolvimento de Sistemas de Informação Orientado a Objetos. Disponível em: http://anhanguera.edu.br/home/index2.php?option=com_docman&task=doc_view&gid=82&Itemid=98, Última consulta em: 10/06/2009.

CRUZ, S. M. S. “Uma ferramenta de Gestão Acadêmica e Escolar Baseada em Serviços Web e Softwares Livres”. In: XXV Congresso Brasileiro de Computação, 2005. São Leopoldo/RS

FARIA, J. M. F; GERALDES. W. B. *SchoolToll: Software de Gestão Escolar para a Criação e Manutenção de Quadro de Horários*. Monografia de Conclusão de Curso – UFLA, 2008.

FREITAS, C. C; GUIMARÃES, P. R.B. *Kairós: Uma Ferramenta Baseada em Algoritmos Genéticos para a Geração de Tabela de Horário Escolar*. Faculdade Ruy Barbosa, 2007.

GAMA, Erich, HELM, Richard, JOHNSON, Ralph, VLISSIDES, J., *Padrões de Projeto: Soluções Reutilizáveis de Software orientado a Objetos*, Trad., Bookman, 2000.

GUIZZARDI, Giancarlo. Um modelo genérico de processo de desenvolvimento de software para e com reuso. Disponível em: <http://www.loa-cnr.it/Guizzardi/Cap2.pdf>. Última consulta em: 17/05/2009

JUNIOR, N.M., *Engenharia de Aplicações no Contexto da Reutilização Baseada em Modelos de Domínio..* Tese de M.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2000.

KANG, K., COHEN, S., HESS, J., et al., (1990), “Feature-Oriented Domain Analysis (FODA) – Feasibility Study”, Software Engineering Institute, CMU/SEI-90-TR-21.

KANG, K. C., LEE, J., DONOHOE, P., (2002), “Feature-Oriented Product Line Engineering”, IEEE Software, v. 9, n. 4 (Jul. /Aug 2002), pp. 58-65.

KUMMEL, G. Z., *Uma abordagem para a Criação de Arquiteturas de Referência de Domínio a partir de da Comparação de Modelos Arquiteturais de Aplicação*. Tese de M.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2007.

LOBO, E. M., *Uma Solução do Problema de Horário Escolar via Algoritmo Genético Paralelo..* Tese de M.Sc., CEFET Minas Gerais, MG, Brasil, 2005.

LOA, 2007, Um modelo genérico de processo de desenvolvimento de software para e com reuso. Disponível em <http://www.loa-cnr.it/Guizzardi/Cap2.pdf>. Última consulta em 02/06/2009.

MAIA, Natanael Elias Nascimento. Engenharia de Software, 2006, Odyssey-MDA: Uma abordagem para a Transformação de Modelos.

MASSEN, T. V. d., LICHTER, H., (2002), "Modeling Variability by UML Use Case Diagrams". In: Proceedings REPL02 – International Workshop on Requirements Engineering for Product Lines, Essen, Germany, September, 2002.

MILER, Nelson., *A Engenharia de Aplicação no Contexto da Reutilização baseada em Modelos de Domínio*, Tese de M.Sc., COOPE/Sistemas, Julho 2000.

MOURA, Ana M., CARVALHO, Jonnathan dos Santos; SILVA, Rogério de Jesus. *Abordagens de Reutilização de Software e sua Aplicação no Domínio de Arrecadação Tributária Municipal*. Monografia de Pós Graduação - CEFET Campos. Dezembro de 2006.

MELLO, A. C. V., MELLO, C. OLIVEIRA., Reutilização de Software: Classificação de Artefatos Reutilizáveis. Disponível em: <http://www.ime.usp.br/~yw/2004/mac5701i/monografias/claudia-acvm.pdf>. Última consulta em: 15/05/2009

Melhor Horário Escolar, 2009. "Ferramenta para Gerenciamento de Horário Escolar". Disponível em: <http://www.horarioescolar.com.br/index.php?pag=apresentacao.htm> Última Consulta em: 21/05/2009.

MURTA, L. G. P., FrameDoc: *Um Framework para a Documentação de Componentes Reutilizáveis*. Monografia para a Conclusão de Curso, 1999.

MURTA, L.G.P., BARROS, M.O., WERNER, C.M.L., 2001, "*FrameDoc: Um Framework para a Documentação de Componentes Reutilizáveis*". In: IV International Symposium on Knowledge Management/Document Management (ISKM/DM'2001), pp. 241-259, Curitiba, PR, Brasil, Agosto.

ODYSSEY, 2007. "Odyssey SDE". Disponível em: <http://reuse.cos.ufrj.br/odyssey>, Última Consulta em: 10/05/2009.

ODYSSEY, 2007. "Odyssey SDE". Disponível em: <http://reuse.cos.ufrj.br/odyssey>. Última consulta em: 15/05/2009

OLIVEIRA, R. F., *Formalização e Verificação de Consistência na Representação de Variabilidades*. Tese de M.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2006.

OLIVEIRA, H. L.P., ROCHA, C. R. P., GONÇALVES, K. M., "Utilização de Sistemas Críticos nas Atividades de Engenharia de Domínio e de Aplicações". In: XV Simpósio Brasileiro de Engenharia de Software, Rio de Janeiro, 2007.

OLIVEIRA, R.F., BLOIS, A.P.T.B., VASCONCELOS, A.P.V., *et al.*, 2005, "Representação de Variabilidades em Componentes de Negócio no Contexto da Engenharia de Domínio". In: Anais do V Workshop de Desenvolvimento Baseado em Componentes, pp. 73-80, Juiz de Fora, MG, Brasil, Novembro.

OLIVEIRA, R.F., 2005, *Metamodelo de Características da Notação Odyssey-FEX - Descrição das Classes*, COPPE/UFRJ, Rio de Janeiro, Brasil, Projeto Reuse - Relatório Técnico 2/2005.

PRESSMAN, R.S., 1997, "Software Reuse". In: MCGRAW-HILL (eds), *Software Engineering: a Practitioner's Approach* 4th ed., pp. 728-755.

PRIETO-DIAZ, R., ARANGO, G., 1991, "Domain Analysis Concepts and Research Directions". In: PRIETO-DIAZ, R., ARANGO, G. (eds), *Domain Analysis and Software Systems Modeling*, IEEE Computer Society Press, pp. 312.

PRESSMAN, R.S., *Engenharia de Software*, Trad., McGraw Hill, 6º ed. , São Paulo, 2006.

ROSETI, M. Z; WERNER, C. M.L., "Aquisição do Conhecimento no Contexto de Análise de Domínio". In: XIII Simpósio de Engenharia de Software, Florianópolis, Outubro de 1999, pp. 429 - 444.

TEIXEIRA, H. V., *Geração de Componentes de Negócio a partir de Modelos de Análise.*, Tese de M.Sc., COOPE/Sistemas, Março 2003.

TEIXEIRA, H., (2003), *Geração de Componentes de Negócio a partir de Modelos de Análise*. Dissertação de MSC., COPPE, UFRJ, Rio de Janeiro, Brasil.

TRIGAUX, J.-C., HEYMANS, P., (2003), “Modelling variability requirements in Software Product Lines: a comparative survey”, Institut d’ Informatique FUNDP.

UFPE, 2007 ABC – Reuso e componentes de Software. Disponível em: http://www.cin.ufpe.br/~aa2/ABC/processo_index.html. Última consulta em: 21/05/2009.

VASCONCELOS, A. P. V., *Uma abordagem de apoio à Criação de Arquiteturas de Referência de Domínio baseada na Análise de Sistemas Legados*. Tese de D.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil 2007.

WERNER, C.M.L., BRAGA, R.M.M., 2005, "A Engenharia de Domínio e Desenvolvimento Baseado em Componentes". In: GIMENES, I.M.S., HUZITA, E.H.M. (eds), *Desenvolvimento Baseado em Componentes: Conceitos e Técnicas*, Rio de Janeiro, pp. 57-104.

WERNER, Cláudia; BRAGA, Regina; ZOPELARI, Mônica; BARROS, Márcio e MURTA, Leonardo. *Odyssey-LE: Uma Infra-Estrutura de Reutilização para o Domínio de Processamento Legislativo*. COOPE/UFRJ – Programa de Engenharia de Sistemas da Universidade Federal do Rio de Janeiro – 2000. Disponível em <http://reuse.cos.ufrj.br/prometeuspublicacoes/enial00.pdf>. Última consulta em 17/05/2009.

XAVIER, J.R., *Criação e Instanciação de Arquiteturas de Software Específicas de Domínio no Contexto de uma Infra-Estrutura de Reutilização*. Tese de M.Sc., COOPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2001.

**ANEXO A: DESCRIÇÃO DOS CASOS DE USO DO FRAMEWORK
IMPLEMENTADO**

CDU 01 - Manter Professor
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Coordenador
Pré-Condição: Cursos devem estar cadastrados
Fluxo Principal:
<ol style="list-style-type: none"> 1. Quando é iniciado um semestre letivo coordenador passa uma lista do funcionário da Coordenação contendo os nomes dos novos professores se existirem. 2. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Professor”. 3. O Funcionário da Coordenação informa os dados a serem cadastrados. 4. O Sistema lista todos os cursos para que o Professor seja associado a um curso. 5. O Sistema é direcionado para a página que contém todos os Professores cadastrados.
Fluxos Alternativos:
<p>1a. A solicitação solicitada é a alteração de dados de um Professor.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de um determinado professor. ➤ O Sistema lista os Professores cadastrados. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os dados Professores cadastrados. <p>2a. A operação solicitada é a exclusão de um Professor.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de um determinado professor. ➤ O Sistema lista os professores cadastrados. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Professor é criado.

CDU 02 - Manter Disciplina
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Coordenador de Curso.
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Coordenador do Curso passa ao Funcionário da Coordenação uma lista de todas as disciplinas que pertencem as matrizes curriculares do curso. 2. O Funcionário da Coordenação acessa o menu de cadastro e escolhe a opção “Disciplina” 3. O Funcionário da Coordenação informa os dados a serem cadastrados. 4. O Funcionário da Coordenação confirma os dados. 5. O Sistema é direcionado para a página que contém todas as disciplinas cadastradas.
Fluxos Alternativos:
<p>1a. A operação solicitada é alteração de dados de uma Disciplina</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada Disciplina. ➤ O Sistema lista todas as disciplinas cadastradas. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os dados Professores cadastrados. <p>2a. O Funcionário da Coordenação solicita a exclusão de uma Disciplina.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada disciplina. ➤ O Sistema lista as disciplinas cadastradas. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Disciplina é criado

CDU 03 - Manter Ambiente de Aprendizagem
Ator: Funcionário da coordenação
Participantes: Funcionário da Coordenação, Coordenador e Coordenação de Turno.
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Coordenador de Turno passa ao Coordenador de Curso os ambientes disponíveis para as aulas daquele curso. 2. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Ambiente de Aprendizagem”. 3. O Sistema lista os tipos de ambientes de aprendizagem existentes. 4. O Funcionário da Coordenação escolhe o tipo de curso a ser cadastrado. 5. O Funcionário da Coordenação informa os dados a serem cadastrados. 6. O Sistema é direcionado para a página que contém todos os ambientes de aprendizagem cadastrados.
Fluxos Alternativos:
<p>1a. A solicitação solicitada é a alteração de dados de um Ambiente de Aprendizagem.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de um determinado ambiente de aprendizagem. ➤ O Sistema lista os tipos de ambientes de aprendizagem. ➤ O Funcionário da Coordenação seleciona o tipo de ambiente de aprendizagem a ser alterado. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os ambientes de aprendizagem cadastrados. <p>2a. A operação solicitada é a exclusão de um Ambiente de Aprendizagem.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação solicita a opção consulta de um cadastro de um determinado ambiente de aprendizagem. ➤ O Sistema lista os ambientes de aprendizagem cadastrados. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Ambiente de aprendizagem é criado.

CDU 04 - Manter Curso
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Curso”. 2. O Sistema lista os tipos de cursos existentes. 3. O Funcionário da Coordenação escolhe o tipo de curso a ser cadastrado. 4. O Funcionário da Coordenação informa os dados a serem cadastrados. 5. O Sistema é direcionado para a página que contém todos os cursos cadastrados.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de um Curso.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação solicita a opção consulta de um cadastro de um determinado curso. ➤ O Sistema lista os tipos de cursos. ➤ O Funcionário da Coordenação seleciona o tipo de curso a ser alterado. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os dados cursos cadastrados. <p>2a. A operação solicitada é a exclusão de um Curso.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação solicita a opção consulta de um cadastro de um determinado curso. ➤ O Sistema lista os cursos cadastrados. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Curso é criado.

CDU 05 - Manter Turma
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Coordenador Curso
Fluxo Principal:
<ol style="list-style-type: none"> 1. Ao início de cada semestre o Coordenador de Curso passa ao Funcionário da Coordenação uma lista das novas turmas. 2. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Turma”. 3. Funcionário da Coordenação informa os dados a serem cadastrados. 4. O Sistema deverá listar os cursos cadastrados para que a turma seja associada um destes. 5. O Funcionário da Coordenação confirma os dados. 6. O Sistema é direcionado para a página que contém todas as turmas cadastradas.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de uma Turma.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada turma. ➤ O Sistema lista as turmas cadastradas. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todas as turmas cadastradas. <p>2a. A operação solicitada é a exclusão de uma Turma.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de cadastro de uma determinada turma. ➤ O Sistema lista as turmas cadastradas. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Turma é criado.

CDU 06 - Manter Competência
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Professor
Pré-Condição: Professor deve ter fornecido suas disciplinas e suas disponibilidades.
Fluxo Principal:
<ol style="list-style-type: none"> 1. Antes do início de um semestre o Professor preenche um formulário, onde informa suas disponibilidades e disciplinas que leciona. 2. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Competência”. 3. O Sistema deverá listar neste cadastro todas as disciplinas e professores cadastrados. 4. O Funcionário da Coordenação deverá fornecer os dados a serem cadastrados. 5. O Funcionário confirma os dados. 6. O Sistema é direcionado para a página que contém todas as competências cadastradas.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de uma Competência.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada competência. ➤ O Sistema lista as competências cadastradas. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todas as competências cadastradas. <p>2a. A operação solicitada é a exclusão de uma Competência.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de cadastro de uma determinada competência. ➤ O Sistema lista as competências cadastradas. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Competência é criado.

CDU 07 - Manter Período
Ator: Coordenador de Curso
Participantes: Coordenador de Curso
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Coordenador de Curso acessa o menu de cadastro e escolhe a opção “Período”. 2. O Sistema deverá neste cadastro listar as matrizes e as disciplinas cadastradas. 3. Funcionário da Coordenação informa os dados a serem cadastrados. 4. O Funcionário confirma os dados. 5. O Sistema é direcionado para a página que contém todos os períodos cadastrados.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de um Período.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de um determinado período. ➤ O Sistema lista os períodos cadastrados. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os períodos cadastrados. <p>2a. A operação solicitada é a exclusão de dados de um Período.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um registro de um determinado período. ➤ O Sistema lista os períodos cadastrados. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Período é criado.

CDU 08 - Manter Módulo
Ator: Coordenador de Curso
Participantes: Coordenador de Curso
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Coordenador de Curso acessa o menu de cadastro e escolhe a opção “Módulo”. 2. O Sistema deverá neste cadastro, listar os cursos técnicos e as disciplinas cadastradas. 3. O Coordenador de Curso informa os dados a serem cadastrados. 4. O Coordenador de Curso confirma os dados. 5. O Sistema é direcionado para a página que contém todos os módulos cadastrados.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de um Módulo.</p> <ul style="list-style-type: none"> ➤ O Coordenador de Curso acessa a opção consulta de um cadastro de um determinado módulo. ➤ O Sistema lista os módulos cadastrados. ➤ O Coordenador de Curso seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Coordenador de Curso informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os módulos cadastrados. <p>2a. A operação solicitada é a exclusão de dados de um Módulo.</p> <ul style="list-style-type: none"> ➤ O Coordenador de Curso acessa a opção consulta de um registro de um determinado módulo. ➤ O Sistema lista os módulos cadastrados. ➤ O Coordenador de Curso seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Módulo é criado.

CDU 09 - Manter Disponibilidade
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Professor
Pré-Condição: Devem existir registros de Professores no sistema e estes já devem ter fornecido suas disponibilidades.
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Funcionário da Coordenação acessa a menu de cadastro e escolhe a opção “Disponibilidade”. 2. Funcionário da Coordenação informa os dados a serem cadastrados. 3. O Sistema deverá listar os professores cadastrados para este ser associada a uma disponibilidade. 4. O Funcionário da Coordenação confirma os dados. 5. O Sistema é direcionado para a página que contém todas as disponibilidades cadastradas.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de uma Disponibilidade.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada disponibilidade. ➤ O Sistema lista as disponibilidades cadastradas. ➤ O Funcionário da Coordenação seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todas as disponibilidades cadastradas. <p>2a. A operação solicitada é a exclusão de uma Disponibilidade.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de cadastro de uma determinada disponibilidade. ➤ O Sistema lista as disponibilidades cadastradas. ➤ O Funcionário Coordenação seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Disponibilidade é criado.

CDU 10 - Manter Matriz Curricular
Ator: Coordenador de Curso
Participantes: Coordenador de Curso
Pré-Condição: Deve existir registro de Cursos no sistema
Fluxo Principal:
<ol style="list-style-type: none"> 1. O Coordenador de Curso acessa a menu de cadastro e escolhe a opção “Matriz Curricular”. 2. O Coordenador de Curso informa os dados a serem cadastrados. 3. O Sistema deverá listar os cursos cadastrados para que a matriz seja associada um destes. 4. O Coordenador de Curso confirma os dados. 5. O Sistema é direcionado para a página que contém todas as matrizes cadastradas.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de uma Matriz Curricular.</p> <ul style="list-style-type: none"> ➤ O Coordenador de Curso acessa a opção consulta de um cadastro de uma determinada matriz curricular. ➤ O Sistema lista as matrizes cadastradas. ➤ O Coordenador de Curso seleciona o registro a ser alterado e escolhe a opção “editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todas as matrizes cadastradas. <p>2a. A operação solicitada é a exclusão de uma Matriz Curricular.</p> <ul style="list-style-type: none"> ➤ O Coordenador de Curso acessa a opção consulta de cadastro de uma determinada matriz curricular. ➤ O Sistema lista as matrizes cadastradas. ➤ O Coordenador de Curso seleciona o registro a ser excluído. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão.
Pós-Condição: Um objeto Matriz Curricular é criado.

CDU 11 - Montar Horário
Ator: Funcionário da Coordenação
Participantes: Funcionário da Coordenação e Coordenador de Curso
Fluxo Principal:
<ol style="list-style-type: none"> 1. Ao início de cada semestre o Coordenador de Curso passa ao Funcionário da Coordenação uma lista das novas turmas. 2. O Funcionário da Coordenação acessa o menu Horário e escolhe a opção “Montar Horário”. 3. O Sistema deverá listar opções de hora início hora fim, ambientes de aprendizagem, turmas, professores e disciplinas para que estes sejam associados a um horário. 4. O Funcionário da Coordenação seleciona e informa os dados necessários para montagem do horário. 5. O Funcionário da Coordenação confirma os dados. 6. O Sistema é direcionado para a página que contém todos os horários cadastrados.
Fluxos Alternativos:
<p>1a. A operação solicitada é a alteração de dados de um Horário.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada turma. ➤ O Sistema lista as turmas cadastradas. ➤ O Funcionário da Coordenação seleciona a turma a ter seu horário alterado e escolhe a opção “Horário por Turma”. ➤ O Sistema lista os horários da turma. ➤ O Funcionário da Coordenação escolhe a opção “Editar”. ➤ O Funcionário da Coordenação informa os dados a serem alterados e confirma a alteração. ➤ O Sistema é direcionado para a página que contém todos os horários das turmas cadastradas. <p>2a. A operação solicitada é a exclusão de um dado do Horário da Turma.</p> <ul style="list-style-type: none"> ➤ O Funcionário da Coordenação acessa a opção consulta de um cadastro de uma determinada turma. ➤ O Sistema lista as turmas cadastradas. ➤ O Funcionário da Coordenação seleciona a turma a ter um dado excluído do seu horário e escolhe a opção “Apagar”. ➤ O Sistema emite uma mensagem solicitando que seja confirmada a exclusão do horário.
Pós-Condição: Um objeto Horário é criado.

**ANEXO B: ALGUMAS TELAS DO SISTEMA IMPLEMENTADO PARA O
FRAMEWORK**

SAGHE-Cadastro de Cursos - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://127.10.0.11/mole/output/cadastro_cursos.php

Guia rápido Últimas notícias

teste 123/SIEP SAGHE-Cadastro de Cursos

CADASTRO DE CURSOS

Selecione o tipo de Curso a ser Cadastrado:

...Selecione o Tipo de Curso... OK

Superior
Extensão
Técnico
Pós-Graduação

[Voltar](#)

Tela de Cadastro de Cursos

SAGHE - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://127.10.0.11/mole/output/professor.php

Guia rápido Últimas notícias

teste 123/SIEP SAGHE

CADASTRO DE PROFESSOR

DADOS A SEREM PREENCHIDOS	PREENCHA TODOS OS DADOS CORRETAMENTE
Matricula	
Nome	
E-mail	
Título	
Regime Trabalho	
Telefone Celular	
Telefone Residencial	
Curso a qual ele pertence	...Selecione um Curso...

Adicionar Deletar

[Voltar](#)

Tela de Cadastro de Professores

SAGHE - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://127.10.0.11/mole/output/horario.php

Guia rápido Últimas notícias

teste 123/SIEP SAGHE

MONTAR HORARIO GERAL

DADOS A SEREM PREENCHIDOS PREENCHA TODOS OS DADOS CORRETAMENTE [Verificar Disponibilidade de Professores]

ANO/SEMESTRE	
HORA INICIO	...Escolha uma hora...
HORA FIM	...Escolha uma hora...
DIA	...Escolha uma dia...
NUMERO AMBIENTE	...Escolha uma ambiente...
TURMA	...Escolha uma Turma...
DISCIPLINA	...Escolha uma Disciplina...
PROFESSOR	...Escolha uma Professor...

Adicionar Deletar Voltar

Tela de Montagem de Horário

SAGHE - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://127.10.0.11/mole/output/turma_list.php

Guia rápido Últimas notícias

teste 123/SIEP SAGHE

SAGHE Sistema de Apoio ao Gerenciamento de Horário Escolar

Id	Cod. da Turma	Nome da Turma	Período/Modulo	Turno	Curso	
5	TL1.GEO	Licenciatura em Geografia-1P/Manha	1	manha	Desenvolvimento de Software	Horarios por Turmas Editar Apagar
4	TDS1.2009	Tecnologia em Desenvolvimento de Software	3	Noite	Desenvolvimento de Software	Horarios por Turmas Editar Apagar

Adicionar Voltar

Tela de Lista de Turmas Cadastradas

**ANEXO C: FORMULÁRIOS PREENCHIDOS COM BASE NAS ENTREVISTAS
REALIZADAS NAS COORDENAÇÕES DE CURSO.**

1- FORMULÁRIO PREENCHIDO JUNTO A COORDENAÇÃO DE INFORMÁTICA

SAGHE- SISTEMA DE APOIO AO GERENCIAMENTO DE HORÁRIO ESCOLAR	
Assunto: Levantamento de requisitos no domínio de horário escolar	
Sector: Informática	Entrevistada: Nádia Chagas Cargo: Pedagoga
Data: 10/11/08	Responsável pelo preenchimento: Elisabelly/Helvio/Renata
Descrição: Entrevista para identificar as atividades envolvidas na elaboração de Horário Escolar. Com a finalidade de identificar variações neste domínio.	
1.0 Quanto ao acesso do beneficiário, público ou parte interessada e as atividades envolvidas	
1.1 Beneficiário ou público diretamente atendido ou parte interessada no processo em análise: (x) Professores () Responsável pela organização do horário escolar () Alunos/ Turma Obs: Não existe um beneficiário direto. Porém tenta-se associar esta tarefa de acordo com o professor.	
1.2 Quando o horário vai ser preparado quais são os pontos principais que podem ser levados em consideração (1) disponibilidade de horário dos professores (2) disponibilidade do dia da semana dos professores (4) carga horária das turmas (5) disponibilidade de laboratórios e salas (6) é levada em consideração a quantidade de horários vagos das turmas? Sim (x) Não () (3) As matrizes curriculares dos cursos	
1.3 Dificuldades para montagem de Horário Escolar. A principal dificuldade é a falta de um auxílio, de uma ferramenta que ajude a otimizar esta atividade. Alocar professores, ambientes e turmas de acordo com suas disponibilidades não é uma tarefa fácil.	
1.4 Em relação as salas disponíveis qual é a técnica utilizada para que os horários das salas não se coincidem? Existe o compartilhamento entre as salas locais de aulas com outros cursos? Como é gerenciado esse compartilhamento? Não existe uma técnica. É verificado as disponibilidades dos ambientes. A coordenação possui ambientes disponíveis para as aulas de seus cursos, algumas salas são gerenciadas pela coordenação de turno, onde esta é responsável pelos horários de alguns ambientes.	
1.5 É organizado um horário de professor, das turmas e dos laboratórios? Sim (x) Não () Quais os passos? Primeiro é organizado o horário das turmas. A partir daí tem-se um horário geral onde são extraídos os horários de Professores e dos Ambientes de Aprendizagem. A coordenação tem posse de uma pasta que contém todos estes horários.	
1.6 Em relação aos dados dos professores, o que mais influencia em relação ao horário? (3) os seus dados pessoais (1) as disciplinas lecionadas por ele (2) carga horária do professor (seu regime)	
1.7 Em relação aos horários dos cursos superiores e técnicos, como é feito para organizá-los? O horário e a carga horária dos professores se diferenciam? Existem professores que ministram aulas para os cursos técnicos e superiores nesta coordenação. Porém, para cada curso o qual eles ministrarão alguma disciplina, os mesmos fornecem suas disponibilidades de forma independente.	

1.8 Se o professor entregou sua disponibilidade de horário depois do quadro de horário já pronto, e o mesmo precise modificar sua disponibilidade. Ou se ele for substituído por um outro professor em meio a um período letivo, sendo que esse também não possa ter a disponibilidade já anexada anteriormente. Como é tratada essa questão?

Tentar adequar da melhor forma possível um novo horário, para não precisar refazer o horário novamente. Neste caso, questões de relacionamento são discutidas. Daí a necessidade de não restringir muito qualquer aplicação que possa ser desenvolvida neste domínio.

2. Quanto a medidas de desempenho existentes

2.1. Tempo médio para finalização do processo __7__ dias. Meta __15__ dias.

2.2. Quantidade de pessoas responsáveis pelo processo:

Normalmente são 2. O funcionário responsável por executar esta atividade na coordenação junto a um bolsista.

3.0 Quais os tipos de Cursos Existentes nesta Coordenação? Como é organizado a sua grade curricular?

Existe nesta coordenação 3 tipos de cursos. Sendo estes:

- Curso técnico de informática;
- Curso de Tecnologia e desenvolvimento de software;
- Curso de bacharelado em sistemas de informação.

Em relação à matriz curricular, o Curso de Tecnologia e o curso de Bacharelado são organizados por matriz, onde cada período corresponde a conjunto de disciplinas a serem cursadas. Já os cursos técnicos não trabalham com o conceito de matriz curricular, este possui uma grade que é organizada em módulos, ou seja, um conjunto de disciplinas a serem cursadas.

4.0 Quais os tipos de Ambientes onde ocorrem as aulas?

As aulas dos cursos pertencentes a esta coordenação ocorrem em salas e laboratórios. As salas possuem alguns recursos disponíveis necessárias as aulas de uma determinada disciplina. Nos laboratórios normalmente ocorrem as aulas que necessitam de alguns softwares específicos que são fundamentais ao andamento da mesma.

2- FORMULÁRIO PREENCHIDO JUNTO A COORDENAÇÃO DE GEOGRAFIA

SAGHE- SISTEMA DE APOIO AO GERENCIAMENTO DE HORÁRIO ESCOLAR	
Assunto: Levantamento de requisitos no domínio de horário escolar	
Setor: Coordenação de Geografia	Entrevistado(a): Guiomar Valdez Cargo: Coordenadora do Curso de Licenciatura em Geografia
Data: 06/05/2009	Responsável pelo preenchimento: Elisabelly/Helvio/Renata
Descrição: Entrevista para identificar as atividades envolvidas na elaboração de Horário Escolar. Com a finalidade de identificar variações neste domínio.	
1.0 Quanto ao acesso do beneficiário, público ou parte interessada e as atividades envolvidas	
1.1 Beneficiário ou público diretamente atendido ou parte interessada no processo em análise: <input type="checkbox"/> Professores <input type="checkbox"/> Responsável pela organização do horário escolar <input checked="" type="checkbox"/> Alunos/ Turma	
1.2 Quando o horário vai ser preparado quais são os pontos principais que podem ser levados em consideração <input type="checkbox"/> disponibilidade de horário dos professores <input type="checkbox"/> disponibilidade do dia da semana dos professores <input type="checkbox"/> carga horária das turmas <input type="checkbox"/> disponibilidade de laboratórios e salas <input type="checkbox"/> é levada em consideração a quantidade de horários vagos das turmas? Sim <input type="checkbox"/> Não <input checked="" type="checkbox"/> <input type="checkbox"/> As matrizes curriculares dos cursos	
1.3 Dificuldades para montagem de Horário Escolar. Certamente, a maior dificuldade encontrada nesta atividade em conseguir conciliar todas as demandas. Matriz curricular do curso, disponibilidades de professores, carga horária das disciplinas e locais de aula.	
1.4 Em relação as salas disponíveis qual é a técnica utilizada para que os horários das salas não se coincidem? Existe o compartilhamento entre as os locais de aulas com outros cursos? Como é gerenciado esse compartilhamento? Os ambientes disponíveis para as aulas são controlados pela coordenação de turno. A cada início de um semestre letivo a coordenação de turno fornece a essa coordenação os ambientes que serão disponibilizados para as aulas deste curso.	
1.5 É organizado um horário de professor, das turmas e dos laboratórios? Sim <input checked="" type="checkbox"/> Não <input type="checkbox"/> Quais os passos? O primeiro passo é organizar os horários das turmas a partir daí, os outros horários são extraídos.	
1.6 Em relação aos dados dos professores, o que mais influencia em relação ao horário? <input type="checkbox"/> os seus dados pessoais <input type="checkbox"/> as disciplinas lecionadas por ele <input type="checkbox"/> carga horária do professor (seu regime)	
1.7 Em relação aos horários dos cursos superiores e técnicos, como é feito para organizá-los? O horário e a carga horária dos professores se diferenciam? Nesta coordenação não possuímos cursos técnicos, logo a disponibilidade do professor é passada para o curso de Licenciatura e para o curso de Pós-Graduação. Ou seja, existem professores que ministram aulas para o curso de Licenciatura e Pós-Graduação.	

1.8 Se o professor entregou sua disponibilidade de horário depois do quadro de horário já pronto, e o mesmo precise modificar sua disponibilidade. Ou se ele for substituído por um outro professor em meio a um período letivo, sendo que esse também não possa ter a disponibilidade já anexada anteriormente. Como é tratada essa questão?

Tentar adequar da melhor forma possível um novo horário, para não precisar refazer o horário novamente. Neste caso, questões de relacionamento são discutidas. Daí a necessidade de não restringir muito qualquer aplicação que possa ser desenvolvida neste domínio.

2. Quanto a medidas de desempenho existentes

2.1. Tempo médio para finalização do processo ____ 5 ____ dias. Meta ____ 15 ____ dias.

2.2. Quantidade de pessoas responsáveis pelo processo:

Normalmente é 1. Apenas o coordenador do curso.

3.0 Quais os tipos de Cursos Existentes nesta Coordenação? Como é organizada a sua grade curricular?

Existe nesta coordenação 3 tipos de cursos. Sendo estes:

- Curso de Licenciatura em Geografia;
- Curso de Pós-Graduação;
- Curso de Extensão.

O curso de Licenciatura é organizado pela matriz curricular.

A Pós-Graduação possui um programa a ser cumprido, o que não foge do conceito de matriz curricular.

Os Cursos de Extensão são cursos referentes a uma disciplina. Este ocorre quando os alunos estão com dificuldade em uma determinada disciplina, então a coordenação organiza um curso de reforço que é ministrado pelo próprio professor que ministra a disciplina. Normalmente estes cursos tem duração de 1 a 2 semanas.

4.0 Quais os tipos de Ambientes onde ocorrem as aulas?

As aulas ocorrem em salas e laboratórios específicos e ainda na Multimídia, de acordo com os grupos de aulas e com as disciplinas.

3- FORMULÁRIO PREENCHIDO JUNTO A COORDENAÇÃO DE TELECOMUNICAÇÕES

SAGHE- SISTEMA DE APOIO AO GERENCIAMENTO DE HORÁRIO ESCOLAR	
Assunto: Levantamento de requisitos no domínio de horário escolar	
Sector: Telecomunicações	Entrevistada: Sérgio Inácio Cargo: Coordenador do Curso Técnico
Data: 29/04/2009	Responsável pelo preenchimento: Elisabelly/Helvio/Renata
Descrição: Entrevista para identificar as atividades envolvidas na elaboração de Horário Escolar. Com a finalidade de identificar variações neste domínio.	
1.0 Quanto ao acesso do beneficiário, público ou parte interessada e as atividades envolvidas	
1.1 Beneficiário ou público diretamente atendido ou parte interessada no processo em análise: <input type="checkbox"/> Professores <input type="checkbox"/> Responsável pela organização do horário escolar <input checked="" type="checkbox"/> Alunos/ Turma	
1.2 Quando o horário vai ser preparado quais são os pontos principais que podem ser levados em consideração <input type="checkbox"/> 1) disponibilidade de horário dos professores <input type="checkbox"/> 4) disponibilidade do dia da semana dos professores <input type="checkbox"/> 2) carga horária das turmas <input type="checkbox"/> 5) disponibilidade de laboratórios e salas <input type="checkbox"/> 6) é levada em consideração a quantidade de horários vagos das turmas? Sim <input type="checkbox"/> Não <input checked="" type="checkbox"/> <input type="checkbox"/> 3) As matrizes curriculares dos cursos	
1.3 Dificuldades para montagem de Horário Escolar. Conseguir conciliar demandas tais como: disponibilidades, matrizes e grades curriculares e locais de aula.	
1.4 Em relação as salas disponíveis qual é a técnica utilizada para que os horários das salas não se coincidem? Existe o compartilhamento entre as salas de aulas com outros cursos? Como é gerenciado esse compartilhamento? A coordenação possui salas específicas para as aulas de seus cursos. Logo, este controle de ambientes de aprendizagem é mantido pela própria coordenação.	
1.5 É organizado um horário de professor, das turmas e dos laboratórios? Sim <input checked="" type="checkbox"/> Não <input type="checkbox"/> Quais os passos? Essa montagem é feita em conjunto, porém a base são os horários das turmas.	
1.6 Em relação aos dados dos professores, o que mais influencia em relação ao horário? <input type="checkbox"/> 3) os seus dados pessoais <input type="checkbox"/> 2) as disciplinas lecionadas por ele <input type="checkbox"/> 1) carga horária do professor (seu regime)	
1.7 Em relação aos horários dos cursos superiores e técnicos, como é feito para organizá-los? O horário e a carga horária dos professores se diferenciam? É variável, depende da formação e a demanda do curso. Existem professores que ministram aulas para o curso técnico e para o curso de tecnólogo, onde para cada um deste ele fornece suas disponibilidades.	

1.8 Se o professor entregou sua disponibilidade de horário depois do quadro de horário já pronto, e o mesmo precise modificar sua disponibilidade. Ou se ele for substituído por um outro professor em meio a um período letivo, sendo que esse também não possa ter a disponibilidade já anexada anteriormente. Como é tratada essa questão?

O primeiro critério é tentar conciliar um horário anterior, se isso não for possível, a tentativa é realizar mudanças tentando aproveitar o máximo todo trabalho já realizado.

2. Quanto a medidas de desempenho existentes

2.1. Tempo médio para finalização do processo __ 3 __ dias. Meta __ 15 __ dias.

2.2. Quantidade de pessoas responsáveis pelo processo:

Normalmente são 3. Os coordenadores dos cursos técnico e tecnólogo juntamente a um funcionário que vem realizando esta atividade na coordenação.

3.0 Quais os tipos de Cursos Existentes nesta Coordenação? Como é organizada a sua grade curricular?

Existe nesta coordenação 2 tipos de cursos. Sendo estes:

- Curso técnico de telecomunicações;
- Curso de Tecnólogo em telecomunicações;

Em relação à matriz curricular, o Curso de Tecnólogo é organizado por uma matriz curricular onde cada período corresponde a um conjunto de disciplinas a serem cursadas. O curso técnico possui uma grade curricular que é organizada em módulos, ou seja, um conjunto de disciplinas a serem cursadas.

4.0 Quais os tipos de Ambientes onde ocorrem as aulas?

Os tipos de ambientes disponíveis para aula nesta coordenação são salas e laboratórios.

4- FORMULÁRIO PREENCHIDO JUNTO À ÁREA DE CONSTRUÇÃO CIVIL E ARQUITETURA.

SAGHE- SISTEMA DE APOIO AO GERENCIAMENTO DE HORÁRIO ESCOLAR		
Assunto: Levantamento de requisitos no domínio de horário Escolar		
Setor: Área de Construção Civil	Entrevistada: Marcelo França/ Isabel	Cargo: Coordenador e Bolsista
Data: 08/07/09	Responsável pelo preenchimento: Elisabelly/Helvio/Renata	
Descrição: Entrevista para identificar as atividades envolvidas na elaboração de Horário Escolar. Com a finalidade de identificar variações neste domínio.		
1.0 Quanto ao acesso do beneficiário, público ou parte interessada e as atividades envolvidas		
1.1 Beneficiário ou público diretamente atendido ou parte interessada no processo em análise: () Professores () Responsável pela organização do horário escolar (x) Alunos/ Turma		
1.2 Quando o horário vai ser preparado quais são os pontos principais que podem ser levados em consideração (1) disponibilidade de horário dos professores (2) disponibilidade do dia da semana dos professores (4) carga horária das turmas (5) disponibilidade de laboratórios e salas (6) é levada em consideração a quantidade de horários vagos das turmas? Sim () Não (x) (3) As matrizes curriculares dos cursos		
1.3 Dificuldades para montagem de Horário Escolar. Alocar os professores aos horários de acordo com suas disponibilidades e fornecer os ambientes de aprendizagens disponíveis para essas aulas.		
1.4 Em relação as salas disponíveis qual é a técnica utilizada para que os horários das salas não se coincidem? Existe o compartilhamento entre as salas de aulas com outros cursos. Como é gerenciado esse compartilhamento. Cada curso possui os locais de aulas definidos de acordo com o seu turno. Esse gerenciamento é feito pela Gerencia de da Área de Construção Civil e Arquitetura.		
1.5 É organizado um horário de professor, das turmas e dos laboratórios? Sim (x) Não () Quais os passos? Primeiro é organizado o horário das turmas. A partir daí tem se um horário geral onde são extraídos os horários de Professores e dos Ambientes de Aprendizagem.		
1.6 Em relação aos dados dos professores, o que mais influencia em relação ao horário? (3) os seus dados pessoais (1) as disciplinas lecionadas por ele (2) carga horária do professor (seu regime)		
1.7 Em relação aos horários dos cursos superiores e técnicos, como é feito para organizá-los? O horário e a carga horária dos professores se diferenciam? Nesta Coordenação existem professores que ministram aulas para os dois cursos técnicos e ainda ministram no curso de Arquitetura que pertence a área. Para cada Coordenação passa a sua disponibilidade e seus dias na semana. Os Coordenadores do Curso Técnico e do Curso de Arquitetura procuram elaborar os horários juntos e acompanhados do Gerente da Área.		

1.8 Se o professor entregou sua disponibilidade de horário depois do quadro de horário já pronto, e o mesmo precise modificar sua disponibilidade. Ou se ele for substituído por um outro professor em meio a um período letivo, sendo que esse também não possa ter a disponibilidade já anexada anteriormente. Como é tratada essa questão?

Tentar organizar da melhor forma possível, para não precisar refazer o horário novamente. Daí a necessidade de não restringir muito qualquer aplicação que possa ser desenvolvida neste domínio.

2. Quanto a medidas de desempenho existentes

2.1. Tempo médio para finalização do processo __ 7 __ dias. Meta __ 15 __ dias.

2.2. Quantidade de pessoas responsáveis pelo processo:

Normalmente são 4. Os Coordenadores do Curso superior de Arquitetura, o Coordenador dos Cursos Técnicos, a Gerente da Área de Construção Civil e Arquitetura e um bolsista que vem realizando esta atividade.

3.0 Quais os tipos de Cursos Existentes nesta Coordenação? Como é organizado a sua grade curricular?

Na área são 3, porém são em Coordenações separadas onde:

Na Coordenação de Construção Civil existem dois cursos técnicos: Construção Civil/Edificações e Construção Civil/ Estradas e Vias.

Na Coordenação de Arquitetura existe apenas um: Arquitetura e Urbanismo.

Em relação a matriz curricular, o Curso de Arquitetura possui uma matriz, onde em cada período existe um conjunto de disciplinas a serem cursadas. Já os Cursos Técnicos seguem um padrão, onde estes possui uma grade de disciplinas por módulo.

4.0 Quais os tipos de Ambientes onde ocorrem as aulas?

Existem as salas que são salas comuns e as salas de desenho, existem os laboratórios que são de informática onde ocorrem as aulas de informática e de algumas disciplinas que exigem determinados software (ex: AUTOCAD) e existe os laboratórios de solos onde são ministradas as aulas relacionadas a construção de obras e existe também um mini-auditório onde normalmente ocorrem as apresentações de trabalhos e eventos da área.

**ANEXO D: HEURÍSTICAS DE MAPEAMENTO DE CARACTERÍSTICAS DA
NOTAÇÃO ODYSSEY-FEX**

Este anexo tem o objetivo de apresentar as Heurísticas de Mapeamento de Características Funcionais para Casos de Uso e Características Conceituais para Classes. Vale ressaltar que nem todas as e heurísticas deste anexo foram realizadas no trabalho, porém uma vez estes apresentados torna-se mais sucinto a o trabalho realizado.

- HEURÍSTICAS DE MAPEAMENTO DE CARACTERÍSTICAS FUNCIONAIS PARA CASOS DE USO BLOIS, (2006)

É APRESENTADO AQUI AS HEURÍSTICAS DESSE MAPEAMENTO PROPOSTO POR BLOIS (2006).

HEURÍSTICA 1

Características conceituais e tecnológicas não devem ser mapeadas para casos de uso.

HEURÍSTICA 2

Uma característica funcional pode ser mapeada para um caso de uso do domínio.

HEURÍSTICA 3

Para cada ponto de variação e suas variantes, devem existir casos de uso com estereótipos <<variant point>> e <<variant>>, respectivamente, relacionados através de herança.

HEURÍSTICA 4

Para cada relacionamento de composição entre características funcionais, deve existir um relacionamento de include entre os casos de uso correspondentes . O caso de uso que origina a relação de include deve representar a característica funcional que representa o todo da relação de composição. As partes da relação de composição devem ser representadas pelo caso de uso destino da relação include.

HEURÍSTICA 5

Para características funcionais relacionadas por herança, devem existir casos de uso relacionados da mesma forma.

HEURÍSTICA 6

Para cada regra de composição do tipo “requer” envolvendo características funcionais, deve existir uma pré-condição especificada no caso de uso que equivale a característica funcional que faz parte do antecedente da regra. Nesta descrição de pré-condição, deve ser registrado que o caso de uso requer um outro que equivale ao conseqüente da regra de composição original.

HEURÍSTICA 7

Para cada regra de composição do tipo “excluir” envolvendo características funcionais, deve existir um estereótipo <<XOR...>> nos casos de uso que correspondem às características envolvidas.

HEURÍSTICA 8

A opcionalidade deve ser transposta para o caso de uso que represente a característica funcional em questão. O caso de uso receberá o estereótipo <<optional>>.


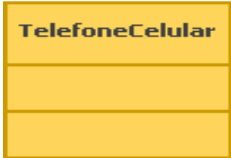
HEURÍSTICA 9

Características de Entidade podem ser mapeadas para atores de casos de uso se tal entidade representa um elemento externo ao domínio. Neste caso, o relacionamento <<communication link>> entre as características de entidade e funcionais, deverá ser mapeada para um relacionamento equivalente entre um ator e um caso de uso.

- HEURÍSTICAS DE MAPEAMENTO DE CARACTERÍSTICAS CONCEITUAIS PARA CLASSES OLIVEIRA(2006)

É APRESENTADO AQUI AS HEURÍSTICAS DESSE MAPEAMENTO PROPOSTO PELA OLIVEIRA (2006). VALE RESSALTAR QUE PARA REPRESENTAR ESTAS HEURÍSTICAS DESTE TIPO DE MAPEAMENTO ESTÁ SENDO USADO COMO EXEMPLO UM DOMÍNIO DE TELEFONE MÓVEL.

HEURÍSTICA 1


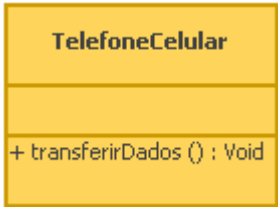
<i>Se existir, no modelo de características, uma característica conceitual, então, pode existir no diagrama de classes uma classe ou atributo que suporte tal característica</i>	
Característica Conceitual	Elemento Correspondente
	

Mapeamento sugerido pela Heurística 1

HEURÍSTICA 2

*Se existir, no modelo de características, um elemento **pacote**, então, pode existir, no diagrama de classes, um **pacote** que suporte tal elemento.*


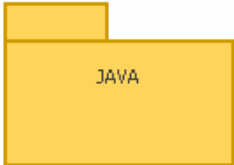
HEURÍSTICA 3

<i>Se existir, no modelo de características, uma característica funcional, então, pode existir no diagrama de classes uma classe, um método ou pacote que suporte tal característica.</i>	
Característica Funcional	Elemento Correspondente
	

Mapeamento sugerido pela Heurística 3

HEURÍSTICA 4

*Se existir, no modelo de características, uma característica **tecnológica**, então, deve existir no diagrama de classes uma **classe**, um **método** ou **pacote** que suporte tal característica.*

Característica Tecnológica	Elemento Correspondente
	

Mapeamento sugerido pela Heurística 4

HEURÍSTICA 5

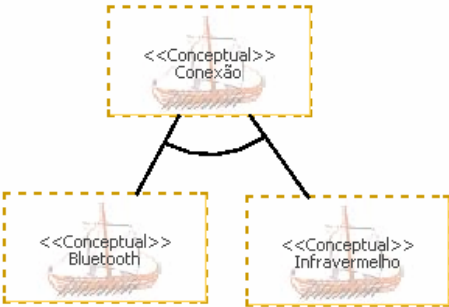

*Características classificadas como **Organizacionais** ou **Não definidas** não são mapeadas para o diagrama de classe*

HEURÍSTICA 6

*Características de **Entidade** não são mapeadas para o diagrama de classe.*

HEURÍSTICA 7

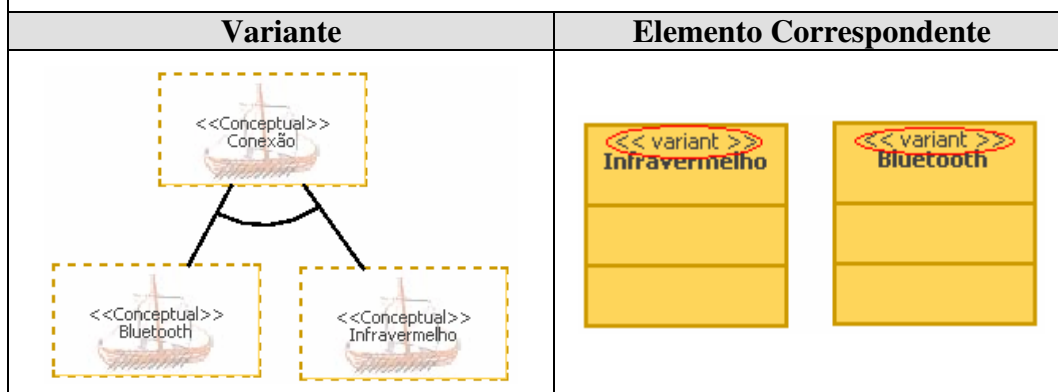
*Se existir, no modelo de características, um **ponto de variação**, pode existir no diagrama de classes uma **classe** ou **interface** que suporte tal elemento, utilizando-se do estereótipo <<vp>>*

Ponto de Variação	Elemento Correspondente
	

Mapeamento sugerido pela Heurística 7

HEURÍSTICA 8

Se existir, no modelo de características, uma característica **variante**, então podem existir, no diagrama de classes, **classes** ou **métodos** que suportem tal elemento, utilizando o estereótipo <<variant>>



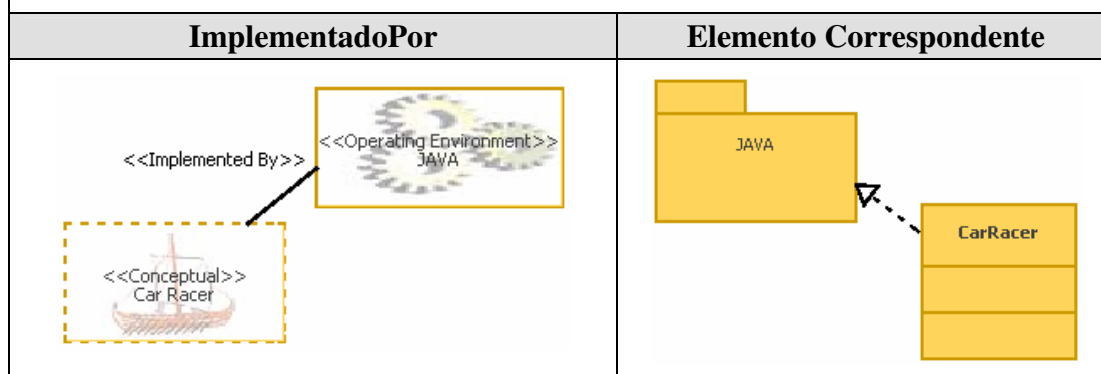
Mapeamento sugerido pela Heurística 8

HEURÍSTICA 9

O relacionamento do tipo **LigaçãoDeComunicação** entre duas características no modelo de características **não é mapeado** no diagrama de classes, uma vez que tal relacionamento envolve pelo menos uma característica do tipo **Entidade**, que não é mapeada.

HEURÍSTICA 10

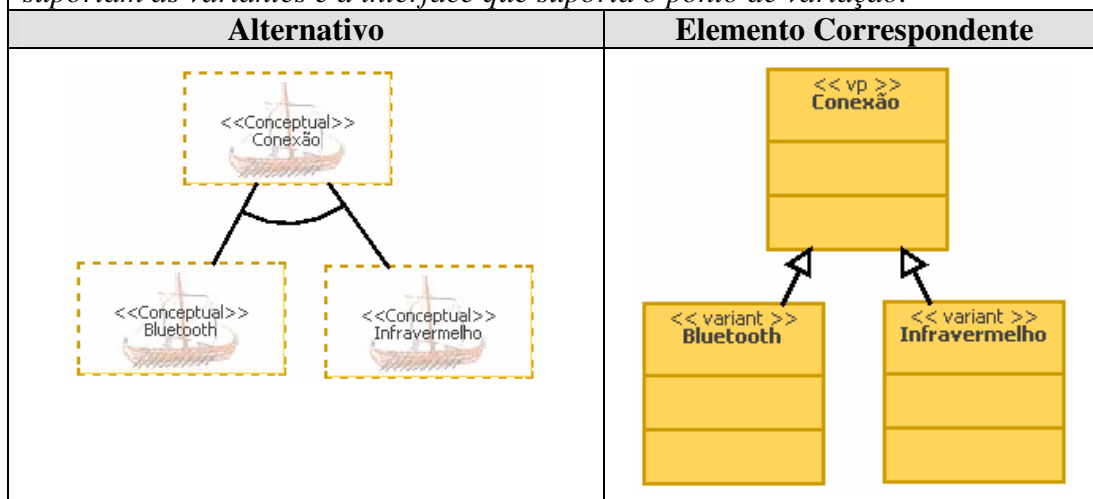
Se existir, no modelo de características, um relacionamento do tipo **ImplementadoPor** entre duas características, então pode existir no diagrama de classes um relacionamento de **associação** ou **implementação** entre as classes que suportam tais características.



Mapeamento sugerido pela Heurística 10

HEURÍSTICA 11

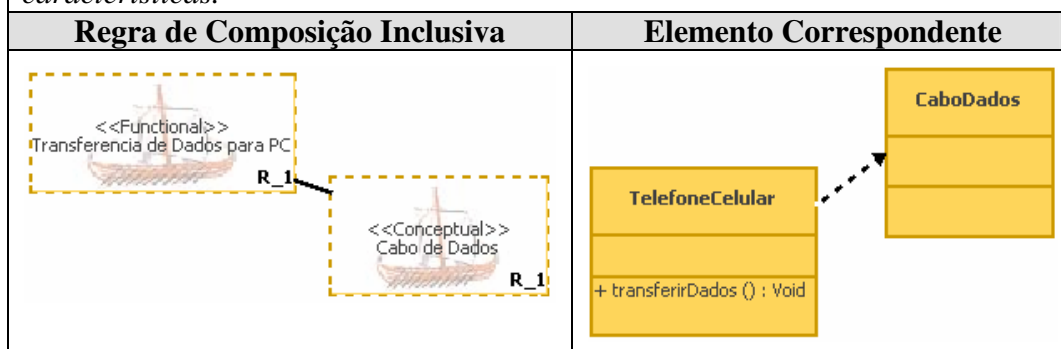
Se existir, no modelo de características, um relacionamento do tipo **Alternativo** entre um ponto de variação e suas variantes, então, deve existir no diagrama de classes um relacionamento de **herança** entre as classes que suportam tais características, se o ponto de variação estiver mapeado para uma **classe**. No caso de o ponto de variação estar mapeado para uma **interface**, deve existir no diagrama de classes um relacionamento de **realização** entre as classes que suportam as variantes e a interface que suporta o ponto de variação.



Mapeamento sugerido pela Heurística 11

HEURÍSTICA 12

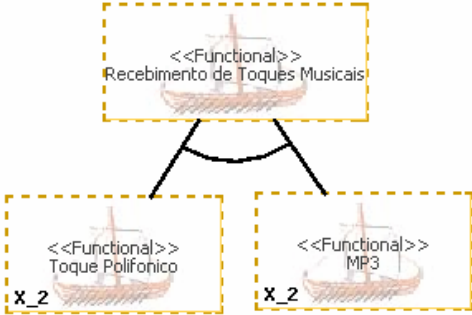
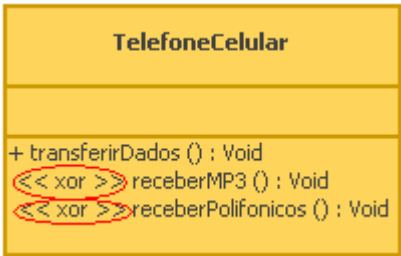
Se existir, no modelo de características, uma **Regra de Composição Inclusiva** entre duas ou mais características, então, deve existir no diagrama de classes um relacionamento de **dependência** ou **associação**, (com multiplicidade mínima igual a 1 para a classe requerida) entre as classes que suportam tais características.



Mapeamento sugerido pela Heurística 12

HEURÍSTICA 13

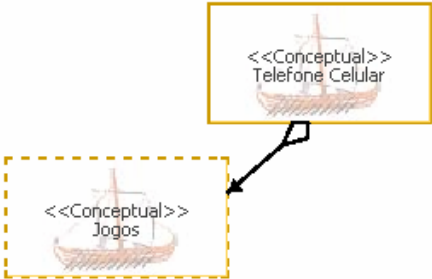
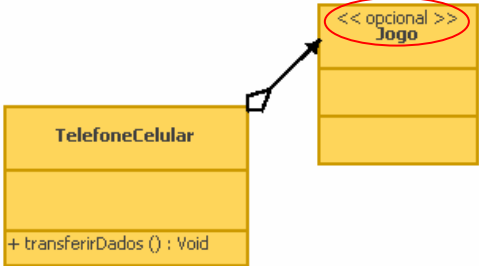
*Se existir, no modelo de características, uma **Regra de Composição Exclusiva** entre duas características, então, deve existir no diagrama de classes **estereótipos** do tipo `<<xor>>` nas classes, métodos ou pacotes que suportam tais características.*

Regra de Composição Exclusiva	Elemento Correspondente
	

Mapeamento sugerido pela Heurística 13

HEURÍSTICA 14

Em todos os casos, deve ser observada a condição de opcionalidade do elemento no modelo de características: elementos não-mandatários no modelo de características devem ser suportados por classes, métodos, atributos e pacotes não-mandatários no diagrama de classe, representados graficamente e devidamente diferenciados dos elementos mandatários.

Características Opcional e Mandatória	Elementos Correspondentes
	

Mapeamento sugerido pela Heurística 14