

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

PLANEJAMENTO DE ALOCAÇÃO DE RECURSOS
HUMANOS EM PROJETOS

JEAN RIGHEZ BASTOS

BLUMENAU
2009

2009/2-12

JEAN RIGHEZ BASTOS

PLANEJAMENTO DE ALOCAÇÃO DE RECURSOS

HUMANOS EM PROJETOS

Trabalho de Conclusão de Curso submetido à Universidade Regional de Blumenau para a obtenção dos créditos na disciplina Trabalho de Conclusão de Curso II do curso de Ciência da Computação — Bacharelado.

Prof. Everaldo Artur Grahl, Mestre - Orientador

PLANEJAMENTO DE ALOCAÇÃO DE RECURSOS

HUMANOS EM PROJETOS

Por

JEAN RIGHEZ BASTOS

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente: _____
Prof. Everaldo Artur Grahl, Mestre – Orientador, FURB

Membro: _____
Prof. Ricardo de Alencar Azambuja, Mestre – FURB

Membro: _____
Prof. Marcel Hugo, Mestre – FURB

Blumenau, 14 de dezembro de 2009

Dedico este trabalho a todos que de alguma forma me incentivaram e apoiaram durante a sua elaboração, especialmente a minha família e a minha namorada.

AGRADECIMENTOS

Aos meus pais Darci e Maria, pelo acesso a educação, amor, carinho, puxões de orelha, apoio e incentivo.

À minha irmã Vanessa, pela paciência e compreensão.

À minha namorada Tuane, pela motivação, compreensão, amor e carinho.

Aos meus amigos, pela compreensão de minha ausência em função do tempo dedicado ao trabalho.

Ao Rafael Stefanos, pela idéia, cobrança e motivação.

Ao meu orientador, Everaldo Artur Grahl, por ter acreditado na conclusão deste trabalho.

Cada escolha, uma renuncia, essa é a vida.

Alexandre Magno Abrão

RESUMO

Este trabalho visa apresentar o estudo sobre alocação de recursos humanos que considere os fatores determinantes em projetos, também apresenta a automatização deste processo, através da utilização do algoritmos de busca Tabu e através de modelos definidos pelo guia Project Management Body Of Knowledge (PMBOK). A ferramenta permite a geração de alocações manual e automática para composição de um calendário de recursos que representa visualmente a alocação. Como forma de comprovação dos resultados, foi realizada uma pesquisa junto a profissionais com experiência em gerência de projetos, a fim de comparar os resultados obtidos. Para sua implementação foram utilizadas as tecnologias ASP.NET MVC Framework, ExtJs, Entity Framework.

Palavras-chave: Alocação de recursos humanos. Planejamento de recursos humanos. Modelo de alocação. Busca Tabu.

ABSTRACT

This paper presents the study on allocation of human resources considering the determinants of projects. The automation of this process, using the Tabu search algorithm and models defined in Project Management Body Of Knowledge (PMBOK) guide. The tool allows the manual and automatic generation of allocation composition of a resource calendar that visually represents the allocation. In order to prove the results, it conducted a survey of professional experience in project management in order to compare the results. Technologies like ASP.NET MVC Framework, ExtJs and Entity Framework were used for the implementation.

Key-words: Allocation of human resources. Human resource planning. Allocation model. Tabu search.

LISTA DE ILUSTRAÇÕES

Figura 1 - Visão geral do gerenciamento de recursos humanos	17
Figura 1 - Matriz de responsabilidades.....	19
Figura 2 - Exemplo de calendário do projeto	20
Figura 3 - Fluxo de solicitação do padrão MVC	24
Figura 4 - Estrutura do Ext JS	25
Figura 5 – Previsão do cronograma.....	27
Figura 6 - Tela principal exibindo <i>menu</i> de cadastros.....	28
Figura 7 - Interface do Resource Management mostrando tabela de alocação de recursos	29
Figura 8 - Diagrama de casos de uso do protótipo	32
Figura 9 – Diagrama de classes da rotina de alocação	34
Figura 10 – Diagrama de classes da rotina de busca	35
Figura 11 - Diagrama de classes da camada de persistência de dados	36
Figura 12 - Diagrama de seqüência para alocação automática de recursos.....	37
Figura 13 - Diagrama de atividades do processo de alocação.....	39
Figura 14 - Diagrama de atividades que representa a busca Tabu	42
Figura 15 - Diagrama entidade relacionamento	44
Quadro 1 - Método OrdenaProjetos, exemplo de utilização da tecnologia LINQ no protótipo.....	46
Quadro 2 - Método CalculaAfinidade implementado na classe AlocacaoController	48
Quadro 3 - Método CalcularAfinidade implementado na classe Afinidade.....	48
Figura 16 - Classe Alocacao.....	50
Figura 17 - Classe Calendario.....	50
Quadro 4 - Método MelhorAlocacaoTabu implementação do algoritmo de busca Tabu	51
Quadro 5 - Interface IRestricaoTabu	52
Figura 18 - Classe AfinidadeRestricao.....	52
Figura 19 - Classe CargoRestricao	52
Figura 20 - Classe DataRestricao.....	52
Quadro 6 - Criação do objeto JsonStore para requisição do dados do calendário de recursos via AJAX	53

Figura 21 - Tela de visualização das alocações	54
Figura 22 - Coluna de profissional reposicionada para melhor visualização	55
Figura 23 - Colunas "04", "05" e "06" ocultas no calendário de recursos.....	55
Figura 24 - Fluxo de funcionamento do protótipo.....	56
Figura 25 - <i>Menu</i> "Cadastros"	56
Figura 26 - Cadastro de cargos	57
Figura 27 - Cadastro de profissionais	58
Figura 28 - Cadastro de profissionais, associação de atributos ao profissional	59
Figura 29 - Cadastro de profissionais, criação de restrições	59
Figura 30 - Cadastros de projetos	60
Figura 31 - Cadastros de projetos, cargos necessários	60
Figura 32 - <i>Menu</i> de alocações	61
Figura 33 - Escolha de projetos e profissionais para execução da alocação automática.....	61
Figura 34 - Alocação manual.....	62
Figura 35 - Lista de afinidades	63
Figura 36 - Calendário de recursos.....	64
Figura 37 - Informações adicionais no calendário de recursos	64
Figura 38 - Relatório de profissionais alocados	65
Figura 39 - Tela de escolha das alocações salvas.....	66
Quadro 7 - Comparativo de ferramentas de alocação de recursos humanos	67
Quadro 8 - Comparativo entre os resultados do protótipo e a pesquisa realizada.....	67
Quadro 9 - Comparativo dos calendários de recursos	69
Quadro 10 - Descrição do caso de uso "UC01 - Cadastrar habilidade"	74
Quadro 11 - Descrição do caso de uso "UC02 - Cadastrar competência".....	74
Quadro 12 - Descrição do caso de uso "UC03 - Cadastrar conhecimento".....	74
Quadro 13 - Descrição do caso de uso "UC04 - Cadastrar profissional".....	75
Quadro 14 - Descrição do caso de uso "UC05 - Cadastrar equipe".....	75
Quadro 15 - Descrição do caso de uso "UC06 - Cadastrar projeto".....	75
Quadro 16 - Descrição do caso de uso "UC07 - Alocar automaticamente profissional".....	76
Quadro 17 - Descrição do caso de uso "UC08 - Alocar manualmente profissional".....	77

Quadro 18 - Descrição do caso de uso "UC09 - Visualizar relatório de alocação profissional x projeto"	77
Quadro 19 - Descrição do caso de uso "UC10 - Cadastrar restrição".....	77
Quadro 20 - Descrição do caso de uso "UC11 - Cadastrar cargo"	78
Quadro 21 - Cenário	79
Quadro 22 - Projetos para alocação	80
Quadro 23 - Profissionais	81
Quadro 24 - Lista de restrições.....	82
Quadro 25 - Instruções de preenchimento	82
Quadro 26 - Calendário de recursos	83

LISTA DE SIGLAS

AJAX - *Asynchronous Javascript And XML*

API - *Application Programming Interface*

DER - Diagrama Entidade Relacionamento

HTTP - *Hypertext Transfer Protocol*

JSON - *JavaScript Object Notation*

LINQ - *Language Integrated Query*

MPS.BR - Melhoria de Processo do Software Brasileiro

MR - Matriz de Responsabilidades

MS-PL - *Microsoft Public Licence*

MVC - *Model View Controller*

PMBOK - *Project Management Body Of Knowledge*

PMI - Project Management Institute

PMP - *Project Management Professional*

RACI - *Responsible, Accountable, Consult, Inform*

RDL - *Report Definition Language*

RF - Requisitos Funcionais

RNF - Requisitos Não Funcionais

RPC - *Remote Procedure Call*

SQL - *Structured Query Language*

TI - Tecnologia da Informação

UML - *Unified Modeling Language*

YUI - *Yahoo User Interface*

SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 OBJETIVOS DO TRABALHO	14
1.2 ESTRUTURA DO TRABALHO	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 PROJECT MANAGEMENT BODY OF KNOWLEDGE (PMBOK)	16
2.2 ALOCAÇÃO DE PROFISSIONAIS EM PROJETOS	18
2.2.1 Planejamento de recursos humanos	18
2.2.2 Calendário de recurso.....	19
2.2.3 Atributos da atividade e recurso.....	20
2.2.4 Modelo de Alocação	21
2.3 ALGORITMO DE PESQUISA TABU.....	21
2.4 ASP.NET MVC FRAMEWORK.....	22
2.5 EXT JS.....	24
2.6 TRABALHOS CORRELATOS	26
2.6.1 Ferramenta para Gerenciamento de Tempo de Projetos	26
2.6.2 Ferramenta de Apoio ao Processo de Gerência de Recursos Humanos do MPS.BR	27
2.6.3 Resource Management	28
3 DESENVOLVIMENTO.....	30
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	30
3.2 ESPECIFICAÇÃO	31
3.2.1 Diagrama de Casos de Uso	31
3.2.2 Diagramas de classes.....	33
3.2.3 Diagrama de seqüência	37
3.2.4 Diagrama de atividades	38
3.2.5 Cálculo de afinidade profissional por projeto	40
3.2.6 Ordenação da lista de projetos	40
3.2.7 Geração de calendário do profissional	40
3.2.8 Preenchimento de restrições de projeto e profissional no calendário	41
3.2.9 Escolha dos melhores calendários.....	41
3.2.10 Apresentação dos resultados	43
3.2.11 Diagrama entidade relacionamento.....	43

3.3 IMPLEMENTAÇÃO	45
3.3.1 Técnicas e ferramentas utilizadas.....	45
3.3.2 Desenvolvimento do protótipo.....	46
3.3.2.1 Afinidade profissional por projeto.....	47
3.3.2.2 Geração de calendário do profissional.....	49
3.3.2.3 Busca Tabu	50
3.3.2.4 Apresentação dos resultados em forma de calendário de recursos.....	53
3.3.3 Operacionalidade da implementação	55
3.3.3.1 Cadastros.....	56
3.3.3.2 Alocação automática.....	61
3.3.3.3 Alocação manual	62
3.3.3.4 Visualização dos resultados	63
3.3.3.5 Salvar resultados	65
3.4 RESULTADOS E DISCUSSÃO	66
4 CONCLUSÕES.....	70
4.1 EXTENSÕES	70
REFERÊNCIAS BIBLIOGRÁFICAS	72
APÊNDICE A – Descrições dos Casos de Uso	74
APÊNDICE B – Formulário da pesquisa para comparação dos resultados.....	79

1 INTRODUÇÃO

Com o crescimento das empresas, tornaram-se fatores críticos a agilidade, pontualidade e qualidade dos produtos. Executar projetos, de maneira cada vez mais eficiente, é um recurso essencial para a sobrevivência da empresa moderna (XAVIER, 2005, p. 2). O gerenciamento inadequado de pessoas é uma das mais significativas contribuições para o fracasso de projetos (SOMMERVILLE, 2003, p. 418).

No guia Project Management Body Of Knowledge (PMBOK), são descritas as nove áreas de conhecimento em gerenciamento de projetos. São elas as gerências de integração, de escopo, de tempo, de custo, de qualidade, de recursos humanos, de comunicações, de riscos e de aquisições do projeto (PROJECT MANAGEMENT INSTITUTE, 2004, p. 9). Dentre essas pode-se ressaltar a gerência de recursos humanos, mais especificamente a alocação de recursos humanos em atividades que compõe o projeto de software.

Segundo Brasil Filho et al. (2006), a alocação de recursos é uma atividade importante e complexa na execução de projetos. Normalmente é um processo não sistemático, uma vez que tipicamente se baseia em experiência pessoal sem o uso de modelos. Os gerentes de projeto têm de solucionar problemas técnicos e não técnicos, utilizando a capacidade das pessoas de sua equipe da maneira mais eficaz possível. Esta tarefa não é simples, nem todas as alocações possibilitam que para o maior número de atividades seja alocado algum profissional, geralmente há uma série de diferentes combinações de alocação possíveis. A falta de um processo que apóie o administrador, de forma a reduzir o esforço necessário para sua realização, certamente é um grande influenciador para o fracasso.

Neste contexto, pretende-se desenvolver uma ferramenta de alocação de recursos humanos que considere os fatores determinantes em projetos, através da utilização de algoritmos de busca e através de modelos definidos pelo PMBOK, utilizando também como base informações de profissionais de gerência de projetos.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver uma ferramenta de auxílio ao planejamento de alocação de recursos humanos em projetos.

1.2 ESTRUTURA DO TRABALHO

O presente trabalho está organizado em quatro capítulos. O capítulo 2 apresenta os aspectos teóricos estudados para o desenvolvimento do trabalho. São abordados temas como gerência de recursos humanos, busca Tabu e tecnologias aplicadas no trabalho. Também são relacionados alguns trabalhos correlatos.

No capítulo 3 relatou-se o desenvolvimento do presente trabalho, detalhando os requisitos do protótipo, a especificação e a implementação. São apresentados os resultados encontrados com a finalização do trabalho.

Por fim, o capítulo 4 traz conclusões deste trabalho, bem como alguns aspectos que ficaram em aberto, servindo de sugestões para futuras extensões.

2 FUNDAMENTAÇÃO TEÓRICA

Na seção 2.1 apresenta-se uma visão geral do guia PMBOK que fundamenta alguns conceitos de gerência de recursos em projetos. Na seção 2.2 apresenta-se uma visão de alocação de profissionais em projetos de TI. Na seção 2.3 mostra-se o algoritmo de pesquisa Tabu. A seção 2.4 e 2.5 mostra respectivamente o *framework* ASP.NET MVC e o *framework* Ext Js utilizados para desenvolvimento do protótipo. Na seção 2.6 são descritos trabalhos correlatos.

2.1 PROJECT MANAGEMENT BODY OF KNOWLEDGE (PMBOK)

O principal objetivo do guia PMBOK é identificar o subconjunto do conjunto de conhecimentos em gerenciamento de projetos que é amplamente reconhecido como boa prática (PROJECT MANAGEMENT INSTITUTE, 2004, p. 3). Foi criado pelo Project Management Institute (PMI), organização sem fins lucrativos de profissionais da área de gerenciamento de projetos. Tem como objetivo promover e ampliar o conhecimento existente sobre gerenciamento de projetos assim como melhorar o desempenho dos profissionais e organizações da área (PROJECT MANAGEMENT INSTITUTE, 2008).

Segundo Xavier (2005, p. 4) o PMBOK sugere quais processos devem ser executados, durante o gerenciamento de projetos, nas áreas de escopo, tempo, custo, recursos humanos, comunicação, risco, aquisições e qualidade, propondo também um conjunto de processos para a integração destas áreas.

Neste contexto podem-se destacar as áreas de gerenciamento de tempo e gerenciamento de recursos humanos. O gerenciamento de tempo descreve os processos relativos ao término do projeto no prazo correto e estimativa de recursos. Já gerenciamento de recursos humanos descreve os processos que organizam e gerenciam a equipe do projeto (PROJECT MANAGEMENT INSTITUTE, 2004, p. 10).

A Figura 1 mostra uma visão geral do gerenciamento de recursos humanos do projeto segundo o PMBOK.



Fonte: Project Management Institute (2004, p. 201).

Figura 1 - Visão geral do gerenciamento de recursos humanos

2.2 ALOCAÇÃO DE PROFISSIONAIS EM PROJETOS

Para Vargas (2003, p. 101) o gerenciamento dos recursos humanos tem como objetivo central fazer o melhor uso dos indivíduos envolvidos no projeto. Todos os resultados do projeto podem ser vistos como fruto das relações humanas e das habilidades interpessoais dos envolvidos. O processo de gestão de recursos humanos garante o fornecimento de uma equipe com experiência, capacitada e qualificada para a realização de processos de ciclo de vida de modo a alcançar os objetivos da organização, do projeto e dos clientes (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2009).

As pessoas que trabalham em uma organização de software são seu maior patrimônio. Elas representam o capital intelectual e é responsabilidade dos gerentes de software garantir que a organização obtenha o melhor retorno de seu investimento em pessoas (SOMMERVILLE, 2003, p. 418).

2.2.1 Planejamento de recursos humanos

O planejamento de recursos humanos determina funções, responsabilidades e relações hierárquicas do projeto (PROJECT MANAGEMENT INSTITUTE, 2004, p. 199). Uma forma de ilustrar as conexões entre um trabalho que precisa ser realizado e membros da equipe do projeto é através de uma Matriz de Responsabilidades (MR) como ilustrado na Figura 1.

Segundo Project Management Institute (2004, p. 206), o formato matricial, às vezes chamado de tabela, permite visualizar todas as atividades associadas a uma pessoa ou todas as pessoas associadas a uma atividade.

A matriz apresentada a seguir recebe a denominação de gráfico RACI (do inglês *Responsible, Accountable, Consult, Inform*), pois os nomes das funções documentadas são Responsável, Reporta-se, Consultoria e Informar. À esquerda ficam trabalhos a serem realizados, atividades, podendo ser exibidos em vários níveis. A coluna pessoa pode exibir pessoas ou grupos.

Gráfico RACI	Pessoa				
Atividade	Ana	Beto	Carlos	Dina	Edu
Definir	A	R	I	I	I
Projetar	I	A	R	C	C
Desenvolver	I	A	R	C	C
Testar	A	I	I	R	I

R = Responsável A = Reporta-se C = Consultoria I = Informar

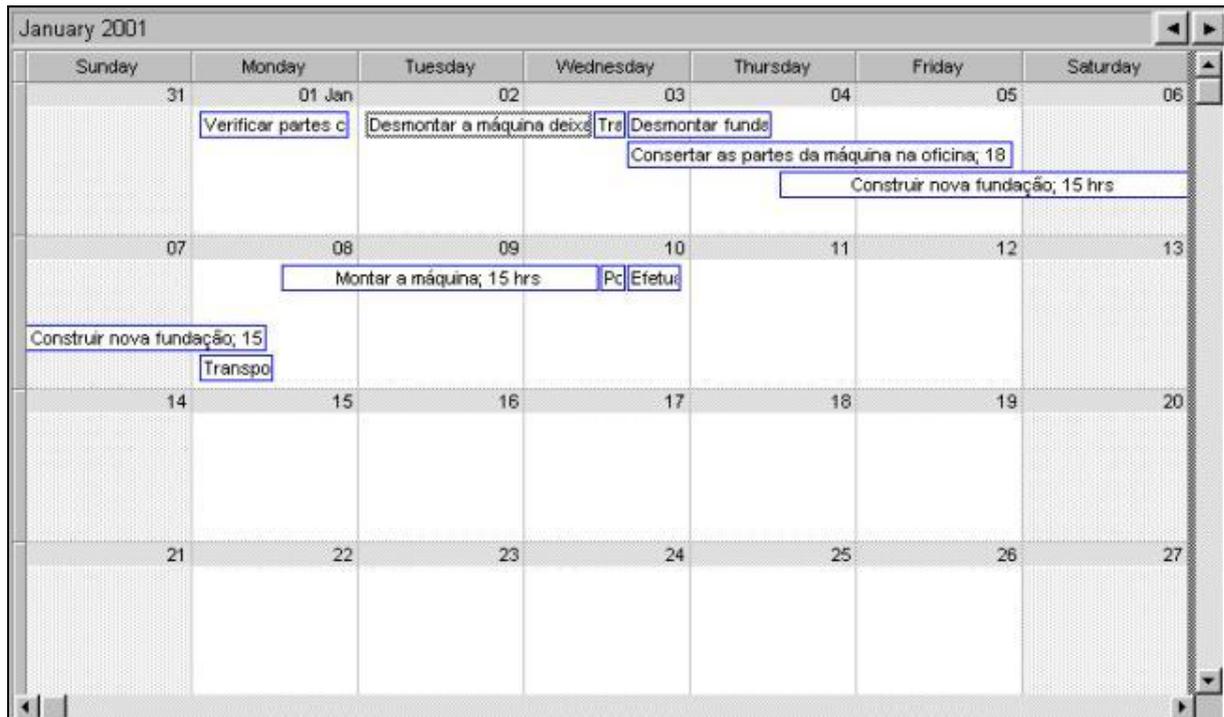
Fonte: Project Management Institute (2004, p. 206).

Figura 1 - Matriz de responsabilidades

2.2.2 Calendário de recurso

Um calendário de recurso do projeto documenta os dias trabalhados e os dias não trabalhados que determinam as datas nas quais um recurso específico, uma pessoa ou material, pode estar ativo ou ocioso (PROJECT MANAGEMENT INSTITUTE, 2004, p. 138). Heldman (2005, p. 173) diz que o calendário de recurso registra também as datas de início das tarefas, citando o recurso alocado à tarefa e a duração esperada. Normalmente o calendário de recurso do projeto identifica feriados específicos de recursos e períodos de disponibilidade de recursos. O calendário de recurso do projeto identifica a quantidade de cada recurso disponível durante cada período de disponibilidade (PROJECT MANAGEMENT INSTITUTE, 2004, p. 138).

A Figura 2 mostra um modelo de calendário do projeto. Pode-se notar que existem dias onde apenas uma tarefa deve ser executada, 01 de janeiro, e dias em que existem várias tarefas a serem cumpridas como dia 04 de janeiro.



Fonte: Heldman (2005, p. 173).

Figura 2 - Exemplo de calendário do projeto

2.2.3 Atributos da atividade e recurso

Heldman (2005, p. 123) diz que os interesses e as características pessoais também devem ser levados em consideração quando se está alocando pessoal. Deve-se considerar vários fatores durante o planejamento da equipe do projeto:

- habilidades necessárias para cada tarefa ou grupo de tarefas;
- capacidade para aprender coisas novas;
- conhecimento;
- personalidade;
- disponibilidade;
- experiência;
- capacidade de trabalhar bem com os outros.

Esses fatores devem coincidir com os atributos das atividades do projeto. Esses atributos são usados para o desenvolvimento do cronograma do projeto e para a seleção, ordenamento e classificação das atividades planejadas do cronograma de várias maneiras dentro dos relatórios. O número de atributos varia por área de aplicação (PROJECT MANAGEMENT INSTITUTE, 2004, p. 130).

2.2.4 Modelo de Alocação

Brasil Filho et al. (2006) propõe um modelo de alocação de recursos humanos em projetos de TI, buscando atender as seguintes restrições:

- a) um profissional pode exercer mais de uma função: um profissional pode exercer várias funções simultaneamente em um projeto;
- a) o profissional deve ser alocado somente aos perfis para os quais ele tem competência: de acordo com o seu conhecimento, habilidades e experiências indica os perfis que podem ser exercidos por um profissional;
- b) o profissional pode estar alocado a mais de um projeto em um mesmo período: o mesmo profissional poderá participar de mais de um projeto ao mesmo tempo;
- c) o percentual mínimo de alocação de cada profissional a um projeto é definido pelo perfil: indica que cada perfil possui um percentual mínimo de alocação em um projeto que deverá ser observado no momento de alocação do profissional ao projeto;
- d) o percentual máximo de alocação do profissional não deverá ultrapassar sua carga horária: permite que se tenham profissionais com cargas-horárias diferentes, as quais, estarão sendo respeitadas dentro do modelo de alocação;
- e) a alocação dos profissionais deverá atender a demanda de perfis dos projetos: cada projeto tem a sua demanda específica¹ de perfis de profissionais, que varia ao longo do tempo de sua execução.

2.3 ALGORITMO DE PESQUISA TABU²

Para a criação de alocações mais otimizadas, considerando todos os fatores que a influenciam, existem alguns algoritmos. Esses tentam encontrar sempre a melhor opção, para cada uma das diversas ocorrências encontradas nessa situação.

Para Silva (2000), a busca Tabu, é uma técnica de procura que consiste em selecionar

¹ As demandas são definidas pelos gerentes de projeto de acordo com o escopo e orçamento definidos e negociados para o projeto (Brasil Filho et al., 2006).

² Tradução literal do inglês: *Tabu search*.

apenas parte das melhores soluções da vizinhança, mantendo uma tabela de movimentos proibidos para impedir que o algoritmo retorne a soluções já visitadas.

Segundo Pigatti (2003, p. 47) vários experimentos computacionais mostraram que a busca tabu é uma técnica de aproximação muito boa, que pode competir com quase todas as demais técnicas conhecidas e que, por sua flexibilidade, pode superar muitos procedimentos clássicos. A vantagem desta técnica é que mesmo que não exista uma solução ótima é garantido que as melhores soluções estão entre as últimas encontradas (SILVA, 2000).

Restrições Tabu são usadas para evitar que o processo de busca cicle por soluções já visitadas. Estratégias de intensificação visam explorar características históricas desejáveis enquanto estratégias de diversificação forçam a busca a examinar regiões não visitadas (PIGATTI, 2003, p. 50).

Segundo Pigatti (2003, p. 46) a busca Tabu é um método iterativo. O passo geral de um método iterativo consiste em construir, partindo de uma solução corrente i , uma próxima solução j e checar se deve parar o processo ou executar um novo passo.

Ainda para Pigatti (2003, p. 50) a busca Tabu utiliza uma coleção de princípios de solução inteligente de problemas, que são baseados no uso de memória adaptativa usada para guiar o processo de busca para sair de ótimos locais e obter soluções próximas do ótimo global.

A seguir são mostrados os passos de execução do algoritmo, onde i é a solução, i^* é a última solução possível, k o contador de iterações, T_t tabela de restrições e A_t tabela de regras a serem atendidas:

- a) escolher uma solução inicial i . Fazer $i^* = i$ e $k = 0$;
- b) fazer $k = k + 1$ e gerar um novo estado para i onde nenhuma condição da T_t seja violada e no mínimo uma das condições de aspiração A_t seja atendida;
- c) se $i < i^*$, então fazer $i^* = i$;
- d) atualizar as condições tabu e de aspiração;
- e) se uma condição de parada for encontrada, então parar. Se não ir para o passo b.

2.4 ASP.NET MVC FRAMEWORK

Para esse trabalho foram utilizadas tecnologias atuais do contexto *web*, a fim de construir-se uma solução computacional recente e que contemple boas práticas de pesquisa e

mercado.

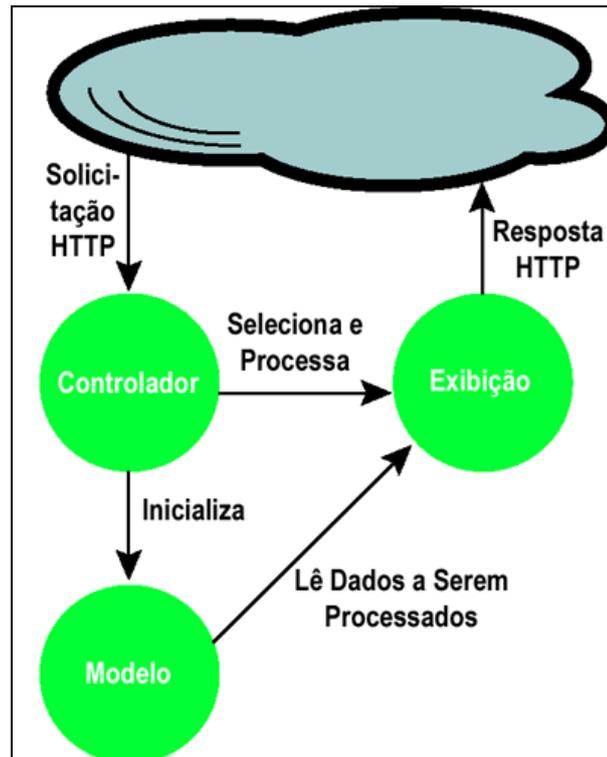
Segundo Microsoft Corporation (2009a) o padrão Model View Controller (MVC) é uma arquitetura projetada para separar os componentes de uma aplicação. Esta separação permite um maior controle sobre partes individuais da aplicação, tornando mais fácil desenvolver, testar e manter.

O padrão especifica onde cada tipo de lógica deve ser colocado na aplicação. A lógica da interface do usuário pertence à *View*. Entrada lógica pertence ao *Controller*. A lógica do negócio pertence ao *Model*. Esta separação ajuda a gerenciar a complexidade quando você constrói uma aplicação, pois permite você se concentrar em um aspecto da aplicação de cada vez. Por exemplo, você pode se concentrar na visão, sem depender de lógica de negócios. (MICROSOFT CORPORATION, 2009b).

Para Microsoft Corporation (2009b) o padrão MVC permite criar aplicativos que separam os diferentes aspectos da aplicação e da lógica (entrada, lógica de negocio e lógica do usuário), enquanto fornece um acoplamento entre esses elementos:

- a) *model*: encapsula alguns dados juntamente com a sua lógica de processamento, e é isolado da lógica de manipulação, que é encapsulado no *controller*. Ele pode definir algumas operações que manipulam esses dados encapsulados. Ele não sabe absolutamente nada sobre a interface gráfica do usuário e não exibe dados ou responde às ações que ocorrem na interface;
- b) *view*: é uma referência para o *model*. Ele usa os métodos apenas de leitura do modelo para consultar e recuperar dados. Que podem ser exibidos nas mais variadas formas, HTML, XML, Windows Forms, etc...
- c) *controller*: o objetivo do *controller* é interagir com o *model* e com o *view*. Ele sabe que o *model* prevê ações a serem executada sobre os dados, e sabe também que a interface vai enviar alguns eventos que podem exigir que estas operações sejam chamadas.

A Figura 3 mostra o fluxo de uma solicitação Hypertext Transfer Protocol (HTTP), em um contexto *web*, utilizando o padrão MVC.



Fonte: Tavares (2008).

Figura 3 - Fluxo de solicitação do padrão MVC

O ASP.NET MVC é uma implementação da arquitetura MVC para o ASP .NET em um framework com o objetivo de criar aplicações *web* no padrão MVC e fornecer uma alternativa ao modelo WebForm do ASP .NET disponível até então (MACORATTI, 2008).

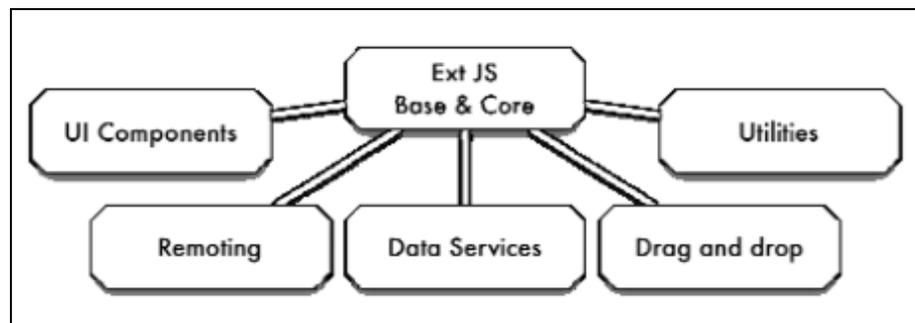
É desenvolvido e mantido pela Microsoft e distribuído sobre a licença Microsoft *Public Licence* (MS-PL). O ASP.NET MVC foi inspirado por outros *web frameworks* como Rails, Django, e Monorail, baseando-se em muitas das principais estratégias que essas outras plataformas MVC utilizam, além de oferecer os benefícios do código compilado e gerenciado e também explorar os novos recursos de linguagem do VB.NET e C# (CONERY et al., 2009, p. 188).

2.5 EXT JS

A biblioteca Ext JS foi originalmente criada como uma extensão do *framework* de JavaScript Yahoo User Interface (YUI). Como uma extensão, ele funcionava junto ao YUI, e fornecia funcionalidades e facilidades que o YUI não disponibilizava em sua Application Programming Interface (API) (BLADES; FREDERICK; RAMSAY, 2008, p. 10).

Blades, Frederick e Ramsay (2008, p. 11) dizem que aos poucos o Ext JS se tornou um verdadeiro *framework* de JavaScript não se limitando mais a ser apenas uma extensão para outros *frameworks*, apesar de ainda poder ser usado assim. O Ext JS fornece de forma fácil de usar, uma interface de usuário, parecido com as interfaces para aplicativos *desktop*. Isto permite aos desenvolvedores *web* concentrar-se na funcionalidade da aplicação em vez de ressalvas técnicas.

Ext JS além de fornecer recursos de interface de usuário também contém uma série de outras recursos a Figura 4 mostra a estrutura do *framework*.



Fonte: Garcia (2009, p. 9)

Figura 4 - Estrutura do Ext JS

Garcia (2009, p. 10) mostra como estão divididas as seis grandes áreas que formam a estrutura do Ext JS:

- a) *base e core*: disponibiliza recursos básicos para o funcionamento do *framework* como comunicação Asynchronous Javascript And XML (AJAX), manipulação de DOM e gestão de eventos;
- b) *UI components*: contém todos os *widgets*³, *gadgets*⁴ e demais estruturas para a interface visual com o usuário;
- c) *remoting*: é a estrutura responsável por executar remotamente, por meio de JavaScript, chamadas de método que são definidos e expostos no servidor, o que é comumente conhecido como chamada de procedimento remoto (*Remote Procedure Call – RPC*);
- d) *data services*: cuida de todas as necessidades de dados, que inclui coleta, análise e carregamento de informação para a interface do usuário. Pode realizar a comunicação entre cliente servidor através de Arrays, XML e JSON (JavaScript

³ *Widget*: termo de origem no inglês, refere-se a programas (leves, na maioria das vezes) que se tornam “atalhos” para serviços e utilidades (AMARAL, 2009).

⁴ *Gadgets*: do inglês a tradução literal é bugiganga, no contexto refere-se a rotinas que auxiliam no uso de estruturas maiores (AMARAL, 2009).

Object Notation);

- e) *drag and drop*: inclui todos os membros necessários para gerir toda a gama das operações de arrastar e soltar. Responsável pela existência da capacidade de arrastar e soltar de um componente do *framework* ou qualquer elemento HTML na página;
- f) *utilities*: é composto de rotinas de utilidade geral, que auxiliam nas mais variadas tarefas rotineiras.

2.6 TRABALHOS CORRELATOS

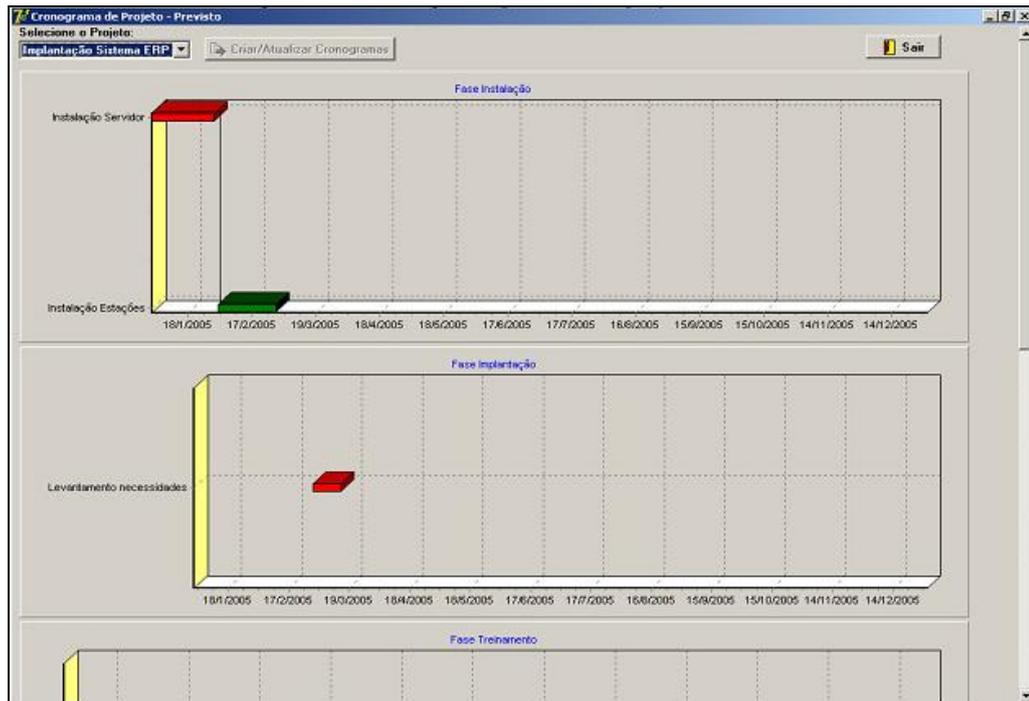
A seguir apresentam-se alguns trabalhos que possuem pontos em comum a este, sendo eles: Ferramenta para Gerenciamento de Tempo de Projetos (SOETHE, 2004), Ferramenta de Apoio ao Processo de Gerência de Recursos Humanos do MPS.BR (DEMO, 2008) e Resource Management (QUICKARROW, 2009).

2.6.1 Ferramenta para Gerenciamento de Tempo de Projetos

O objetivo geral deste trabalho foi desenvolver uma ferramenta de apoio ao gerenciamento de projetos (SOETHE, 2004, p. 54). Foram utilizadas as práticas sugeridas pelo PMBOK para fundamentar a pesquisa.

A ferramenta considera todas as atividades da Gerência de Tempo definidas pelo PMBOK, que são: Definição das Atividades, Seqüenciamento das Atividades, Estimativa da Duração das Atividades, Desenvolvimento do Cronograma e Controle do Cronograma. Além disso permite que seja visualizado as atividades consideradas críticas no projeto. (SOETHE, 2004, p. 54).

Foi utilizado Delphi 7 e Interbase 6 para implementação do protótipo. A Figura 5 mostra a tela de previsão do cronograma do projeto, onde são mostradas todas as fases, atividades e dependências das atividades.



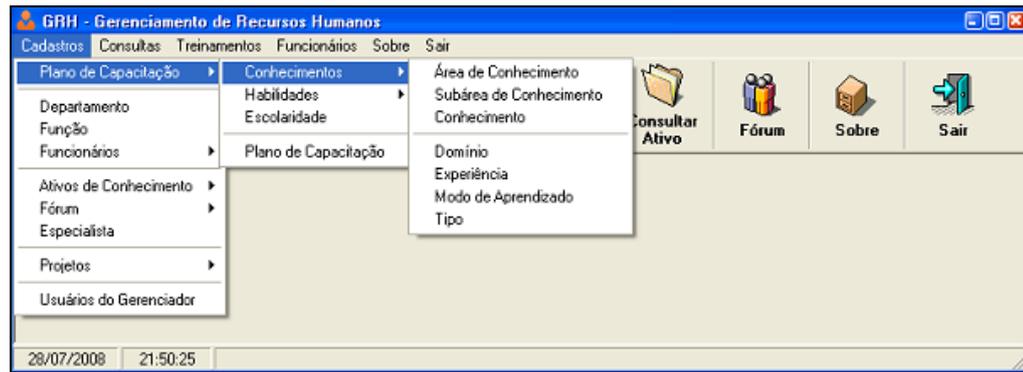
Fonte: Soethe (2004, p. 49).

Figura 5 – Previsão do cronograma

2.6.2 Ferramenta de Apoio ao Processo de Gerência de Recursos Humanos do MPS.BR

Demo (2008) define uma ferramenta para gerência de recursos humanos que utiliza como modelo um processo do programa de Melhoria de Processo do Software Brasileiro (MPS.BR). “Este modelo se baseia nos padrões de qualidade aceitos internacionalmente e utiliza o conhecimento existente nos modelos de melhoria de processo já disponíveis, porém foi adaptado ao contexto das empresas brasileiras” (DEMO, 2008, p. 6) (Figura 6).

Na sua implementação foi utilizado o ambiente Delphi 6 e MySQL para armazenar os dados.



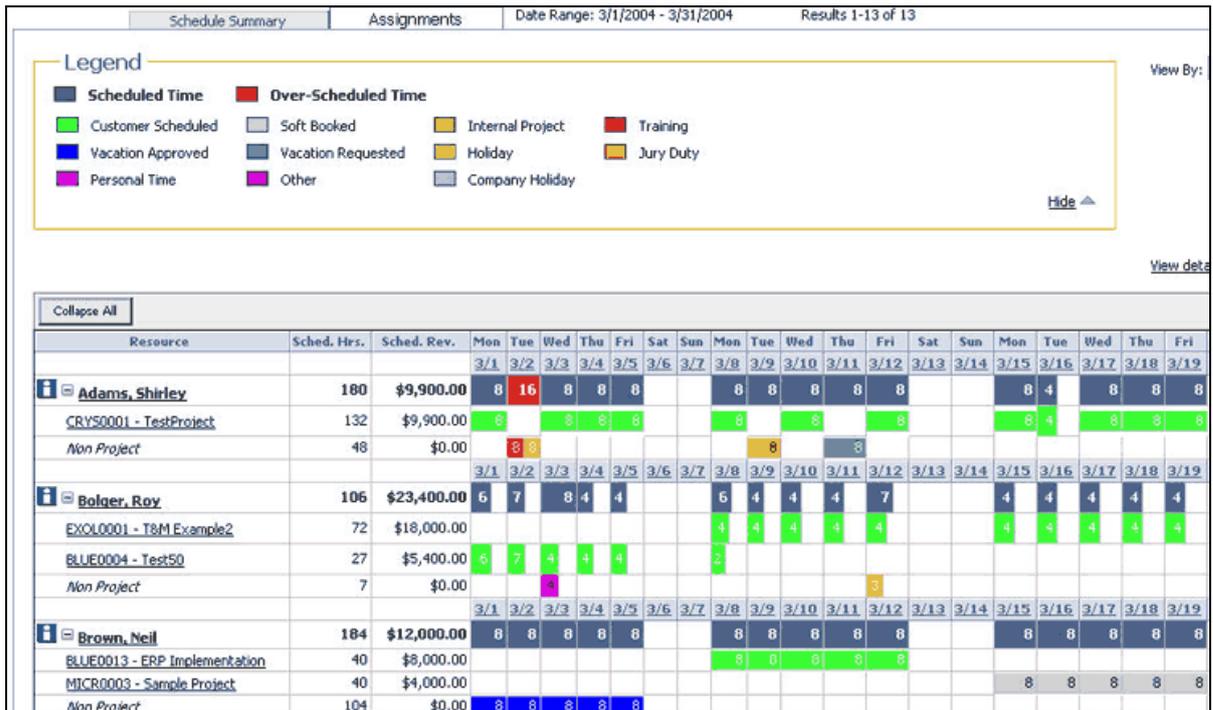
Fonte: Demo (2008, p. 70).

Figura 6 - Tela principal exibindo *menu* de cadastros

2.6.3 Resource Management

Este sistema foi desenvolvido e distribuído pela empresa QuickArrow, sob licença proprietária. É um conjunto de ferramentas de gestão de recursos que permitem aos gestores de projeto pesquisar e atribuir recursos adequados aos projetos com base em seu trabalho, habilidade, certificações, localização e disponibilidade (QUICKARROW, 2009).

Possui recursos de visualização, previsões e planejamento de alocação. Na Figura 7 é mostrado o calendário de recursos, exibindo graficamente as alocações por data e informações de valores e quantidades de horas de cada recurso.



Fonte: QuickArrow (2009).

Figura 7 - Interface do Resource Management mostrando tabela de alocação de recursos

3 DESENVOLVIMENTO

As seções seguintes descrevem a especificação, implementação e a operacionalidade do protótipo. A operacionalidade é apresentada com imagens das telas e exemplos de utilização. Por fim, são indicados os resultados obtidos com este trabalho.

3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

A partir do estudo das características das ferramentas pesquisadas, pode-se definir os requisitos do protótipo, que deve conter um módulo de cadastros, onde todas as informações necessárias para a execução da alocação serão cadastradas e mantidas, e um módulo de operação, onde o usuário poderá executar e visualizar as alocações. A seguir são apresentados os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF):

- a) sistema deverá permitir o cadastros de habilidades (RF);
- b) sistema deverá permitir o cadastro de competências (RF);
- c) o sistema deverá permitir o cadastro de conhecimentos (RF);
- d) o sistema deverá permitir o cadastro de equipes (RF);
- e) o sistema deverá permitir o cadastro de profissionais, relacionando-o com habilidades, competências e conhecimentos pré-cadastrados (RF);
- f) o sistema deverá permitir o cadastro de projetos, relacionando-o com profissionais (RF);
- g) o sistema deverá permitir alocação manual e automática de recursos (RF);
- h) o sistema deverá permitir a emissão de relatório de projetos / profissionais alocados (RF);
- i) o sistema deverá utilizar o algoritmo de busca Tabu para realizar a alocação de recursos (RNF);
- j) o sistema deverá apresentar uma interface em forma de tabela para alocação de recursos (RNF);
- k) o sistema deverá ser implementado em C# ASP.NET, utilizando o ambiente de programação Microsoft Visual Studio 2008 (RNF);
- l) o sistema deverá utilizar SQL Server para armazenar os dados (RNF);

m) o sistema deverá ser desenvolvido para web (RNF).

3.2 ESPECIFICAÇÃO

Na seqüência é apresentada a especificação do protótipo. Foi utilizada a ferramenta Microsoft SQL Server Management Studio para criação do Diagrama Entidade Relacionamento (DER). Para a criação do diagrama de classes da camada de persistência de dados foi utilizado o Microsoft Visual Studio. Os demais diagramas foram concebidos utilizando a ferramenta Enterprise Architect.

Foram utilizados conceitos da orientação a objetos e da Unified Modeling Language (UML) para a criação dos diagramas de casos de uso, classe e de atividades.

3.2.1 Diagrama de Casos de Uso

Os casos de uso do protótipo estão divididos em dois grupos: Cadastros, onde o ator informa as informações e configurações necessárias para a execução do sistema e Alocação que é o processo de distribuição de datas de disponibilidade ou restrições de projetos e recursos.

O diagrama de casos de uso do protótipo pode ser observado na Figura 8 e as descrições de seus casos de uso podem ser encontradas no Apêndice A. Para criação do diagrama foi utilizado a ferramenta Enterprise Architect.

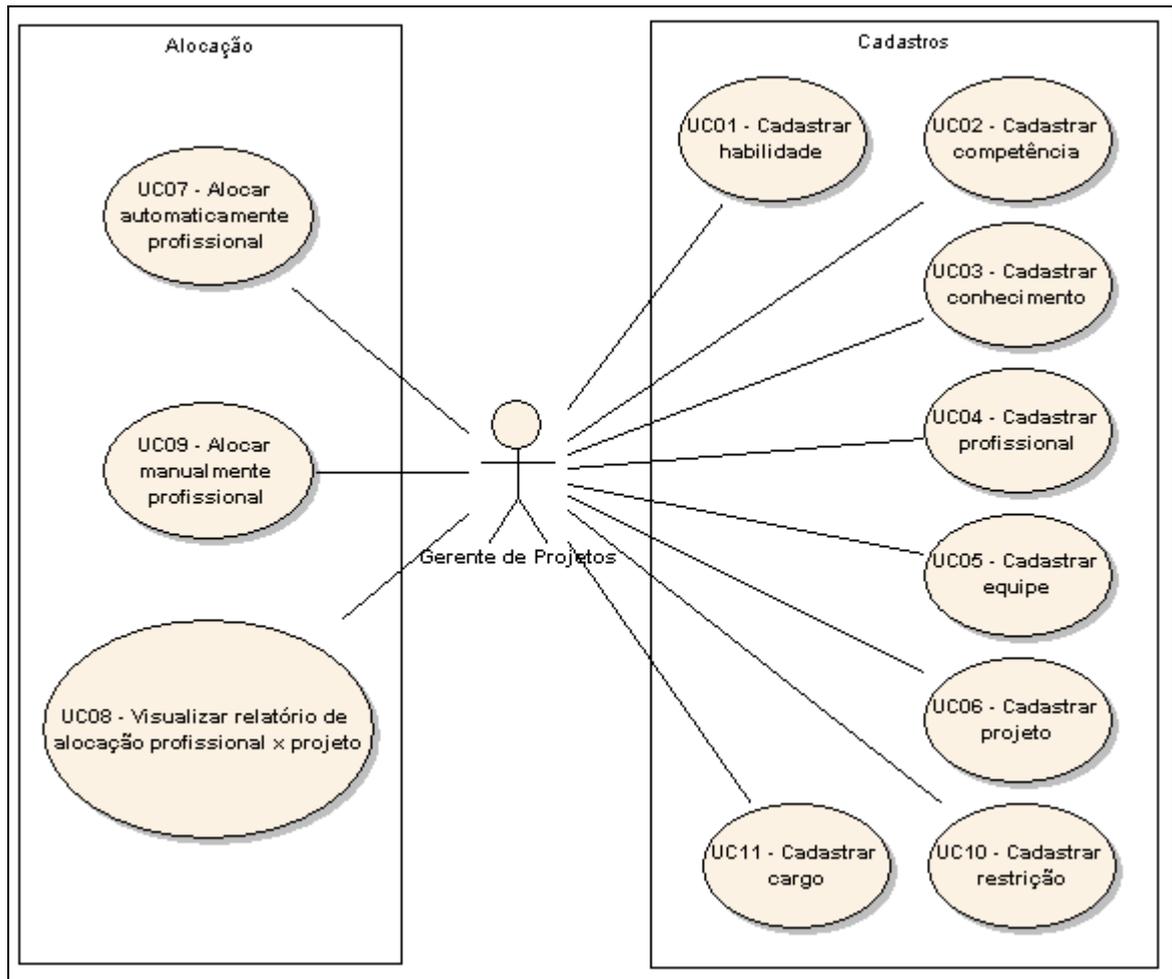


Figura 8 - Diagrama de casos de uso do protótipo

A seguir são apresentados breves comentários sobre cada caso de uso:

- a) UC01 – cadastrar habilidade: permite o cadastro e manutenção de habilidades utilizadas em profissionais e projetos;
- b) UC02 - cadastrar competência: permite o cadastro e manutenção de competências utilizadas no cadastro de profissionais e projetos;
- c) UC03 – cadastrar conhecimento: permite o cadastro e manutenção de conhecimentos utilizados em profissionais e projetos;
- d) UC04 – cadastrar profissional: permite o cadastro e manutenção dos profissionais, relacionando habilidades, competências, conhecimentos, restrições, equipe e cargo;
- e) UC05 – cadastrar equipe: permite o cadastro e manutenção de equipes de profissionais;
- f) UC06 – cadastrar projeto: permite o cadastro e manutenção de projetos relacionando habilidades, competências, conhecimentos, restrições e demais parâmetros para execução da alocação como: cor, data de início, duração, etc.;

- g) UC07 – alocar automaticamente profissional: permite a alocação automática dos profissionais em projetos selecionados;
- h) UC08 – alocar manualmente profissional: permite ao ator criar a alocação dos profissionais;
- i) UC09 – visualizar relatório de alocação profissional x projeto: permite ao ator visualizar a alocação dos profissionais em projetos selecionados;
- j) UC10 – cadastrar restrição: permite cadastrar e manter tipos de restrições que podem ser usadas em profissionais e projetos;
- k) UC11 – cadastrar cargo: permite o cadastro e manutenção de cargos, utilizado para profissionais.

3.2.2 Diagramas de classes

O diagrama de classes auxilia na visualização de como as classes estão estruturadas e quais suas relações. A seguir são apresentados três diagramas de classes que mostram como está estruturada a implementação do protótipo.

O primeiro, Figura 9, fornece uma visão das classes que compõe as rotinas de geração das alocações, a classe `AfinidadeColecao` manipula os valores resultantes do cruzamento dos atributos dos projetos com dos recursos. Ela implementa a rotina que sumariza todas as afinidades encontradas.

A classe `Alocacao` juntamente com a classe `Calendario` manipulam dados referente a datas, tanto de projetos como de profissionais, essa datas indicam se pode existir atividades ou restrições. `Alocacao` reúne os dados do profissional, projeto e calendário, é responsável por calcular os intervalos de atividades e ociosidades do recurso.

A classe `Dia` contém uma coleção de restrições, caso existam, para a data que representa, assim podem existir varias restrições coincidentes. Através do método `SituacaoDiaPorData` da classe `Calendario` é possível determinar a situação do profissional para aquele projeto na data deseja. Essa rotina é utilizada para montagem do calendário de recursos na camada de apresentação.

A classe `AlocacaoController` recebe as requisições do navegador, e processa parte dos dados. Porém, o acesso a camada de persistência de dados e manipulação dessas informações, é realizada pela camada de modelos.

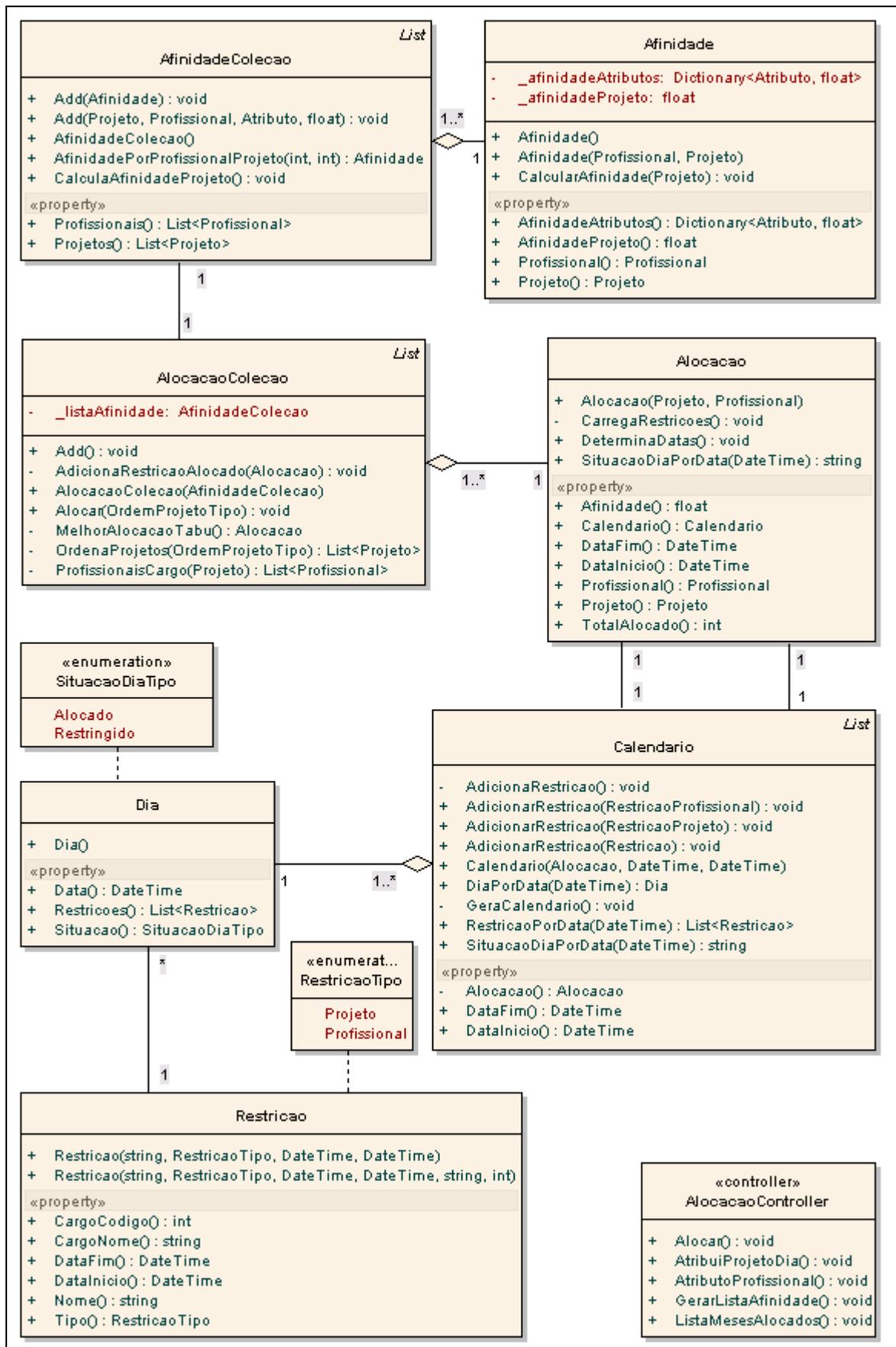


Figura 9 – Diagrama de classes da rotina de alocação

A Figura 10 apresenta as classes da implementação do algoritmo de busca Tabu, a

classe `ListaTabu` implementa a coleção de restrições para as alocações, o item 3.3.2.3 mostra detalhes da implementação.

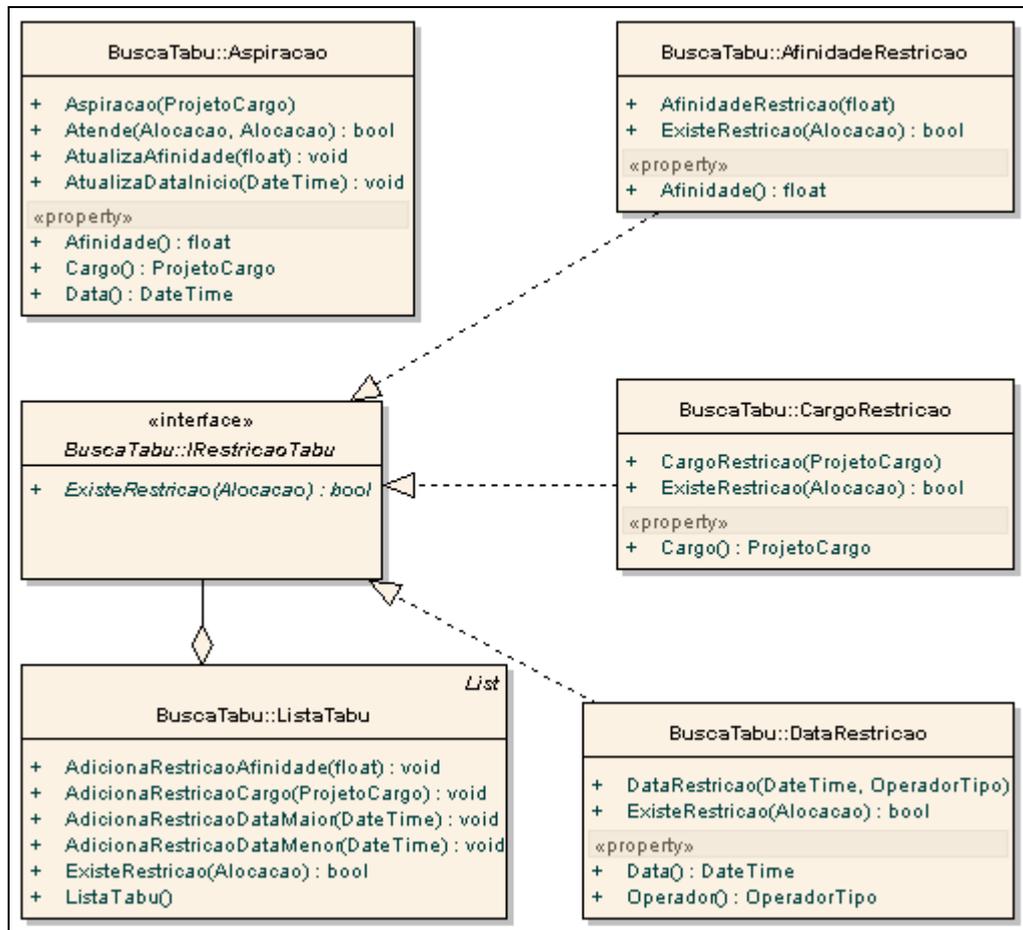


Figura 10 – Diagrama de classes da rotina de busca

A Figura 11 mostra as classes da camada de persistência de dados do protótipo, implementam métodos de acesso a dados abstraindo a geração dos comandos Structured Query Language (SQL).

3.2.3 Diagrama de seqüência

Nessa sessão é apresentado o diagrama de seqüência que representa o conjunto de ações que o programa executa para realizar a alocação automática de recursos. Para criação do mesmo foi utilizado a ferramenta Enterprise Architect, utilizando-se a especificação UML.

A Figura 12 mostra o diagrama, que equivale ao caso de uso UC07 – Alocar automaticamente profissional. Através dele pode-ser observar os passos e mensagens trocadas demonstrando o processo de geração das alocações.

O usuário escolhe a opção para gerar a alocação na tela do sistema, que envia uma requisição ao servidor de aplicação. Esta requisição é atendida pelo *controller* *AlocacaoController*. O processo de alocação é executado então pelas classes do *model* que ao final do processamento retornam ao *controller* os resultados.

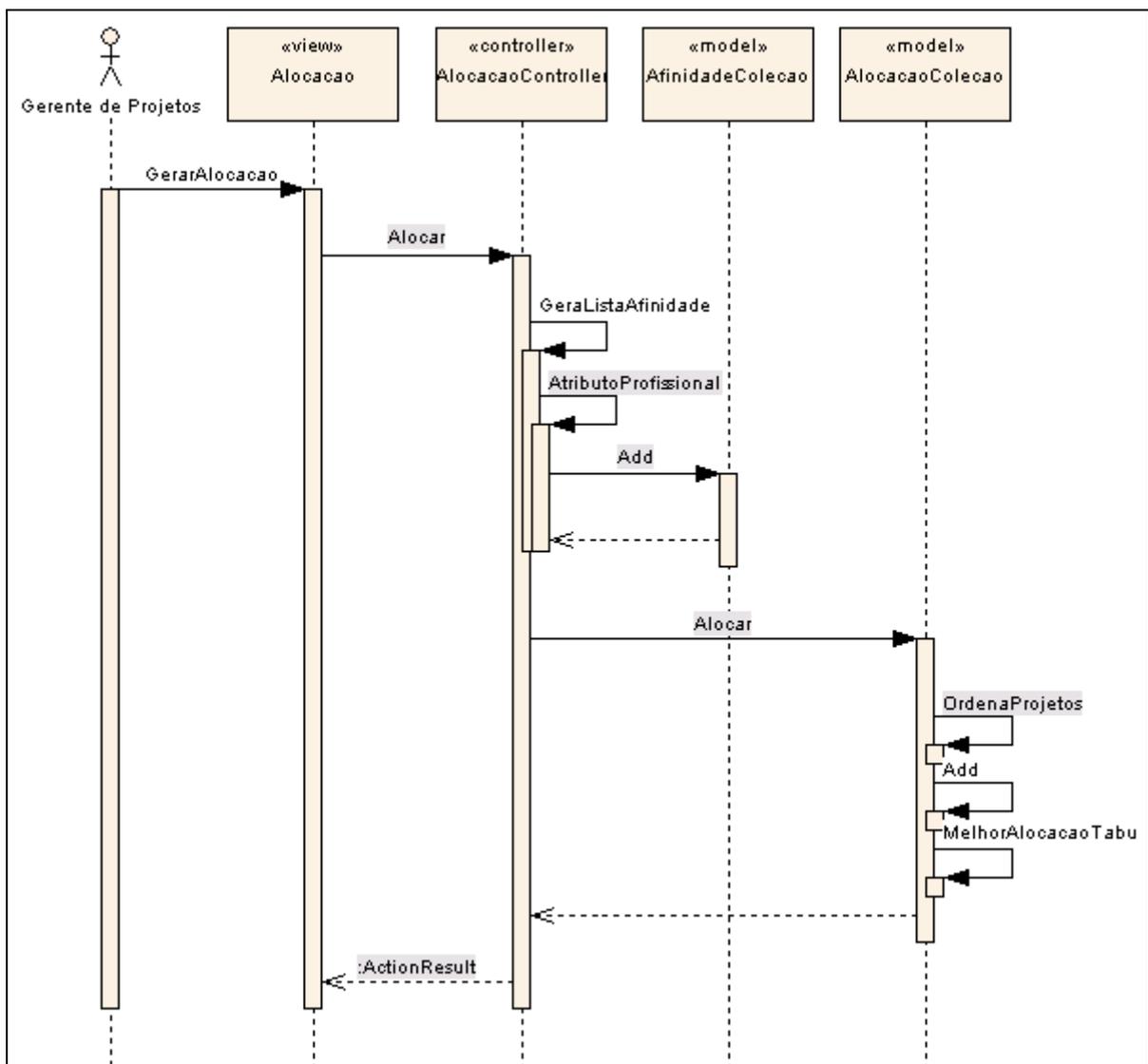


Figura 12 - Diagrama de seqüência para alocação automática de recursos

3.2.4 Diagrama de atividades

O processo de alocação automática dos recursos é dividido em seis fases principais:

- a) cálculo de afinidade profissional versus projeto;
- b) ordenação da lista de projetos;
- c) geração de calendário do profissional;
- d) preenchimento de restrições de projeto e profissional no calendário;
- e) escolha dos melhores calendários (alocações);
- f) apresentação dos resultados.

Cada fase está detalhada em subseções a seguir. A Figura 13 foi construída na ferramenta Enterprise Architect e apresenta o diagrama de atividades do processo automático de alocação com uma visão geral.

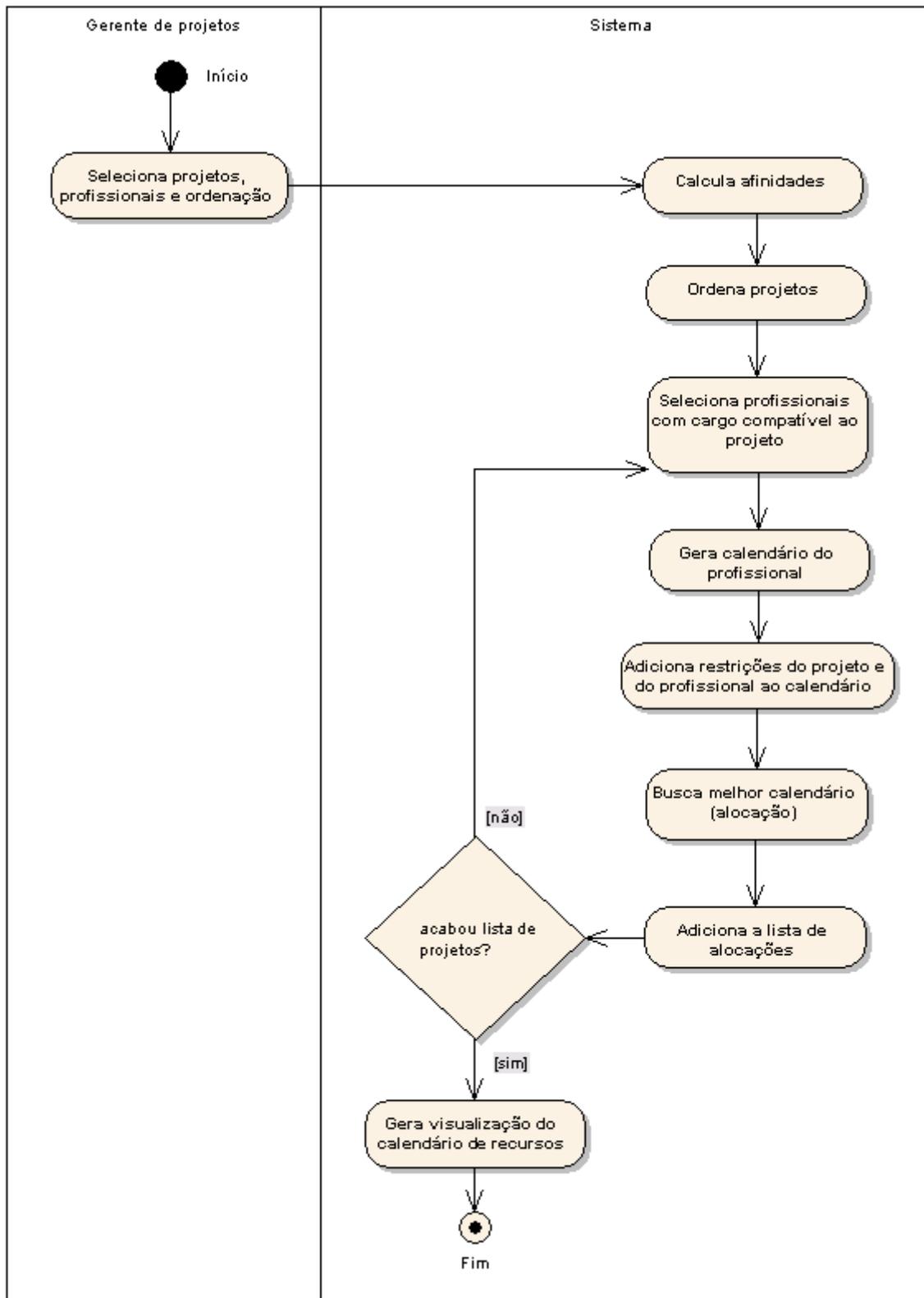


Figura 13 - Diagrama de atividades do processo de alocação

3.2.5 Cálculo de afinidade profissional por projeto

Esse processo é responsável pela geração da lista de afinidade entre profissional e projeto. Nessa etapa os atributos cadastrados nos profissionais e nos projetos são considerados. A lista de afinidade é uma coleção que contém o produto cartesiano entre profissionais e projetos atribuindo um valor de afinidade entre os envolvidos. Dessa forma pode-se determinar quanto o perfil do profissional é adequado ao perfil do projeto.

A lista de afinidades é utilizada pela busca Tabu para determinar qual é o profissional mais indicado a ser alocado para determinado período, considerando restrições e necessidades de projeto e profissional. A determinação desse valor de afinidade é mostrada no item 3.3 deste capítulo.

3.2.6 Ordenação da lista de projetos

De acordo com a opção informada pelo usuário é determinada a ordem em que os projetos serão alocados. Isso se mostra imprescindível diante da necessidade de priorizar determinados projetos de acordo com fatores influenciadores como prazo, concorrência, entre outros.

Optou-se em limitar as opções de ordenação em somente três sendo elas: data de início crescente dos projetos, tamanho em quantidade de horas decrescente e por fim tamanho em quantidade de horas crescente.

3.2.7 Geração de calendário do profissional

Calendário do profissional é o nome dado ao mapeamento de todos os dias que estão dentro do intervalo de início e fim do projeto ao qual o profissional está sendo alocado. Ou seja, todos os dias pertencentes ao período são considerados e analisados para determinação se o profissional pode ser alocado ou se existem restrições.

O calendário do profissional também pode ser chamado de alocação, porém nessa fase do processo não se sabe se essa alocação é a mais indicada. Assim quando a mesma for analisada pela busca Tabu e descartada, será ignorada e não utilizada.

O calendário do profissional contém informações de início e fim dos períodos que o profissional pode trabalhar no projeto, também contém o total de horas que esse período representa. Na seção 3.3 é mostrado em detalhes como ocorre o processo de geração dos calendários para cada profissional.

3.2.8 Preenchimento de restrições de projeto e profissional no calendário

As restrições cadastradas para projetos e profissionais são consideradas nessa fase do processo. Aqui elas são adicionadas aos calendários dos profissionais de forma a sinalizar que em determinada data, ou intervalo, o profissional não poderá trabalhar. Não poderá ser alocado em nenhum projeto (férias, atestados médico, folgas, etc...) ou que determinado projeto não poderá existir atividades.

Para projetos podem existir restrições gerais que limitam as atividades para todos os profissionais (feriados, fins de semana, problemas de ambiente, datas de impostas pelo cliente, etc...) ou em nível de cargos, onde somente determinados cargos serão restringidos. Por exemplo, no início de um projeto, na fase de levantamento de requisitos não existirá atividades relacionadas à programação. Assim pode-se criar uma restrição para os cargos relacionados à programação, evitando que os mesmos contenham profissionais alocados nesse período.

3.2.9 Escolha dos melhores calendários

A Figura 14 mostra o fluxo de execução da rotina de escolha das alocações através da busca Tabu. Essa rotina é responsável pela escolha dos profissionais que melhor atendem as necessidades de cada projeto em determinado período. A busca Tabu atua sobre os calendários dos profissionais gerados duas fases antes, ela analisa qual calendário é o mais adequado para a situação e o determina como alocado, desconsiderando os demais.

Detalhes da implementação e execução são apresentados no item 3.3.

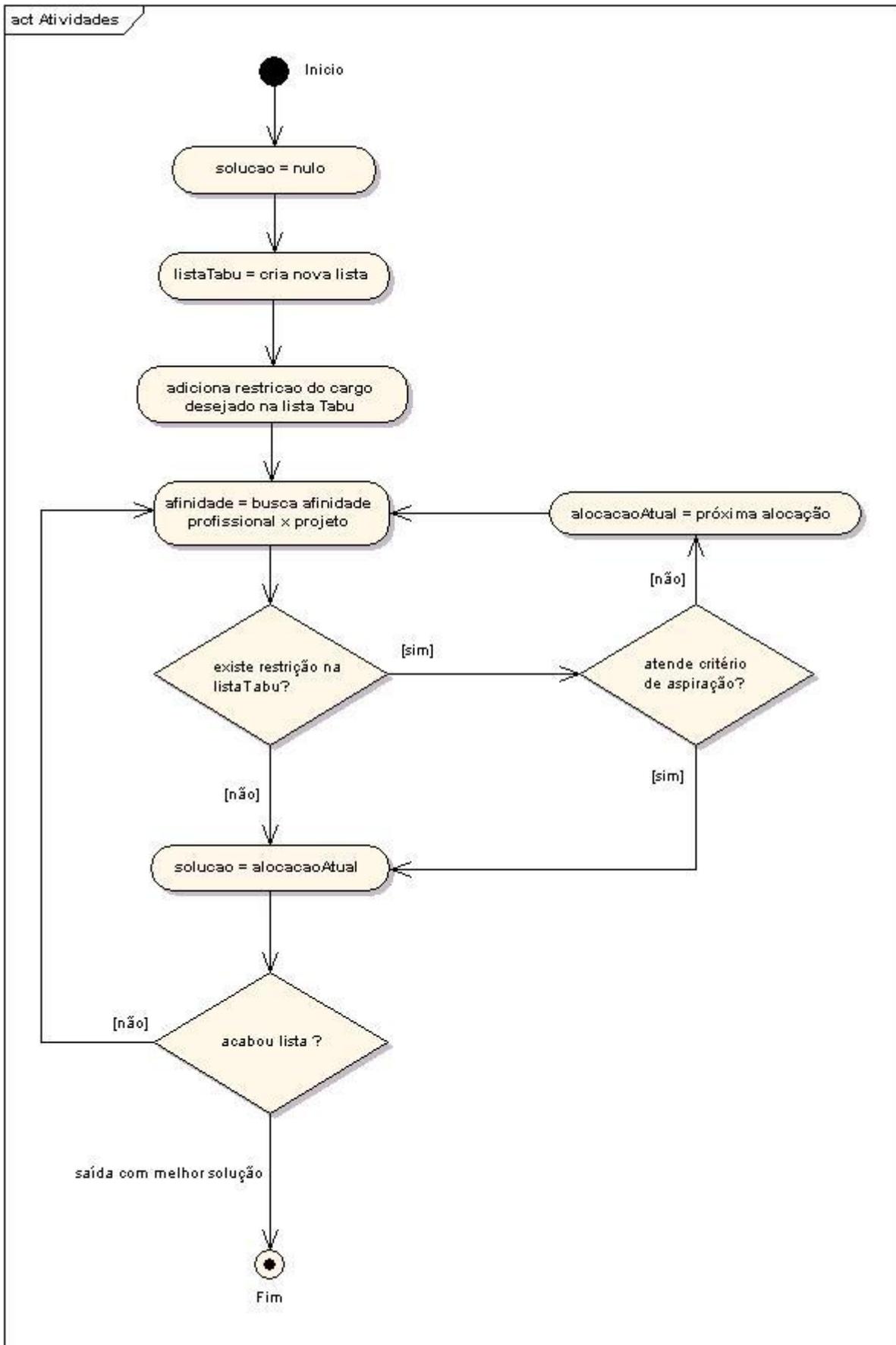


Figura 14 - Diagrama de atividades que representa a busca Tabu

3.2.10 Apresentação dos resultados

Após todo o processo de alocação ser realizado é necessário mostrar ao usuário os resultados. Essa fase é responsável por mostrar a alocação de forma amigável ao usuário, isso se dá em forma de calendário de recursos, onde são listados os profissionais escolhidos e os respectivos projetos alocados.

Informações sobre início e fim de intervalos de alocação ou restrição são agrupadas de forma a tornar possível seu entendimento. No item 3.3 é mostrado o esse processo e também algumas telas, onde se pode observar o resultado final dos processamentos.

3.2.11 Diagrama entidade relacionamento

Para representar os objetos no banco de dados, foi utilizada a ferramenta Microsoft SQL Server Management Studio. A ferramenta permite que as alterações realizadas no diagrama reflitam em tempo real nas entidades na base de dados. Assim não é necessário a criação e / ou execução de *scripts* para manutenção dos artefatos.

A Figura 15 mostra o diagrama de entidade relacionamento físico, onde pode-se observar as ligações entre as entidades. A chave ao lado do campo indica que o mesmo é ou faz parte da chave primária da tabela. Já a chave que aparece nas ligações indica que o campo está sendo levado para outra tabela como chave estrangeira.

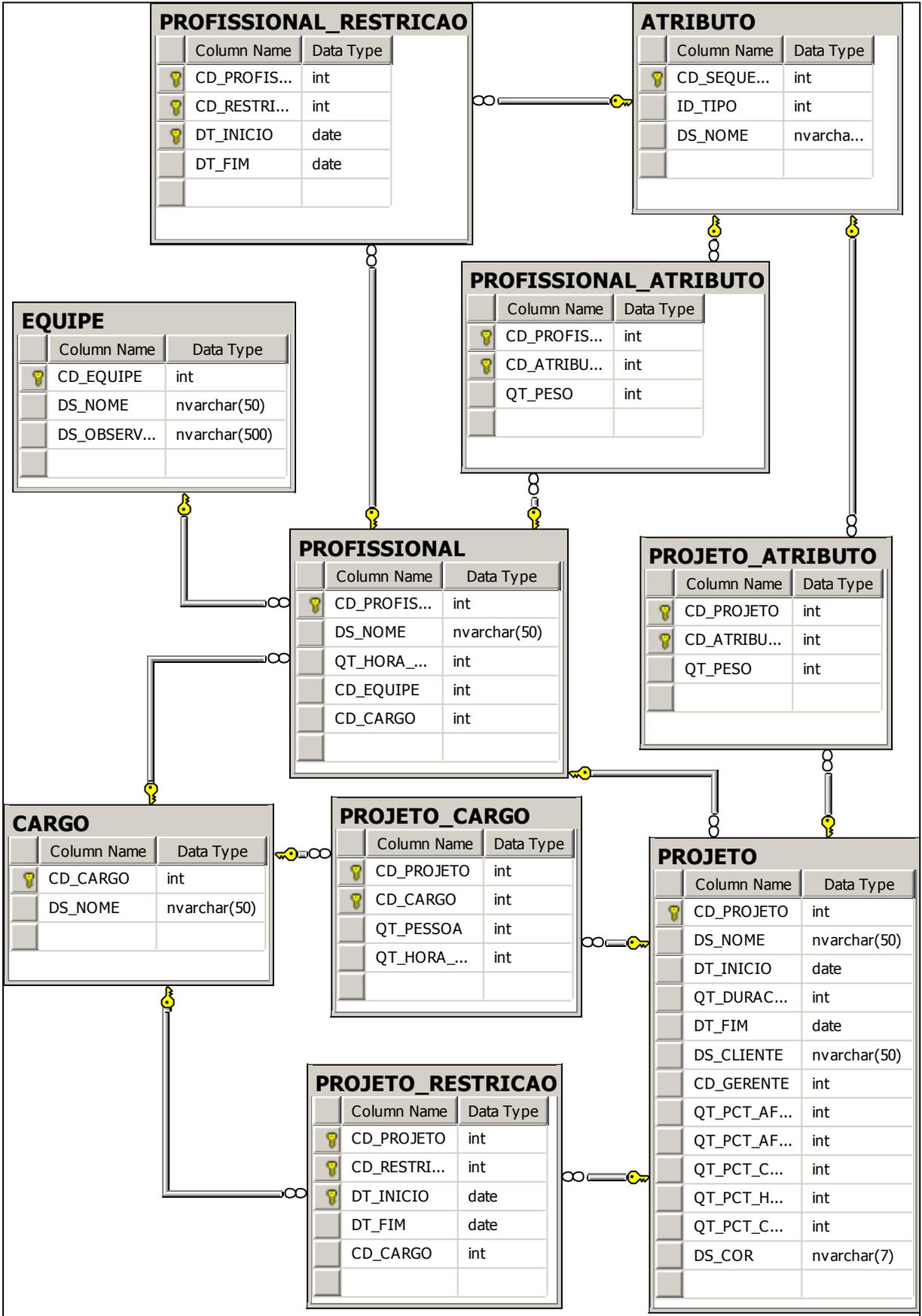


Figura 15 - Diagrama entidade relacionamento

3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação. Sendo apresentados blocos de códigos fontes e algumas telas da interface do usuário.

3.3.1 Técnicas e ferramentas utilizadas

O protótipo foi desenvolvido na linguagem C# / ASP.NET sobre a plataforma Microsoft .NET (versão 3.5). Para codificação foi utilizado o ambiente Microsoft Visual Studio 2008, também foram utilizados três *frameworks* a fim de agilizar o desenvolvimento e obter um melhor resultado. A seguir estão listados esses *frameworks* bem como um resumo de suas funcionalidades:

- a) ASP.NET MVC Framework: é a implementação da Microsoft para o modelo de arquitetura *model, view e controller* para a plataforma ASP.NET. Mais informações sobre esse *framework* foram mostradas no item 2.4 na fundamentação teórica;
- b) ExtJS: biblioteca Java Script que auxilia a criação da interface do usuário, acelerando o processo de desenvolvimento e fornecendo recursos avançados de interface, na fundamentação teórica (item 2.5) também foram apresentados maiores informações sobre o *framework*;
- c) Microsoft Entity Framework: é uma tecnologia de acesso a dados que ajuda diminuir o espaço semântico entre o mundo orientado a objetos e o relacional. O Entity Framework é uma biblioteca de alto nível que permite aumentar a abstração no acesso aos dados, dando a flexibilidade para que o modelo de dados faça sentido para a aplicação e para o desenvolvedor. O Entity Framework faz a leitura das entidades da base de dados e gera as classes para a aplicação. O código gerado não deve ser alterado pelo programador, pois a cada alteração na base dados esse código será re-gerado substituindo qualquer alteração manual. Permite também através da criação de diagrama a geração das classes para a aplicação. Nesse caso o desenvolvedor deverá fazer manualmente o relacionamento entre classes e entidades. No item 3.2.2 foi apresentado um diagrama gerado pelo *framework*.

Outra tecnologia bastante utilizada no desenvolvimento do protótipo foi Language Integrated Query (LINQ), segundo Microsoft Corporation (2009c) é uma extensão do .NET Framework 3.5 que engloba consulta, atribuição e operações de transformação de dados em praticamente qualquer fonte de dados. Através de uma sintaxe semelhante ao SQL é possível realizar pesquisas em bloco de informações, filtrar dados (vertical ou horizontalmente), quando necessário através da associação de operadores.

No protótipo essa tecnologia foi largamente utilizada para manipulação das informações no banco de dados e coleções em memória. O Quadro 1 mostra o método `OrdenaProjetos` que realiza a ordenação dos projetos na rotina de alocação automática de recursos, utilizando o LINQ para criar uma coleção de projetos na ordem definida pelo usuário.

```
private List<Projeto> OrdenaProjetos(OrdemProjetoTipo ordem)
{
    switch (ordem)
    {
        case OrdemProjetoTipo.DataInicio:
            return _listaAfinidade.Projetos.
                OrderBy(p => p.DataInicio).ToList();
        case OrdemProjetoTipo.TamanhoMaior:
            return _listaAfinidade.Projetos.
                OrderByDescending(p => p.Duracao).ToList();
        case OrdemProjetoTipo.TamanhoMenor:
            return _listaAfinidade.Projetos.
                OrderBy(p => p.Duracao).ToList();
        default: // ou OrdemProjetoTipo.Aleatorio
            return _listaAfinidade.Projetos;
    }
}
```

Quadro 1 - Método `OrdenaProjetos`, exemplo de utilização da tecnologia LINQ no protótipo.

Para criação do relatório foi utilizado a ferramenta `fyiReporting`, que trabalha com Report Definition Language (RDL), um padrão proposto pela Microsoft para definição de relatórios. A ferramenta permite a geração de relatórios para os formatos HTML, PDF, Excel, RTF, XML, .Net Control, Web Archive, e impressão direta. Para seu uso no protótipo, optou-se em apenas gerar no formato PDF.

3.3.2 Desenvolvimento do protótipo

A seguir são apresentados detalhes da implementação as rotinas que compõe a solução do problema.

3.3.2.1 Afinidade profissional por projeto

Como já apresentado no item 3.2.5 esse processo é responsável pela geração da lista de afinidade entre profissional e projeto. Nessa etapa os atributos cadastrados nos profissionais e nos projetos são considerados. É o primeiro a ser executado no momento em que o usuário inicia o processo de alocação automática.

A rotina consiste em cruzar os atributos cadastrados para os projetos com os atributos cadastrados para os profissionais, de acordo com os projetos e profissionais selecionados pelo usuário para alocação. Se o profissional possuir o atributo exigido pelo projeto então é determinado o valor de afinidade caso contrário o mesmo é inserido na lista com valor zero.

Para determinar o valor de afinidade é calculada a diferença entre os pesos cadastrados para o atributo do projeto com o do profissional. Essa diferença pode ser positiva (profissional tem menos conhecimento, habilidade ou competência que o projeto exige) ou negativa (profissional tem mais conhecimento, habilidade ou competência que a exigência do projeto). Através de uma equação de 1º grau é determinado a percentagem que a diferença representa para o peso exigido. Esse valor percentual é aplicado à outra equação de 1º grau que utiliza os valores de ajuste informados pelo usuário, no cadastro do projeto, para cada situação (excesso ou falta de atributo). O Quadro 2 mostra a implementação do cálculo, que é executado para cada cruzamento de atributos, tendo assim como parâmetros de entrada um atributo de projeto e um atributo de profissional.

```

private float CalculaAfinidade(AtributoProjeto atributoProj,
    AtributoProfissional atributoProf)
{
    int diferenca = atributoProj.Peso - atributoProf.Peso;
    float pct = (100 * Math.Abs(diferenca)) / atributoProj.Peso;
    if (diferenca > 0)
    {
        // Falta de afinidade
        float ajuste =
            (pct * atributoProj.Projeto.AfinidadeInferior) / 100;
        return 100 - ajuste;
    }
    else if (diferenca < 0)
    {
        // Sobra de afinidade
        float ajuste =
            ((pct - 100) * atributoProj.Projeto.AfinidadeSuperior) / 100;
        return 100 - ajuste;
    }
    else
    {
        // Afinidade ideal
        return 100;
    }
}

```

Quadro 2 - Método CalculaAfinidade implementado na classe AlocaçãoController

O Quadro 3 mostra o método CalcularAfinidade responsável por somar todos os atributos calculados até então e gerar o valor de afinidade do profissional para com o projeto. Esse valor também é encontrado através de uma equação de 1º grau que considera os pesos, de cada tipo de atributo, informados pelo usuário no cadastro do projeto.

```

public void CalcularAfinidade(Projeto projeto)
{
    // Conhecimento
    float conhecimento =
        _afinidadeAtributos.Where(e => e.Key.Tipo == 0).Sum(a => a.Value);
    // Habilidade
    float habilidade =
        _afinidadeAtributos.Where(e => e.Key.Tipo == 1).Sum(a => a.Value);
    // Competencia
    float competencia =
        _afinidadeAtributos.Where(e => e.Key.Tipo == 2).Sum(a => a.Value);

    _afinidadeProjeto =
        ((conhecimento * projeto.PesoConhecimento) / 100) +
        ((habilidade * projeto.PesoHabilidade) / 100) +
        ((competencia * projeto.PesoCompetencia) / 100);
}

```

Quadro 3 - Método CalcularAfinidade implementado na classe Afinidade

A determinação de um valor de afinidade é crucial para que a busca encontre o recurso que melhor se adapta as exigências do projeto. Imagine a existência de dois projetos, um de alta complexidade e um simples, e que exista dois recursos a serem utilizados para execução

das atividades, um recurso possui boa experiência no trabalho enquanto o outro ainda é um aprendiz. A busca identifica que o profissional de maior experiência possui atributos acima dos exigidos pelo projeto simples, o que seria um desperdício de recurso, e que o aprendiz está abaixo do esperado para o projeto complexo, o que provavelmente trará problemas de prazo e até qualidade ao longo da execução das atividades.

Assim a busca tentará alocar sempre o profissional mais apto ao projeto, no item 3.3.2.3 é mostrado em detalhes como a busca Tabu executa essa análise sobre os valores calculados nessa rotina.

3.3.2.2 Geração de calendário do profissional

O processo de geração do calendário do profissional é responsável por determinar os dias em que o profissional poderá ser alocado e também os dias em que existirão restrições, sejam elas do recurso ou do projeto.

A criação do calendário para um profissional ocorre em duas etapas: criação de uma alocação e adição de restrições. A primeira etapa consiste em criar uma alocação (Figura 16) para cada profissional, essa alocação será posteriormente analisada pela busca Tabu e se não for adequada ao projeto será descartada. Cada alocação possui um calendário (Figura 17) que contém todos os dias a partir da data de início até a data fim do projeto. No construtor da classe `Alocacao`, após a criação do calendário, o método `CarregaRestricoes` é invocado adicionando ao calendário todas as restrições cadastradas para o projeto e para o profissional.

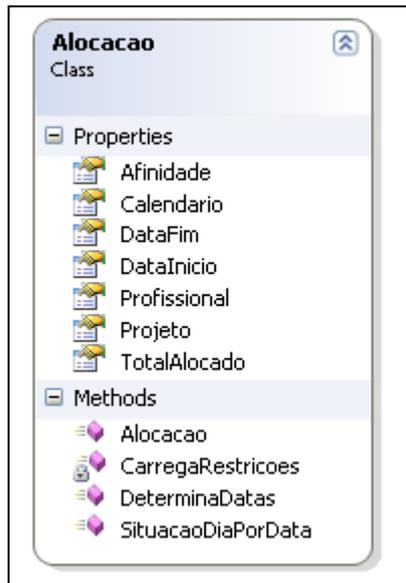


Figura 16 - Classe Alocacao

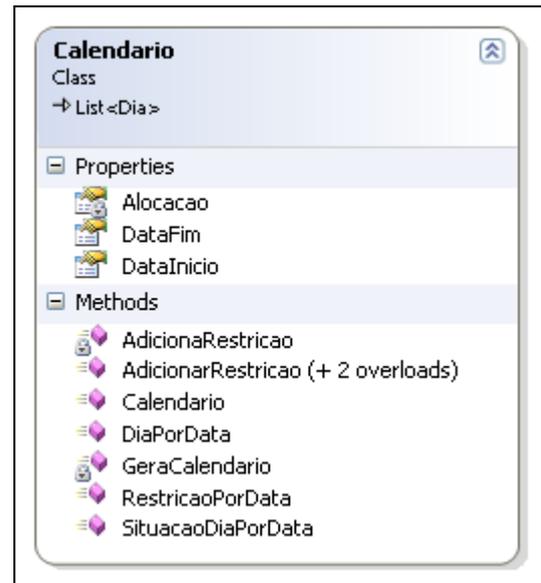


Figura 17 - Classe Calendario

A segunda etapa do processo de geração do calendário ocorre ao término da criação da alocação, é quando são adicionados ao calendário os dias em que o profissional já está alocado para outros projetos, esses dias são marcados em forma de restrição para o calendário, tornando impossível que o profissional seja alocado para outro projeto em um destes dias. O protótipo considera que um recurso só poderá ser alocado em um projeto por dia.

Após a finalização deste processo, o método `DeterminaDatas` da classe `Alocacao`, calcula as datas de início e fim da alocação do profissional para o projeto. Isso é realizado com base na quantidade de horas informado pelo usuário para o cargo do recurso no cadastro do projeto. Assim, mesmo que o recurso tenha disponibilidade para o trabalho não será alocado além desse valor. A propriedade `TotalAlocado` contém esse total de horas alocadas para o profissional. Esse valor não será exatamente igual ao informado pelo usuário, pois o protótipo não considera um dia parcial de trabalho e sim o dia completo. Logo poderá existir a quantidade de horas trabalhadas pelo recurso em um dia, como diferença.

As alocações geradas são armazenadas em uma lista em memória e serão utilizadas pela busca Tabu para verificação da mais apropriada para o projeto corrente.

3.3.2.3 Busca Tabu

Como já mencionado no item 3.2.9 essa rotina é responsável pela escolha dos profissionais que melhor atendem as necessidades de cada projeto em determinado período. Esse processo é executado após a geração dos calendários para cada profissional em

determinado projeto. Como parâmetros de entrada essa rotina recebe a lista de alocações geradas no processo anterior, o cargo desejado para ser alocado e o projeto que irá receber a alocação.

A busca Tabu precisa de uma alocação inicial. Para isso é utilizado a primeira alocação da lista de alocações. A partir desse ponto a busca Tabu irá comparar essa alocação com as próximas e à medida que for encontrando alocações melhores, utilizará esta para as próximas comparações.

O Quadro 4 mostra o método `MelhorAlocacaoTabu`, que através da implementação do algoritmo de busca Tabu, encontra a melhor alocação de acordo com as restrições e necessidades do projeto para o período exigido.

```
private Alocacao MelhorAlocacaoTabu(List<Alocacao> alocacaoProjeto,
    ProjetoCargo cargo, Projeto projeto)
{
    // Solução inicial
    Alocacao solucao = null;

    // Inicializa lista Tabu (restrições) para o projeto
    ListaTabu tabu = new ListaTabu();

    // Cargo atual
    tabu.AdicionaRestricaoCargo(cargo);
    // Aspiração (desejado) para melhor resultado
    Aspiracao aspiracao = new Aspiracao(cargo);
    foreach (Alocacao alocacao in alocacaoProjeto)
    {
        Afinidade afinidade = _listaAfinidade.AfinidadePorProfissionalProjeto
            (alocacao.Projeto.Codigo, alocacao.Profissional.Codigo);
        alocacao.Afinidade = afinidade.AfinidadeProjeto;

        // Determina nova possível solução
        // Se não esta na lista Tabu (restrições) OU atende aspiração
        if (!tabu.ExisteRestricao(alocacao) ||
            aspiracao.Atende(alocacao, solucao))
        {
            solucao = alocacao;
            // Atualiza lista tabu com data inicial da alocacao,
            // ou seja, só sera melhor alocacao caso a data de inicio
            // se for maior que a atual
            tabu.AdicionaRestricaoDataMenor(alocacao.DataInicio);
            // Adiciona restrição da nova afinidade
            tabu.AdicionaRestricaoAfinidade(afinidade.AfinidadeProjeto);
        }
    }
    // Retorna melhor resultado
    return solucao;
}
```

Quadro 4 - Método `MelhorAlocacaoTabu` implementação do algoritmo de busca Tabu

Para determinar se uma alocação é melhor que outra, a busca considera os seguintes fatores: valor de afinidade do profissional o projeto, data de início da alocação, total de horas

alocados e cargo do profissional. Essa comparação é realizada através da lista Tabu e da aspiração desejada. A cada interação se melhores resultados para esses itens forem encontrados a busca atualiza a alocação escolhida. As próximas alocações só serão escolhidas caso atendam todos esses itens de uma melhor forma.

O objeto `tabu` do tipo `ListaTabu` é a coleção de restrições existentes para o projeto. Essa classe é a implementação de uma lista de objetos do tipo `IRestricaoTabu` (Quadro 5). Contém apenas um método, `ExisteRestricao`, com a finalidade de retornar ao chamador se existe ou não uma restrição. Para a alocação passada como parâmetro, com aquele item da coleção, assim vários tipos de restrições podem ser adicionados a lista Tabu bastando ser criada uma classe que implemente a interface. Para o protótipo foram criadas três classes de restrições: `AfinidadeRestricao` (Figura 18), `CargoRestricao` (Figura 19) e `DataRestricao` (Figura 20) sendo que cada classe possui os atributos necessários para sua validação.

```
using AlocacaoRecursos.Models.Alocacao;

namespace AlocacaoRecursos.Controllers.BuscaTabu
{
    public interface IRestricaoTabu
    {
        bool ExisteRestricao(Alocacao alocacao);
    }
}
```

Quadro 5 - Interface `IRestricaoTabu`

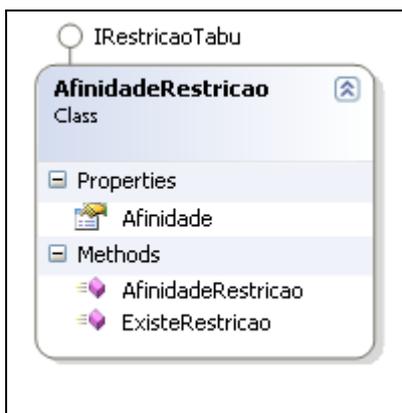


Figura 18 - Classe `AfinidadeRestricao`

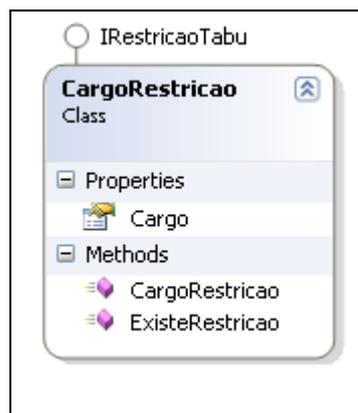


Figura 19 - Classe `CargoRestricao`



Figura 20 - Classe `DataRestricao`

As restrições vão sendo adicionadas a lista Tabu a cada alocação considerada como aceitável. Assim a lista vai se tornando cada vez mais restritiva considerando como melhor alocação somente as que atenderem todas as restrições já analisadas até então. Caso uma alocação não atenda a todas as restrições ela é avaliada pelo critério de aspiração, que na implementação é representado pelo objeto `aspiracao`, onde pode ser considerada aceitável

mesmo existindo restrições. Isso faz com que primeiro, as melhores opções (ou ideais) para solução sejam encontradas. Caso nenhum resultado seja determinado, a busca retorna uma alocação não ideal porém que atende o melhor possível a situação de acordo com as existentes.

3.3.2.4 Apresentação dos resultados em forma de calendário de recursos

Após o término da rotina de busca pelas melhores alocações, os resultados ficam armazenados na memória do servidor de aplicação, onde o protótipo está em execução. À medida que o usuário vai navegando pelos calendários de recursos gerados, os dados são enviados ao navegador através de requisições AJAX. O Quadro 6 mostra o código JavaScript para criação de um objeto JsonStore do *framework* ExtJs para realizar a carga dos dados do calendário de recursos.

```
var stoListaCalendario = new Ext.data.JsonStore({
    url: '/Alocacao/Calendario',
    autoLoad: true,
    baseParams: {
        mes: mes,
        ano: ano
    },
    root: 'data',
    fields: ['NomeProfissional', 'CodigoProfissional', 'TitleProfissional', 'MesAno',
        'D01', 'D02', 'D03', 'D04', 'D05', 'D06', 'D07', 'D08', 'D09', 'D10',
        'D11', 'D12', 'D13', 'D14', 'D15', 'D16', 'D17', 'D18', 'D19', 'D20',
        'D21', 'D22', 'D23', 'D24', 'D25', 'D26', 'D27', 'D28', 'D29', 'D30',
        'D31',
        'D01Cor', 'D02Cor', 'D03Cor', 'D04Cor', 'D05Cor', 'D06Cor', 'D07Cor',
        'D08Cor', 'D09Cor', 'D10Cor', 'D11Cor', 'D12Cor', 'D13Cor', 'D14Cor',
        'D15Cor', 'D16Cor', 'D17Cor', 'D18Cor', 'D19Cor', 'D20Cor', 'D21Cor',
        'D22Cor', 'D23Cor', 'D24Cor', 'D25Cor', 'D26Cor', 'D27Cor', 'D28Cor',
        'D29Cor', 'D30Cor', 'D31Cor']
});
```

Quadro 6 - Criação do objeto JsonStore para requisição do dados do calendário de recursos via AJAX

As requisições AJAX são enviadas diretamente as classes *controllers* que executam o processamento, através de consultas, utilizando a tecnologia LINQ, sobre os resultados na memória do servidor e retornam os dados no formato JSON, que são interpretados pelos componentes do ExtJs.

Os resultados das alocações são mostrados em forma de calendário de recursos, onde se pode observar a lista de profissionais com suas respectivas alocações. Esse método de visualização utiliza cores para diferenciar alocações entre projetos diferentes tornando a

visualização agradável e de fácil compreensão.

A Figura 21 mostra a tela principal de alocação. A direita fica a listagem dos projetos selecionados pelo usuário, ao centro é mostrado, de acordo com a aba selecionada, o calendário de recursos para a data selecionada na listagem da esquerda (Meses Alocados).

Ainda na barra da esquerda pode-se acessar os cadastros de projetos, profissionais, equipes e restrições. Na parte inferior da tela encontra-se uma área de texto onde são exibidos eventuais erros e mensagens no processo de alocação. Todas as barras podem ser ocultadas a fim de aumentar a área central de visualização, tornando mais agradável a utilização do sistema.

Recursos	Lista de afinidades	Lista de alocados	Calendário 01/2009															Projetos
Meses Alocados	Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	Projetos
01/2009	Jean Bastos	-	-	A	A	A	A	A	A	A	A	A	A	A	A	A	-	Projeto 1
02/2009	Briatori	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Projeto 2
03/2009	Nelson	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
04/2009	Jose Zeca	A	A	A	A	A	A	A	A	A	A	A	A	A	R	R	A	A
05/2009	Antonio Boleiro	-	-	A	A	A	A	A	A	A	A	A	A	A	A	A	A	-
06/2009	Doberval Coimbra	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
07/2009	Antonio Tavares	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
08/2009	Carlos Santos	A	A	A	A	A	A	A	A	A	A	A	A	R	R	A	A	
09/2009																		
10/2009																		
11/2009																		

Figura 21 - Tela de visualização das alocações

Para melhorar a usabilidade do protótipo, pode-se arrastar a coluna “Profissional” do calendário de recursos para outra posição dentro da tabela. A Figura 22 mostra a coluna reposicionada ao final da tabela, o que torna mais rápida a visualização da linha do profissional para dias ao final do mês.

	25	Professional	26	27	28	29	30	31
-		Jean Bastos	-	-	-	-	-	-
-		Briatori	-	-	-	-	-	-
-		Nelson	-	-	-	-	-	-
A		Jose Zeca	A	A	A	A	A	A
-		Antonio Boleiro	-	-	-	-	-	-
-		Doberval Coimbra	-	-	-	-	-	-
-		Antonio Tavares	-	-	-	-	-	-
A		Carlos Santos	A	A	A	A	A	A

Figura 22 - Coluna de profissional reposicionada para melhor visualização

Outra funcionalidade que auxilia a utilização do calendário de recursos é a possibilidade de ocultar qualquer coluna da tabela, para isso basta escolher quais colunas serão mostradas e quais serão escondidas, isso é realizado no menu ao clicar sobre a coluna. A Figura 23 mostra a escolha das colunas “04”, “05” e “06” como ocultas, isso permite ao usuário remover da tela informações que não lhe são necessárias.

Professional	01	02	03	07	08	09	10	11	12
Jean Bastos	-	-	A	A	A	-	-	-	-
Briatori	-	-	-	-	-	-	-	-	-
Nelson	-	-	-	-	-	-	-	-	-
Jose Zeca	A	A	A	A	A	-	-	-	-
Antonio Boleiro	-	-	A	A	A	A	A	A	A
Doberval Coimbra	-	-	-	-	-	-	-	-	-
Antonio Tavares	-	-	-	-	-	-	-	-	-
Carlos Santos	A	A	A	A	A	A	A	A	R

Figura 23 - Colunas "04", "05" e "06" ocultas no calendário de recursos

3.3.3 Operacionalidade da implementação

Esta seção apresenta a funcionalidade e operacionalidade do protótipo, mostrando suas principais características através de um estudo de caso, onde será mostrado todo o processo de alocação para um projeto, dos cadastros e salvar e restaurar alocações.

A Figura 24 mostra o fluxo de funcionamento do protótipo, onde é iniciado pelo

cadastramento de projetos, profissionais, atributos, restrições, cargos e equipes. Após o usuário tem a opção de executar o processo de alocação automaticamente, onde o sistema apresenta as melhores alocações encontradas, ou manualmente onde o usuário cria as alocações. Então o usuário pode apenas visualizar os resultados, ou ajustar os parâmetros para tentar novas alocações e por fim pode salvar as alocações para uso posterior. Nas sessões seguintes são apresentados todos os passos do processo.

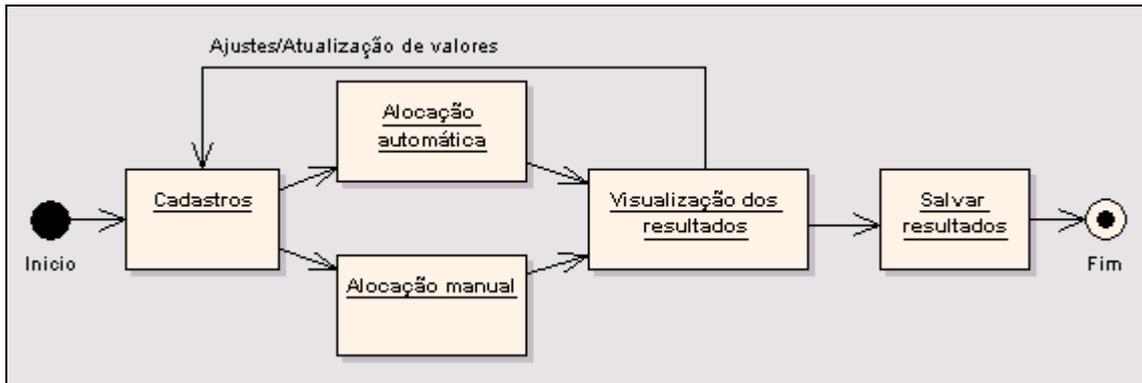


Figura 24 - Fluxo de funcionamento do protótipo

3.3.3.1 Cadastros

Através do *menu* “Cadastros”, localizado no lado esquerdo da tela de alocações, é possível chamar as telas para cadastro ou manutenção dos cargos, competências, conhecimentos, equipes, habilidades, restrições, profissionais e projetos (Figura 25).

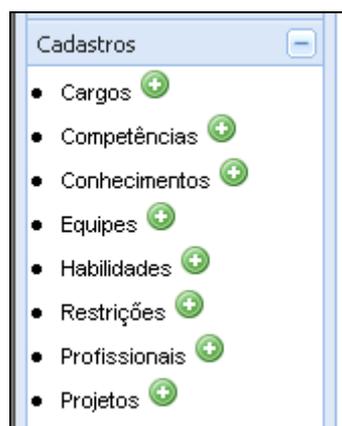


Figura 25 - Menu "Cadastros"

Cargos, competências, conhecimentos, equipes, habilidades e restrições são utilizados nos cadastros de profissionais e projetos. Assim, o usuário deve respeitar essa seqüência para conseguir efetuar o cadastramento. A Figura 26 mostra o cadastramento do cargo “Gerente de Projetos” no sistema. As telas de cadastros são mostradas sobre a tela principal, de forma a

permitir que o usuário possa trabalhar somente na tela aberta.

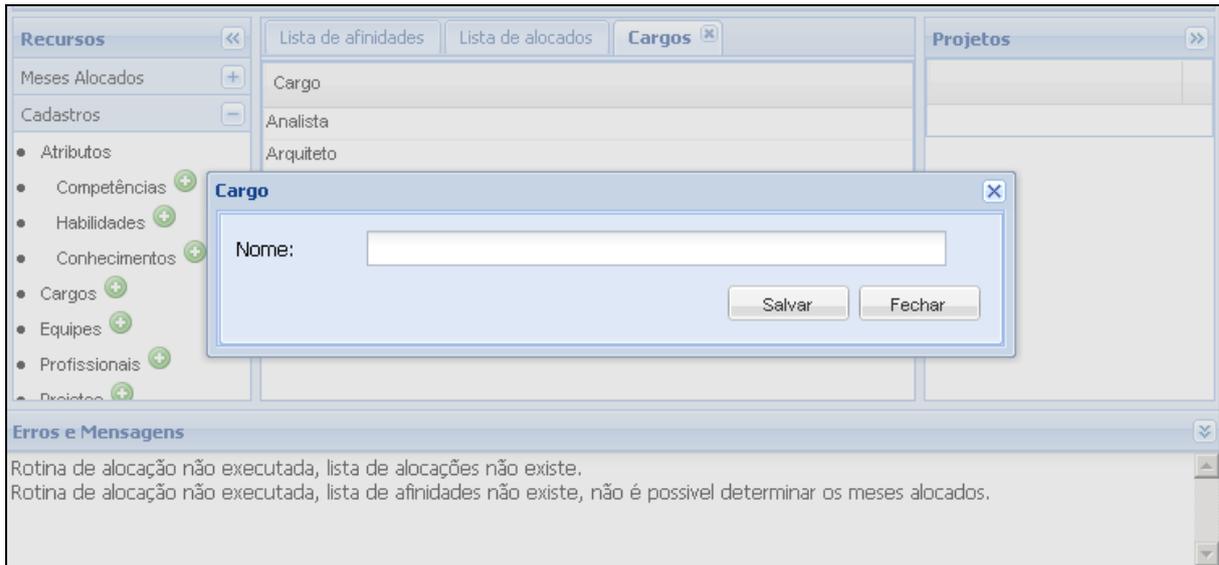


Figura 26 - Cadastro de cargos

Após o cadastramento dos atributos, o próximo passo que o usuário devera efetuar é o cadastramento dos profissionais e projetos. Essa etapa pode ser realizada várias vezes de acordo com as mudanças que irão ocorrer nos projetos durante sua execução (alterações de prazos, surgimento de novas restrições, etc..).

No cadastro do profissional deve ser informado nome, quantidade de horas de trabalho desse profissional, cargo e equipe a qual pertence. Após clicar em “Salvar” o profissional é criado, porém ainda não possui nenhum atributo associado a ele. A Figura 27 mostra o cadastramento do profissional “Jose dos Santos” já utilizando o cargo “Gerente de Projetos” cadastrado anteriormente.

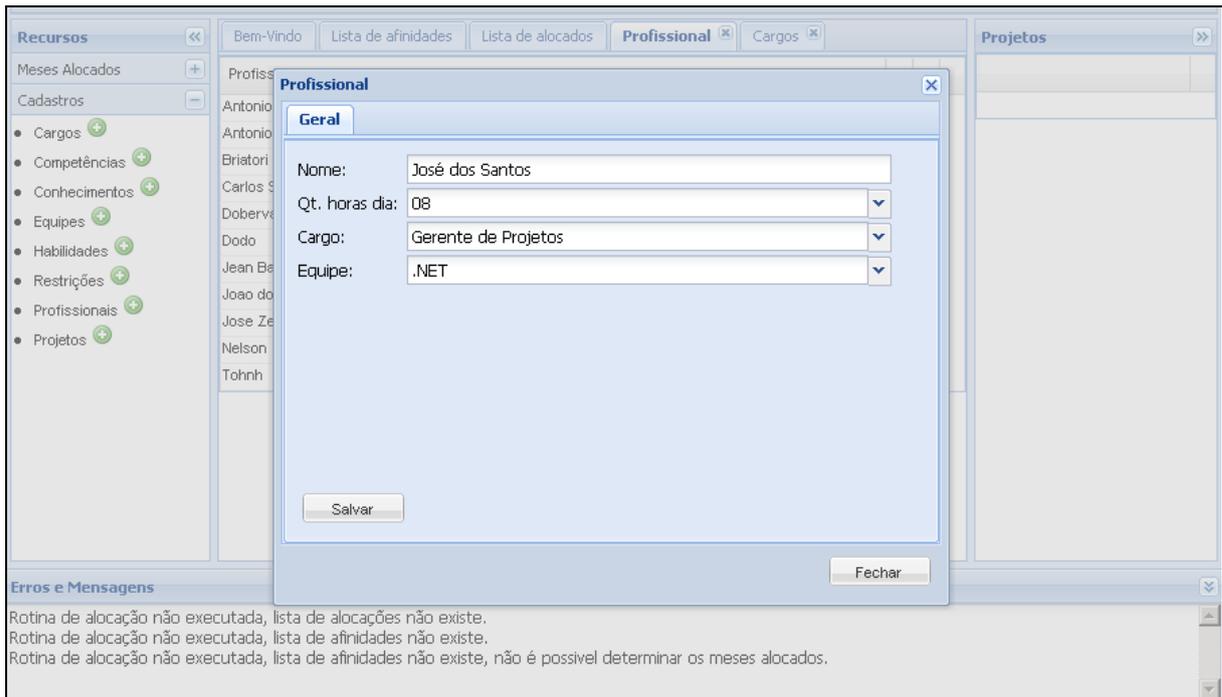


Figura 27 - Cadastro de profissionais

A Figura 28 mostra o processo de associação dos atributos, anteriormente cadastrados ao profissional. As abas com os atributos só aparecem quando o usuário escolher a opção “Editar” na lista de profissionais que é mostrada após a escolha de “Profissionais” no *menu* de cadastros. A associação dos atributos é realizada através da checagem do atributo, habilitando assim a lista de níveis. O usuário escolhe então, qual é o nível do profissional para aquele atributo (os níveis iniciam em um, ruim, e vão até dez, ótimo). Ao escolher o sistema já armazena o novo valor sem a necessidade de novas ações. A associação dos atributos competências, habilidades e conhecimentos tanto para profissionais como para projetos é realizado dessa forma.

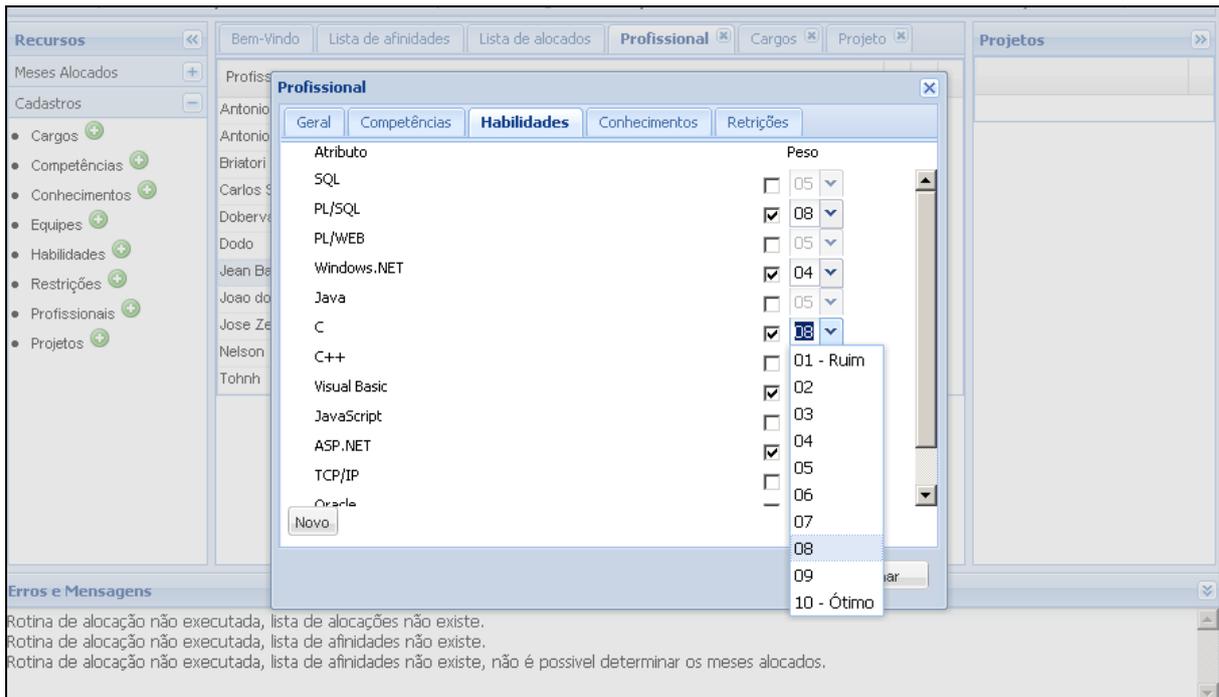


Figura 28 - Cadastro de profissionais, associação de atributos ao profissional

Para a criação de restrições para um profissional ou para um projeto o usuário escolhe a opção “Restrições”. Após informa o tipo da restrição (já cadastrada anteriormente) a data de início e a data de fim, ao escolher “Adicionar” o sistema cria a restrição (Figura 29).

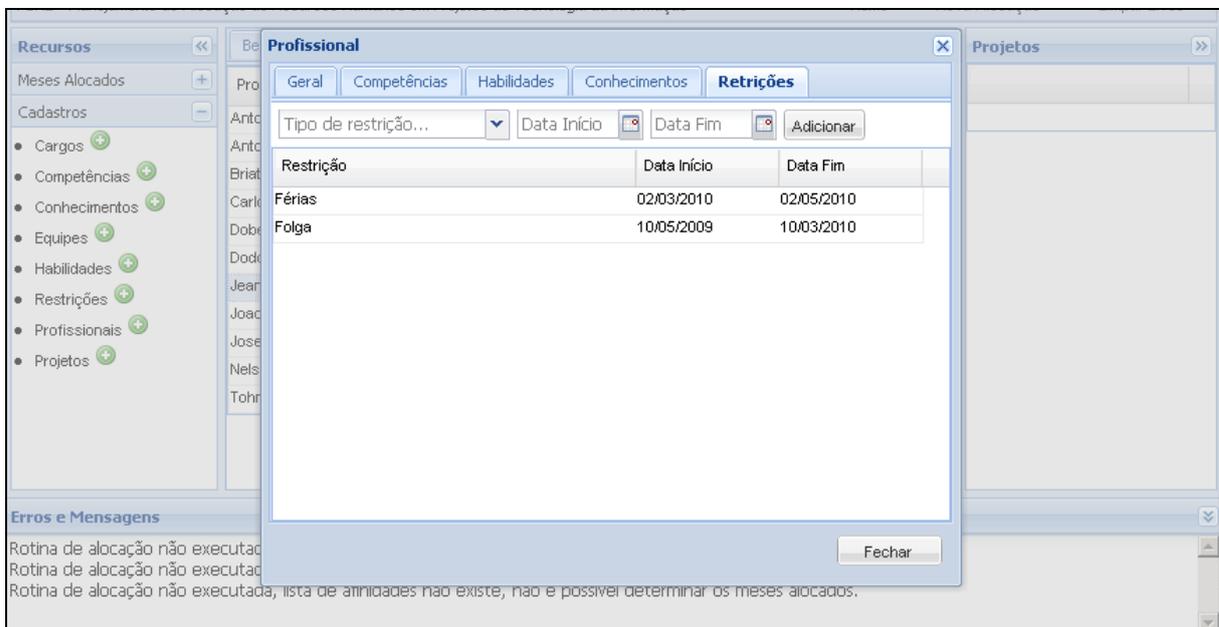


Figura 29 - Cadastro de profissionais, criação de restrições

Como mostra a Figura 30 o cadastro de projetos pode ser dividido em três grupos: informações do projeto, informações de alocação e atributos. Informações do projeto são dados como nome, datas, cliente e gerente, as informações de alocação referem-se aos parâmetros utilizados no processo de alocação automática (ajustes de afinidade, proporção de uso dos atributos e cor para representar o projeto).

Projeto

Nome: Projeto 1

Data Início: 03/01/2009

Data Fim: 01/06/2009

Cliente: Cliente 1

Gerente: Briatori

Alocação

Afinidade inferior: 50

Afinidade superior: 80

% Conhecimento: 33

% Habilidade: 33

% Competência: 33

Cor: yellow

Salvar

Fechar

Figura 30 - Cadastros de projetos

Também está incluído nesse grupo o cadastro de cargos (Figura 31), que informa quais cargos são necessários, quantidades e total de horas que cada recurso será alocado.

Projeto

Cargos

Cargo	Quantidade Pessoas	Quantidade Horas
Programador	2	100
Analista	1	100

Fechar

Figura 31 - Cadastros de projetos, cargos necessários

3.3.3.2 Alocação automática

Após os cadastros realizados o usuário pode optar por realizar a alocação automática. Essa opção está disponível na parte superior direita da tela (Figura 32). O usuário tem acesso também a opção de “Alocação Manual”, que é detalhada na seção seguinte e a opção “Limpar Erros” que apaga a lista de erros e mensagens na parte inferior da tela.

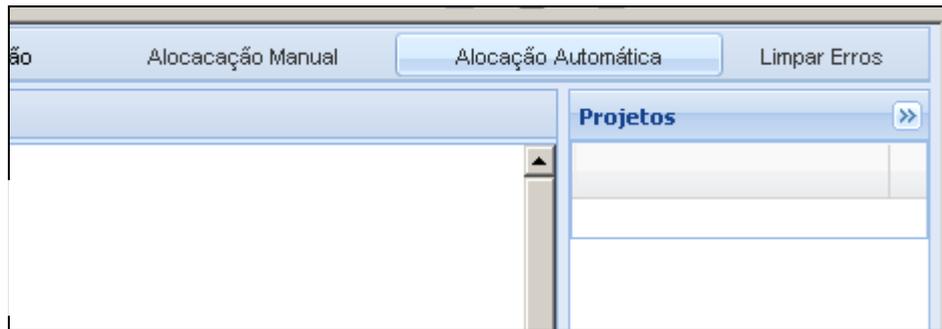


Figura 32 - Menu de alocações

O usuário informa quais projetos, ordem e profissionais que serão utilizados no processo de alocação (Figura 33). O processo irá desconsiderar os itens não marcados nessa etapa, a ordem, como já apresentado no item 3.2.6 determinada a seqüência a qual os projetos serão alocados.

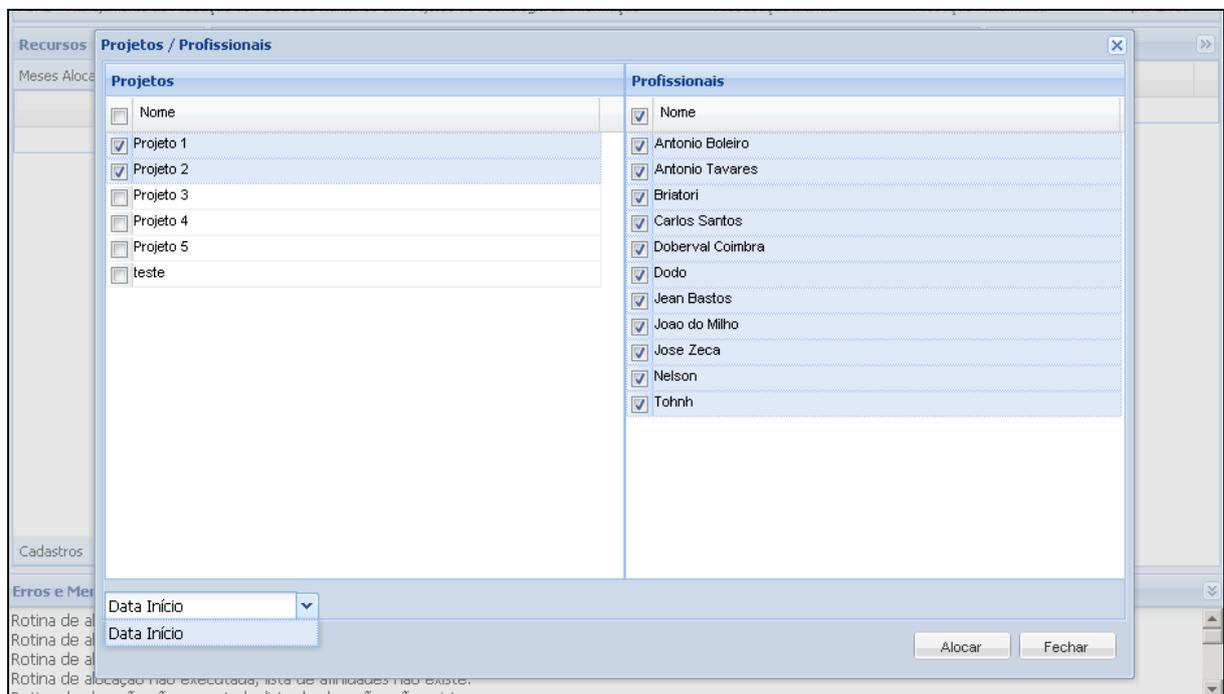


Figura 33 - Escolha de projetos e profissionais para execução da alocação automática

Por fim o usuário escolhe a opção “Alocar”. Nesse momento o protótipo gera a alocação, que pode demorar alguns segundo de acordo com a quantidade de projetos e profissionais selecionados. Após o término do processamento é mostrado a “Lista de

afinidades”, “Lista de alocados” e o calendário de recursos. Esses itens são apresentados na seção 3.3.3.4.

Após a geração automática é possível alterar a alocação gerada. O processo de alocação manual é apresentado a seguir (item 3.3.3.3). As alterações manuais não consideram as restrições cadastradas para o projeto e /ou para o profissional, assim as alocações serão criadas de acordo com os comandos do usuário.

3.3.3.3 Alocação manual

Para criar uma alocação manualmente o usuário escolhe a opção “Alocação Manual” no *menu* superior. O sistema irá exibir uma tela para seleção dos projetos e profissionais, em seguida o sistema gera a lista de afinidades e exibe a lista de alocados em branco, pois nesse instante ainda não existem alocações criadas.

O usuário escolhe o mês que deseja visualizar no *menu* da direita. Neste momento é apresentado o calendário de recursos. O usuário escolhe o projeto que deseja alocar e arrasta para cima do calendário de recursos (Figura 34). São apresentadas as restrições para todas as datas fora do intervalo do projeto. O usuário clica no dia de início do intervalo de alocação e logo após no dia de término. O protótipo mostra então a alocação criada. Nesse momento a lista de alocações é atualizada e a alocação criada é apresentada.

Profis...	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
Jean Bastc	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
João dos S	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
Carlos Prac	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
Maria Crarr	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
Leandro Pr	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
Antonio Sil	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-
Luis Inácio	R	R	R	R	R	R	R	R	R	R	-	-	-	-	-	-

Figura 34 - Alocação manual

O processo de alocação manual não considera as restrições, nem atributos cadastrados

para projetos e profissionais, ficando a critério do usuário a escolha das datas e projetos.

3.3.3.4 Visualização dos resultados

Os resultados das alocações são apresentados em forma de listas e do calendário de recursos, que está dividido em meses, de acordo com o intervalo entre a menor e a maior data dos projetos.

A “Lista de afinidades” (Figura 35) mostra o produto cartesiano entre profissionais e projetos com os valores calculados de afinidade entre ambos. O cálculo de afinidade foi apresentado no item 3.2.5.

Profissional	Projeto	Afinidade
Dodo	Projeto 1	79.035
Jean Bastos	Projeto 2	56.1
Joao do Milho	Projeto 2	56.1
Briatori	Projeto 2	33
Nelson	Projeto 2	0
Jose Zeca	Projeto 2	0
Antonio Boleiro	Projeto 2	0
Doberval Coimbra	Projeto 2	0
Antonio Tavares	Projeto 2	0
Carlos Santos	Projeto 2	0
Tohnh	Projeto 2	0

Figura 35 - Lista de afinidades

A “Lista de alocados” contém a relação de todos os profissionais que foram alocados em um dos projetos selecionados com data de início e fim de cada alocação. Também é apresentado o total de horas que o profissional foi alocado.

A Figura 36 mostra o calendário de recursos do mês de janeiro de 2009. O usuário seleciona o mês desejado no *menu* há esquerda da tela (“Meses Alocados”). Uma nova aba é criada mostrando o calendário para o mês escolhido. Pode-se navegar entre os meses já visualizados sem a necessidade de carregar todo o calendário novamente do servidor de aplicação. Basta clicar sobre a aba que o mesmo é mostrado instantaneamente, pois nesse instante não ocorre uma nova requisição ao servidor.

Recursos		Bem-Vindo	Lista de afinidades	Lista de alocados	Calendário 01/2009	Calendário 02/2009	Projetos																
Meses Alocados	Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
	Jean Bastos	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
01/2009	Joao do Milho	-	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
02/2009	Briatori	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
03/2009	Nelson	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
04/2009	Jose Zeca	A	A	A	A	A	A	A	A	A	A	A	R	R	A	A	A	A	A	A	A	A	A
05/2009	Antonio Bolei	-	A	A	A	A	A	A	A	A	A	A	A	A	A	A	-	-	-	-	-	-	-
06/2009	Doberval Coin	-	A	A	A	A	A	A	A	A	A	A	A	A	A	A	-	-	-	-	-	-	-
07/2009	Antonio Tava	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
08/2009	Carlos Santos	A	A	A	A	A	A	A	A	A	A	A	R	R	A	A	A	A	A	A	A	A	A
09/2009	Tohnh	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10/2009	Dodo	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Figura 36 - Calendário de recursos

O calendário utiliza as cores informadas no cadastro dos projetos para diferenciar as alocações dos recursos. Isso torna a compreensão do calendário mais fácil e rápida. Ao passar o ponteiro do mouse sobre qualquer um dos itens do calendário são exibidas as informações sobre a alocação, restrição ou ociosidade (Figura 37).

A	R	R	A	A	A	A	A	A
A	A	A	A	A	A	A	A	A
A	A	De: 12/01/2009 Até: 13/01/2009		-	-	-	-	-
-	-	Restrição: Fim de semana		-	-	-	-	-
-	-	Tipo: Projeto		-	-	-	-	-
A	R	Cargo: Não Há		-	-	-	-	-
-	-	-	-	-	-	-	-	-

Figura 37 - Informações adicionais no calendário de recursos

Outra maneira de visualizar os resultados obtidos é através do relatório de profissionais alocados (Figura 38). Para gerar o relatório basta escolher a opção “Profissionais Alocados” no *menu* “Relatórios”. Será exibida uma nova janela com o relatório gerado no formato “.pdf”. O usuário pode então salvar, imprimir, enviar por e-mail de acordo com suas necessidades.

Profissionais Alocados					
Projeto	Profissional	Cargo	Data Inicio	Data Fim	Total Alocado (horas)
Projeto 1					
	Antonio Boleiro	Programador	03/01/2009	15/01/2009	104
	Doberval Coimbra	Programador	03/01/2009	15/01/2009	104
	Joao do Milho	Analista	03/01/2009	28/01/2009	104
					312
Projeto 2					
	Jose Zeca	Programador	01/01/2009	22/02/2009	408
	Carlos Santos	Analista	01/01/2009	27/02/2009	504
					912

Figura 38 - Relatório de profissionais alocados

3.3.3.5 Salvar resultados

O protótipo permite salvar os calendários gerados para visualização posterior. As informações são serializadas e salvas na base de dados podendo ser restauradas quando necessário. O processo persistência de uma alocação inicia quando o usuário seleciona a opção “Salvar Calendário” no *menu* “Ações”. O sistema pede então um nome para identificar os dados. No momento da restauração, após o usuário escolher “Salvar” todos objetos relacionados com a alocação atual são salvos.

O processo de restauração de uma alocação se dá de forma parecida. O usuário escolhe a opção “Abrir Calendário” o sistema exibe uma lista das alocações salvas (Figura 39). O usuário efetua um duplo clique sobre a alocação desejada e o sistema restaura os dados.

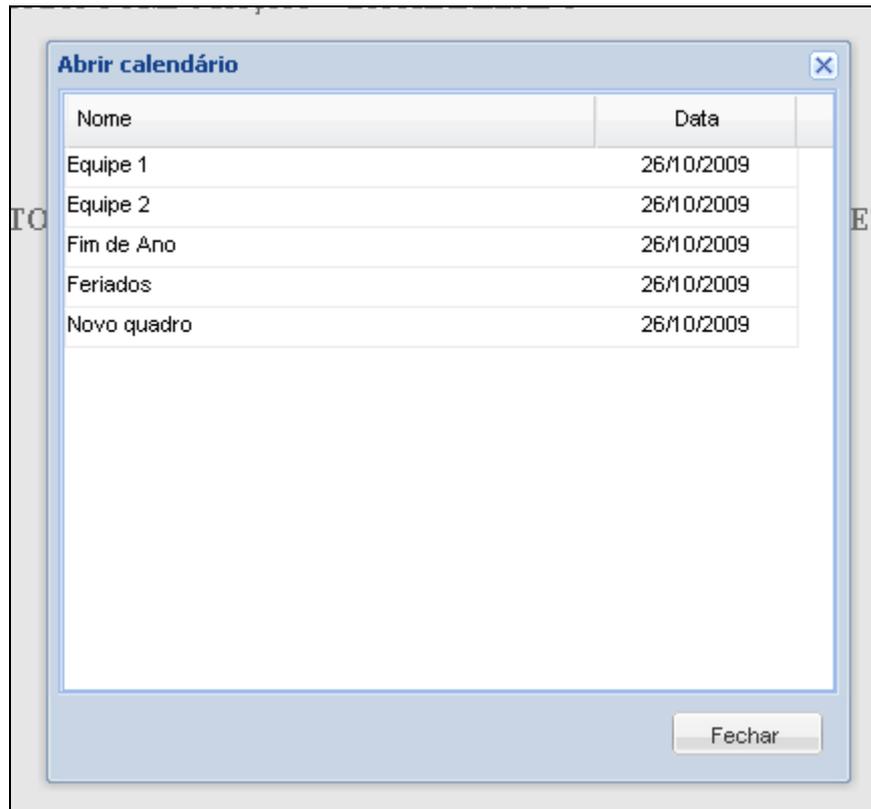


Figura 39 - Tela de escolha das alocações salvas

3.4 RESULTADOS E DISCUSSÃO

O estudo realizado sobre o guia PMBOK, e trabalhos correlatos significaram muito para a realização do presente trabalho. A partir deste estudo foi possível realizar uma análise global e verificar os pontos positivos e negativos de cada caso em particular para obter o maior aproveitamento possível.

Em comparação aos trabalhos correlatos apresentados, o trabalho desenvolvido pode ser diretamente comparado a ferramenta Resource Management. As demais ferramentas têm outros objetivos divergentes do trabalho atual, porém possuem funcionalidades, pontos em comum, que foram utilizados como apoio para construção deste.

	Protótipo	Resource Management
Plataforma	<i>Web</i>	<i>Web</i>
Idioma	Português	Inglês
Geração automática de alocações	Sim	Não
Geração de relatório de alocações	Sim	Sim
Visualização das alocações através de calendário de recursos	Sim	Sim
Cadastro de perfil dos profissionais	Sim	Sim
Controle de custos	Não	Sim
Integração com softwares de cronograma	Não	Sim

Quadro 7 - Comparativo de ferramentas de alocação de recursos humanos

Outra forma de análise dos resultados obtidos, foi a realização uma pesquisa junto a profissionais com experiência em gerência de projetos de TI. Dois gerentes de projetos, com certificação Project Management Professional (PMP) emitida pelo PMI, e um analista de sistemas preencheram os formulários.

A pesquisa consiste em apresentar um cenário que contempla quatro projetos, seis profissionais e uma lista de restrições tanto de projetos como de recursos, onde há a necessidade de criar um calendário de recursos com as respectivas alocações (projetos vs. recursos), considerando a afinidade de cada profissional para a realização das atividades do projeto.

O tamanho e ciclo de vida dos projetos foram reduzidos para viabilizar a resolução de forma rápida e manual. No Apêndice B é apresentado o formulário utilizado para realização da pesquisa. O Quadro 8 mostra a comparação dos resultados obtidos na pesquisa com os resultados gerados pelo protótipo. “Pesquisa A”, “Pesquisa B” e “Pesquisa C” representam os formulários preenchidos manualmente pelos entrevistados, um formulário para cada pessoa.

	Protótipo	Pesquisa A	Pesquisa B	Pesquisa C
Tempo de resolução do problema	25 segundos	30 minutos	40 minutos	2,5 horas
Total de erros nas alocações	0	4	0	0
Quantidade de profissionais alocados de acordo com a afinidade para o projeto	5	4	5	5

Quadro 8 - Comparativo entre os resultados do protótipo e a pesquisa realizada

A comparação do tempo de resolução, desconsidera o tempo de cadastro das informações no protótipo e o tempo de leitura e entendimento do cenário pelos entrevistados.

Analisando os calendários de recursos gerados (Quadro 9), pode-se observar a situação em que o protótipo atrasou a data de início do Projeto 2 (em verde) para que o profissional

com maior afinidade pudesse ser alocado. Porém não extrapolou o prazo do projeto. Já o entrevistado, Pesquisa B, optou por alocar um profissional de afinidade inferior ao projeto e iniciar a atividade em paralelo deixando um período de ociosidade ao fim do projeto.

O protótipo escolheu o melhor caminho, nessa situação. Pois além de alocar o recurso mais adequado ao projeto, também otimizou o tempo deste recurso.

Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
João dos Santos	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	R	R	A	A	A	A	A	R	R	A	A	A	A	-	-	-
Carlos Prado	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	R	R	A	A	-	-	-	-
Maria Cramer	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Leandro Prato	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	A	R	R	A	A	A	A	A	R	R
Antonio Silva	-	-	-	A	A	A	A	A	R	R	A	A	A	A	A	R	R	A	A	A	A	-	-	-	-	-	-	-	-	-	-
Luis Inácio	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	-	-

Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
João dos Santos	R1	R2	R2						R4	R4					R1	R5	R5	R1	R1	R1	P1	P1	R6	R6	P1	P1	P1	P1	P1	R7	R7
Carlos Prado	R1	R2	R2						R4	R4					R5	R5	R5	R2	R2	R2	R2	R2	R6	R6							
Maria Cramer	R1	R2	R2		R3	R3			R4	R4					R5	R5	R5	R2	R2	R2	R2	R2	R6	R6							
Leandro Prato	R1	R2	R2	P1	P1	P1	P1	P1	R4	R4	P1	P1	P1	P1		R5	R5			R3	P3	P3	R6	R6	P3	P3	P3	P3	R7	R7	
Antonio Silva	R1	R2	R2						R4	R4		R2	R2	R2	R2	R5	R5					R6	R6	R6	R6						
Luis Inácio	R1	R2	R2						R4	R4					R5	R5	R5						R6	R6	P2	P2			P1	P1	P1

Quadro 9 - Compativo dos calendários de recursos

4 CONCLUSÕES

Através do estudo dos modelos definidos pelo PMBOK e informações de profissionais de gerência de projetos, pode-se construir todo o processo de alocação de recursos humanos. A ferramenta pode ser usada para encontrar cenários mais otimizados para as alocações. O emprego da busca Tabu se mostrou eficiente para encontrar as melhores alocações para cada projeto.

Através da pesquisa realizada junto a profissionais com experiência em gerência de projetos, pode-se constatar que a ferramenta apresenta um ganho significativo de tempo. Mediante cenários mais complexos, pode-se também obter melhores resultados através da diminuição de incidência de erros e otimização dos resultados.

O objetivo do trabalho foi atingido. O protótipo possui uma restrição para alocações simultâneas, de um mesmo recurso em um mesmo intervalo de datas, pois somente permite que cada recurso seja alocado em um único projeto por dia, o que pode não ser a realidade em algumas empresas.

O sistema foi desenvolvido na linguagem de programação C# (ASP.NET) e banco de dados SQL Server. Foram utilizadas técnicas e tecnologias atuais, como Entity Framework, ExtJs, ASP.NET MVC Framework e LINQ, que mostraram-se muito eficientes, atendendo todas as necessidades que surgiram.

A divisão em camadas facilitou o projeto e desenvolvimento do trabalho, Já a especificação, através do uso da UML, auxiliou na definição do escopo e na codificação da ferramenta.

4.1 EXTENSÕES

Como sugestões para continuação deste trabalho e melhoria da ferramenta, pode-se citar:

- a) adicionar novos relatórios, permitindo diferentes visões dos resultados gerados;
- b) criar controle de acesso às funções do sistema, assim diferentes usuários poderiam utilizar a ferramenta;
- c) permitir exportar os resultados para outras ferramentas de cronograma de projetos;

- d) tornar seqüencial e dependentes as atividades do projeto, para que uma inicie após o termino da outra sem a necessidade de cadastro da data;
- e) permitir alocações por hora, considerando alocação de um recurso, em dois projetos no mesmo dia.

REFERÊNCIAS BIBLIOGRÁFICAS

AMARAL, Fabio Eduardo Paganin Reis. **O que é Gadget? E Widget, é a mesma coisa?**. [S.l.], 2009. Disponível em: <<http://www.baixaki.com.br/info/1959-o-que-e-gadget-e-widget-e-a-mesma-coisa-.htm>>. Acesso em: 20 ago. 2009.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR ISO/IEC 12207:2009**: sistemas e engenharia de software – processos de ciclo de vida do software. Rio de Janeiro, 2009.

BLADES, Steve Cutter; FREDERICK, Shea; RAMSAY, Colin. **Learning Ext JS**. Birmingham: Packt Publishing, 2008.

BRASIL FILHO, Amaury Teófilo et al. Otimização da alocação de profissionais em projetos de tecnologia da informação. SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 38., 2006, Goiana. **Anais...** João Pessoa: UFPB, 2006. p. 2292-2302.

CONERY, Rob et al. **Professional ASP.NET MVC 1.0**. Indianapolis: Wiley Publishing, 2009.

DEMO, Éverton. **Ferramenta de apoio ao processo de gerência de recursos humanos do MPS.BR**. 2008. 92 f. Trabalho de Conclusão de Curso - (Graduação em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2008. Disponível em: <http://www.bc.furb.br/docs/MO/2008/331521_1_1.pdf>. Acesso em: 28 mar. 2009.

GARCIA, Jesus. **Ext JS in action**. Greenwich: Manning, 2009. Versão preliminar em PDF. Disponível em: <<http://www.manning.com>>. Acesso em: 31 ago. 2009.

HELDMAN, Kim. **Gerência de projetos: fundamentos: um guia prático para quem quer certificação em gerência de projetos**. Tradução Luciana Amaral Teixeira. Rio de Janeiro: Elsevier, 2005.

MACORATTI, José Carlos. **ASP .NET – MVC: introdução**. [S.l.], [2008?]. Disponível em: <http://www.macoratti.net/08/06/asp_mvc1.htm>. Acesso em: 18 ago. 2009.

MICROSOFT CORPORATION. **ASP.NET Model View Controller (MVC)**. [S.l.], [2009?]a. Disponível em: <<http://msdn.microsoft.com/en-us/library/dd394709.aspx>>. Acesso em: 30 ago. 2009.

_____. **ASP.NET MVC overview: the official Microsoft ASP.NET site**. [S.l.], [2009?]b. Disponível em: <<http://www.asp.net/learn/mvc/tutorial-01-cs.aspx>>. Acesso em: 31 ago. 2009.

_____. **LINQ**. [S.l.], [2009?]c. Disponível em: <<http://msdn.microsoft.com/en-us/netframework/aa904594.aspx>>. Acesso em: 14 out. 2009.

PIGATTI, Alexandre Altoé. **Modelos e algoritmos para o Problema de Alocação Generalizada (PAG) e aplicações**. 2003. 74 f. Dissertação (Mestrado em Informática) - Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro.

PROJECT MANAGEMENT INSTITUTE. **Um guia do conjunto de conhecimentos em gerenciamento de projetos: guia PMBOK**. 3. ed. Newtown Square: PMI, 2004.

_____. **About us**. Newtown Square, [2008]. Disponível em: <<http://www.pmi.org>>. Acesso em: 28 mar. 2009.

QUICKARROW. **Resource management, utilization, scheduling and planning software tools**. Austin, [2009]. Disponível em: <<http://www.quickarrow.com/solutions/resourcemanagement.asp>>. Acesso em: 24 mar. 2009.

SILVA, Pedro Nuno Macedo Leite da. **Sistema multi-agente para geração de horários no ensino secundário**. 2000. [sem páginas]. Dissertação (Mestrado em Inteligência Artificial e Computação) - Universidade do Porto, Porto, 2000. Disponível em: <<http://paginas.fe.up.pt/~eol/MIACC/2000/GHES>>. Acesso em: 28 mar. 2009.

SOETHE, Fabio. **Ferramenta para gerenciamento de tempo de projetos**. 2004. 57 f. Trabalho de Conclusão de Curso – (Graduação em Sistemas de Informação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2004. Disponível em: <http://www.bc.furb.br/docs/MO/2004/307534_1_1.pdf>. Acesso em: 20 mar. 2009.

SOMMERVILLE, Ian. **Engenharia de software**. 6. ed. Tradução André Maurício de Andrade Ribeiro. São Paulo: Addison Wesley, 2003.

TAVARES, Chris. **ASP.NET MVC: criando aplicativos web sem Web Forms**. [S.l.], 2008. Disponível em: <<http://msdn.microsoft.com/pt-br/magazine/cc337884.aspx>>. Acesso em: 19 ago. 2009.

VARGAS, Ricardo Viana. **Gerenciamento de projetos: estabelecendo diferenciais competitivos**. 5. ed. Rio de Janeiro: Brasport, 2003.

XAVIER, Carlos Magno da Silva. **Gerenciamento de projetos: como definir e controlar o escopo de projeto**. São Paulo: Saraiva, 2005.

APÊNDICE A – Descrições dos Casos de Uso

Do Quadro 10 ao Quadro 20 são apresentadas as descrições dos casos de uso do protótipo. Optou-se por descrever de forma detalhada apenas os casos de uso mais críticos da aplicação.

UC01 – Cadastrar habilidade	
Descrição	Permite o cadastro e manutenção de habilidades. Habilidades são atributos de profissionais e projetos. Esse atributo é considerado para geração da afinidade que o recurso tem com os projetos, assim pode-se determinar quais recursos são mais recomendados para realizar as atividades. Habilidade pode ser vista como uma tecnologia a qual o recurso tem conhecimento/domínio, ou então, qual tecnologia o projeto exige. Exemplos de habilidades: C#, ASP.NET, Java, Oracle, SQL Server, etc...
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	A habilidade deve ter sido criada ou atualizada.

Quadro 10 - Descrição do caso de uso "UC01 - Cadastrar habilidade"

UC02 - Cadastrar competência	
Descrição	Permite o cadastro e manutenção de competências. Competências são atributos de profissionais e projetos. Esse atributo é considerado para geração da afinidade que o recurso tem com os projetos, assim pode-se determinar quais recursos são mais recomendados para realizar as atividades. Competência pode ser vista como características de cada recurso em atividades em geral, ou então exigência do projeto para com essas características, são exemplos de competências: Trabalho em Equipe, Comprometimento, Organização, etc...
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	A competência deve ter sido criada ou atualizada.

Quadro 11 - Descrição do caso de uso "UC02 - Cadastrar competência"

UC03 – Cadastrar conhecimento	
Descrição	Permite o cadastro e manutenção de conhecimentos. Conhecimentos são atributos de profissionais e projetos. Esse atributo é considerado para geração da afinidade que o recurso tem com os projetos, assim pode-se determinar quais recursos são mais recomendados para realizar as atividades. Conhecimentos
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	O conhecimento deve ter sido criado ou atualizado.

Quadro 12 - Descrição do caso de uso "UC03 - Cadastrar conhecimento"

UC04 – Cadastrar profissional	
Descrição	Permite o cadastro e manutenção dos profissionais, no cadastro do profissional são informados todos os dados de parametrização dos atributos (habilidades, conhecimentos e competências) bem como restrições para esse profissional. Além das informações básicas do profissional como nome, cargo e equipe.
Atores	Gerente de Projetos
Pré-condições	Devem existir habilidades, competências, conhecimentos, restrições, equipes e cargos previamente cadastrados.
Pós-condições	O profissional deve ter sido criado ou atualizado

Quadro 13 - Descrição do caso de uso "UC04 - Cadastrar profissional"

UC05 – Cadastrar equipe	
Descrição	Permite o cadastro e manutenção de equipes de profissionais. Equipe auxilia o usuário a agrupar os recursos melhorando a identificação do mesmo. A equipe não é considerada no processo de alocação sendo utilizada somente em telas e relatórios.
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	A equipe deve ter sido criada ou atualizada

Quadro 14 - Descrição do caso de uso "UC05 - Cadastrar equipe"

UC06 – Cadastrar projeto	
Descrição	Permite o cadastro e manutenção de projetos. No cadastro do projeto são informados todos os parâmetros relacionados a atributos, alocação e ao projeto em si, como nome, cliente, datas de início e fim. Também ficam as informações de cargos exigidos juntamente com quantidade de recursos para sua execução. Habilidades, Conhecimentos, Competências e Restrições também são informadas no cadastro, parâmetros do cálculo de afinidade e demais informações pertinentes ao projeto são manipuladas por esse caso de uso.
Atores	Gerente de Projetos
Pré-condições	Devem existir cargos, habilidades, competências, conhecimentos e restrições previamente cadastrados.
Pós-condições	O projeto deve ter sido criado ou atualizado.

Quadro 15 - Descrição do caso de uso "UC06 - Cadastrar projeto"

UC07 – Alocar automaticamente profissional	
Descrição	Permite a alocação automática dos profissionais em projetos selecionados.
Atores	Gerente de Projetos
Cenário principal	<ol style="list-style-type: none"> 1. O caso de uso inicia quando o ator seleciona a opção “Alocação Automática” no <i>menu</i> superior da tela principal; 2. O sistema apresenta a lista de projetos e profissionais cadastrados; 3. O ator seleciona os projetos, profissionais e a ordem de alocação; 4. O ator escolhe a opção “Alocar”; 5. O sistema verifica se existem projetos e profissionais selecionados e também a ordem informada; 6. O sistema gera a alocação; 7. O sistema fecha a janela de seleção, retornando a tela principal; 8. O sistema apresenta em forma de abas na tela principal a lista de afinidades, lista de alocações e em forma de <i>menus</i> a lista de projetos envolvidos e os meses da alocação.
Fluxos alternativos	<p>Não existem projetos, profissionais selecionados ou ordem informada</p> <p>5a1. O sistema destaca as listas indicando a obrigatoriedade da seleção</p> <p>5a2. O sistema retorna ao passo 4.</p>
Exceções	<p>No passo 6: Erro no na geração da alocação</p> <p>O sistema exibe o erro ocorrido na caixa de mensagens da tela principal;</p> <p>O sistema termina a operação.</p>
Regras de Negócio	<p>Não deve ser possível gerar alocação se a ordem não for informada ou ao menos um projeto ou profissional esteja selecionado;</p> <p>As ordens de alocação devem ser “Data de Inicio”, “Menor Tamanho” e “Maior Tamanho”.</p>
Pré-condições	Habilidades, conhecimentos, competências, restrições, cargos, projetos e profissionais previamente cadastrados.
Pós-condições	Profissionais alocados.

Quadro 16 - Descrição do caso de uso "UC07 – Alocar automaticamente profissional"

UC08 – Alocar manualmente profissional	
Descrição	Permite ao ator criar a alocação dos profissionais.
Atores	Gerente de Projetos
Cenário principal	<ol style="list-style-type: none"> 1. O caso de uso inicia quando o ator seleciona a opção “Alocação Manual” no <i>menu</i> superior da tela principal; 2. O sistema apresenta a lista de projetos e profissionais cadastrados; 3. O ator seleciona os projetos e profissionais; 4. O ator escolhe a opção “Alocar”; 5. O sistema verifica se existem projetos e profissionais selecionados; 6. O sistema fecha a janela de seleção, retornando a tela principal; 7. O sistema apresenta em forma de abas na tela principal a lista de afinidades, lista de alocações e em forma de <i>menus</i> a lista de projetos envolvidos e os meses da alocação; 8. O ator seleciona o mês desejado; 9. O sistema exibe o calendário de recursos do mês escolhido; 10. O ator seleciona o projeto desejado no <i>menu</i> de projetos; 11. O ator clica sobre o dia desejado no calendário; 12. O sistema armazena em memória a alocação criada;
Fluxos alternativos	<p>Não existem projetos ou profissionais selecionados</p> <p>5a1. O sistema destaca as listas indicando a obrigatoriedade da seleção</p> <p>5a2. O sistema retorna ao passo 4.</p> <p>O ator escolhe a opção de nova alocação</p> <p>8a1. O sistema oculta as abas e <i>menus</i>;</p> <p>8a2. O sistema termina a operação.</p>
Exceções	<p>No passo 8, 9, 10 e 11: Erro no na geração da alocação</p> <p>O sistema exibe o erro ocorrido na caixa de mensagens da tela principal;</p> <p>O sistema termina a operação.</p>
Pré-condições	Habilidades, conhecimentos, competências, restrições, cargos, projetos e profissionais previamente cadastrados.
Pós-condições	Profissionais alocados.

Quadro 17 - Descrição do caso de uso "UC08 – Alocar manualmente profissional"

UC09 – Visualizar relatório de alocação profissional x projeto	
Descrição	Permite ao ator visualizar a alocação dos profissionais em projetos. Apresenta na forma de tabela a relação de todos os projetos selecionados para alocação vigente (aberta atualmente), essa relação permite ao usuário saber quanto tempo cada recurso ficara alocado e qual as datas de inicio e fim de cada alocação.
Atores	Gerente de Projetos
Pré-condições	Alocação automática ou manual previamente realizada.
Pós-condições	Não há.

Quadro 18 - Descrição do caso de uso "UC09 – Visualizar relatório de alocação profissional x projeto"

UC10 – Cadastrar restrição	
Descrição	Permite cadastrar e manter tipos de restrições que podem ser usadas em profissionais e projetos. Restrições são datas onde não podem haver atividades, o tipo de restrição é usado para identificar o motivo pelo qual um projeto ou recurso não esta alocado. Exemplos de tipos de restrições: Férias, Final de Semana, Feriado, Atestado Médico, etc...
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	A restrição deve ter sido criada ou atualizada.

Quadro 19 - Descrição do caso de uso "UC10 – Cadastrar restrição"

UC11 – Cadastrar cargo	
Descrição	Permite o cadastro e manutenção de cargos. No cadastro de projetos é informado quais cargos são necessários para execução das atividades do mesmo, esses cargos também são informados no cadastro dos profissionais, logo a rotina de alocação procura por recursos que atendam as necessidades de cargos de acordo com o projeto. Exemplos de cargo: Programados, Analista de Negócio, Analista de Sistemas, Testador, etc...
Atores	Gerente de Projetos
Pré-condições	Não há.
Pós-condições	O cargo deve ter sido criado ou atualizado.

Quadro 20 - Descrição do caso de uso "UC11 – Cadastrar cargo"

APÊNDICE B – Formulário da pesquisa para comparação dos resultados

No Quadro 21 é mostrado a introdução da pesquisa realizada, contendo o cenário apresentado aos entrevistados. No Quadro 22, Quadro 23, Quadro 24, Quadro 25 e Quadro 26 são apresentados demais informações fornecidas aos entrevistados.

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

PLANEJAMENTO DE ALOCAÇÃO DE RECURSOS HUMANOS EM PROJETOS DE TECNOLOGIA DA INFORMAÇÃO

Tempo para resolução: _____

Objetivo:

Comparar os resultados gerados por pessoas, diretamente envolvidas no processo de gerência de projetos, com os resultados obtidos através do protótipo desenvolvido.

Cenário:

Existem quatro projetos para serem coordenados, na **Erro! Fonte de referência não encontrada.** é apresentado cada projeto bem como data de início, quantidade de horas para cada tipo de atividade, e tipo de recursos que necessita para sua execução.

A **Erro! Fonte de referência não encontrada.** mostra quais recursos você dispõe para alocar nos projetos mencionados, bem como competências, conhecimentos e habilidades de cada profissional (atributos). Tanto projetos como profissionais podem possuir restrições que impeçam a execução de atividades nas datas restritas (**Erro! Fonte de referência não encontrada.**), logo deve-se respeitar esses intervalos.

Tanto projetos como profissionais possuem atributos que podem auxiliar a definir qual é o profissional mais indicado para cada projeto. Os atributos possuem um nível que mensura o domínio (profissional) ou exigência (projeto) do mesmo. Esse nível inicia em um (1, ruim) e vai até dez (10, ótimo).

O ciclo de desenvolvimento do projeto foi reduzido em análise, codificação e teste e deve ser considerado a seqüência e duração respectivamente. Deve-se evitar ao máximo períodos de ociosidade, tanto de recursos como de projetos (desde que não impacte nas datas dos mesmos).

Os resultados de sua alocação deverão ser apresentados em forma de calendário de recursos (página 4). É permitido o uso de qualquer técnica para a elaboração dos resultados.

Quadro 21 - Cenário

Cód.	Projeto	Data inicio	Data fim	Atividade	Quantidade	Atributos	Nível
P1	Projeto 1	01/01/2010	03/02/2010	Análise	9 dias	.NET	8
				Codificação	10dias	Oracle	5
				Teste	4 dias	Ministrar Treinamentos Externos	6
P2	Projeto 2	12/01/2010	31/01/2010	Análise	4 dias	JavaScript	8
				Codificação	5 dias	SQL	6
				Teste	2 dias	Trabalho em Equipe	3
				-	-	Flexibilidade	6
P3	Projeto 3	20/01/2010	28/02/2010	Análise	11 dias	ASP.NET	8
				Codificação	12 dias	SQL	4
				Teste	4 dias	Comunicação	9
P4	Projeto 4	08/02/2010	18/02/2010	Análise	4 dias	.NET	9
				Codificação	3 dias	Coordenar Projetos	4
				Teste	1 dia	-	-

Quadro 22 - Projetos para alocação

Código	Profissional	Cargo	Atributo	Nível
PF1	João dos Santos	Programador	.NET / ASP.NET	8
			JavaScript	7
			Oracle	6
			SQL Server	5
			Ministrar Treinamentos Externos	9
			Pró-Atividade	7
PF2	Carlos Prado	Programador	.NET	6
			Java	5
			Oracle	7
			Comunicação	9
			Trabalho em Equipe	6
PF3	Maria Cramer	Programador	Java	10
			JavaScript	4
			SQL	6
			Iniciativa	8
			Trabalho em Equipe	5
PF4	Leandro Prato	Analista	Modelar Processos de Negócio	9
			ASP.NET	5
			Efetuar Documentação Funcional	8
			Coordenar Projetos	8
			Flexibilidade	4
PF5	Antonio Silva	Analista	Windows.NET	3
			Java	7
			Visual Basic	3
			Flexibilidade	9
			Modelar Processos de Negócio	6
PF6	Luis Inácio	Testador	ASP.NET	8
			Java	7
			Organização	10
			Trabalho em Equipe	6
			Oracle	3
			SQL	9

Quadro 23 - Profissionais

Código	Restrição	Data inicio	Data fim	Tipo	Cód. Projeto/Profissional
R1	Confraternização Universal	01/01/2010	01/01/2010	Projeto	Todos
R2	Fim de semana	02/01/2010	03/01/2010	Projeto	Todos
R3	Folga	05/01/2010	06/01/2010	Profissional	PF3
R4	Fim de semana	09/01/2010	10/01/2010	Projeto	Todos
R5	Fim de semana	16/01/2010	17/01/2010	Projeto	Todos
R6	Fim de semana	23/01/2010	24/01/2010	Projeto	Todos
R7	Fim de semana	30/01/2010	31/01/2010	Projeto	Todos
R8	Fim de semana	06/02/2010	07/02/2010	Projeto	Todos
R9	Atestado médico	11/02/2010	12/02/2010	Profissional	PF6
R10	Fim de semana	13/02/2010	14/02/2010	Projeto	Todos
R11	Carnaval	16/02/2010	16/02/2010	Projeto	Todos
R12	Fim de semana	20/02/2010	21/02/2010	Projeto	Todos
R13	Fim de semana	27/02/2010	28/02/2010	Projeto	Todos

Quadro 24 - Lista de restrições

Instruções de preenchimento:

Para preenchimento do calendário de recursos siga as seguintes instruções:

- a) Cada profissional poderá ser alocado em somente um projeto por dia;
- b) Alocações paralelas não são permitidas;
- c) Utilize o código do projeto para representar a alocação de um recurso no mesmo;
- d) Utilize o código da restrição para indicar que naquele dia o recurso não executara as atividades;
- e) Dias em branco ou com um traço ("-") são considerados como ociosos;
- f) Considere que um dia de trabalho tem 8 horas.

Quadro 25 - Instruções de preenchimento

Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
João dos Santos																															
Carlos Prado																															
Maria Cramer																															
Leandro Prato																															
Antonio Silva																															
Luis Inácio																															

Mês: Janeiro de 2010

Profissional	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28		
João dos Santos																														
Carlos Prado																														
Maria Cramer																														
Leandro Prato																														
Antonio Silva																														
Luis Inácio																														

Mês: Fevereiro de 2010

Quadro 26 - Calendário de recursos