UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

LEANDRO SANDRI DE VASCONCELOS

Sistema de Alocação e Gerenciamento de Ambientes do Instituto de Informática da UFRGS

Trabalho de Graduação.

Prof. Dr. Marcelo Soares Pimenta
Orientador

Porto Alegre, dezembro de 2011.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos Alexandre Netto
Vice-Reitor: Prof. Rui Vicente Oppermann
Pró-Reitora de Graduação: Profa. Valquiria Link Bassani
Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb
Coordenador do CIC: Prof. Raul Fernando Weber
Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

À minha família pelo apoio incondicional e confiança depositada.
Ao Professor Marcelo Pimenta pela orientação.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS **6**
LISTA DE FIGURAS **7**
LISTA DE TABELAS **8**
RESUMO **9**
ABSTRACT **10**
1 INTRODUÇÃO **11**
1.1 Motivação **11**
1.2 Objetivos **11**
1.3 Estrutura do Trabalho **12**
2 GESTÃO DE AMBIENTES NO INSTITUTO DE INFORMÁTICA **13**
2.1 Situação Atual **13**
2.2 Requisitos para um Sistema de Gerenciamento de Ambientes **15**
2.2.1 Gestão de Usuários 15
2.2.2 Gestão de Ambientes 15
2.2.3 Alocação para as Disciplinas de Graduação e Pós-Graduação 16
2.2.4 Alocação sob Demanda 16
2.2.5 Plataforma Web 16
2.3 IAlocador **16**
3 TECNOLOGIAS UTILIZADAS NO PROJETO **18**
3.1 Ferramenta CASE **18**
3.2 Banco de Dados **19**
3.3 Linguagem de Programação **19**
3.4 Framework de Desenvolvimento **20**
3.5 Sistema de Controle de Versão **21**
4 A INFORMATIZAÇÃO DA GESTÃO DE AMBIENTES DO INSTITUTO DE INFORMATICA **22**
4.1 Arquitetura **22**
4.2 Modelagem **23**
4.2.1 Diagrama de Casos de Uso 23
4.2.2 Modelo do Banco de Dados 25
4.3 Desenvolvimento **30**
4.3.1 Primeira Iteração 30
4.3.2 Segunda Iteração 32
4.3.3 Terceira Iteração 34
4.3.4 Quarta Iteração 40
4.4 Utilização do sistema **43**
4.4.1 Carga Semestral 43
4.4.2 Alocação Sob Demanda 46

4.5 Validação 51
5 CONCLUSÃO 52
5.1 Avaliação das ferramentas utilizadas 52
5.2 Trabalhos Futuros 53
REFERÊNCIAS 55

LISTA DE ABREVIATURAS E SIGLAS

AJAX	*Asynchronous Javascript and XML*
BD	Banco de Dados
CEI	Centro de Empreendimentos do Instituto de Informática da UFRGS
DDL	*Data Definition Language*
EA	*Enterprise Architect*
INF	Instituto de Informática da UFRGS
OO	Orientação a Objetos
PAS	Problema de Alocação de Salas
SGBD	Sistema de Gerenciamento de Banco de Dados
UFRGS	Universidade Federal do Rio Grande do Sul

LISTA DE FIGURAS

Figura 2.1: Exemplo de uma folha do bloco de um ambiente. 14
Figura 4.1: Estrutura estática de uma aplicação com Yiiframework. 23
Figura 4.2: Diagrama de casos de uso 23
Figura 4.3: Diagrama com as entidades envolvidas no cadastro de usuários. 26
Figura 4.4: Diagrama com as entidades envolvidas na modelagem dos ambientes. 26
Figura 4.5: Diagrama com as entidades envolvidas na modelagem das alocações. 27
Figura 4.6: Diagrama com as entidades envolvidas na carga semestral. 29
Figura 4.7: Página de acesso do sistema 32
Figura 4.8: Tela de administração de prédios. 33
Figura 4.9: Tela de cadastro de ambientes. 34
Figura 4.10: Tela do consulta de ambientes. 35
Figura 4.11: Tela popup de detalhamento do ambiente. 36
Figura 4.12: Tela de visualização semanal da ocupação do ambiente. 37
Figura 4.13: Tela de confirmação de reserva. 38
Figura 4.14: Tela popup de disciplinas. 39
Figura 4.15: Tela inicial do sistema. 40
Figura 4.16: Tela do primeiro passo da carga semestral. 41
Figura 4.17: Vetor auxiliar do algoritmo de alocação 42
Figura 4.18: Pseudocódigo do algoritmo de alocação 43
Figura 4.19: Tela de administração de modelos 44
Figura 4.20: Tela do primeiro passo da carga semestral – Parte 1. 44
Figura 4.21: Tela do primeiro passo da carga semestral – Parte 2. 44
Figura 4.22: Tela do primeiro passo da carga semestral – Parte 3. 45
Figura 4.23: Tela do primeiro passo da carga semestral – mensagem de erro. 45
Figura 4.24: Tela do segundo passo da carga semestral. 46
Figura 4.25: Tela do segundo passo da carga semestral – Mensagem de erro. 46
Figura 4.26: Tela do segundo passo da carga semestral – Mensagem de sucesso. 46
Figura 4.27: Tela do primeiro passo da alocação sob demanda – Menu “Ambientes”.. 47
Figura 4.28: Tela do segundo passo da alocação sob demanda – Pesquisa. 47
Figura 4.29: Tela do segundo passo da alocação sob demanda – Resultado. 48
Figura 4.30: Tela do terceiro passo da alocação sob demanda. 49
Figura 4.31: Tela do quarto passo da alocação sob demanda. 50
Figura 4.32: Tela do quinto passo da alocação sob demanda. 50

LISTA DE TABELAS

Tabela 3.1: Principais diferenças entre MySQL e SQLite 19

RESUMO

Este trabalho descreve a concepção e o desenvolvimento de um sistema que visa atender uma necessidade real da Secretaria do Instituto de Informática da UFRGS: auxiliar a gerência de seus ambientes.

Inicialmente é descrita a situação atual (procedimento manual) deste gerenciamento, apresentando as atividades relacionadas neste processo, suas deficiências e sugestões de melhorias. A seguir, são discutidos os requisitos para um sistema que visa auxiliar esta gestão. Na sequência, resumem-se as tecnologias empregadas no desenvolvimento deste trabalho. Na descrição da solução proposta são apresentadas todas as etapas do projeto, desde o levantamento dos requisitos junto com os interessados até implementação, passando pela análise e os modelos que foram úteis neste projeto. Por fim, uma descrição passo a passo é também apresentada.

Palavras-Chave: UFRGS, web, INF, ambiente, disciplina.

Allocation and Management Environments System of Informatics Institute of UFRGS

ABSTRACT

This work describes the design and development of a system that aims to attend a real need of UFRGS'S Secretariat of Institute of Informatics: helping the management of their rooms.

Initially is described the current situation (manual procedure) of this management, by presenting related activities in this process, its weakness and suggestions for improvements. Then, are discussed the requirements for a system that aims to help this administration. Following, are summarized the technologies employed in this work. In the description of the proposed solution are presented all steps of the project, from requirements gathering with stakeholders to implement, through the analysis and the models that were useful in this project. Finally, a step by step description is also presented.

.

Keywords: UFRGS, web, INF, environment, discipline.

1 INTRODUÇÃO

Gestão de ambientes consiste em realizar o acompanhamento constante do funcionamento de toda a infraestrutura de uma instituição, garantindo a qualidade de serviços aos usuários. O Instituto de Informática da UFRGS oferece aos seus usuários serviços de qualidade, fruto de uma gestão realizada por pessoas competentes que buscam manter a excelência reconhecida internacionalmente. Porém, algumas atividades desta gestão são realizadas com um nível de dificuldade desnecessário. Este trabalho visa melhorar um destes processos: a gestão de ambientes, permitindo a consulta, a reserva e a alocação de ambientes compartilhados do INF.

Na sequência, serão apresentadas a motivação, os objetivos e a estrutura deste trabalho.

1.1 Motivação

Este trabalho tem como principal motivação a melhoria no processo de gestão dos ambientes do Instituto. Entende-se como melhoria, a informatização do processo, minimizando o tempo necessário para as atividades relacionadas e o abandono do trabalho manual e antiquado utilizado, até então, para este fim.

Este trabalho faz parte de uma campanha promovida por alguns professores do INF (entre os quais o Professor Pimenta, orientador deste trabalho) intitulada ‘Casa de ferreiro, espeto de ferro’, cujo objetivo maior é que vários trabalhos desenvolvidos em disciplinas do INF sejam direcionados a resolução de problemas reais do INF, gerando sistemas realmente úteis e com usuários reais e não aplicações hipotéticas (*toy-applications*). A alocação de ambientes do INF é apenas um dos problemas reais sendo investigados.

Outro problema já investigado é a situação da Comissão de Pesquisa do Instituto de Informática da UFRGS (COMPESQ). Na ocasião, a COMPESQ necessitava de um sistema para auxiliar o gerenciamento dos projetos de pesquisa do INF. Este problema foi resolvido através do trabalho de graduação do aluno Matias Schuler Guenter (GUENTER, 2010).

1.2 Objetivos

O principal objetivo é o desenvolvimento de uma ferramenta capaz de cumprir todas as tarefas relacionadas à gestão dos ambientes. Esta ferramenta deve agregar funcionalidades que substituam os processos atuais, mantendo ou superando a qualidade existente, além de promover melhorias nos processos.

Adicionalmente, para o desenvolvimento deste trabalho, é necessária a utilização da infraestrutura existente no INF. Isto não é apenas para evitar aquisições ou instalações de novas linguagens/plataformas, mas também porque a Administração de Redes possui regras rígidas quanto à adoção de algumas tecnologias. Assim, a tecnologia adotada não pode ser rejeitada pela Administração de Rede do INF senão, mesmo que o sistema funcione, seu uso não será permitido. Para isto, foi necessária a definição e o estudo das tecnologias (sobretudo a linguagem de programação) a serem usadas no trabalho e que sejam efetivamente suportadas pelos servidores do Instituto.

1.3 Estrutura do Trabalho

Para cumprir os objetivos citados, este trabalho foi dividido em cinco capítulos. Após a introdução, no capítulo dois, será apresentado a situação atual (manual) do gerenciamento dos ambientes e os requisitos para um sistema que visa auxiliar esta gestão. Este capítulo apresenta as atividades relacionadas neste processo, suas deficiências e sugestões para melhorias.

No capitulo três são apresentadas as tecnologias empregadas no desenvolvimento deste trabalho. Para cada item apresentado, será dada uma breve explicação da tecnologia, a razão de sua escolha e, em alguns casos, uma comparação com tecnologias alternativas, atividade que foi necessária antes da escolha definitiva.

Na sequência do trabalho, no capítulo quatro, serão apresentadas todas as etapas do projeto, desde o levantamento de requisitos junto com os interessados até a implementação, passando pela análise e seus modelos, extremamente úteis neste projeto. Neste capítulo é apresentado um algoritmo construído para a alocação das disciplinas, apresentando outros algoritmos existentes e a justificativa por se ter optado pela construção de um algoritmo próprio. Uma descrição de funcionamento passo a passo do sistema é também apresentada.

Finalmente, o capítulo cinco apresenta as conclusões do trabalho.

2 GESTÃO DE AMBIENTES NO INSTITUTO DE INFORMÁTICA

O Instituto de Informática possui, aproximadamente, cento e oitenta e cinco ambientes distribuídos em quatro prédios. Dentre estes ambientes, alguns são compartilhados pela comunidade, como dezesseis salas de aula, oito laboratórios, dois anfiteatros, três auditórios e três salas de reuniões. O restante compreende ambientes de uso mais restrito e em geral não compartilhado, como gabinetes de professores, salas da administração, laboratórios de pesquisa, biblioteca e salas destinadas a empresas incubadas no Centro de Empreendimentos do Instituto de Informática da UFRGS (CEI).

Fica claro que a gerência de tantos recursos demanda uma administração eficiente e eficaz. A eficiência é alcançada com a minimização dos erros e a alta qualidade dos procedimentos executados. A eficácia, por sua vez, se restringe ao objetivo final alcançado com êxito.

Este capítulo tem o objetivo de apresentar a situação atual desta administração e os requisitos para um sistema que visa auxiliar esta tarefa, minimizando os erros, facilitando a gerência e obtendo um serviço de qualidade.

2.1 Situação Atual

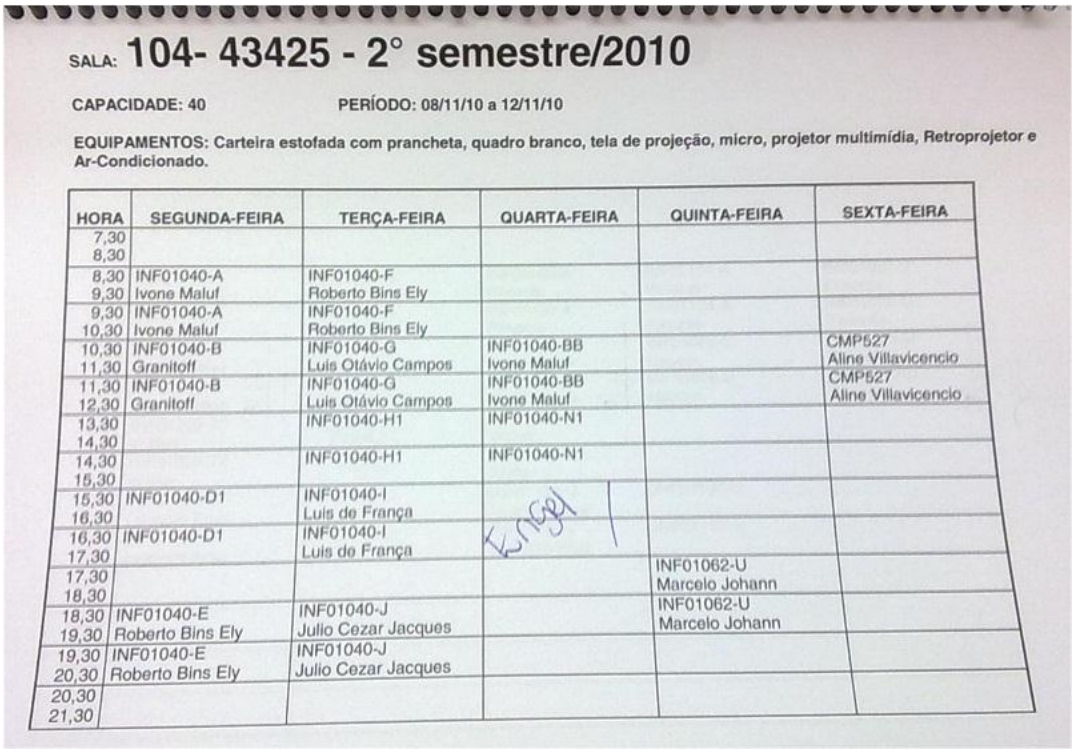
Atualmente, a gestão dos ambientes está sob a responsabilidade da Secretaria da Direção e a Chefia dos Laboratórios. Dentre as suas responsabilidades, as que são relevantes para este projeto são:

- Alocação de salas e laboratórios para as disciplinas de graduação e pós-graduação
- Alocação de salas e laboratórios durante o semestre à medida que são necessárias
- Alocação de salas para reuniões, palestras, eventos e apresentações.

Estas alocações devem obedecer alguns critérios, que se referem a recursos que os ambientes devem possuir para suprir as necessidades daquele que o utilizará. Por exemplo, o ambiente deve possuir sistema de som com microfone; ou é preciso um projetor multimídia; ou então, deve ser capaz de comportar certo número de ocupantes.

Todas estas preocupações referentes à gestão dos ambientes são feitas de forma manual. Os responsáveis contam apenas com o auxílio de planilhas eletrônicas que facilitam um pouco o trabalho. O fruto deste encontra-se na sala da secretaria, onde existem vários blocos de papel com a agenda semestral dos ambientes.

Cada bloco refere-se a um ambiente. Seu conteúdo compreende identificação, capacidade, recursos disponíveis e a tabela de alocações. Esta tabela – onde as linhas representam os horários e as colunas os dias da semana – é preenchida com os dados do interessado em ocupar o ambiente. A imagem abaixo ilustra uma folha de um desses blocos.



SALA: 104- 43425 - 2° semestre/2010

CAPACIDADE: 40 PERÍODO: 08/11/10 a 12/11/10

EQUIPAMENTOS: Carteira estofada com prancheta, quadro branco, tela de projeção, micro, projetor multimídia, Retroprojetor e Ar-Condicionado.

HORA	SEGUNDA-FEIRA	TERÇA-FEIRA	QUARTA-FEIRA	QUINTA-FEIRA	SEXTA-FEIRA
7,30 8,30					
8,30 9,30	INF01040-A Ivone Maluf	INF01040-F Roberto Bins Ely			
9,30 10,30	INF01040-A Ivone Maluf	INF01040-F Roberto Bins Ely			
10,30 11,30	INF01040-B Granitoff	INF01040-G Luis Otávio Campos	INF01040-BB Ivone Maluf		CMP527 Aline Villavicencio
11,30 12,30	INF01040-B Granitoff	INF01040-G Luis Otávio Campos	INF01040-BB Ivone Maluf		CMP527 Aline Villavicencio
13,30 14,30		INF01040-H1	INF01040-N1		
14,30 15,30		INF01040-H1	INF01040-N1		
15,30 16,30	INF01040-D1	INF01040-I Luis de França			
16,30 17,30	INF01040-D1	INF01040-I Luis de França			
17,30 18,30				INF01062-U Marcelo Johann	
18,30 19,30	INF01040-E Roberto Bins Ely	INF01040-J Julio Cezar Jacques		INF01062-U Marcelo Johann	
19,30 20,30	INF01040-E Roberto Bins Ely	INF01040-J Julio Cezar Jacques			
20,30 21,30					

Figura 2.1: Exemplo de uma folha do bloco de um ambiente.

Nota-se na figura acima, horários preenchidos com o código de uma disciplina e o nome de um professor. Isto significa que, neste horário, a sala está reservada para a disciplina informada e a mesma será ministrada pelo professor também informado. Tais informações são inseridas nestes documentos no início do semestre, de forma manual, por um funcionário do Instituto. Os departamentos de Informática Teórica (INT) e Aplicada (INA), além dos programas de pós-graduação em Computação (PPGC) e Microeletrônica (PPGM), enviam para a secretaria a demanda necessária para o semestre letivo. Através destes documentos é realizada a distribuição das disciplinas nos ambientes disponíveis, respeitando as necessidades requeridas de cada disciplina. Lembrando, tudo isto é feito manualmente.

Mas a necessidade de gestão dos ambientes não termina com essa distribuição inicial. Ao longo do semestre, por diversas razões, são necessárias alocações esporádicas. Seja para um professor ministrar uma aula extra em laboratório, ou aplicar uma prova em um ambiente mais espaçoso, ou uma reunião para um dos colegiados do INF. Esta reserva é feita de duas maneiras:

1. Interessado entra em contato com a secretaria do INF e solicita a reserva de um ambiente para um determinado horário. Este contato pode ser via telefone ou e-mail.
2. Interessado se desloca até a secretaria para efetuar a reserva.

Em ambos os casos, existe a necessidade de procurar em todos os documentos por um ambiente no horário pretendido, que esteja vago e que possua os recursos necessários (projetor, quadro, etc.).

Ficou ainda mais evidente que este processo manual e centralizado pela secretaria não condiz com o centro de excelência internacional que é o Instituto de Informática. A secretaria necessitaria de um sistema que auxilie este gerenciamento. Na seção subsequente, são apresentados os requisitos para que este sistema supra as necessidades atuais e mantenha a eficácia alcançada com todo este trabalho realizado pela secretaria.

2.2 Requisitos para um Sistema de Gerenciamento de Ambientes

A engenharia de requisitos é uma fase crucial de todo e qualquer projeto. Uma de suas atividades consiste na obtenção dos requisitos do sistema, podendo ser alcançada através de entrevistas e questionários direcionados aos usuários. Esta atividade é de fundamental importância para que se desenvolva exatamente o esperado pelo mesmo.

Para o desenvolvimento deste projeto, algumas reuniões foram realizadas com os principais interessados, mais evidentemente maior atenção foi dada à secretaria do INF. Ademais, alguns documentos foram disponibilizados como forma de auxílio, tanto para demonstrações dos problemas enfrentados atualmente, quanto para ratificação do que foi exposto nas entrevistas.

Nas subseções seguintes estão expostos os requisitos levantados nesta fase do projeto.

2.2.1 Gestão de Usuários

Um requisito básico do sistema consiste na alteração do funcionamento de reservas sob demanda. O software precisa disponibilizar aos funcionários e utilizadores dos recursos do Instituto, uma ferramenta capaz de efetuar a reserva de um ambiente sem a necessidade do envolvimento dos membros da secretaria. Para isso, faz-se necessário a inclusão de um gerenciamento de usuários.

Através de cadastros, os usuários são incluídos no sistema. Estes passam a contar com credenciais que os habilitarão a utilizar o mesmo. Com isso, além de oferecer um controle sobre o grupo de pessoas que poderá utilizar o software, estas pessoas contam com a garantia de que ninguém utilizará seu nome para efetuar reservas.

2.2.2 Gestão de Ambientes

Todo e qualquer ambiente possui uma série de recursos disponíveis aos usuários. Estes recursos englobam projetores multimídia, computadores, sistema de som com microfones, poltronas com prancheta, climatização, entre outros. Além disso, algumas salas possuem usos específicos, como palestras, apresentações, eventos, aulas teóricas, aulas práticas e reuniões.

O gerenciamento desses recursos foi definido como mais um requisito do sistema. Os usuários administradores (usuários com mais privilégios) devem ser capazes de cadastrar e administrar estes ambientes, definindo seus recursos e finalidades. Estes ambientes ficariam então disponíveis no sistema para alocações.

2.2.3 Alocação para as Disciplinas de Graduação e Pós-Graduação

O principal objetivo que este sistema visa atingir é a mitigação do trabalho manual no gerenciamento dos ambientes. Muito deste trabalho está na distribuição das aulas das disciplinas de graduação e pós-graduação, executado no início de todos os semestres letivos.

Visando atingir este objetivo, incluiu-se como requisito esta distribuição semestral. O usuário administrador deve ser capaz de cadastrar todas as disciplinas ministradas no semestre em questão, incluindo:

- Quem as ministrará,
- Os recursos necessários que devem estar presente no ambiente escolhido,
- Os dias da semana que ocorrerão as aulas e
- O horário de início e fim das aulas

Além disso, existem disciplinas com códigos diferentes, mas que são ministradas pelo mesmo professor no mesmo horário. Esta situação deve ser detectada pelo sistema e atribuir somente um ambiente no horário escolhido para ambas as disciplinas.

2.2.4 Alocação sob Demanda

Após a inclusão de um usuário no sistema, efetuada através da gerência de usuários descrito anteriormente, este deve estar apto a efetuar reservas em seu nome. Somente o responsável pela reserva, ou um administrador, pode cancelar ou alterar a mesma.

A opção de reserva deverá estar disponível através de uma consulta das alocações do ambiente. Este, por sua vez, deve ser encontrado através de consulta de acordo com recursos necessários.

Desta forma, o ato de reservar um ambiente é melhorado, comparado à maneira que é feita hoje, dispensando as procuras manuais nos blocos da secretaria. E melhoraria ainda mais se acabasse com a necessidade de entrar em contato com a secretaria para este fim. Surge então o próximo requisito, descrito a seguir.

2.2.5 Plataforma Web

Devidamente cadastrados, os usuários podem acessar o sistema para consultar ocupações dos ambientes e efetuar suas reservas. Administradores podem gerenciar usuários e ambientes. E tudo isto deve ser feito de qualquer lugar.

A solução é a plataforma web. Sem a necessidade de instalação, todos os usuários acessam o sistema através de navegadores. O sistema deve possuir um conjunto de páginas, cada qual com sua funcionalidade e juntas compondo a solução para a gestão de ambientes do INF.

2.3 IAlocador

Com o intuito de suprir esta deficiência da secretaria, alguns professores propuseram a implementação de um sistema de alocação como trabalhos em suas disciplinas. Entre estes, a Professora Érika Fernandes Cota.

No primeiro semestre de 2010, a Professora Érika, na disciplina de Técnicas de Construção de Programas, propôs como trabalho a construção de um sistema para tal

fim. Devido ao tempo disponível aos alunos para esta atividade, este trabalho teve continuidade no segundo semestre da mesma disciplina, aproveitando os melhores trabalhos do semestre anterior. Deste esforço nasceu o sistema IAlocador.

Este sistema, disponível para a comunidade, atende alguns requisitos, mas não todos. O primeiro requisito encontrado é o seu desenvolvimento voltado à plataforma web. Além disso, seu uso está disponível apenas a usuários cadastrados. Outro requisito disponibilizado pelo sistema é a reserva de ambientes sob demanda, onde usuários cadastrados efetuam reservas em seu nome para os diversos fins. Por fim, o usuário pode consultar os horários das salas, tanto disponíveis como ocupados (mapa de ocupação dos ambientes – visão semanal).

Mas nem todos os requisitos levantados com os interessados são encontrados neste sistema. Em primeiro lugar, não é possível visualizar o mapa de ocupação sem estar autenticado. Esta funcionalidade é útil para que alunos possam consultar os ambientes designados às disciplinas que lhe interessam. Em segundo lugar, os horários disponíveis para reservar estão em um intervalo bem definido, hoje utilizado para as aulas. Porém, pode ser necessário reservar um espaço por um período menor de tempo, como uma reunião, por exemplo. Se não houver a opção de reserva por um tempo mais curto, o ambiente ficará ocioso até o horário final da mesma.

Este sistema está em fase de construção. As funcionalidades estão sendo desenvolvidas e liberadas para a comunidade gradativamente. Para isto, a Prof.ª Érika conta com um bolsista de iniciação científica, a saber, Alexandre T. Salle, que está trabalhando neste projeto.

Estas foram as observações que puderam ser citadas sobre este sistema desenvolvido pelos alunos da Prof.ª Érika. Isto porque, tanto orientador quanto o autor deste trabalho tiveram acesso a essa informação no final do semestre, dias antes da apresentação do mesmo. Por esta razão, o IAlocador não será mais citado no restante deste documento.

3 TECNOLOGIAS UTILIZADAS NO PROJETO

A vasta lista de ferramentas disponíveis para o desenvolvimento de um sistema torna a escolha de cada objeto da arquitetura uma tarefa árdua. Qual a linguagem de programação que será usada? Qual o sistema de gestão de banco de dados será empregado? Existe algum framework que possa auxiliar e agilizar o processo? Estas são apenas algumas perguntas que são feitas antes de ser escrita a primeira linha de código.

Neste projeto, algumas perguntas já nasceram com resposta. Isto porque o sistema será instalado sobre uma plataforma existente. Logo, a linguagem de programação deveria ser aceita por esta plataforma, bem como o banco de dados.

Assim, o projeto teve poucas escolhas, restritas ao processo de análise e modelagem do sistema, bem como a codificação do mesmo. Nas linhas que seguem deste capítulo, pretende-se enumerar e tecer comentários acerca das tecnologias empregadas.

3.1 Ferramenta CASE

Do inglês *Computer-Aided Software Engineering*[1] (CASE), são ferramentas que auxiliam nas atividades de engenharia de software. Desde a análise de requisitos até a programação e testes, são ferramentas automatizadas com propósitos em todas as etapas do ciclo de desenvolvimento de um sistema.

A ferramenta CASE escolhida para a especificação do sistema foi a *Enterprise Architect* (EA), da *Sparx Systems*. O uso desta ferramenta se justifica somente pela familiaridade do autor com a mesma, utilizada há mais de dois anos na vida profissional, uma vez que não se trata de uma ferramenta livre e tampouco foi comparada com outras ferramentas existentes.

O diagrama de casos de uso, essencial no início do projeto para organizar os requisitos do sistema, foi facilmente desenhado com o auxílio desta ferramenta. O diagrama de classes também foi utilizado na especificação do sistema.

Outra funcionalidade da ferramenta explorada no projeto foi a especificação do modelo relacional. Através de diagramas, foi possível modelar as tabelas e suas relações, especificando todas as restrições de integridade necessárias. Além disso, a ferramenta suporta a modelagem do esquema e a geração de scripts DDL (do inglês, *Data Definition Language*) de onze Sistemas de Gerenciamento de Banco de Dados (SGBD). Entre eles, o SGBD utilizado neste projeto e citado na seção posterior.

---

[1] http://www.sparxsystems.com/products/ea/index.html acessado em novembro de 2011.

3.2 Banco de Dados

Um banco de dados é uma coleção de dados relacionados com algum significado inerente. Esta coleção é organizada com o propósito de servir de base para recuperação de informações. Com o crescimento de tais informações, impulsionado pelo aumento do processamento de dados que a informática gerou, surgiu a necessidade de gerenciar esses dados: surgem os Sistemas de Gerenciamento de Banco de Dados.

O principal objetivo de uma SGBD é prover um ambiente adequado e eficiente para armazenar e recuperar informações de um banco de dados. O gerenciamento de dados envolve a definição de estruturas para o armazenamento das informações e o fornecimento de ferramentas para manipulá-las.

Para este projeto, o SGBD utilizado foi o MySQL[2], uma das opções disponíveis nos servidores do Instituto de Informática. Esta escolha se justifica quando se leva em conta apenas uma característica do mesmo: gerência de usuários e permissões. A outra opção é o uso do SQLite[3], um motor de banco de dados que apenas lê e escreve diretamente em arquivos do disco rígido. Sua aplicação não é recomendada para substituir um SGBD e sim uma alternativa à leitura de arquivos. Ambos possuem licença gratuita.

A tabela abaixo mostra as diferenças entre o MySQL e o SQLite.

Tabela 3.1: Principais diferenças entre MySQL e SQLite

MySQL	SQLite
Difícil de configurar e necessária configuração de usuário	Fácil de configurar e em muitos casos nenhuma instalação necessária
Adequado para gerenciamento de usuários e permissões	Não adequado quando o gerenciamento de usuários é necessário
Adequado para transações de concorrência sobre os dados	Não adequado onde as transações de concorrência são necessárias
Recomendado para aplicações em larga escala	Não recomendado para aplicações em larga escala
Recomendado para aplicações cliente-servidor	Não recomendado para arquiteturas cliente-servidor

Fonte: SQLITE (2011)

3.3 Linguagem de Programação

Surgidas a pouco mais de meio século, as linguagens de programação permitem que o programador tenha mais produtividade, permitindo a ele expressar mais facilmente suas intenções. São projetadas para possuir uma sintaxe de mais alto nível, tornando-as mais legíveis para os humanos.

---

[2] http://www.apachefriends.org/pt_br/xampp-windows.html acessado em março de 2011.

[3] http://www.sqlite.org/download.html acessado em fevereiro de 2011.

Seguindo estes objetivos e tentando aproximar o mundo real do mundo virtual, surgiram as linguagens de programação orientadas a objetos. Na programação orientada a objetos, o programador é responsável por moldar os objetos e explicar para estes como devem interagir. Os objetos interagem através de mensagens e cabe ao programador especificar quais serão estas mensagens, bem como qual será a ação que cada objeto deverá realizar ao receber aquela mensagem.

Por volta de 1994, surgiu uma linguagem voltada ao desenvolvimento de páginas pessoais para a internet com o nome de *Personal Home Page*: o PHP[4]. Mas somente em sua terceira versão contava com alguns recursos de orientação a objetos. Atualmente, em sua quinta versão, a linguagem possui todos os recursos de orientação a objetos, se tornando uma das linguagens mais utilizadas para o desenvolvimento web.

Esta foi a linguagem escolhida para o desenvolvimento deste projeto, por razões técnicas, uma vez que é a linguagem suportada pelos servidores do Instituto de Informática. Além disso, PHP é distribuída sob licença gratuita.

Os desenvolvedores contam ainda com uma ferramenta de apoio no desenvolvimento de softwares: o ambiente integrado de desenvolvimento (IDE – do inglês *Integrated Development Enviroment*). O ambiente escolhido para trabalhar com a linguagem PHP foi o Netbeans IDE[5]. Sua escolha baseia-se no seu suporte a linguagem escolhida e também possuir licença gratuita.

3.4 Framework de Desenvolvimento

Um Framework, em desenvolvimento de software, é uma abstração que captura a funcionalidade comum a várias aplicações. Em outras palavras, é um esqueleto que contém partes comuns a uma classe de problemas. As partes faltantes deverão ser acrescentadas ao esqueleto para que este passe a resolver o problema proposto.

Para o desenvolvimento deste projeto, utilizou-se o Yiiframework[6] – framework PHP de alto desempenho para desenvolvimento de aplicações web. O Yii vem com suporte a recursos avançados, tais como:

- *Model-View-Controller* (MVC) – padrão de arquitetura de software que visa separar a lógica de negócio da lógica da apresentação,
- *Data Access Object* (DAO) – padrão para persistência de dados que permite separar regras de negócio das regras de acesso ao banco de dados,
- *Active Record* (AR) – padrão de projeto que permite mapear tabelas de banco de dados relacionais em classes,
- Autenticação e controle de acesso baseado em regras,
- Cache,
- Segurança preventiva contra ataques a páginas web, como *SQL injection* e *Cross-site scripting* (XSS),

---

[4] http://www.apachefriends.org/pt_br/xampp-windows.html acessado em março de 2011.

[5] http://netbeans.org/downloads/ acessado em março de 2011.

[6] http://www.yiiframework.com/download/ acessado em março de 2011.

- Tratamento de erro e registro de eventos,
- Geração automática de código,
- Validação de formulários baseado em regras e
- Integração com JQuery e AJAX.

A escolha deste framework baseou-se em suas características, pela simplicidade de instalação e pela excelente documentação disponível, indispensável para a configuração e utilização de seus recursos. O Yii framework é um software livre e distribuído sob os termos da licença BSD.

3.5 Sistema de Controle de Versão

Um software de controle de versão tem como finalidade a gerência das diferentes versões de um documento qualquer. São muito utilizados no desenvolvimento de software, onde existem vários arquivos sendo acessados e modificados por muitos. É uma ferramenta extremamente útil tanto em grandes projetos comerciais quanto em pequenos projetos pessoais.

As principais vantagens do seu uso são:

- Controle do histórico – facilidade de desfazer uma modificação e possibilidade de resgatar uma versão mais antiga, permitindo analisar as diferenças entre duas versões.
- Trabalho em equipe – possibilidade de mais de uma pessoa trabalhar sobre o mesmo conjunto de arquivos, minimizando tempo gasto para edições e integrações dos mesmos.
- Marcação e resgate de versões estáveis – capacidade de marcar uma versão do projeto como estável, podendo recuperá-la quando e se necessário.
- Ramificação – viabilidade de dividir um projeto em outros menores, sem que o desenvolvimento de um interfira no outro.

Este projeto contou desde o seu início com um controlador de versionamento denominado Subversion[7], também conhecido como svn. O cliente instalado nas máquinas de desenvolvimento foi o Tortoise SVN[8]. Sua escolha se deve a familiaridade com a mesma e sua licença gratuita.

---

[7] http://subversion.apache.org/ acessado em setembro de 2010.

[8] http://tortoisesvn.net/downloads.html acessado em setembro de 2010.

4 A INFORMATIZAÇÃO DA GESTÃO DE AMBIENTES DO INSTITUTO DE INFORMATICA

Este capítulo tem o intuito de apresentar o desenvolvimento do trabalho. Partindo da arquitetura do sistema, passando pelos modelos utilizados como auxílio e organização, até a implementação, onde serão detalhadas suas iterações.

4.1 Arquitetura

Aqui se destaca os principais aspectos da arquitetura da solução desenvolvida. Como deve ser realizada para a plataforma web, a escolha de um framework é de grande valia. E para esta solução, o escolhido foi o Yiiframework, já apresentado anteriormente.

O Yii implementa o padrão de projeto Modelo-Visão-Controlador[9] (MVC), amplamente utilizado neste tipo de programação. Este padrão visa separar a lógica de negócio da lógica de apresentação. O modelo representa a informação e as regras de negócio. A visão contém os elementos da interface do usuário. Por sua vez, o controlador gerencia a comunicação entre a visão e o modelo.

Além do MVC, o Yii introduz um controlador de frente, outro padrão de projeto que gerencia o processamento das requisições. Para cada uma dessas requisições, ele coleta algumas informações para depois despachar para o controlador adequado. Entre estas informações está a autenticação e a autorização. A autenticação verifica os dados do usuário que realizou o pedido. A autorização compreende o controle de acesso, verificando se o responsável pela requisição possui privilégios suficientes para que a mesma seja processada. A figura abaixo ilustra a estrutura estática de uma aplicação com o Yii.

---

[9] http://pt.wikipedia.org/wiki/MVC acessado em outubro de 2011.

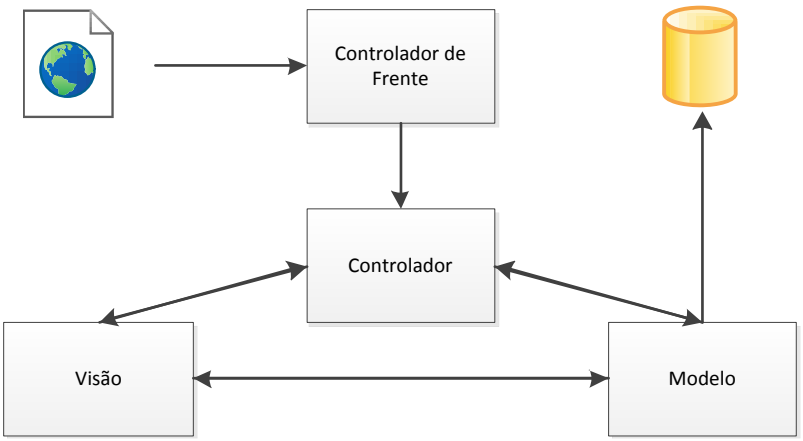


Figura 4.1: Estrutura estática de uma aplicação com Yiiframework.

4.2 Modelagem

Esta seção tem como objetivo apresentar a modelagem do sistema. A implementação do sistema foi planejada através de diagramas de caso de uso e de entidade relacionamento. O primeiro serviu para organizar as ideias e discutir melhor sobre as funcionalidades do sistema. O segundo descreve o modelo de dados, ou seja, as tabelas do banco de dados e os relacionamentos entre si.

4.2.1 Diagrama de Casos de Uso

O diagrama de casos de uso oferece uma visão geral do sistema ao analista. Nele é possível definir as funcionalidades disponíveis para cada ator. Um ator é uma entidade externa que interage com o sistema. Esta entidade pode ser um(ns) usuário(s) ou outro(s) sistema(s). A figura 4.2 apresenta os casos de uso que representam as funcionalidades previstas, definidas sobre os requisitos levantados.

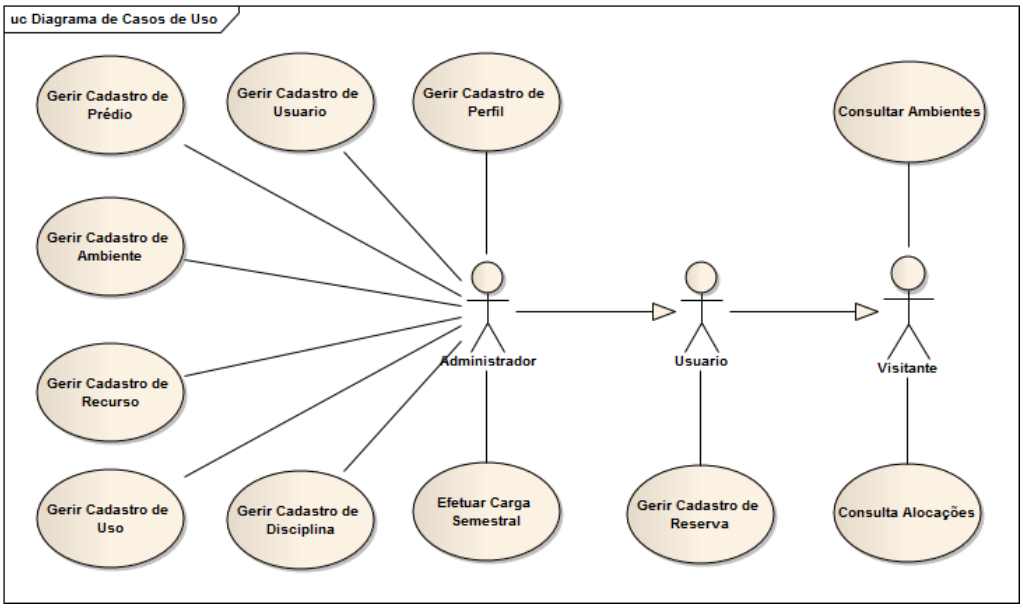


Figura 4.2: Diagrama de casos de uso

Primeiramente vamos focar nos atores descritos na figura 4.2. O visitante é um usuário com privilégio apenas de consulta. E uma consulta restrita apenas a ambientes e alocações. Não possui cadastro no sistema e, portanto, não é capaz de se autenticar no mesmo. Esta funcionalidade está prevista para o ator usuário. Devidamente cadastrado e autenticado, pode efetuar reservas em seu nome e gerenciá-las. Por fim, o administrador é o ator com permissão para todas as funcionalidades do sistema. Ele é quem gerencia os usuários, ambientes e efetua a carga das disciplinas no inicio dos semestres.

Uma das funcionalidades é a gerencia de perfis. Os perfis são destinados a classificação dos usuários. O sistema pode possuir vários perfis, mas somente os administradores tem o acesso total ao sistema.

A gerência de usuários se restringe ao cadastro daqueles que poderão se autenticar no sistema. Para o cadastro ser aceito, o usuário precisa ter um endereço eletrônico ativo do Instituto de Informática. Esta restrição foi adicionada, pois o sistema deve servir de apoio àqueles que usufruem dos ambientes do INF.

A gestão destes ambientes está representada na figura 4.2 pelos casos de uso: Gerir cadastro de prédio, gerir cadastro de ambientes, gerir cadastro de usos e gerir cadastro de recursos. Todos os ambientes do Instituto estão localizados em algum prédio. Por isto, e pela possibilidade de novos prédios serem incorporados, permite-se a gestão deles. Além disso, todo ambiente possui uma série de recursos disponíveis. Estes recursos podem aumentar ou diminuir. Logo, é preciso ter um controle sobre estes recursos em cada ambiente, permitindo alterar seus recursos de acordo com a viabilidade dos mesmos. Por último, os ambientes possuem uma ou mais finalidades (ou usos, como sugere o título do caso de uso). Como a lista de usos possíveis dos ambientes pode mudar, existe a necessidade de gerência destes.

O próximo caso de uso a ser apresentado é a gerência do cadastro das disciplinas. Como a grande maioria das reservas serão efetuadas para aulas, tanto das disciplinas da graduação quanto da pós-graduação, o sistema permite o cadastro de disciplinas e o gerenciamento das mesmas (editar, excluir, pesquisar, etc.).

Seguindo com a descrição dos casos de uso, o próximo se refere a principal funcionalidade do sistema, a carga semestral. Um administrador pode cadastrar as disciplinas que serão oferecidas durante o semestre letivo, incluindo a disciplina, quem a ministrará, o horário de inicio e fim da aula, os dias da semana que ocorrerão as aulas, os recursos necessários no ambiente e a capacidade da turma. Depois de cadastrado todas as disciplinas, o sistema se encarrega de encontrar o ambiente adequado para todas. Adicionalmente, os administradores podem utilizar este cadastro como modelo para futuras cargas.

Durante o semestre, os usuários podem efetuar reservas em seu nome, podendo gerenciá-las (caso de uso “Gerir Cadastro de Reserva” apresentado na figura acima). Os usuários somente alteram reservas por eles efetuadas. Apenas os administradores são capazes de gerenciar todas as reservas do sistema.

Os dois casos de uso restantes se referem a consultas. E ambos estão disponíveis aos visitantes. O caso de uso “Consultar Ambientes” descreve a funcionalidade de busca nos ambientes cadastrados. A partir do resultado destes ambientes, é possível visualizar o seu mapa de alocação. Este mapa, referenciado pelo caso de uso “Consultar Alocações”, permite uma visualização semanal do ambiente, informando as alocações existentes e os horários vagos. Esta visualização tem como objetivo substituir os atuais

blocos de papel existentes na secretaria, cuja função já foi apresentada em outro momento.

4.2.2 Modelo do Banco de Dados

Como esta aplicação utiliza um banco de dados para o armazenamento de informações, torna-se necessária a definição do modelo relacional. Este modelo baseia-se nos conceitos de entidade e relação. O primeiro é um elemento constituído de atributos onde, na prática, darão origem a uma tabela e suas colunas. O segundo define uma associação entre duas entidades.

As relações podem assumir diferentes cardinalidades. As cardinalidades definem as quantidades mínima e máxima de relacionamentos que uma entidade mantém com outra. Estas cardinalidade podem ser:

- Um para um (1:1) – uma entidade A mantém uma e apenas uma relação com uma entidade B.
- Um para muitos (1:N) – uma entidade A mantém pelo menos uma ou muitas relações com a entidade B.
- Muitos para muitos (M:N) – uma entidade A mantém muitas relações com a entidade B e essa mantém muitas relações com a entidade A.

Para facilitar a visualização, o modelo será apresentado em pedaços. Os pedaços foram escolhidos com o intuito de agrupar algumas funcionalidades em comum.

A figura 4.3 ilustra as tabelas envolvidas nas informações dos usuários. O cadastro de um usuário é realizado informando-se nome, perfil, endereço eletrônico, senha e situação. Como é possível utilizar o perfil para categorizar os usuários, foi criada a tabela PERFIL_USUARIO, que possui um atributo definindo um código único e uma descrição. A tabela USUARIO possui uma relação 1:1 com a tabela PERFIL_USUARIO. Os atributos ID_PERFIL de ambas as tabelas fazem parte desta relação. O mesmo ocorre com as tabelas USUARIO e SITUACAO_USUARIO, onde o atributo ID_SITUACAO de ambas as tabelas participam da relação. A finalidade desta relação é definir a situação do cadastro do usuário no sistema (ativo, bloqueado, senha temporária, etc.).

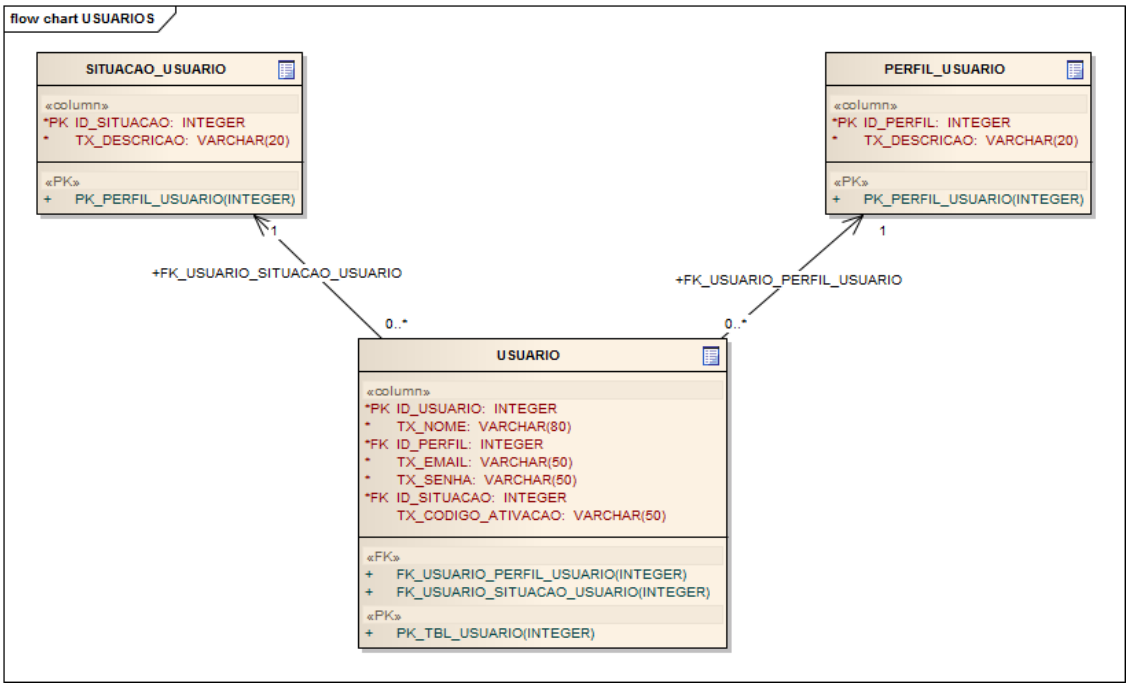


Figura 4.3: Diagrama com as entidades envolvidas no cadastro de usuários.

O sistema deve ser capaz de gerenciar os ambientes e seus recursos. Um ambiente possui uma série de recursos e usos. A lista de recursos de um ambiente pode aumentar ou diminuir com o tempo. O mesmo ocorre com os usos. Um ambiente pode permitir um novo uso ou deixar de permiti-lo. A figura 4.4 a seguir ilustra a modelagem do banco de dados para permitir que estas informações possam ser armazenadas.

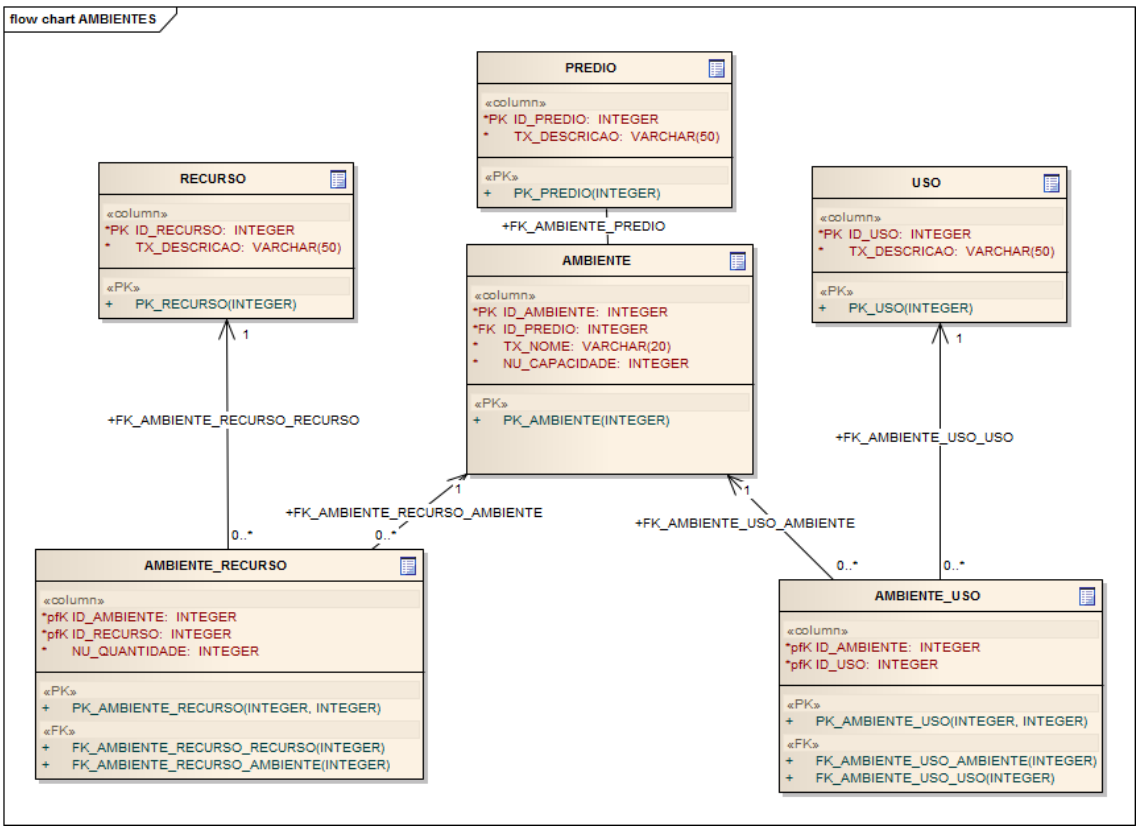


Figura 4.4: Diagrama com as entidades envolvidas na modelagem dos ambientes.

Os ambientes estão dispostos em prédios. A tabela PREDIO tem a função de armazenar as informações dos prédios do Instituto. A tabela AMBIENTE possui uma relação de 1:1 com a tabela prédio, onde o atributo ID_PREDIO de ambas as tabelas fazem parte da relação. Outros dados como o nome do ambiente (201, ANFV, etc.) e a sua capacidade são os demais atributos da tabela AMBIENTE.

Os ambientes possuem um ou mais usos. Apenas alguns ambientes podem estar destinados às aulas teóricas, assim como outros para aula práticas, reuniões, eventos e qualquer outro tipo de uso que se queira definir. Para que essa informação pudesse ser armazenada no banco de dados, foram criadas as tabelas USO e AMBIENTE_USO. A primeira serve somente para cadastrar os usos disponíveis. A segunda é utilizada para definir a lista de usos que cada ambiente possui.

Assim como os usos, os ambientes podem possuir uma lista de recursos. Esses recursos podem possuir quantidades diferentes. Igualmente aos usos, foram criadas as tabelas RECURSO e AMBIENTE_RECURSO. A primeira serve apenas para se cadastrar os recursos que podem ser cadastrados para um ambiente. A segunda serve para cadastras estes diversos recursos para um determinado ambiente. Nesta, o atributo NU_QUANTIDADE é responsável pela definição da quantidade disponível para o recurso em questão.

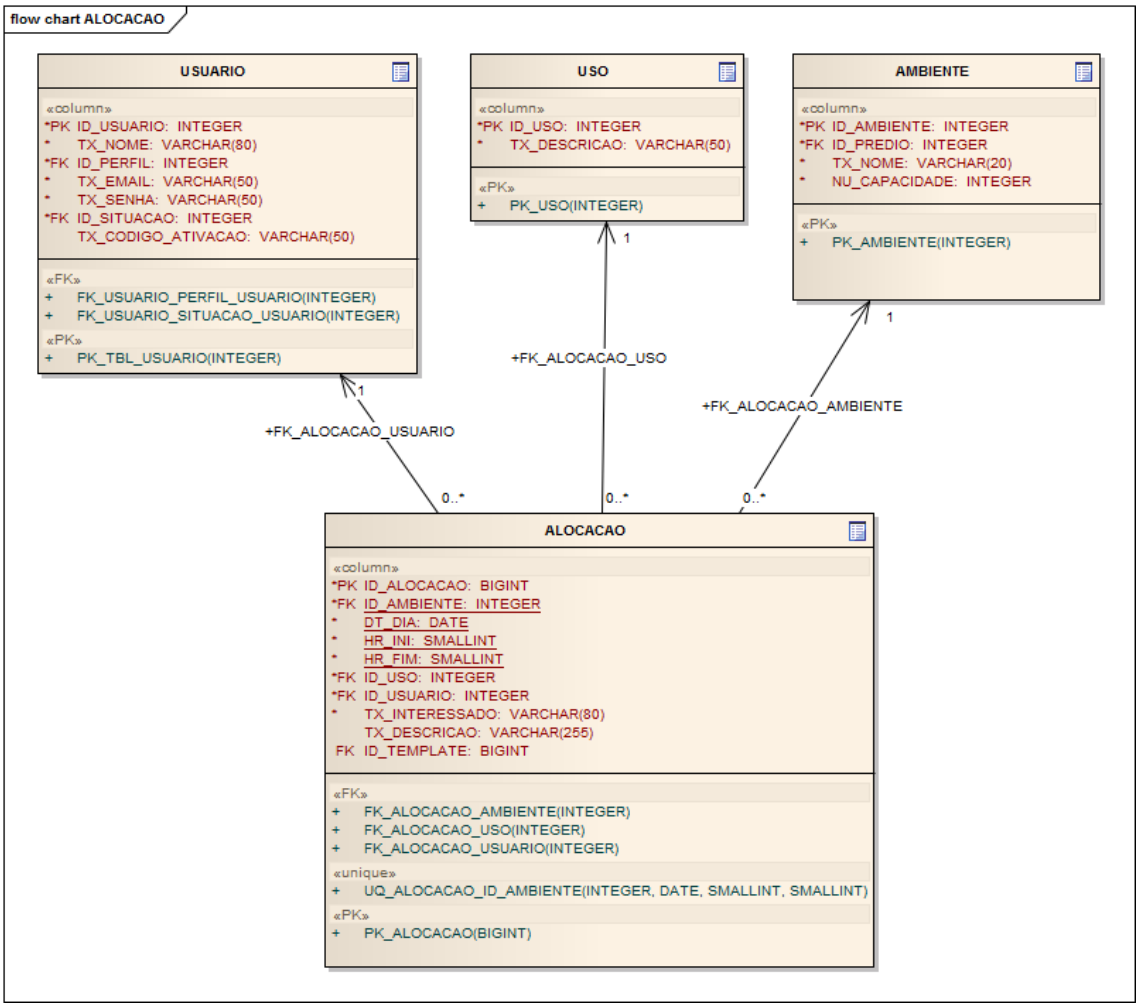


Figura 4.5: Diagrama com as entidades envolvidas na modelagem das alocações.

Na figura 4.5 acima, estão apresentadas as tabelas envolvidas no armazenamento das alocações dos ambientes. Toda alocação deve possuir um usuário responsável. Por isso, a tabela ALOCACAO possui uma relação com a tabela USUARIO, onde o atributo ID_USUARIO de ambas as tabelas fazem parte desta relação. Nem sempre esse usuário refere-se aquele que utilizara o ambiente alocado. Este responsável diz respeito a quem efetuou a reserva no sistema. O utilizador do ambiente alocado está armazenado no atributo TX_INTERESSADO da tabela ALOCACAO.

Toda reserva deve ter uma finalidade. Esta finalidade precisa estar disponível no ambiente a ser alocado. Exatamente por isso existe uma relação entre a tabela ALOCACAO e a tabela USO, onde o atributo ID_USO de ambas as tabelas fazem parte desta relação.

Como uma reserva está associada a um ambiente, existe uma relação entre a tabela ALOCACAO e AMBIENTE, onde o atributo ID_AMBIENTE de ambas as tabelas fazem parte desta relação. As demais informações essenciais para uma alocação são armazenadas nos outros atributos da tabela ALOCACAO. Dentre essas informações estão o dia, hora inicial e hora final da reserva.

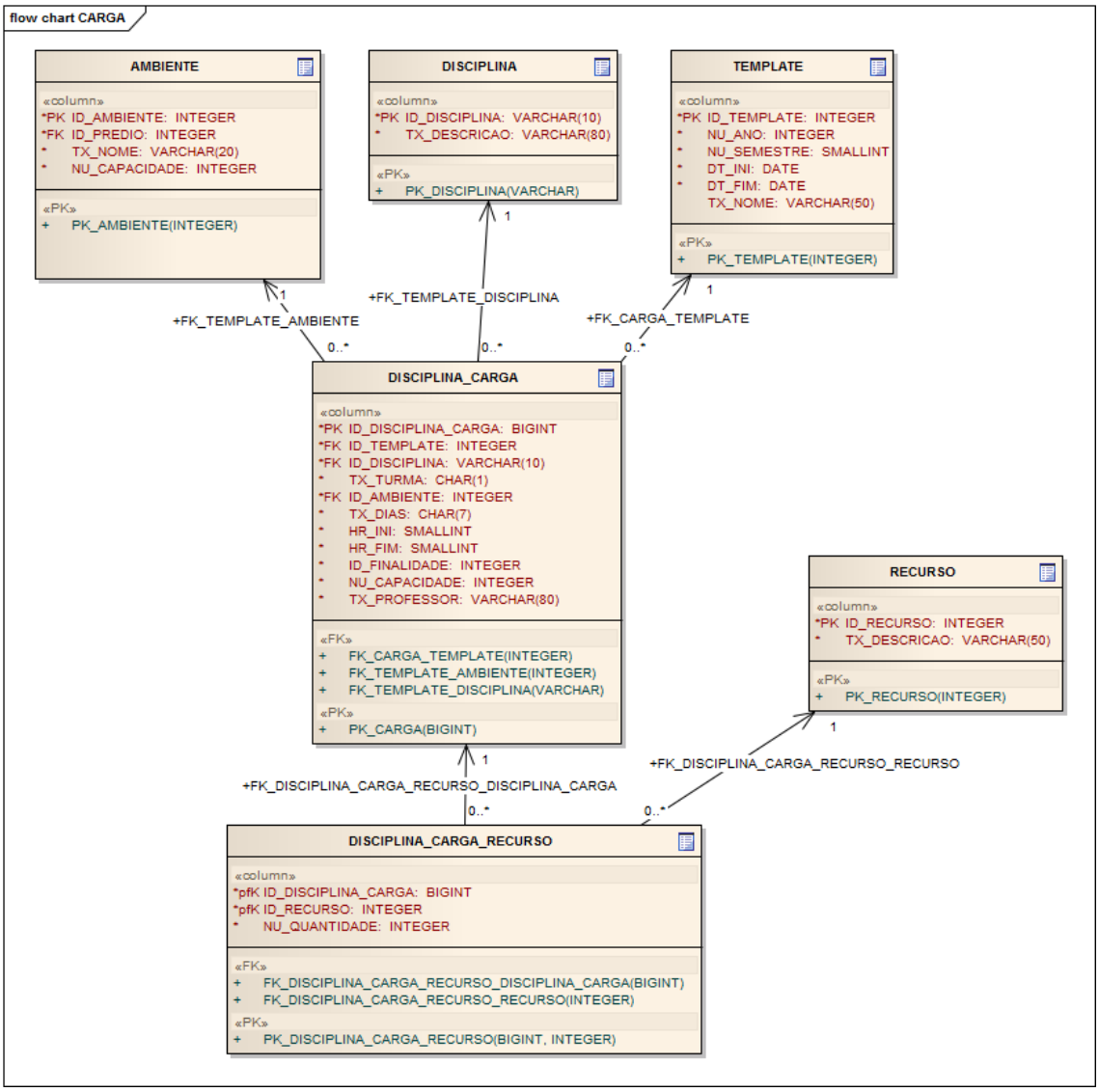


Figura 4.6: Diagrama com as entidades envolvidas na carga semestral.

Por fim, a figura 4.6 acima ilustra as tabelas envolvidas na criação da carga efetuada no início dos semestres letivos. Tais informações são armazenadas de forma que se possa utilizar uma carga como modelo para outras cargas.

Para isso, foi criada a tabela TEMPLATE. Nesta tabela são definidos o ano, semestre, dia inicial e final do semestre e um atributo nome a ser utilizado pelo usuário para controle próprio.

No momento da criação de uma carga semestral, o usuário define as disciplinas que farão parte da mesma. Para cada disciplina é possível informar os dias da semana que as aulas serão ministradas, o uso (aula prática, teórica, etc.), a hora inicial e final, a capacidade da turma, o professor e os recursos que precisam estar disponíveis no ambiente. Todas estas informações são armazenadas nos atributos da tabela DISCIPLINA_CARGA, com exceção dos recursos.

Os recursos são armazenados através da relação da tabela DISCIPLINA_CARGA com a tabela DISCIPLINA_CARGA_RECURSO, onde o atributo ID_DISCIPLINA_CARGA de ambas as tabelas fazem parte desta relação. A tabela DISCIPLINA_CARGA_RECURSO possui uma relação com a tabela RECURSO já

apresentada anteriormente. O atributo ID_RECURSO de ambas as tabelas fazem parte desta relação.

O sistema permite o cadastro das disciplinas oferecidas pelos cursos de graduação e pós-graduação do INF. Para armazenar estas informações foi criada a tabela DISCIPLINA. A tabela DISCIPLINA_CARGA se relaciona com a tabela DISCIPLINA para garantir que somente disciplinas cadastradas estão sendo utilizadas para alocações.

4.3 Desenvolvimento

Até aqui foram apresentadas as etapas iniciais do projeto. O levantamento de requisitos junto ao cliente, a modelagem do sistema e a definição da arquitetura e tecnologias a serem empregadas. Agora, será apresentado como ocorreu o planejamento e a execução do desenvolvimento da aplicação.

O desenvolvimento apresentado a seguir segue princípios ágeis, com pouca modelagem e focando principalmente na codificação da aplicação. Como o framework escolhido utiliza as tabelas do banco de dados para gerar as classes do sistema, esta modelagem para criação e organização das classes pode ser dispensada. Além disso, o framework gera automaticamente o controlador e as telas das operações CRUD. Este recurso ratifica que um método ágil se encaixa muito bem neste tipo de desenvolvimento. Embora gerado automaticamente, todas as classes podem ser alteradas para melhor atender a necessidade envolvida em cada uma.

Dentre os princípios ágeis, aqueles que justificam sua escolha neste trabalho estão:

- Softwares funcionais entregues frequentemente
- Cooperação constante entre pessoas que entendem do “negócio” e desenvolvedores
- Design do software deve prezar pela excelência técnica
- Simplicidade
- Software funcional mais do que documentação extensa

Para conquistar o objetivo deste trabalho, resolveu-se dividi-lo em pequenas iterações seguindo princípios ágeis citado anteriormente. Cada uma envolve o desenvolvimento das funcionalidades e os testes de sistema, cuja meta é executar o mesmo sob o ponto de vista do usuário final, varrendo as funcionalidades em busca de falhas.

Como a aplicação seria desenvolvida com o auxílio de um framework novo no mercado, com uma linguagem de programação nova para o desenvolvedor, decidiu-se utilizar funcionalidades mais simples nas primeiras iterações. Seguindo esta estratégia, chegou-se a uma divisão com quatro iterações. A seguir, será apresentada cada uma delas.

4.3.1 Primeira Iteração

O Yii oferece uma ferramenta que pode ser usada para automatizar a criação de código. Uma das opções desta ferramenta é a criação de um esqueleto de uma aplicação. Este esqueleto já contém a funcionalidade de autenticação de usuários. Por isso, uma das funcionalidades incluídas nesta iteração é a gerência de acesso ao site.

Os usuários presentes no esqueleto inicial são usuários estáticos, ou seja, a aplicação não consulta nenhuma estrutura de arquivos nem um banco de dados para validar o formulário de acesso. Assim, foi possível estudar o framework ajustando esta página. Nesta etapa, já foi possível entender algumas particularidades do framework e da linguagem PHP:

- A comunicação entre o controlador e o modelo
- A amarração dos atributos do modelo com os campos de entrada de dados de um formulário
- O controle de acesso e suas regras
- Uso de variáveis de sessão

Outra opção da ferramenta de geração de código é a criação das operações básicas de cadastro, conhecidas como CRUD (do inglês, Create, Read, Update e Delete), através de um banco de dados. A ferramenta se conecta no mesmo e gera as páginas das operações, o controlador e o modelo (incluindo as relações).

Com o intuito de fechar o ciclo de autenticação de um usuário do sistema, incluiu-se nesta iteração o cadastro de usuários e, por consequência, o cadastro de perfis e situações do usuário. Estes últimos pelo simples fato de estarem relacionados com o usuário.

Uma vez gerado o código necessário para esses objetos, é necessária a personalização dos mesmos. Dentre as personalizações estão:

- Definição das permissões de acesso
- Nomenclatura dos atributos do modelo
- Integração dos formulários de acordo com as relações entre os modelos
- Inserir regras de negócio nas ações do controlador

O cadastro de perfis e de situações de usuários não possui nenhuma regra de negócio a ser adicionada, além das regras já definidas pelo modelo. O mesmo não acontece no cadastro de usuários. O sistema deve aceitar apenas o cadastro de funcionários do INF. Para que este requisito seja satisfeito, no momento do cadastro do novo usuário, é necessário informar um endereço eletrônico pertencente ao domínio do Instituto. Este endereço deverá ser validado pelo novo usuário, visitando a página de validação recebida no e-mail informado. Enquanto esta validação não for efetuada, o usuário não conseguirá se autenticar na aplicação.

Com todas estas etapas cumpridas, chegou-se a primeira versão do sistema, que engloba o cadastro de usuários que terão acesso ao mesmo e a autenticação em si. A figura abaixo ilustra o formulário de acesso.

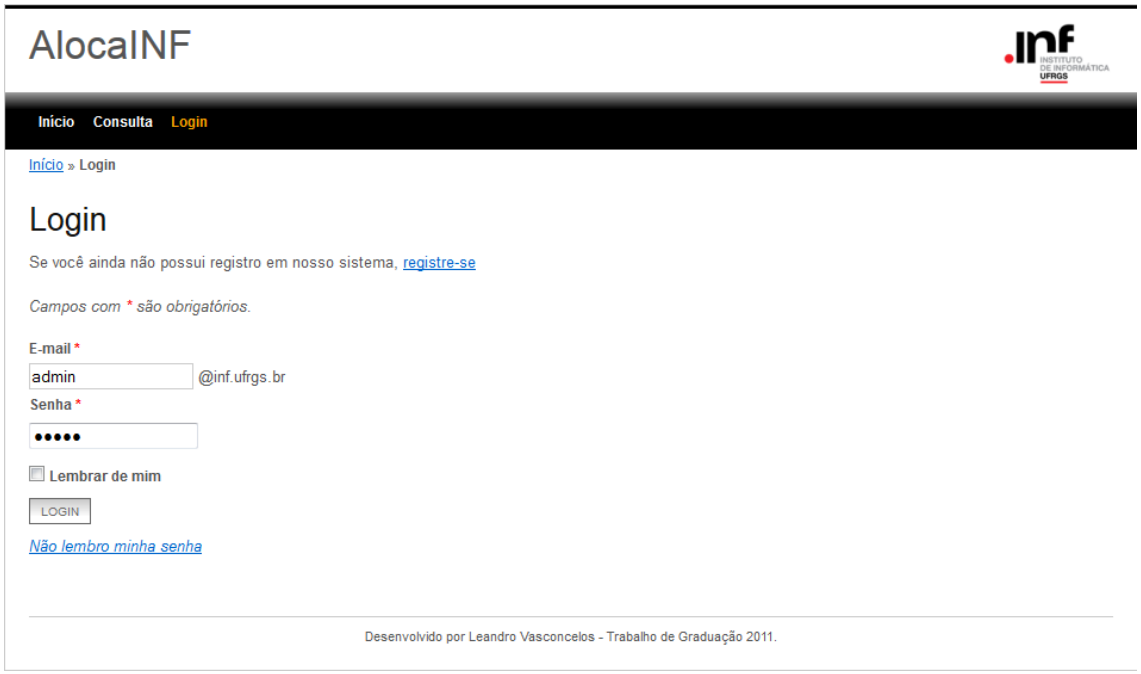


Figura 4.7: Página de acesso do sistema

4.3.2 Segunda Iteração

O próximo passo no desenvolvimento envolve a gestão dos ambientes. Eles são as peças imprescindíveis do sistema. Por causa deles que esta ideia surgiu. E por esta razão, tomou-se a decisão de englobar na segunda iteração o cadastro dos ambientes e seus relacionados.

Como apresentado em outra ocasião, um ambiente está presente em um prédio e possui uma série de recursos e usos. Por esta razão, o primeiro passo desta iteração é a gestão dos prédios do Instituto. Esta gestão envolve o formulário de criação e edição, página de visualização de um prédio, opção de exclusão e uma tela de administração, contemplando uma tabela com todos os prédios cadastrados, ilustrada na figura abaixo.

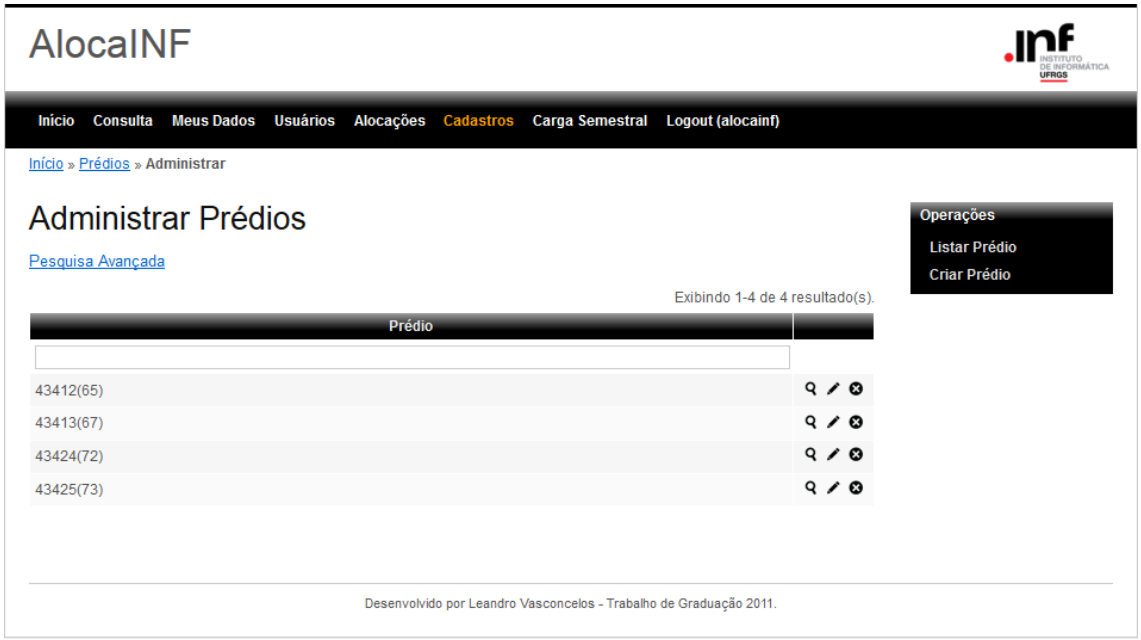


Figura 4.8: Tela de administração de prédios.

De forma análoga funciona o cadastro de recursos e o cadastro de usos, com as mesmas operações citadas no cadastro de prédios. Esta gestão de recursos e usos, de forma independente, serve para o cadastro de ambientes. Ou seja, só é possível adicionar um recurso (ou uso) a um ambiente se o mesmo estiver cadastrado como um recurso (ou uso) possível.

A última funcionalidade desta iteração é o gerenciamento de ambientes. Aqui, surgiram oportunidades para aprender um pouco mais sobre o framework de desenvolvimento, uma vez que o ambiente possui relações com outras entidades, como mostrado na seção sobre a modelagem do sistema. Como exemplo desta relação, será descrito o formulário de cadastro de um novo ambiente. A figura 4.9 ilustra esta tela.

O primeiro campo destina-se a coletar a informação do prédio ao qual o ambiente pertence. Como dito, somente prédios cadastrados no sistema podem ser utilizados em um ambiente. Logo, foi necessária uma consulta para obter os prédios disponíveis e, assim, apresentá-los ao usuário.

O mesmo acontece com a lista de recursos e usos, apresentadas de forma a possibilitar a escolha múltipla de ambos. No controlador, houve a necessidade de incluir esta lógica de negócio, uma vez que o gerador de código do Yii se restringe apenas a definir as relações.

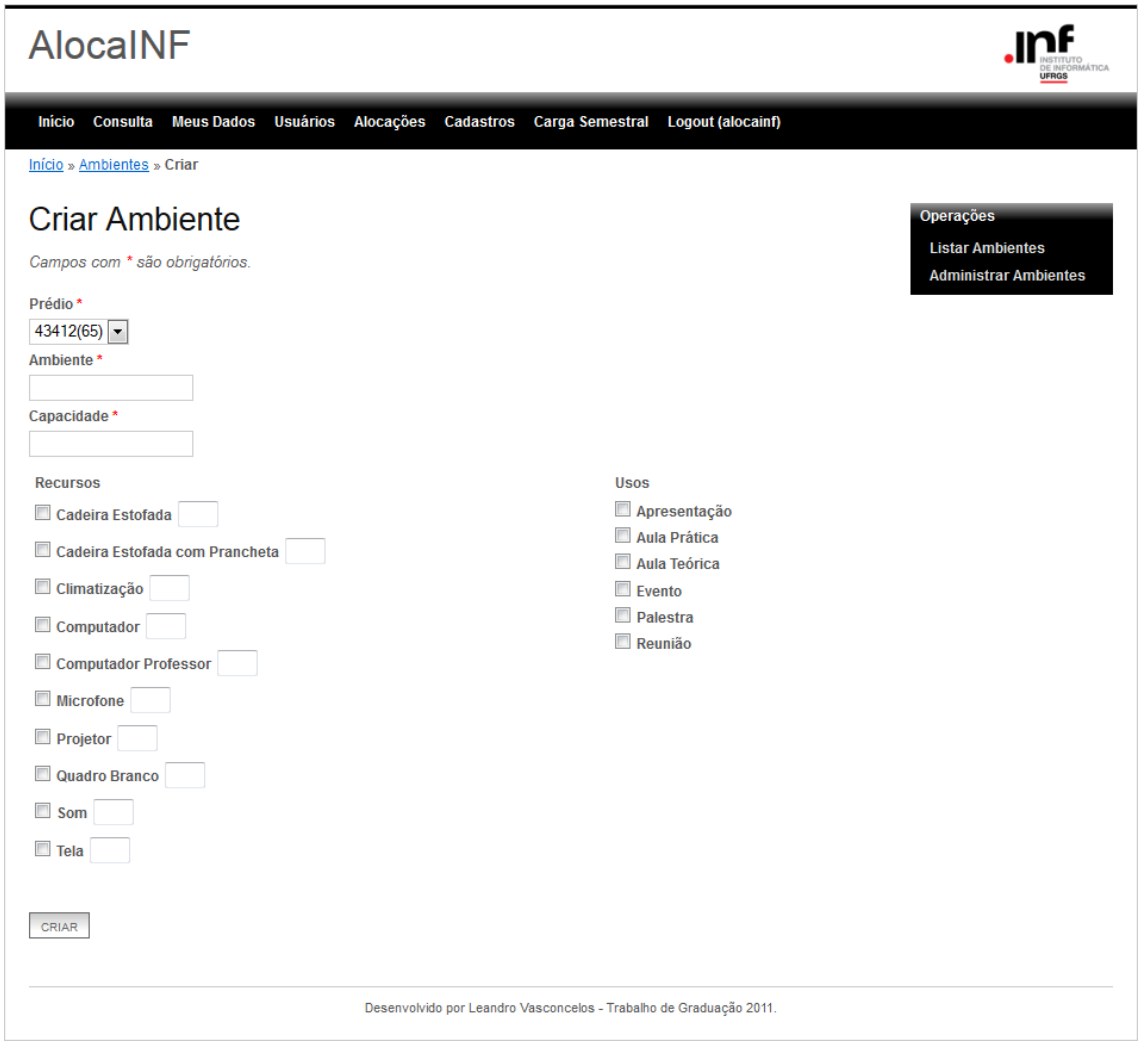


Figura 4.9: Tela de cadastro de ambientes.

Agora, o sistema possui uma nova versão, que contempla as funcionalidades de gestão de usuários e a gestão de ambientes. Logo, a aplicação possui os requisitos para a inclusão de uma nova funcionalidade: alocação de ambiente por um usuário. Esta e outras funcionalidades estão incluídas na próxima iteração.

4.3.3 Terceira Iteração

A primeira funcionalidade implementada foi a consulta de ambientes. Esta consulta está disponível para qualquer usuário do sistema, inclusive aos não cadastrados, conhecidos como visitantes. Na tela em questão é possível efetuar uma consulta avançada, informando um prédio, a capacidade do ambiente (inclusive com operadores), os recursos e usos. A figura 4.10 a seguir ilustra esta tela.

AlocaINF

Início Consulta Meus Dados Usuários Alocações Cadastros Carga Semestral Logout (alocainf)

Início » Ambientes » Administrar

Consultar Ambientes

Pesquisa Avançada

Prédio Selecione

Capacidade Ex.: >=30

Recursos

- Cadeira Estofada
- Cadeira Estofada com Prancheta
- Climatização
- Computador
- Computador Professor
- Microfone
- Projetor
- Quadro Branco
- Som
- Tela

Usos

- Apresentação
- Aula Prática
- Aula Teórica
- Evento
- Palestra
- Reunião

PESQUISAR

Exibindo 1-10 de 31 resultado(s).

Prédio	Ambiente	Capacidade	
43412(65)	214	15	
43412(65)	220	20	
43412(65)	222	15	
43412(65)	ANFA	76	
43412(65)	ANFV	76	
43413(67)	101	45	
43413(67)	102	45	
43413(67)	103	45	
43413(67)	104	45	
43413(67)	105	15	

< Anterior Ir à página: 1 2 3 4 Próximo >

Desenvolvido por Leandro Vasconcelos - Trabalho de Graduação 2011.

Figura 4.10: Tela do consulta de ambientes.

Como podemos observar pela figura acima, os ambientes são listados na tabela paginada. Esta, por sua vez, apresenta apenas o prédio, o ambiente, sua capacidade e duas imagens representando ações. A primeira imagem, representada por uma lupa, permite visualizar em um pequeno frame o detalhamento completo do ambiente. A figura 4.11 ilustra este frame. A segunda imagem, representada por um calendário, direciona o usuário para o mapa semanal de ocupação do ambiente, funcionalidade detalhada a seguir.

Figura 4.11: Tela popup de detalhamento do ambiente.

No mapa semanal de alocação, o usuário é capaz de visualizar todas as reservas e, consequentemente, todos os horários livres do ambiente em questão. Esta tela permite visualizar o detalhamento completo do ambiente, tal qual explicado quando foi apresentada a consulta de ambientes. O usuário pode navegar entre as semanas do ano através das opções “Retroceder Semana” e “Avançar Semana”. A figura abaixo ilustra a tela descrita.

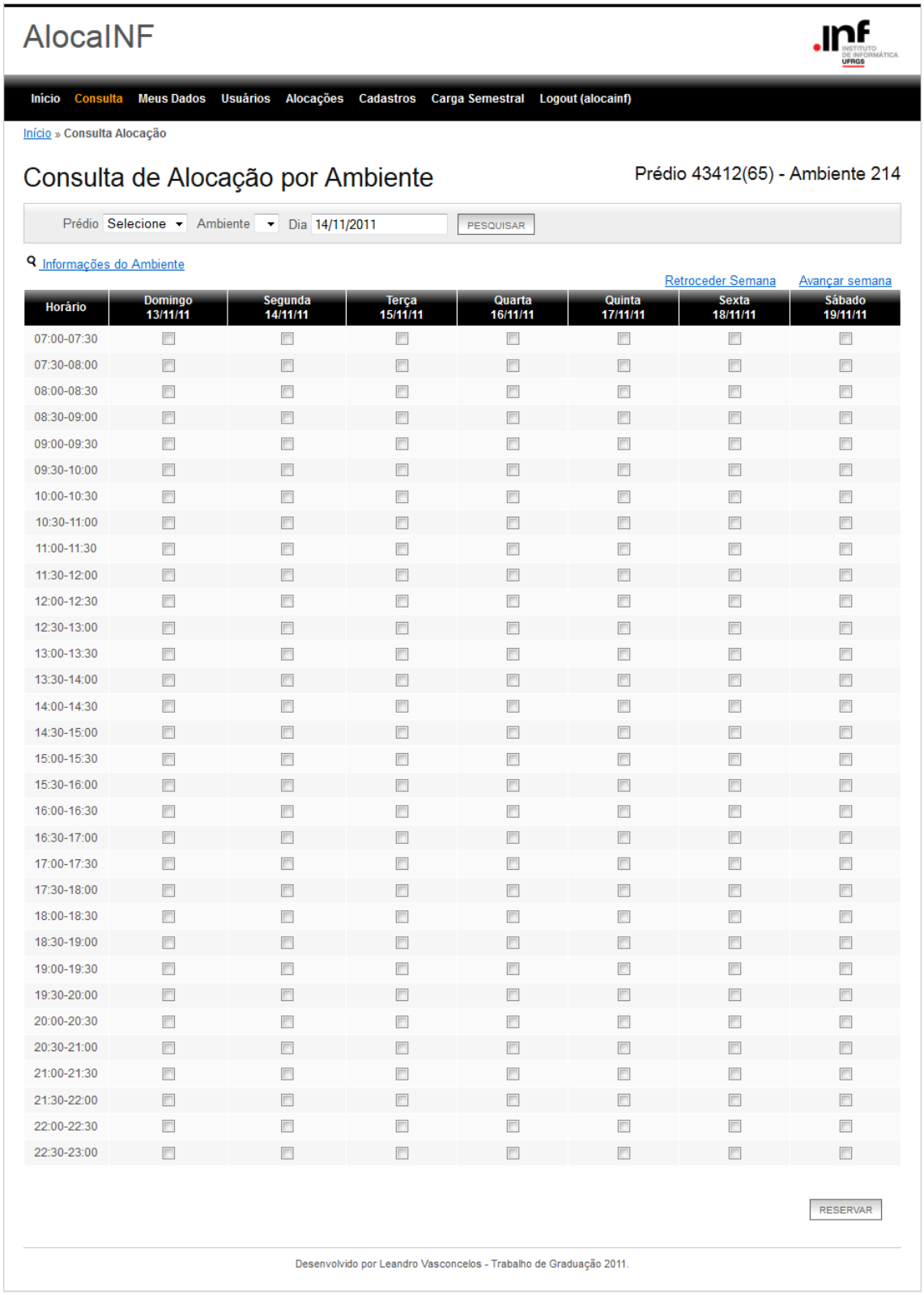


Figura 4.12: Tela de visualização semanal da ocupação do ambiente.

Esta tela permite ainda, para usuários autenticados, a possibilidade de escolher, dentre os horários vagos, aqueles que lhe interessa e reservar. Caso se deseja efetuar uma reserva, o usuário será direcionado a outra tela para preencher os demais dados necessários para confirmar a reserva em seu nome. A figura 4.12 ilustra esta tela.

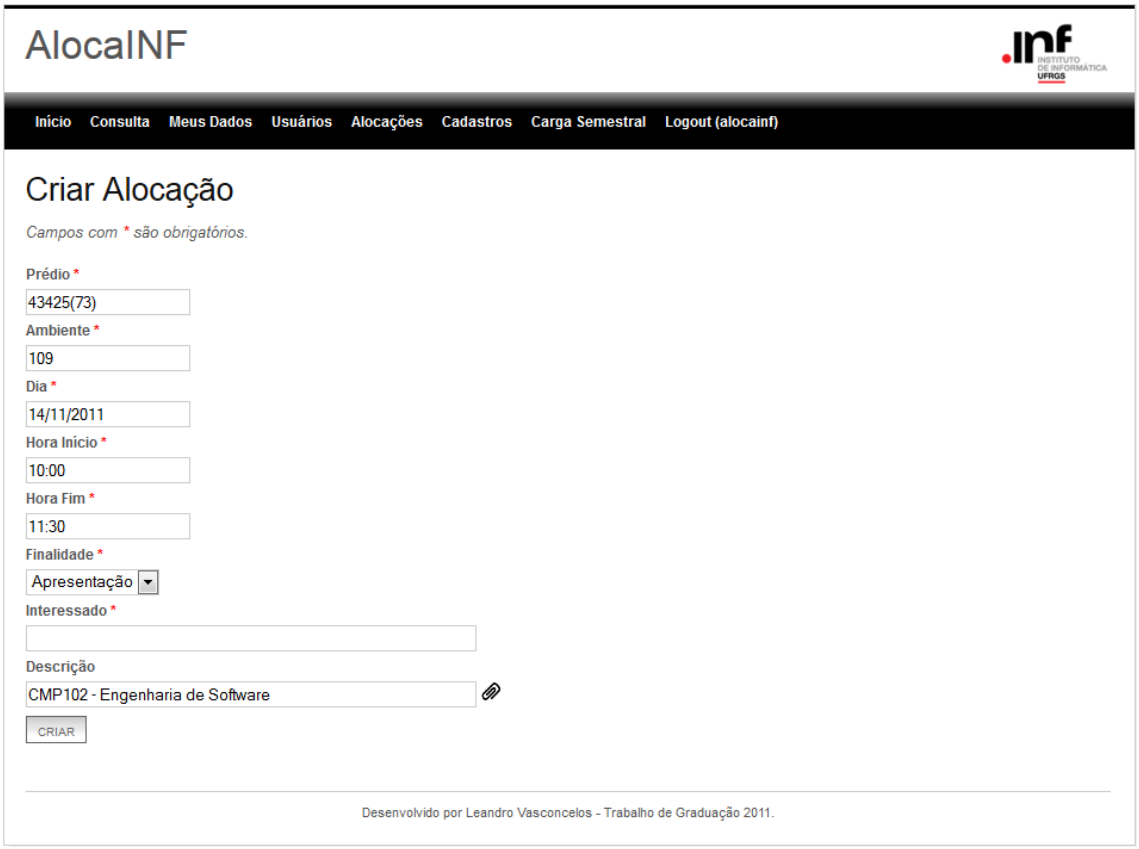


Figura 4.13: Tela de confirmação de reserva.

No segundo passo da reserva, é possível adicionar como descrição uma disciplina. Esta funcionalidade foi considerada, uma vez que alguns professores necessitam reservar ambientes esporadicamente. Seja para uma aula especial em laboratório, ou uma prova em um local mais espaçoso, etc.. Para manter a consistência, adicionou-se uma tela *popup* com a lista das disciplinas cadastradas no sistema. Na figura 4.12 é possível observar esta tela, onde o usuário escolhe a disciplina que deseja anexar na reserva. Para tal funcionalidade, foi necessário implementar o cadastro das disciplinas, que envolve as operações básicas de cadastro. Como não difere em nada com outros cadastros simples do sistema, não será detalhada aqui.

Figura 4.14: Tela popup de disciplinas.

Para que esta versão esteja de acordo com o planejado, foi implementada a página inicial da aplicação. Esta página é estática, ou seja, serve apenas para apresentar informações ao usuário contendo, no máximo, links para redirecionar a outras páginas do sistema . Apenas algumas informações ao usuário, como mostra a figura abaixo.

Figura 4.15: Tela inicial do sistema.

4.3.4 Quarta Iteração

Até aqui, o sistema permite gerir usuários, ambientes e reservas, além de permitir consultas avançadas sobre ambientes e suas alocações. Agora, será descrita a solução para o requisito mais complexo do sistema: a carga semestral.

Justamente por ser considerado mais difícil, foi implementado na ultima iteração deste desenvolvimento. Isto porque envolve uma lógica de negócio muito complexa, com consultas avançadas ao banco de dados e tratamentos minuciosos nos dados manipulados.

O primeiro passo para a carga semestral é o cadastro das disciplinas e turmas. Para cada turma adicionada é possível especificar:

- Os recursos necessários no ambiente,
- A capacidade do ambiente,
- Os dias da semana que haverá aula,
- O horário de início e fim das aulas e
- O professor que ministrará as aulas.

Todos estes dados são salvos no banco como um modelo. Além destes dados, os modelos possuem:

- Ano
- Semestre
- Data de inicio do semestre letivo
- Data de fim do semestre letivo
- Nome para uso do usuário

A figura abaixo ilustra esta tela. Nota-se que cada turma adicionada é armazenada em uma tabela. Cada linha representa uma turma e é permitida a edição e exclusão da mesma, ações simbolizadas pelas imagens na última coluna da tabela.

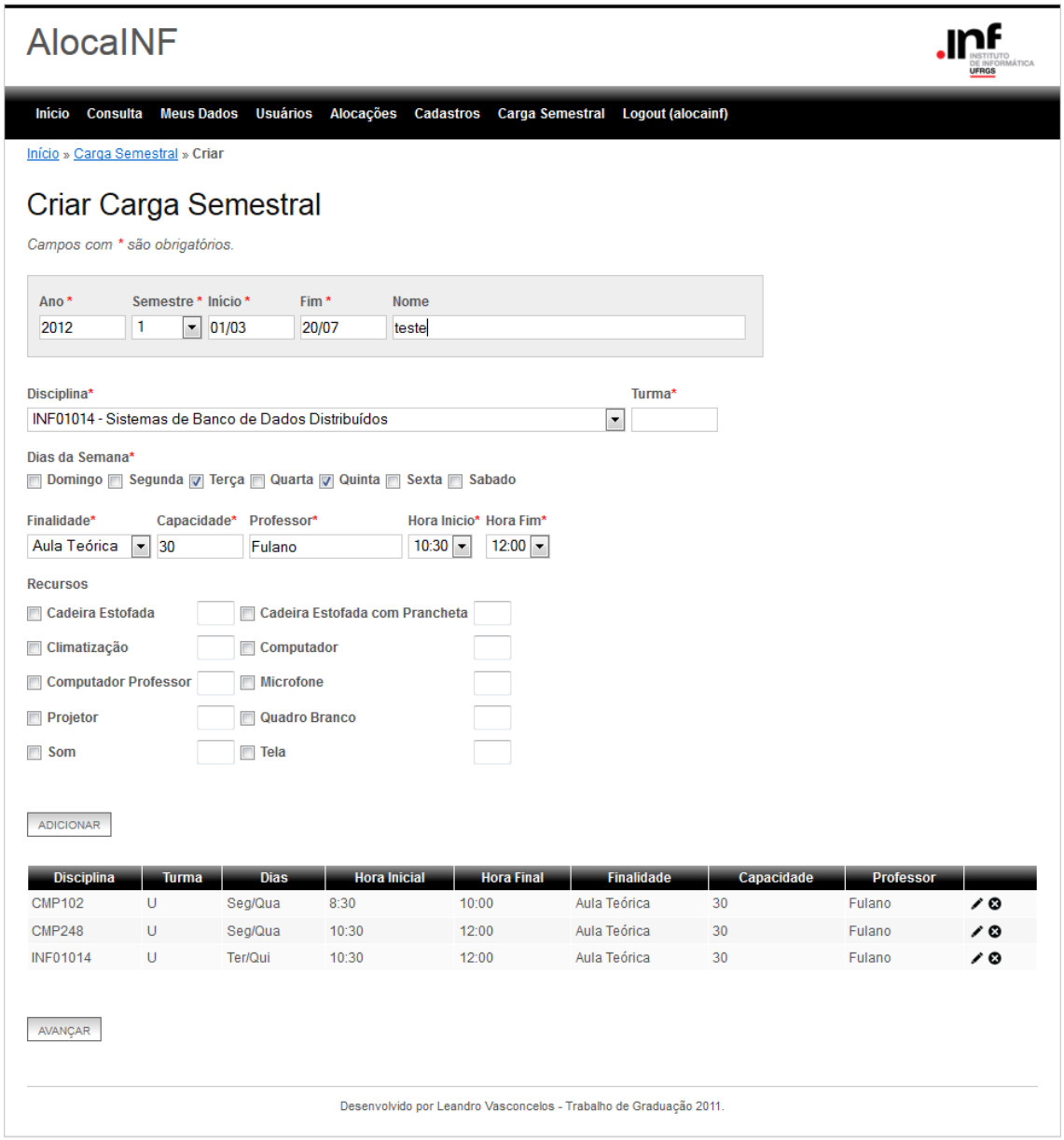


Figura 4.16: Tela do primeiro passo da carga semestral.

Com todas as informações inseridas, o próximo passo é tentar alocar as disciplinas nos ambientes existentes, respeitando os requisitos informados.

Muitos trabalhos foram e ainda são realizados propondo um algoritmo capaz de realizar esta alocação. Este problema é conhecido como Problema de Alocação de Salas (PAS) e os trabalhos realizados utilizam métodos heurísticos para resolvê-lo, uma vez que este problema é classificado como NP-Difícil. Os métodos heurísticos procuram por soluções em tempo viável para problemas onde os algoritmos exatos não terminam em tempo hábil, embora não garantam encontrar uma solução ótima.

Em OLIVEIRA (2006) foi proposto o uso de Algoritmos Genéticos e Recozimento Simulado para o PAS. O objetivo deste trabalho está na comparação destas duas heurísticas para o problema, buscando otimizar a distância percorrida pelos alunos para se deslocarem de uma sala para a próxima.

Em SOUZA (2002) foi proposto a combinação de duas heurísticas para a resolução do problema de alocação: Recozimento Simulado e Busca Tabu. Além dos requisitos básicos deste processo de alocação (não alocar duas aulas na mesma sala e no mesmo horário, por exemplo), este trabalho visa alocar em uma mesma sala alunos de um mesmo curso e período; otimizar a capacidade da sala de acordo com o número de alunos presentes; e deixar pelo menos uma sala vazia em pelo menos um horário ao longo do dia para limpeza.

Os dois trabalhos citados acima visam alguma otimização que, atualmente, não se aplicam ao problema encontrado neste trabalho. Não é um requisito do sistema a otimização da distância percorrida pelos alunos para o deslocamento de uma sala a outra, como foi proposto em OLIVEIRA (2006). Além disso, a alocação de alunos de um mesmo curso e período em uma mesma sala, bem como a necessidade de haver uma sala vazia em um horário para limpeza também não se aplicam ao problema encontrado neste trabalho, como proposto em SOUZA (2002). Por estas razões, decidiu-se implementar um novo algoritmo capaz de realizar a alocação das disciplinas.

O algoritmo construído utiliza a recursividade para resolver este problema, varrendo todas as turmas e alocando-as nos ambientes possíveis. A entrada do algoritmo é uma lista das turmas cadastradas, onde cada turma possui uma lista de ambientes possíveis. A lista de turmas é ordenada de forma que as turmas com menos ambientes possíveis fiquem nas primeiras posições.

hora \ sala	1	2	3	4	5
1				INF01002-A	
2		INF01001-U		INF01002-A	
3		INF01001-U			
4		INF01001-U		INF01004-A	
5				INF01004-A	

Figura 4.17: Vetor auxiliar do algoritmo de alocação

O algoritmo possui ainda um vetor bidimensional, onde as linhas são os horários possíveis e as colunas são os ambientes possíveis. A figura 4.17 ilustra o vetor auxiliar. Nesta figura, é possível observar que no ambiente 2 e nos horários 2, 3 e 4 está alocada a turma U da disciplina INF01001. Assim como a turma A da disciplina INF01002 e da disciplina INF01004, alocadas na sala 4 nos horários 1, 2 e 4, 5, respectivamente. Nas demais posições da tabela não há turmas alocadas. A colisão ocorre quando se tenta alocar uma turma em um horário e um ambiente já em uso. Ou seja, se nas entradas deste vetor correspondente aos horários e ao ambiente já houver uma turma alocada.

Durante a execução, o algoritmo percorre a lista das turmas tentando alocar a mesma nos ambientes previamente selecionados segundo os critérios adicionados (recursos disponíveis no ambiente, capacidade e finalidade do mesmo). Caso seja encontrado um ambiente disponível no horário da turma, o algoritmo segue para a próxima turma da lista. Caso não seja possível alocar a turma em nenhum dos ambientes disponíveis, o

algoritmo retorna (recursivamente) para as turmas anteriores para tentar aloca-las em um ambiente diferente, retomando a sequencia de turmas da lista. A figura 4.18 ilustra o pseudocódigo do algoritmo de alocação.

```
procedimento Alocar(index)
   turma ← listaTurmas[index];
   j ← 0; //indice da lista de ambientes disponíveis da turma
   alocou ← falso; //indica se alocou a turma em um ambiente
   enquanto (j < tamanho(turma.listaAmbientes) E alocou = falso)
      //verifica no vetor auxiliar se ambiente está ocupado
      ocupado ← AmbienteAlocado(turma.id, turma.listaAmbiente[j], turma.horario);
      se (ocupado = falso)
         turma.ambienteSelecionado ← turma.listaAmbientes[j];
         alocou ← verdadeiro E Alocar(index + 1);
         se (alocou = falso)
            //retira o vetor auxiliar a turma corrente
            RetirarDoVetorAuxiliar(turma.id, turma.listaAmbiente[j], turma.horario);
      senao
         alocou ← falso;
      j ← j + 1;
   retorna alocou;
```

Figura 4.18: Pseudocódigo do algoritmo de alocação

A execução do algoritmo termina com sucesso quando todas as turmas estão alocadas. Se não for possível alocar as turmas depois de percorrer todas as possibilidades, o mesmo termina sinalizando o fracasso na alocação.

Para finalizar a implementação do modelo, incluiu-se uma tela de administração. Nesta tela é possível visualizar os modelos criados e todas as suas turmas. Além de visualizar, o usuário pode utilizar um modelo para uma nova carga semestral.

4.4 Utilização do sistema

Nesta seção serão apresentados dois cenários presentes no gerenciamento dos ambientes. Primeiramente é apresentado um passo a passo da alocação semestral das disciplinas ministradas no INF. Em seguida, são apresentadas as etapas envolvidas na reserva de um ambiente disponível em um horário requerido. Em ambos os casos, a descrição contará com o auxílio de imagens obtidas das telas do sistema para cada passo citado.

4.4.1 Carga Semestral

O primeiro passo para criar uma carga semestral é clicar no item de menu “Carga Semestral”. Neste instante, o usuário será redirecionado a pagina de administração de modelos de cargas semestrais. Esta tela apresenta os diversos modelos já criados, uma vez que o sistema permite utilizar um modelo anterior para uma nova carga. A tabela possui dois ícones em sua última coluna, destacados no detalhe A da figura 4.19. Cada ícone refere-se a uma ação relacionada com o modelo da linha em questão. O primeiro ícone, representado por uma lupa, direciona o usuário para uma página de visualização completa do modelo. O segundo, representado por um “x”, remove o modelo do sistema. Aqui, vamos criar um novo modelo. Para isto, o usuário deve clicar no item “Criar Modelo” no menu lateral destacado no detalhe B da figura 4.19.

Figura 4.19: Tela de administração de modelos

A tela de criação da carga semestral está ilustrada na figura 4.16 da seção 4.3.4. Para melhor ilustração deste passo a passo, serão apresentados os pedaços desta tela.

Figura 4.20: Tela do primeiro passo da carga semestral – Parte 1.

Esta tela possui um formulário para que o usuário informe o ano, o semestre, o dia inicial e final das aulas, além do nome que será dado ao modelo. Este formulário está ilustrado na figura 4.20.

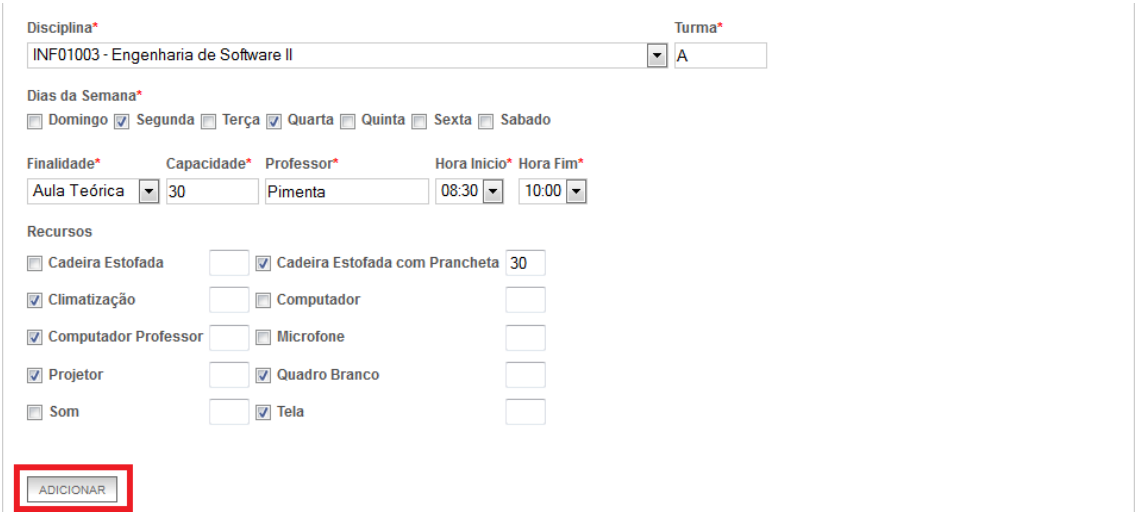


Figura 4.21: Tela do primeiro passo da carga semestral – Parte 2.

Abaixo deste formulário, encontram-se os dados para a inserção das disciplinas que irão fazer parte deste modelo, conforme ilustrado na figura 4.21 acima. O usuário insere as disciplinas, suas turmas, os dias da semana que haverá aulas, a finalidade da alocação (aula teórica, prática, etc.), a capacidade da turma, o professor, os horários de início e fim e os recursos que devem estar presentes no ambiente a ser escolhido. Para cada disciplina inserida, o usuário clica no botão "Adicionar" destacado na figura 4.21. Isto feito, a disciplina será armazenada na tabela ilustrada na figura 4.22 abaixo, podendo ser editada ou removida.

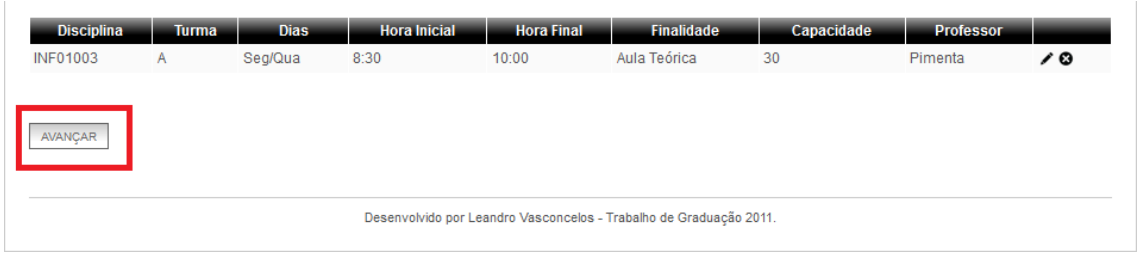


Figura 4.22: Tela do primeiro passo da carga semestral – Parte 3.

Após a inserção de todas as disciplinas, o usuário deve avançar para o próximo passo clicando no botão "Avançar" destacado na figura 4.22. Neste momento, o sistema tentará encontrar ambientes para todas as disciplinas inseridas. Caso não tenha sucesso nesta procura, uma mensagem de erro é apresentada ao usuário no topo deste primeiro passo, como ilustra a figura 4.23. Caso contrário, o usuário é direcionado para uma tela de resumo descrita a seguir.

Figura 4.23: Tela do primeiro passo da carga semestral – mensagem de erro.

Se o sistema conseguiu encontrar pelo menos um ambiente para cada disciplina (atendendo os requisitos desta) e arranjar tais ambientes para alocar todas elas, o usuário é redirecionado para uma nova página onde estão resumidas as informações do primeiro passo. A tela possui os dados do modelo e uma tabela similar a tabela do primeiro passo. A diferença entre elas está na última coluna. No resumo, a última coluna mostra o ambiente selecionado para a cada disciplina. A figura 4.24 ilustra esta tela.

Neste momento o sistema informa ao usuário que todas as disciplinas inseridas puderam ser distribuídas nos ambientes do INF. Porém, ainda não foram reservados os horários nestes ambientes. Isto será feito quando o usuário clicar no botão "Finalizar", destacado no detalhe B da figura 4.24. Caso não queira finalizar a carga, o usuário pode voltar para o passo anterior clicando no botão "Voltar", destacado no detalhe B da mesma figura.

Para finalizar com este passo a passo, ao clicar no botão "Finalizar", o sistema tentará reservar os ambientes para as disciplinas presentes neste modelo. Caso encontre uma colisão, uma mensagem de erro é apresentada contendo os detalhes da reserva

encontrada (ver figura 4.25). Se nenhuma colisão for encontrada, uma mensagem de sucesso é apresentada ao usuário (ver figura 4.26).

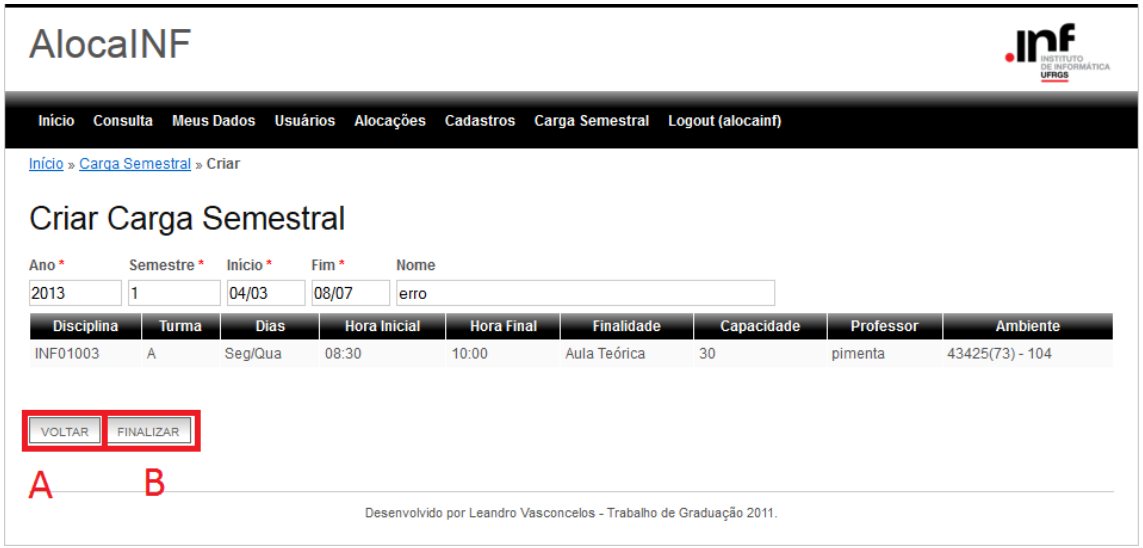


Figura 4.24: Tela do segundo passo da carga semestral.

Figura 4.25: Tela do segundo passo da carga semestral – Mensagem de erro.

Figura 4.26: Tela do segundo passo da carga semestral – Mensagem de sucesso.

4.4.2 Alocação Sob Demanda

Este cenário visa ilustrar o passo a passo executado pelo usuário para que este consiga reservar um ambiente para um determinado fim. Visando explorar as consultas disponíveis no sistema, este cenário se inicia com a consulta por um ambiente que atenda alguns requisitos do usuário.

Figura 4.27: Tela do primeiro passo da alocação sob demanda – Menu "Ambientes"

Para pesquisar os ambientes, o usuário deve clicar no item de menu "Consulta Ambientes" destacado na figura 4.27 acima. Neste momento, o usuário é direcionado para a tela de consulta de ambientes, ilustrada na figura 4.10 da seção 4.3.3. Esta tela apresenta os ambientes do INF em uma tabela na parte inferior da tela. Acima desta há um formulário para filtrar os ambientes de acordo com os parâmetros escolhidos (prédio, capacidade, recursos e usos). Após escolhidos os parâmetros para filtrar os dados, o usuário clica no botão "Pesquisar" no destaque da figura 4.28 (novamente são ilustradas partes da tela para facilitar esta leitura).

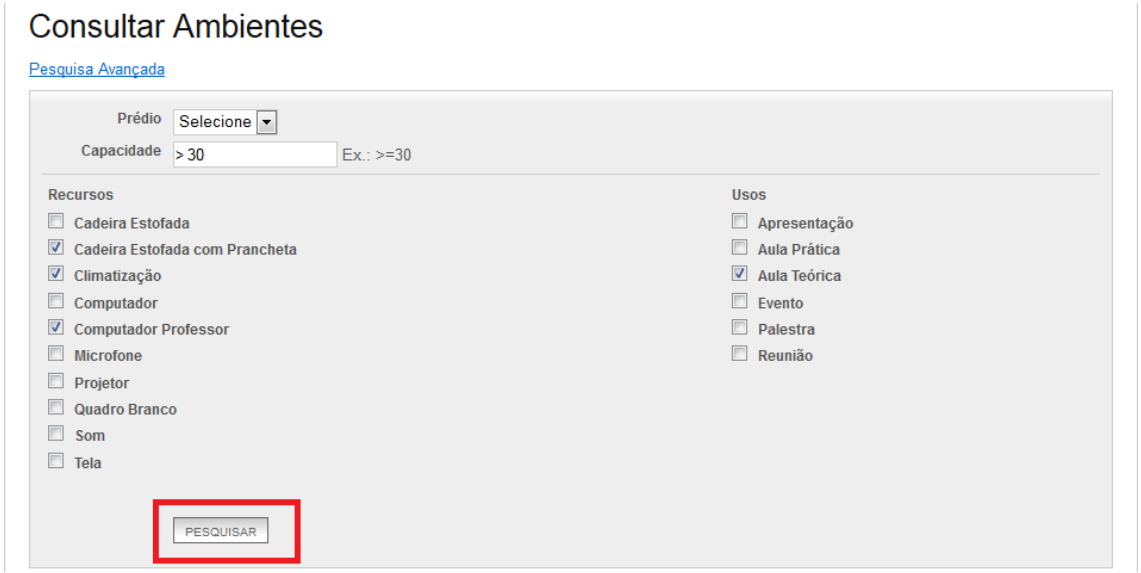


Figura 4.28: Tela do segundo passo da alocação sob demanda – Pesquisa.

Na figura acima está ilustrada uma pesquisa por ambientes com capacidade maior que trinta ocupantes, cadeira estofada com prancheta, climatização, computador do professor e disponível para aulas teóricas. O resultado desta pesquisa está ilustrado na figura 4.29 a seguir. Na tabela utilizada para apresentação dos ambientes há uma coluna com dois ícones (destacados na figura 4.29): o primeiro, representado por uma lupa abre uma *popup* com todas as informações do ambiente (figura 4.11 da seção 4.3.3); o

segundo, representado por um calendário, direciona o usuário para o mapa de ocupação deste ambiente, ilustrado na figura 4.30.

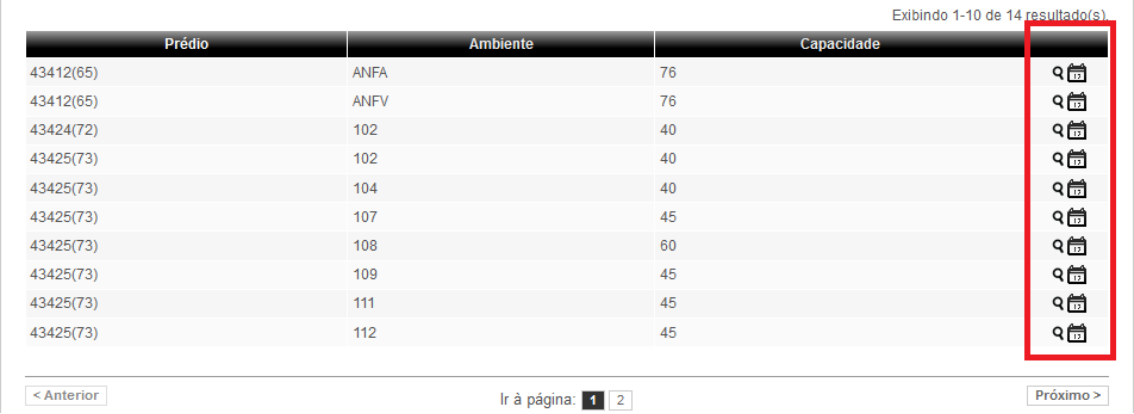


Figura 4.29: Tela do segundo passo da alocação sob demanda – Resultado.

Na tela de ocupação do ambiente, o usuário pode avançar e retroceder as semanas através dos links no canto superior direito da tabela (destaque A da figura 4.30). Na tabela de horários desta tela, um horário vago é representado por um *checkbox* e um horário ocupado é representado pelo nome do responsável pela reserva. Para disponibilizar mais informações aos usuários, uma *tooltip* é apresentada com outras informações da reserva, conforme detalhe B da mesma figura.

Para efetuar uma reserva, o usuário escolhe dentre os horários vagos selecionando as células referentes aos horários que deseja. Estes horários devem ser consecutivos em relação ao dia escolhido, como está destacado no detalhe C da figura 4.30. Isto feito, o usuário clica no botão “Reservar” no canto inferior da tabela, conforme destaque D da figura. Assim, o usuário é direcionado a uma página onde são requeridos os demais dados necessários para finalizar a alocação. Tal tela está ilustrada na figura 4.31.

Para finalizar a reserva, o usuário deve informar a finalidade da ocupação, o interessado em ocupar o ambiente (campo disponível para usuários administradores, únicos com permissão para efetuar reservas em nome de outros) e uma descrição para uso próprio. Nesta descrição, é possível anexar o código de uma disciplina. Para isto, o usuário deve clicar na imagem representada por um clipe (detalhe A da figura 4.31). Uma tela *popup* será apresentada com as disciplinas cadastradas (ilustrada na figura 4.14 da seção 4.3.2). Ao escolher uma, automaticamente o campo “Descrição” é preenchido com o código e o nome da disciplina. Com todos os campos preenchidos o usuário clica no botão “Criar” destacado do detalhe B da figura 4.31. O sistema efetua a reserva e direciona o usuário a uma página com a apresentação dos dados da reserva, como ilustra a figura 4.32.

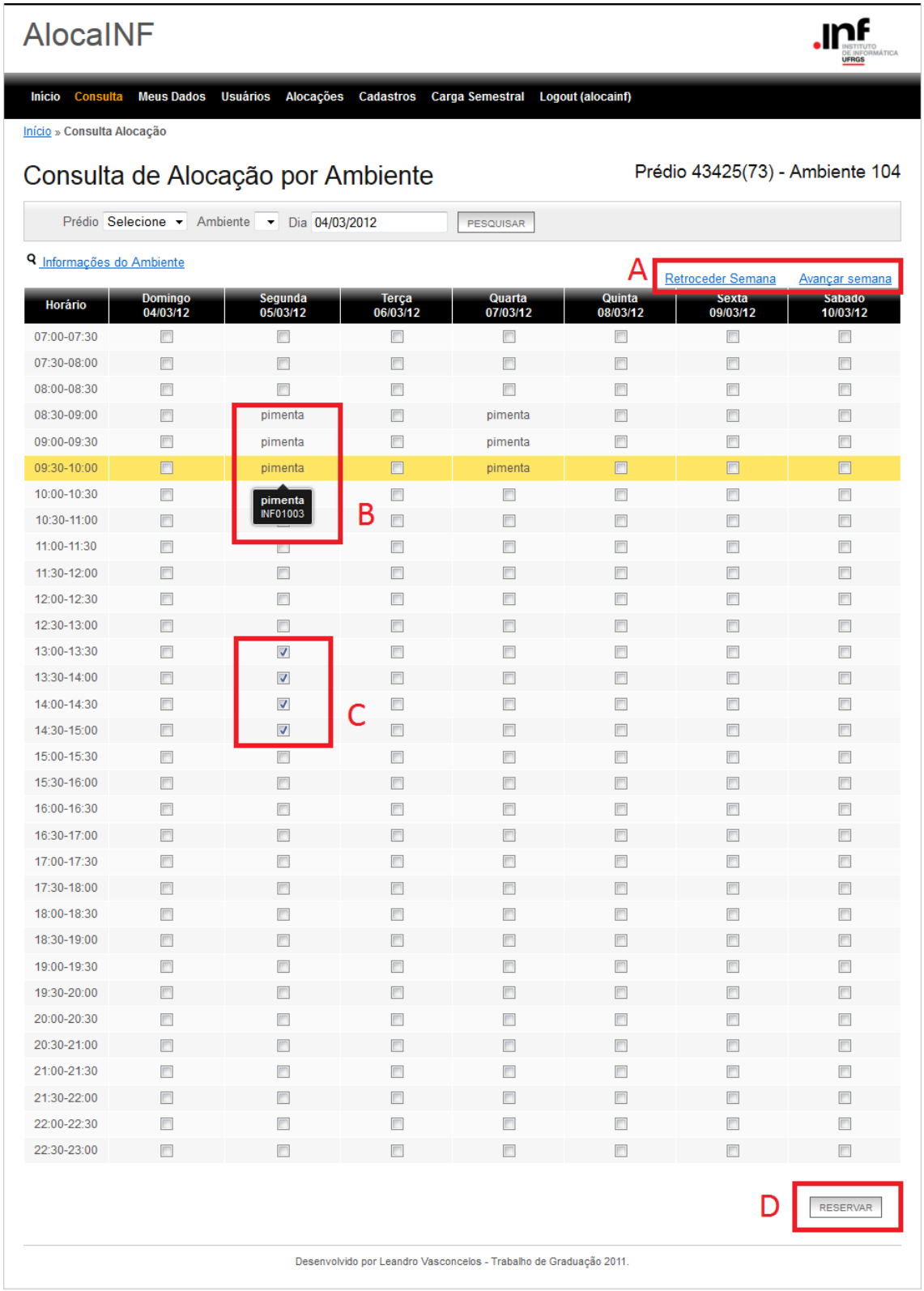


Figura 4.30: Tela do terceiro passo da alocação sob demanda.

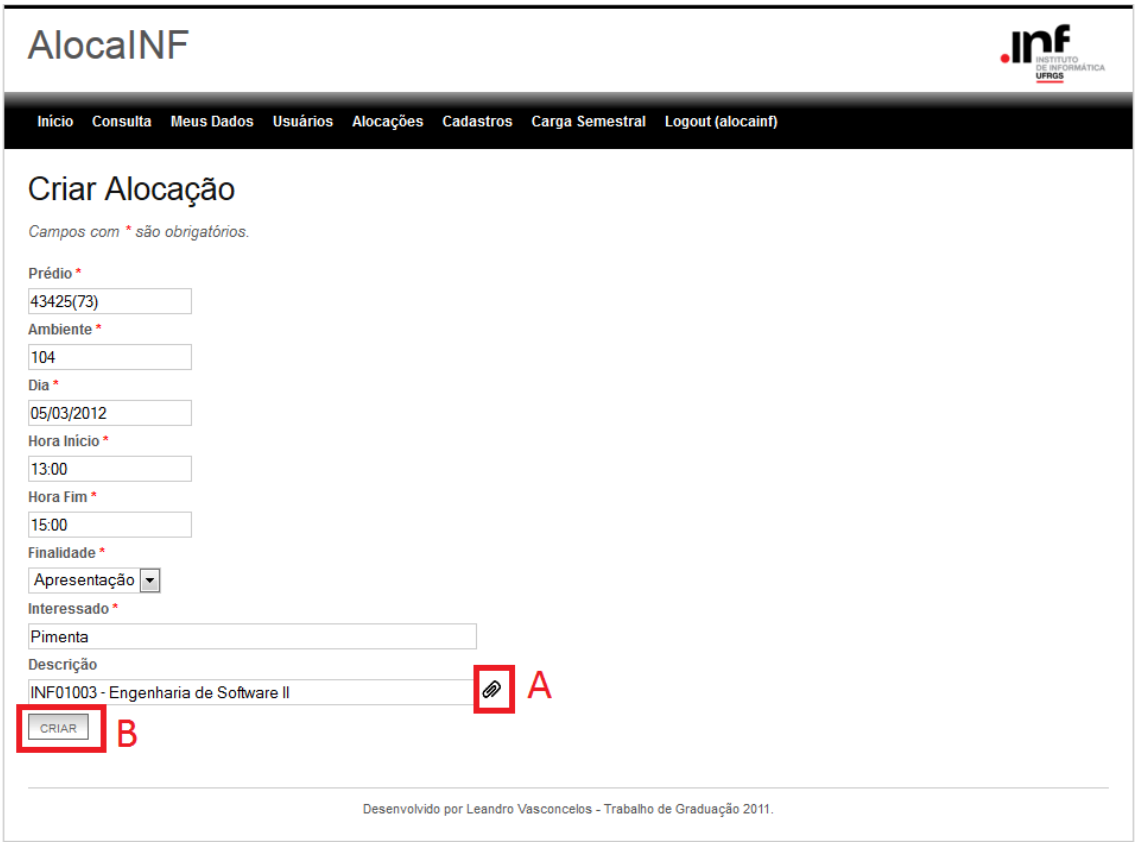


Figura 4.31: Tela do quarto passo da alocação sob demanda.

Figura 4.32: Tela do quinto passo da alocação sob demanda.

Neste capítulo procurou-se descrever dois cenários comuns no gerenciamento de ambientes. Com o auxílio de figuras, primeiramente efetuou-se uma carga semestral e, posteriormente, uma reserva sob demanda, comum no decorrer do semestre letivo. O próximo capítulo apresenta as conclusões deste trabalho.

4.5 Validação

Ao final do desenvolvimento do sistema, marcou-se uma reunião com os interessados para a apresentação do mesmo. Esta reunião tem como objetivo validar o trabalho realizado, garantindo que o sistema atenda os requisitos levantados.

Em um primeiro momento foram apresentadas todas as funcionalidades contidas no sistema. Para cada uma destas funcionalidades, os interessados puderam tirar suas dúvidas referentes a execução das mesmas, simular tarefas, e observar o resultado obtido.

Um ponto positivo destacado foi a possibilidade de se criar uma nova reserva a partia da visão semanal do ambiente. Outro ponto que mereceu destaque foi a flexibilização das informações contidas nos ambientes, uma vez que estes podem sofrer alterações de recursos disponíveis e finalidades. Além disso, ressaltou-se a possibilidade de visualização detalhada dos ambientes através de telas *popup*, disponíveis em algumas telas do sistema.

O único ponto negativo refere-se a pesquisa por ambientes disponíveis em um dia e horário pré-definidos. Nas primeiras etapas deste trabalho, esta funcionalidade não foi levantada nem pelos interessados e nem pelo autor, ficando esquecida. No momento desta reunião, já havia um sistema (IAlocador, citado anteriormente) que possuía esta funcionalidade. Por esta razão, esta deficiência surgiu na reunião. Por não fazer parte da especificação deste trabalho, os interessados pediram apenas que esta funcionalidade seja tratada como uma melhoria.

Do ponto de vista dos usuários interessados neste trabalho, o objetivo deste foi atingido com êxito. O sistema permite, de forma confiável, gerenciar os ambientes e as alocações dos mesmos.

5 CONCLUSÃO

Neste trabalho, visou-se aprimorar os processos de gerência dos ambientes do Instituto, disponibilizando uma ferramenta capaz de auxiliar tais processos, mantendo a qualidade encontrada nos dias atuais.

Em um primeiro momento procurou-se investigar e entender os processos envolvidos nesta gestão e suas deficiências. Em seguida, foram obtidos os requisitos para melhorar tais atividades. De posse destes, os mesmo foram analisados e modelados para futuramente serem inseridos em uma aplicação cujo objetivo é auxiliar a gestão realizada.

Todos os passos foram cumpridos buscando manter ou aprimorar a eficiência e a eficácia destes processos. O resultado foi um sistema capaz de minimizar o tempo dispendido para a realização dos mesmos, minimizando os erros e maximizando a qualidade do serviço prestado pela Secretaria do Instituto de Informática.

O sistema, por ser voltado à plataforma web, permite que os usuários realizem reservas de salas sem sair de seus gabinetes, um exemplo da minimização do tempo. O algoritmo de distribuição das turmas oferecidas também contribui como exemplo de tal minimização, sendo ainda exemplo da mitigação de erros mencionado anteriormente. As consultas de ambientes e suas reservas, aberta para qualquer visitante do sistema, demonstra uma melhoria agregada ao sistema. Por fim, a união de todas as funcionalidades incluídas neste projeto eleva a qualidade supracitada.

5.1 Avaliação das ferramentas utilizadas

Neste trabalho foi possível utilizar algumas ferramentas disponíveis para analistas e desenvolvedores, com o intuito de facilitar a análise e o desenvolvimento de sistemas. Algumas delas utilizadas pelo autor em outras oportunidades. Outras, sendo utilizadas pela primeira vez neste trabalho. Aqui, serão citadas estas últimas, com a finalidade de apresentar a experiência obtida com os seus usos.

O desenvolvimento deste sistema precisou ser codificado na linguagem de programação PHP. Atualmente, o PHP é muito utilizado no desenvolvimento de aplicações voltadas a plataforma web. Este uso está aumentando a medida que a mesma evolui com padrões de desenvolvimento em voga nos dias atuais, como a orientação ao objeto. De forma sucinta, pode-se dizer que a linguagem possui uma sintaxe simples e muito bem adaptada para desenvolvimento seguindo o padrão MVC. Por ser muito utilizada pela comunidade em geral, possui uma documentação completa e disponível em muitos sítios da internet, além de uma bibliografia vasta.

Ainda para o desenvolvimento utilizou-se um framework relativamente novo, denominado YiiFramework. Este framework, voltado a linguagem PHP e ao desenvolvimento seguindo o padrão MVC, fornece ao desenvolvedor muitas funcionalidades prontas para o uso, diminuído o tempo de codificação de sistemas. Ela possui um gerador de código capaz de, através de um banco de dados relacional, gerar as classes e as telas das operações básicas de cadastro (CRUD). Apesar de ser uma ferramenta nova, possui uma documentação muito boa e está sendo muito utilizada pela comunidade de desenvolvedores. Este fato aumenta o número de fóruns e de colaboradores, que disponibilizam novas soluções para usos futuros. Por ter sido o primeiro framework em uma linguagem de programação desconhecida, o autor considera uma ferramenta simples de entender, sendo capaz de começar a utilizá-la após algumas horas de estudo. Vale lembra que é muito importante que o padrão MVC deve ser previamente conhecido para que as ideias sejam assimiladas mais facilmente.

5.2 Trabalhos Futuros

Apesar de ter sido aprovado pelos interessados, o sistema precisa ser melhorado. Algumas ideias serão apresentadas aqui como trabalhos a serem realizados futuramente.

Primeiramente se faz necessário o teste do algoritmo de alocação. Por ter sido utilizado um algoritmo novo para resolver este problema, o mesmo precisa ser testado para garantir que a solução encontrada seja válida. Uma avaliação da sua complexidade também é recomendada como trabalho futuro, bem como a comparação com outros algoritmos existentes que se propõe a resolver o mesmo problema.

Com relação ao sistema, algumas melhorias podem ser incluídas para melhorar a usabilidade do mesmo:

- No mapa de ocupação de ambientes, permitir que reservas do usuário autenticado possam ser removidas com um simples clicar do mouse sobre a alocação ou alterar o horário para um novo disponível na mesma visão semanal
- No mapa de ocupação de ambientes, permitir que usuários autenticados selecionem mais de um bloco de horário para reserva, permitindo finalidades diferentes para cada um dos selecionados
- Permitir que um usuário autenticado solicite um horário reservado por outro usuário. O sistema envia um e-mail para o usuário que possui a reserva sugerindo outros ambientes disponíveis no mesmo horário e com os mesmos recursos do ambiente previamente reservado
- Incluir a funcionalidade de agendamento de horário entre os usuários cadastrados. Esta funcionalidade permitiria que os usuários selecionassem dias e horários que possuem disponibilidade para um evento. Tendo um horário disponível por todos os participantes, o sistema procura ambientes livres nos horários selecionados. esta funcionalidade pode ser melhor vista no sistema Doodle[10]
- Integração do sistema com outros sistemas do INF, permitindo que a autenticação seja realizada utilizando as mesmas credenciais do Instituto

[10] http://doodle.com/ acessado em dezembro de 2011.

- Incluir formulário para manutenção dos ambientes, onde o alocador possa informar sobre defeitos e/ou manutenções em recursos disponíveis. Atualmente este trabalho é realizado através de formulário impresso disponível em todos os ambientes do Instituto
- Incluir visualização panorâmica dos ambientes

Ainda há muito trabalho a ser realizado para melhorar a administração do Instituto de Informática da UFRGS, mas algumas soluções já estão sendo propostas. A partir de agora, é importante que o trabalho não pare e que outras soluções sejam apresentadas para melhorar esta instituição de ensino de excelente qualidade.

REFERÊNCIAS

APACHE FRIENDS. **Apache Friends – XAMPP para Windows**. Disponível em: <http://www.apachefriends.org/pt_br/xampp-windows.html>. Acesso em: março de 2011.

GUENTER, M. S. **Sistema de Gerenciamento de Projetos de Pesquisa COMPESQ/INF**. 2010. 31 f. Trabalho de Conclusão (Graduação em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

NETBEANS. **Download do Netbeans IDE 7.0.1**. Disponível em: <http://netbeans.org/downloads/>. Acesso em: março de 2011.

OLIVEIRA, A. C. **Uso do Algoritmo Genético e Recozimento Simulado para o Problema de Alocação de Salas.** 2006. 82 f. Monografia (Graduação em Ciência da Computação) – Departamento de Ciência da Computação, UFLA, Minas Gerais.

SPARX SYSTEM. **Enterprise Architect – UML Design Tools and UML CASE Tools for Software Development**. Disponível em: <http://www.sparxsystems.com/products/ea/index.html>. Acesso em: novembro de 2010.

SQLITE. **Appropriate Uses for SQLite.** Disponível em: <http://www.sqlite.org/whentouse.html>. Acesso em: fevereiro de 2011.

SOUZA, Marcone Jamilson Freitas ; MARTINS, Alexandre Xavier ; ARAÚJO, C. R. . Experiências com a utilização de Simulated Annealing e Busca Tabu na resolução do Problema de Alocação de Salas. In: **XXXIV Simpósio Brasileiro de Pesquisa Operacional**, 2002, Rio de Janeiro. XXXIV SBPO. Rio de Janeiro : SOBRAPO, 2002. p. 1100-1110.

SUBVERSION. **Apache Subverion.** Disponível em: <http://subversion.apache.org/>. Acesso em: setembro de 2010.

TORTOISESVN. **TortoiseSVN – Downloads.** Disponível em: <http://tortoisesvn.net/downloads.html>. Acesso em: setembro de 2010.

YII FRAMEWORK. **Download Yii Framework**. Disponível em: <http://www.yiiframework.com/download/>. Acesso em: março de 2011.

YII FRAMEWORK. **The Definitive Guide to Yii**. Disponível em: <http://www.yiiframework.com/doc/guide/1.1/pt/index>. Acesso em: março de 2011.