

UNIVERSIDADE FEDERAL DO PIAUÍ  
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS  
CURSO BACHARELADO EM SISTEMAS DE INFORMAÇÃO

JOSÉ DENES LIMA ARAÚJO

**SISTEMA WEB PARA CONTROLE DE ALOCAÇÃO DE ESPAÇOS FÍSICOS DA  
UNIVERSIDADE FEDERAL DO PIAUÍ – CAMPUS SENADOR HELVÍDIO NUNES  
DE BARROS**

PICOS - PI

2015

JOSÉ DENES LIMA ARAÚJO

SISTEMA WEB PARA CONTROLE DE ALOCAÇÃO DE ESPAÇOS FÍSICOS DA  
UNIVERSIDADE FEDERAL DO PIAUÍ – CAMPUS SENADOR HELVÍDIO NUNES DE  
BARROS

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Sistemas de Informação do Campus Senador Helvídio Nunes de Barros da Universidade Federal do Piauí como parte dos requisitos para obtenção do Grau de Bacharel em Sistemas de Informação, sob orientação do Professor Ivenilton Alexandre de Souza Moura.

PICOS - PI

2015

## FICHA CATALOGRÁFICA

Serviço de Processamento Técnico da Universidade Federal do Piauí  
Biblioteca José Albano de Macêdo

**A663s** Araújo, José Denes Lima.

Sistema Web para controle de alocação de espaços físicos da  
Universidade Federal do Piauí – Campus Senador Helvídio  
Nunes de Barros / José Denes Lima Araújo. - 2014.

CD-ROM : il. ; 4 ¾ pol. (62 p.)

Monografia(Bacharelado em Sistemas de Informação) –  
Universidade Federal do Piauí. Picos-PI, 2014.

Orientador(A): Prof. Esp. Ivenilton Alexandre de Sousa Moura

1. Sistema Web. 2. Ruby on Rails. 3. Controle de Alocação  
de Espaços Físicos. 4. I. Título.

**CDD 005.1**

JOSÉ DENES LIMA ARAÚJO

SISTEMA WEB PARA CONTROLE DE ALOCAÇÃO DE ESPAÇOS FÍSICOS DA  
UNIVERSIDADE FEDERAL DO PIAUÍ – CAMPUS SENADOR HELVÍDIO NUNES DE  
BARROS

Trabalho de Conclusão de Curso apresentado  
ao Curso de Bacharelado em Sistemas de  
Informação do Campus Senador Helvídio  
Nunes de Barros da Universidade Federal do  
Piauí como parte dos requisitos para obtenção  
do Grau de Bacharel em Sistemas de  
Informação, sob orientação do Professor  
Ivenilton Alexandre de Souza Moura.

Data de Aprovação:

06/01/2015

Ivenilton Alexandre de Souza Moura

UFPI

Alan Rafael Ferreira dos Santos

UFPI

Frank César Lopes Vêras

UFPI

PICOS - PI

2015

Dedico este trabalho a minha família, em especial aos meus pais, Gdalva Florês de Lima Araújo e José Edimar de Araújo, por sempre acreditarem no meu potencial e me apoiarem durante todo o tempo.

## **AGRADECIMENTOS**

A minha família pelo incentivo, carinho e confiança. Em especial aos meus pais, pelos ensinamentos, conselhos e pelo esforço em sempre proporcionar a seus filhos tudo de melhor.

Ao meu orientador, Ivenilton Alexandre, por sempre estar disposto a dividir o seu conhecimento com seus estudantes. Agradeço pela paciência e pelo apoio, sempre me ajudando da melhor forma possível para realização desse trabalho.

A todos os professores do curso de Sistemas de Informação. Em especial aos professores Leonardo Sousa e a Patrícia Medyna, que mesmo não sendo meus orientadores estiveram sempre a disposição para tirar minhas dúvidas. Agradeço também a dedicação com que coordenam o curso.

Aos colegas e amigos que sempre me apoiaram e me fizeram crescer como pessoa e como profissional.

A todos vocês muito obrigado!

“Para realizar grandes conquistas, devemos não apenas agir, mas também sonhar; não apenas planejar, mas também acreditar.”

Anatole France

## RESUMO

A utilização de sistemas de informações está aumentando cada vez mais. Esse aumento ocorre pela necessidade em um maior controle das informações que são produzidas e mantidas pelas pessoas e organizações. Nessas últimas, públicas ou privadas, existe uma variedade de tarefas/processos que podem ser automatizadas com o uso de sistemas de informações. Essa automatização é benéfica no sentido de aumentar o controle, a velocidade e a segurança no acesso e manipulação das informações. O objetivo desse trabalho é apresentar o desenvolvimento de um sistema *web* para a Universidade Federal do Piauí, Campus Senador Helvídio Nunes de Barros, que automatize o processo de alocação de espaços físicos. Esse permitirá um maior controle na utilização dos espaços físicos e também que as pessoas façam solicitações de reservas pela *internet*. Para o seu desenvolvimento foram utilizadas: a metodologia de desenvolvimento *Scrum*, a modelagem UML, a linguagem de programação *Ruby*, o *Framework Ruby on Rails* e o *Framework* de *front-end Bootstrap*. Por fim, foram mostradas as principais funcionalidades para verificar sua conformidade com as necessidades do problema.

**Palavras-chave:** Sistema *Web*, *Ruby On Rails*, Controle de Alocação de Espaços Físicos.

## ABSTRACT

The usage of computer information systems has been increased over the past years. This happens thanks to the need of controlling information produced and kept for people and organizations. In any organization, whether it's a private or public one, there's a variety of tasks and processes that can be automated by computer information systems. This automation benefits the control, and at the same time increases speed and security in the access of these information. The objective of this thesis is to introduce the development of a web system to the Federal University of Piauí, Campus Senador Helvécio Nunes de Barros, which consist in the automation of the process of physical spaces allocation. This tool will allow a bigger control in the usage of physical spaces and also allow users to request reservations through the Internet. To the development of this work, the following were used: *Scrum* software development model; programming language *Ruby*; and *Ruby on Rails* Framework featuring *Bootstrap front-end*. At last, the main functionalities to check the conformity with the needs regarding the problem are disposed in this work.

**Keywords:** Web System, Ruby on Rails, Physical Space Allocation

## LISTA DE ILUSTRAÇÕES

<b>Figura 1</b> - Diagrama de Pacotes e os Subsistemas Identificados.....	31
<b>Figura 2</b> - Diagrama de Casos de Uso do Subsistema solicitacaoReservaEspacoFisico .....	35
<b>Figura 3</b> - Diagrama de Casos de Uso do Subsistema controlePatrimonio .....	36
<b>Figura 4</b> - Diagrama de classes.....	41
<b>Figura 5</b> – Diagrama de atividades de solicitação de reserva eventual .....	42
<b>Figura 6</b> – Diagrama de atividades de solicitação de reserva fixa .....	43
<b>Figura 7</b> - Diagrama de atividades de gerenciamento de reservas .....	44
<b>Figura 8</b> - Menus dos usuários solicitantes: (a) usuário externo, (b) usuário estudante, (c) usuário funcionário público nível 1, (d) usuário funcionário público nível 2 .....	46
<b>Figura 9</b> - Menus dos usuários administradores: (a) usuário funcionário público nível 2 e chefe do setor de patrimônio, (b) usuário funcionário público nível 3.....	47
<b>Figura 10</b> - Listagem das reservas eventuais de um usuário .....	48
<b>Figura 11</b> - Página com informações de reservas fixas e eventuais aprovadas, feriados e recessos.....	48
<b>Figura 12</b> - Página de escolha dos horários e datas da reserva eventual .....	49
<b>Figura 13</b> - Listagem das reservas fixas de um chefe de setor .....	50
<b>Figura 14</b> - Página com reservas fixas aprovadas e horários livres.....	51
<b>Figura 15</b> - Página de escolha dos dias e horários da reserva fixa .....	51
<b>Figura 16</b> - Página de gerenciamento de reservas eventuais .....	53
<b>Figura 17</b> - Exemplo de choque de horário de reservas eventuais .....	53
<b>Figura 18</b> - Diagrama Entidade Relacionamento .....	62

## LISTA DE QUADROS

<b>Quadro 1</b> - Subsistemas.....	31
<b>Quadro 2</b> - Requisitos funcionais .....	32
<b>Quadro 3</b> - Requisitos não funcionais .....	32
<b>Quadro 4</b> - Regras de negócio .....	33
<b>Quadro 5</b> - Atores.....	34

## LISTA DE ABREVIATURAS E SIGLAS

CSHNB	Campus Senador Helvídio Nunes de Barros
CSS	<i>Cascading Style Sheets</i>
DRY	Don't Repeat Yourself
HTML	<i>HiperText Markup Language</i>
IRB	<i>Interactive Ruby</i>
MVC	Model-View-Controller
SAD	Sistema de Apoio a Decisão
SAE	Sistema de Apoio ao Executivo
SCAD	Sistema de Consulta e Apoio a Decisão
SCAEF	Sistema de Controle de Alocação de Espaços Físicos
SE	Sistemas de Escritório
SIG	Sistema de Informação Gerencial
SIGAA	Sistema Integrado de Gestão de Atividades Acadêmicas
SPT	Sistema de Processamento de Transações
STC	Sistemas de Trabalho com Conhecimento
UFPI	Universidade Federal do Piauí
UML	<i>Unified Modeling Language</i>
XP	<i>Extreme Programming</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>14</b>
<b>1.1</b>	<b>Motivação</b> .....	<b>14</b>
<b>1.2</b>	<b>Objetivo</b> .....	<b>15</b>
<b>1.3</b>	<b>Organização do Documento</b> .....	<b>15</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b> .....	<b>16</b>
<b>2.1</b>	<b>Sistemas de Informação e Seus Tipos</b> .....	<b>16</b>
2.1.1	Sistemas de Informação .....	16
2.1.2	Classificação dos Sistemas .....	17
2.1.3	Sistemas de Informações Gerenciais .....	18
2.1.4	Sistemas <i>Web</i> .....	18
<b>2.2</b>	<b>Tecnologias</b> .....	<b>19</b>
2.2.1	Linguagem <i>Ruby</i> .....	20
2.2.2	<i>Framework Ruby on Rails</i> .....	21
2.2.3	<i>Framework de Front-end Bootstrap</i> .....	23
<b>2.3</b>	<b>Engenharia de Software</b> .....	<b>24</b>
2.3.1	Modelos de Processo .....	24
2.3.2	<i>Scrum</i> .....	26
2.3.3	Engenharia de requisitos .....	27
2.3.4	UML .....	28
<b>3</b>	<b>ESPECIFICAÇÃO DO SISTEMA</b> .....	<b>30</b>
<b>3.1</b>	<b>Requisitos do Sistema</b> .....	<b>31</b>
<b>3.2</b>	<b>Diagramas de Casos de Uso</b> .....	<b>33</b>
<b>3.3</b>	<b>Diagrama de Classes</b> .....	<b>40</b>
<b>3.4</b>	<b>Diagrama de Atividades</b> .....	<b>41</b>
<b>4</b>	<b>FUNCIONAMENTO DO SISTEMA</b> .....	<b>45</b>
<b>4.1</b>	<b>Usuários do sistema</b> .....	<b>45</b>
<b>4.2</b>	<b>Solicitação de reserva eventual</b> .....	<b>47</b>
<b>4.3</b>	<b>Solicitação de reserva fixa</b> .....	<b>50</b>
<b>4.4</b>	<b>Gerenciamento de reservas fixas e eventuais</b> .....	<b>52</b>
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b> .....	<b>55</b>
	<b>REFERÊNCIAS</b> .....	<b>56</b>
	<b>APÊNDICES</b> .....	<b>58</b>

<b>APÊNDICE A – Entrevista com o Chefe de Setor do Patrimônio .....</b>	<b>59</b>
<b>APÊNDICE B – Diagrama Entidade Relacionamento .....</b>	<b>62</b>

# 1 INTRODUÇÃO

## 1.1 Motivação

Sistema de Informação é um conjunto de elementos como: pessoas, computadores, processos, redes de computadores, software, hardware, entre outros que tem por função organizar e disseminar informações (O'BRIEN, 2006). A utilização de sistemas de informação vem aumentando a cada dia, tanto por pessoas como por organizações para as mais diversas finalidades.

Nas organizações, públicas ou privadas, os sistemas de informações são utilizados em todos os níveis. Desde o operacional processando atividades rotineiras, até o estratégico onde são utilizados para auxiliar na tomada de decisões. A utilização de sistemas supracitados traz uma série de benefícios como: o aumento na velocidade para realizar uma tarefa, a facilidade e rapidez no acesso à informação, aumento da segurança e, finalmente, a diminuição de erros.

A Universidade Federal do Piauí (UFPI) utiliza alguns sistemas de informações para automatizar suas atividades. No entanto, a maioria dos sistemas utilizados estão relacionados ao ensino, pesquisa ou extensão como, por exemplo, o Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA<sup>1</sup>). Porém, existem várias atividades administrativas que ainda são realizadas sem o uso de nenhum controle informatizado. Essas atividades são controladas por meio de formulários e pedidos escritos de próprio punho.

No Campus Senador Helvídio Nunes de Barros (CSHNB) da UFPI uma das atividades que é controlada por meio de formulários preenchidos à mão é a alocação de espaços físicos. Os interessados em realizar uma reserva devem deslocar-se ao setor de patrimônio onde um funcionário responsável realiza uma busca, em meio a um conjunto grande de fichas de papel, para então descobrir se é possível realizar a alocação. É notória a demora nesse processo, pois, deve-se contar a quantidade de tempo gasto para o interessado chegar até o setor de patrimônio, somado ao tempo gasto para encontrar e registrar informações no referido conjunto de fichas. Além disso, existe o aumento da possibilidade de erros nas reservas, por causa da natureza desta forma de controle.

Em razão do que foi exposto anteriormente, pode-se citar como exemplos de erros e dificuldades que acontecem com frequência: espaços reservados em um mesmo horário para

---

<sup>1</sup> SIGAA é um sistema de controle acadêmico desenvolvido pela Universidade Federal do Rio Grande do Norte

mais de uma pessoa, perda de informações, repetição de informações, ineficiência no controle sobre os espaços que estão sendo utilizados e os que estão livres, entre outros.

## **1.2 Objetivo**

O objetivo desse trabalho é desenvolver um sistema *Web* para a Universidade Federal do Piauí, Campus Senador Helvídio Nunes de Barros, que automatize o processo de reserva, alocação e mantenha informações acerca dos espaços físicos (salas, auditórios, laboratórios) do referido Campus.

## **1.3 Organização do Documento**

Após a introdução que relatou sobre o problema que se pretende resolver e o objetivo do sistema serão apresentados os próximos capítulos, que estão organizados da seguinte forma:

- Capítulo 2 – Referencial Teórico: Fornece o embasamento teórico para o trabalho. São mostrados conceitos relacionados a sistemas de informação, engenharia de software e tecnologias utilizados para o desenvolvimento do mesmo.
- Capítulo 3 – Especificação do Sistema: Será mostrada a análise de requisitos, os diagramas produzidos para o entendimento do problema e o desenvolvimento do sistema.
- Capítulo 4 – Funcionamento do Sistema: São mostrados os usuários do sistema, como esses se relacionam e as principais funcionalidades que o sistema apresenta.
- Capítulo 5 – Considerações Finais: Apresenta-se a conclusão do trabalho e indicações de trabalhos futuros.

## 2 REFERENCIAL TEÓRICO

### 2.1 Sistemas de Informação e Seus Tipos

#### 2.1.1 Sistemas de Informação

Para melhor compreensão dos conceitos de sistemas de informação, deve-se explicar o conceito de sistema. Conforme O'Brien (2006), sistema pode ser entendido como um grupo de elementos inter-relacionados ou em interação que formam um todo unificado e trabalha rumo a uma meta comum, recebendo insumos e produzindo resultados em um processo organizado de transformação.

Percebe-se que sistema possui três componentes básicos: entrada (o que o sistema recebe do meio em que está inserido), processamento (transformação das entradas) e saída (resultado dessa transformação). Os sistemas estão contidos dentro de outros e são abertos, pois trocam energia com o meio em que estão.

Os sistemas de informações são sistemas que estão dentro das organizações e trocam energia (informação) com seu ambiente. Recebem dados brutos como entrada, os processam e retornam informações úteis que são utilizadas pelas organizações. Laudon e Laudon (2004, p.7) afirmam que:

Sistema de informação pode ser definido tecnicamente como um conjunto de componentes inter-relacionados que coleta (recupera), processa, armazena, e distribui informações destinadas a apoiar a tomada de decisões, a coordenação e o controle de uma organização.

Para O'Brien (2006, p.6), "Sistema de informação é um conjunto organizado de pessoas, hardware, software, redes de comunicações e recursos de dados que coleta, transforma e dissemina informações em uma organização."

Pode-se perceber que o principal papel de um sistema de informação é capturar, organizar e disseminar as informações em uma organização e com isso ter um maior controle dos processos. Com isso nota-se que os sistemas de informações são utilizados para resolver os problemas de onde estão inseridos, coletando, organizando e analisando a grande quantidade de informações produzidas. Dessa forma, auxiliando a tomada de decisões nos diferentes níveis organizacionais e contribuindo também para o controle da organização como um todo.

### 2.1.2 Classificação dos Sistemas

Os sistemas de informação fornecem informações para todos os níveis administrativos de uma organização. Cada nível possui um conjunto diferente de informações que possuem sua importância. Laudon e Laudon (2004) mostram que podemos classificar os sistemas de informação de acordo com o nível organizacional que o sistema atende e também apresentam os tipos de sistemas de informação associados a cada nível.

São quatro tipos principais de níveis de sistemas de informação:

- Sistemas do nível operacional: acompanham atividades e transações elementares da organização;
- Sistemas do nível de conhecimento: suportam os trabalhadores do conhecimento e de dados da organização;
- Sistemas do nível gerencial: atendem às atividades de monitoração, controle, tomada de decisões e procedimentos administrativos dos gerentes médios;
- Sistemas do nível estratégico: são sistemas destinados as decisões de alto nível.

Para cada nível de sistemas de informação pode-se associar um ou mais tipos de sistemas:

- Sistemas de processamento de transações (SPTs): atendem ao nível operacional da organização. Realiza e registra transações rotineiras ao funcionamento da empresa;
- Sistemas de trabalho com conhecimento (STCs): encontrados no nível de conhecimento da organização, utilizam projetos, especificações e bases de conhecimento em geral para produzir modelos e gráficos;
- Sistemas de escritório (SE): encontrados no nível de conhecimento da organização. Aumentam a produtividade na manipulação de dados em um escritório;
- Sistemas de informação gerencial (SIG): atendem ao nível gerencial da organização. Fornecem relatórios aos gerentes. Apoiam as funções de planejamento, controle e decisão no nível gerencial;
- Sistemas de apoio a decisão (SAD): atendem ao nível de gerencia da organização. Ajudam os gerentes a tomar decisões não-usuais, que se alteram com rapidez e que não são facilmente especificadas com antecedência;
- Sistemas de apoio ao executivo (SAEs): atendem ao nível estratégico da organização. São destinados a apoiar a alta gerência em tarefas estratégicas, como o planejamento de longo prazo.

### 2.1.3 Sistemas de Informações Gerenciais

Laudon e Laudon (2004) afirmam que os SIGs possuem um foco maior em eventos internos da organização apoiando funções de planejamento, controle e decisão no nível gerencial. Esse tipo de sistema resume e relata operações básicas da empresa.

Segundo Stair e Reynolds (2006) o principal objetivo de um SIG é ajudar uma empresa a alcançar seus objetivos, fornecendo detalhes a seus administradores em relação a operações diárias da organização. A saída da maioria dos SIGs é através de relatórios. Garcia e Bazzoti (2006, p.9) mostram os possíveis benefícios de um SIG:

- Redução dos custos das operações;
- Melhoria no acesso das informações, proporcionando relatórios mais precisos e rápidos, com menor esforço;
- Melhoria na produtividade;
- Melhoria nos serviços realizados e oferecidos;
- Melhoria na tomada de decisões, por meio do fornecimento de informações mais rápidas e precisas;
- Estímulo de maior interação dos tomadores de decisão;
- Fornecimento de melhores projeções dos efeitos das decisões;
- Melhoria na estrutura organizacional, para facilitar o fluxo de informações;
- Melhoria na estrutura de poder, proporcionando maior poder para aqueles que entendem e controlam os sistemas;
- Redução do grau de centralização de decisões na empresa; e
- Melhoria na adaptação da empresa para enfrentar os acontecimentos não previstos.

Pode-se perceber que é inegável a grande quantidade de benefícios que a implantação de um SIG proporciona em uma organização. Esses auxiliam em várias tarefas, diminuindo seu tempo de execução e a quantidade de erros, assim proporcionando maior segurança na tomada de decisões.

### 2.1.4 Sistemas *Web*

Os sistemas de informações podem ser do tipo *desktop* ou *web*. O que determina se um sistema será de um desses tipos é o estudo detalhado do problema que será resolvido e as

necessidades que os usuários possuem. Determinados problemas são resolvidos com sistemas *desktop* e outros com sistemas *web*.

A *internet* é uma rede de computadores que conecta uma grande variedade de dispositivos de todas as partes do mundo. A *web* é um conjunto de páginas que são acessíveis pela internet que usam a tecnologia de hipertexto para percorrer as páginas através de *links*. Pode-se descrever a *web* como um sistema de informação em hipertexto, gráfico, distribuído, independente de plataforma, dinâmico, interativo e global, utilizado na internet (RODRIGUES, 2010).

Segundo Pauli (2013), o termo aplicação *web* tem significados diferentes para as pessoas dependendo do contexto. As pessoas usam os termos aplicação *web*, *web site*, sistemas baseado em *web*, *software* baseado em *web* muitas vezes com o mesmo significado. Pode-se entender aplicação *web* como um software baseado em *web* que realize funcionalidades ou ações de acordo com entradas de usuários e que interaja com banco de dados para manter informações.

Pode-se entender sistema *web* como um sistema de informação desenvolvido para funcionar utilizando tecnologia *web*, podendo estar acessível através da internet ou funcionando na intranet (rede interna de uma organização). Se esse estiver acessível através da internet pode ser utilizado pelos usuários em qualquer lugar em que esses se encontrem e em qualquer momento. Já, se estiver acessível através da intranet, só pode ser utilizado enquanto os usuários estiverem na organização em que o sistema se encontre.

## 2.2 Tecnologias

Para o desenvolvimento de sistemas *web* e *desktop* (citados na seção 2.1.4) existem tecnologias que são comuns a ambos como, por exemplo: linguagens de programação e banco de dados. Porém, existem diferenças que são relacionadas principalmente à arquitetura. Os sistemas *desktop* geralmente são utilizados em apenas uma máquina ou em um conjunto de máquinas que se encontram num mesmo local. Já os sistemas *web* se encontram em um servidor e podem ser acessados de qualquer lugar que tenha acesso a *internet*. Existem tecnologias que são utilizadas para desenvolvimento *web* como:

- *Framework* para aplicações *web*: É um conjunto de classes implementadas usadas para auxiliar o desenvolvimento. Essas concentram códigos comuns associados a atividades que são sempre realizadas em desenvolvimento *web* (acesso a banco, gerenciamento de sessão, entre outras). E isso faz com que diminua a sobrecarga relacionada a essas

atividades, pois o programador não precisa implementá-las a cada projeto. Como exemplo de *frameworks* para aplicações *web* tem-se: *Ruby on Rails*, *Django*, *CakePHP* e *Spring MVC*;

- *Framework* para *front-end*: Segue o mesmo conceito de *framework* citado no tópico anterior só que está relacionado apenas ao desenvolvimento de *layouts* de sistemas *web*. Como exemplo de *frameworks* para *front-end* tem-se: *Bootstrap*, *Foundation* e *LESS Framework*.

As principais tecnologias utilizadas durante esse projeto são abordadas a seguir.

### 2.2.1 Linguagem *Ruby*

A linguagem *Ruby* foi criada por Yukihiro Matsumoto no ano de 1995 no Japão. Segundo Flanagan e Matsumoto (2008), *Ruby* teve inspiração em outras linguagens da época como *LISP*, *Smalltalk*, *Perl* e *Python*, mas possui uma sintaxe fácil de aprender para programadores C e Java.

*Ruby* possui algumas características que a tornam uma linguagem bastante flexível como: simplicidade de sintaxe, multiparadigma e multiplataforma. Os paradigmas de programação apresentados nesta linguagem são funcional, orientado a objetos, reflexivo e imperativo. Eles podem ser utilizados pelo programador de acordo com suas necessidades. Os códigos *Ruby* podem ser executados em vários sistemas operacionais. Algumas das características citadas anteriormente possibilitam a escrita de códigos mais curtos e fáceis de ser entendidos.

*Ruby* foi escrito na linguagem de programação C. É uma linguagem interpretada e não compilada. Para executar um código *Ruby* no computador é necessária a instalação de um interpretador. Segundo Flanagan e Matsumoto (2008), o interpretador pode possuir a implementação de referência que define a linguagem MRI (“Mat’s *Ruby implementation*”) que é baseada em C ou alguma implementação alternativa como *JRuby* (baseada em Java), *IronRuby* (baseada em .NET), *Rubinius* (baseada em Smalltalk), *Cardinal* (baseado em Perl), entre outras.

Um recurso considerado indispensável pela maioria dos desenvolvedores *Ruby* é o IRB (*Interactive Ruby*). Nele o programador pode fazer os mais variados testes com os recursos da linguagem. De acordo com Thomas (2009), IRB é um “shell” *Ruby* parecido com o conceito de um shell de um sistema operacional e esse fornece um ambiente onde é possível

fazer testes com a linguagem em tempo real. RUBY-LANG (2014) apresenta outras características do *Ruby* onde se destacam as seguintes:

- Capacidade de tratamento de exceções;
- Possui *garbage collector* para todos os objetos *Ruby*;
- Escrever extensões C em *Ruby* é mais fácil do que em *Perl* ou *Python*, com uma API refinada para chamar *Ruby* desde o código C;
- O *Ruby* pode carregar bibliotecas de extensão (*extension libraries*) dinamicamente se um Sistema Operacional o permitir;
- O *Ruby* tem um sistema de *threading* independente do Sistema Operacional;
- O *Ruby* é altamente portátil: é desenvolvido principalmente em ambiente GNU/Linux, mas trabalha em muitos tipos de ambientes UNIX, Mac OS X, Windows 95/98/Me/NT/2000/XP, DOS, BeOS, OS/2.

Souza (2012) afirma que a criação do *Rails* foi o que fez a linguagem se tornar um sucesso. E que existe um mercado grande em torno da linguagem e a previsão é que ele continue crescendo.

Em muitas linguagens é comum o uso de bibliotecas para acelerar o desenvolvimento. Em *Ruby* as bibliotecas são distribuídas através de arquivos chamados *gems*, que possuem a extensão *.gem*, onde ficam localizados o código fonte e também informações e metadados.

### 2.2.2 *Framework Ruby on Rails*

*Ruby on Rails* é um *framework* de desenvolvimento *web* que facilita a criação, implantação e manutenção de aplicações *web*. Foi criado em 2003 por David Heinemeier Hansson. Harlt (2012) afirma que este se tornou uma dos mais poderosos e populares *frameworks* para a construção de aplicações *web* dinâmicas e existem motivos que levaram a ferramenta a ter esse sucesso e ser de qualidade: *Ruby on Rails* possui código aberto, disponível sob a licença MIT permissiva; possui um *design* compacto e elegante; através da exploração da maleabilidade da linguagem *Ruby*, o *Rails* cria uma linguagem de domínio específico para escrever aplicações *web*. Assim tarefas comuns como geração de HTML, geração de modelos de dados e de URIs são mais fáceis com o *Rails* e isto faz com que o resultado seja um código legível e conciso.

*Ruby on Rails* é um *metaframework*, ou seja, é um *framework* composto de *frameworks*. Entre os *frameworks* pode-se citar como exemplo (CAELUM, 2014):

- *Action Pack*: Responsável pela apresentação e controle. Nele está contido o *Action Controller* que gerencia os controladores processando solicitações de entrada, extraíndo os parâmetros e despachando para a ação pretendida; o *Action View* que gerencia as *views*, controlam a renderização de *templates* e *partials* e incluem suporte para *AJAX*; e o *Action Dispatch* que trabalho com o gerenciamento de rotas;
- *Active Record*: É a base para os modelos. Fornece independência de banco de dados. Possui métodos que escrevem consultas dependendo do banco de dados que a aplicação funcionar e estiver configurado;
- *Action Mailer*: Responsável pelo envio de e-mails. Permite que a aplicação envie e-mails para seus usuários;
- *Action Support*: Possui classes *Ruby* utilitárias que são usadas no *Rails*.

O *Rails* segue a arquitetura MVC (*Model-View-Controller*), que separa a lógica de negócio da lógica de entrada e de apresentação associada com uma interface (HARTL, 2012). Com isso fica claro onde se encaixa cada tipo de código o que torna a manutenção mais fácil. De acordo com Ruby, Thomas e Hansson (2011) o modelo MVC se divide em três componentes: modelos (*Models*), apresentações (*Views*) e controladores (*Controllers*) onde o programador desenvolve-os como pedaços separados de uma funcionalidade e interliga todos juntos como o programa é executado. Esses componentes são definidos da seguinte forma pelos autores:

- Modelo (*Model*): É responsável por manter o estado dos dados. Também impõe todas as regras de negócio que se aplicam aos dados;
- Apresentação (*View*): É responsável por gerar uma interface de usuário, normalmente com base em dados do modelo;
- Controle (*Controller*): É responsável por orquestrar a aplicação. Recebem eventos do mundo exterior, interagem com o modelo e mostram através da apresentação ao usuário.

Por seguir o modelo MVC pode-se perceber que os projetos desenvolvidos em *Ruby on Rails* estão bem organizados nas três camadas. Cada uma com sua responsabilidade, agilizando assim o processo de desenvolvimento e facilitando futuras manutenções necessárias.

*Ruby on Rails* utiliza na sua camada de apresentação (*View*) linguagens que a maioria dos navegadores compreende e que se tornaram padrões para o desenvolvimento *web*. Essas são utilizadas no *front-end* (HARTL, 2012). As linguagens são o HTML (*HiperText Markup Language*), o CSS (*Cascading Style Sheets*) e o *javascript*:

- **HTML:** Estrutura as páginas na internet. É uma linguagem de marcação que define o formato e o significado do texto. Trabalha com o conceito de hipertexto que são as ligações entre as páginas de internet que permite navegar entre elas (RODRIGUES, 2010). Com o HTML é possível inserir todos os elementos necessários numa página como textos, imagens, tabelas, vídeos entre outros elementos;
- **CSS:** Adiciona estilo a estrutura (espaçamento, cor, fonte) criada pelo HTML. Segundo Schmitt (2010) o CSS fornece uma maneira simples de estilizar o conteúdo em uma página web. O CSS formata os elementos definidos no HTML. Rodrigues (2010) afirma que com o uso do CSS é possível separar os elementos das páginas definidas no HTML das informações que descrevem como esses elementos serão exibidos;
- **Javascript:** Responsável pelo comportamento das páginas. Rodrigues (2010, p.60) afirma que “*javascript* é uma linguagem de programação utilizada para criar pequenos programas para realizar ações em uma página *web*”.

*Rails* também segue duas outras filosofias: DRY – *dont repeat yourself* (não se repita) e convenções sobre configurações. A primeira evita que o programador escreva o mesmo código várias vezes, ou seja, ele pode centralizar determinado código e utilizá-lo em vários locais. Na segunda, o *Rails* tem convenções de configurações que quando seguidas evitam que o programador escreva códigos desnecessários.

### 2.2.3 Framework de *Front-end Bootstrap*

*Bootstrap* é um *framework* de código aberto desenvolvido por Mark Otto e Jacob Thornton que a lançaram enquanto eram funcionários do *Twitter*, pois havia uma necessidade de padronizar os conjuntos de ferramentas de *front-end* de engenheiros em toda a empresa (SPURLOCK, 2013).

Com o *Bootstrap* o desenvolvedor não precisa definir a partir do zero os elementos básicos da interface como botões. Nele estão à disposição vários elementos de *interface* de uso comum. *Bootstrap* é um kit de ferramentas com as convenções padrão de classes com documentação limpa e prática para dar vida a códigos que estão prontos para serem usados e

personalizados de acordo com as necessidades dos desenvolvedores. Não é uma solução mágica para resolver o problema de reutilização de elementos da *interface*, mas é um bom início (MAGNO, 2013).

Pelas facilidades que apresenta em reutilização de componentes prontos o *bootstrap* vem sendo utilizado cada vez mais pelos desenvolvedores, pois permite uma maior agilidade na hora de montar as interfaces dos seus sistemas *web*. Não é preciso criar e dar estilo a cada componente da sua interface. Basta associar os elementos das páginas HTML às classes que existem prontas no *bootstrap* para reutilizar-se botões, menus, barras de navegação, formulários e outros componentes comumente utilizados em *interfaces web*.

## 2.3 Engenharia de Software

“A engenharia de *software* é uma disciplina da engenharia que se ocupa de todos os aspectos da produção de *software*, desde os estágios iniciais de especificação do sistema até a manutenção desse sistema, depois que ele entrou em operação.” (SOMMERVILLE, 2005, p.5). É um conjunto de métodos e técnicas que possibilita o desenvolvimento de *softwares* com qualidade. Esses métodos e técnicas não estão relacionados apenas ao desenvolvimento de softwares, mas também a atividades que dão apoio a esse desenvolvimento. A seguir são apresentados mais conceitos relacionados a engenharia de *software*.

### 2.3.1 Modelos de Processo

*Softwares* estão sendo utilizados pelas pessoas para as mais diversas finalidades e em praticamente todas as áreas como: entretenimento, indústria, transporte, ciência e telecomunicações. Esses *softwares*, desde os mais simples até os mais complexos, passam por um processo de desenvolvimento.

Sommerville (2011, p.18) afirma que “um processo de *software* é um conjunto de atividades relacionadas que levam à produção de um produto de *software*”. Segundo Pressman (2011), processo de *software* é um conjunto de atividades, ações e tarefas para o desenvolvimento de um *software*. Cada uma dessas atividades, ações e tarefas estão contidas dentro de uma metodologia ou modelo que determina seu relacionamento com o processo e seu relacionamento umas com as outras.

Modelos de processo prescritivos foram propostos para trazer ordem a área de desenvolvimento de *software*. Esses modelos de processos buscam a estrutura e a ordem e

recebem a denominação “prescritivos”, pois prescrevem um conjunto de atividades e um fluxo para realizá-las (PRESSMAN, 2011). Abaixo são mostrados alguns exemplos de modelos de processos prescritivos:

- **Cascata:** É também conhecido como ciclo de vida clássico. Possui uma abordagem sequencial, pois segue uma sequencia de passos realizando um após o outro, e sistemática, pois possui a mesma sequencia de passos todas as vezes que é aplicado. Possui as fases de levantamento de necessidades dos usuários, planejamento das atividades, modelagem e projeto, desenvolvimento da aplicação, teste, implantação e suporte. Uma fase só pode ser iniciada após o término e aprovação da fase anterior a ela;
- **Espiral:** O produto é desenvolvido em uma série de iterações que passam pelos processos de planejamento, ativação, análise de riscos avaliação e desenvolvimento. A cada nova iteração passa-se por todos esses processos (PÁDUA, 2000). Isso possibilita adicionar novas funcionalidades a cada iteração quando identificada as necessidades;
- **Evolucionário:** Esse tipo de modelo também é iterativo. Ele parte da ideia que softwares evoluem, pois as regras de negócios mudam até o desenvolvimento e entrega do produto final. A cada iteração desse modelo são desenvolvidas versões cada vez mais completas dos *softwares*.

Os modelos de processos prescritivos citados anteriormente gastam muito tempo no planejamento, projeto e documentação. Com isso surgiram as metodologias ágeis que permitiam aos desenvolvedores focarem no *software* e não na concepção e documentação. De acordo com Sommerville (2011) as metodologias ágeis são adequadas no desenvolvimento de sistemas onde seus requisitos mudam rapidamente durante o desenvolvimento. Abaixo são mostrados alguns exemplos de metodologias ágeis de desenvolvimento:

- **XP (*Extreme Programming*):** O XP apresenta quatro atividades: planejamento, projeto, codificação e testes. Em todas elas a um acompanhamento contínuo e a realização de pequenos ajustes para adequar o desenvolvimento. O XP possui alguns princípios básicos: *feedback* rápido, presumir simplicidade, mudanças incrementais, abraçar mudanças e trabalho de qualidade que devem ser seguidos;
- **Scrum:** Sabbagh (2013, p.40) mostra uma definição para *Scrum*:

É um *framework* Ágil, simples e leve, utilizado para a gestão do desenvolvimento de produtos complexos imersos em ambientes complexos. *Scrum* é embasado no

empirismo e utiliza uma abordagem iterativa e incremental para entregar valor com frequência e, assim, reduzir os riscos do projeto.

- Desenvolvimento de *software* adaptativo: Se concentra na colaboração humana e na auto-organização das equipes. São definidos planos de ciclos adaptáveis e a medida que são executados os planos podem sofrer ajustes para a realidade da equipe de desenvolvimento (PRESSMAN, 2011).

### 2.3.2 *Scrum*

Segundo Pressman (2011) *Scrum* é um método de desenvolvimento ágil de *software* e seus princípios estão de acordo com o manifesto ágil e são usados para orientar as atividades de desenvolvimento dentro de um processo que incorpora as seguintes atividades: requisitos, análise, projeto, evolução e entrega. Em cada uma dessas atividades ocorrem tarefas a serem realizadas dentro de um padrão de processo chamado *sprint*. O *scrum* possui um conjunto de padrões de processos onde cada um desses padrões possui um conjunto de ações de desenvolvimento. Os padrões de processos são:

- *Backlog*: Lista de requisitos ou funcionalidades que estão dispostos em ordem de prioridade para serem implementados. Essa lista pode ser atualizada a qualquer momento;
- *Sprints*: São unidades de trabalhos que implementam um requisito do *backlog* e possui um prazo para ser entregue;
- Reuniões *Scrum*: Reuniões pequenas realizadas todos os dias. É realizada para se saber como anda o desenvolvimento ou se há algum problema;
- *Demos*: Entrega de parte do *software* ao cliente para que a funcionalidade implementada possa ser utilizada e avaliada por ele. A *demo* não precisa necessariamente ter toda a funcionalidade planejada, mas pode ter funções que podem ser entregues respeitando o prazo combinado.

*Scrum* vem ganhando espaço entre equipes de desenvolvimento de *software*. De acordo com a VersionOne (2013) uma pesquisa realizada em 2013, o *Scrum* e suas variantes são utilizados por 73 % dos participantes e o *Scrum* é utilizado por 53%. Mesmo sendo bem adotado não significa que *Scrum* resolve tudo. De acordo Sabbagh (2013) *Scrum* trará benefícios se bem utilizado e seu uso aumenta a qualidade do produto e melhora a produtividade das equipes. Ainda cita uma série de benefícios no uso do *Scrum*, que são:

- Entregas frequentes de retorno ao investimento dos clientes;
- Redução dos riscos do projeto;
- Maior qualidade no produto gerado;
- Mudanças utilizadas como vantagem competitiva;
- Visibilidade do progresso do projeto;
- Redução do desperdício;
- Aumento de produtividade.

Com o *Scrum* ocorrem entregas frequentes de retorno ao investimento do cliente, ou seja, a cada parte implementada do projeto é levada ao cliente uma versão funcional para ser testada e validada. Com isso o cliente pode dar um *feedback* do sistema e indicar mudanças ou detalhes adicionais. Assim o cliente acompanha todo o desenvolvimento diminuindo o número de erros.

Independente da metodologia de desenvolvimento utilizada, uma das fases mais importantes e que deve ser feita com todo cuidado é o levantamento de requisitos.

### 2.3.3 Engenharia de requisitos

A engenharia de requisitos é uma ação da engenharia de *software* que se inicia na atividade de comunicação e continua na modelagem. É um conjunto de tarefas e técnicas que levam ao entendimento dos requisitos. Fornece a maneira apropriada para entender o que o cliente deseja, verificando as necessidades, a viabilidade, negociando soluções, especificando as soluções, validando a especificação e gerenciando as necessidades (PRESSMAN, 2011).

Requisitos são descrições dos serviços que devem ser fornecidos por um sistema e suas restrições operacionais. Os requisitos podem ser divididos em dois tipos: (SOMMERVILLE, 2011).

- Requisitos do usuário: São requisitos compreendidos pelos usuários do sistema por isso devem está em um nível de abstração mais alto. Mostram as funções que o sistema deve fornecer e as restrições que possui;
- Requisitos do sistema: Fornecem uma descrição mais detalhada do sistema. Define o que o sistema deve fazer e não como fazer.

Os requisitos de sistemas podem ser classificados em:

- Requisitos funcionais: São declarações de serviços que o sistema deve fornecer, descrevendo o que o sistema deve fazer, como o sistema deve reagir a entradas

específicas e como o sistema deve se comportar em situações específicas (SOMMERVILLE, 2011);

- Requisitos não funcionais: São as funções e restrições oferecidas pelo sistema. Quantificam determinados aspectos dos sistemas (PÁDUA, 2000);
- Requisitos de domínio: Se originam do domínio de aplicação do sistema e refletem características e restrições desse domínio. São as regras de negócios.

Durante a fase de levantamento de requisitos pode ser utilizada uma linguagem de modelagem para ajudar tanto os desenvolvedores como os usuários a visualizar as funcionalidades que se deseja implementar.

#### 2.3.4 UML

UML – Unified Modeling Language (Linguagem de Modelagem Unificada) é uma linguagem visual utilizada para modelar *softwares* baseados no paradigma de orientação a objetos. Essa linguagem tem o objetivo de ajudar os engenheiros de *software* a definirem as características dos sistemas como seus requisitos, comportamento, estrutura lógica, a dinâmica dos seus processos e até mesmo necessidades físicas (GUEDES, 2011). De acordo com Silva e Videira (2001) as principais características que a UML apresenta são:

- Independente do domínio da aplicação;
- Independente do processo ou metodologia de desenvolvimento;
- Independente das ferramentas de modelagem;
- Apresenta mecanismos de extensão;
- Possui um conjunto muito significativo de diferentes diagramas.

A utilização da modelagem durante o desenvolvimento de sistemas é uma das formas de comunicação entre os desenvolvedores e também desses com os usuários do sistema. Segundo Guedes (2011, p.23) “Durante a análise de requisitos, uma linguagem de modelagem auxilia a levantar questões que não foram concebidas durante as entrevistas iniciais”. A UML possui vários diagramas e cada um desses apresenta o sistema de uma visão diferente. Alguns dos diagramas oferecidos pela UML são:

- Diagramas de casos de uso: Representam o sistema da visão do usuário. Por isso possui uma linguagem simples. Mostra os usuários do sistema e quais funcionalidades esses possuem acesso;

- Diagrama de classes: Especifica a estrutura das classes utilizadas pelo sistema. Mostra os dados que as classes possuem, seus métodos e como essas classes se relacionam e trocam informações;
- Diagrama de seqüências: Apresenta como o sistema se comporta. É mostrado como os objetos trocam mensagem entre si para chamadas de métodos e em que ordem acontece;
- Diagrama de atividades: Descreve os passos para realizar uma determinada atividade. Mostra todo o fluxo de controle da atividade.

### 3 ESPECIFICAÇÃO DO SISTEMA

Esse capítulo mostra as especificações do Sistema de Controle de Alocação de Espaços Físicos proposto nesse trabalho, que de agora em diante será tratado como SCAEF. Será mostrado o levantamento de requisitos, diagramas de casos de uso, diagrama de classes e principais diagramas de atividades do sistema para que seja possível identificar todos os usuários e os detalhes do sistema.

O SCAEF é um módulo de um sistema integrado onde outros desenvolvedores trabalham em outros módulos. O sistema de controle de diárias e o sistema de controle de alocação de veículos são exemplos de módulos integrados. Os três juntos funcionam integrados e nessa condição recebem o nome de Sistema de Consulta e Apoio à Decisão (SCAD).

O SCAEF é um sistema de informação gerencial *web*. Para o seu desenvolvimento foi utilizada a linguagem de programação *Ruby*. A sua utilização foi ocasionada por ser uma linguagem que possui uma sintaxe bastante fácil de ler (tornando manutenções futuras mais fáceis) e por ser multiparadigma, tornando a resolução de problemas mais fácil e flexível dependendo da habilidade do programador. É a linguagem base para o framework *Ruby on Rails*. A utilização do *Rails* foi ocasionada por ser um *framework* que possui uma grande quantidade de recursos, possuir uma comunidade grande que produz bibliotecas para as mais diversas finalidades e apresentar fácil integração com outros *frameworks*. Para o desenvolvimento da interface do SCAEF foi utilizado o *framework Bootstrap* por esse apresentar vários elementos que compõem páginas *web* e uma flexibilização na forma de utilização desses, permitindo o desenvolvimento de páginas personalizadas ou o uso de temas já prontos.

A metodologia de desenvolvimento utilizada foi o *Scrum*, pois essa metodologia possui entre suas características o acompanhamento do desenvolvimento de perto por parte dos usuários, sempre analisando o que está sendo feito e validando.

Com o SCAEF é possível realizar reservas de dois tipos: fixas e eventuais que serão explicadas em detalhes mais adiante. Essas reservas são solicitadas por determinados usuários e avaliadas por outros, esses últimos possuem funções de administração. Com isso, pode-se dividir o SCAEF em dois subsistemas. A Figura 1 mostra o diagrama de pacotes que os representa. O subsistema “solicitacaoReservaEspacoFisico” representa a parte do sistema utilizado para solicitação e manutenção de reservas. O subsistema “controlePatrimonio” representa a parte de administração do SCAEF.

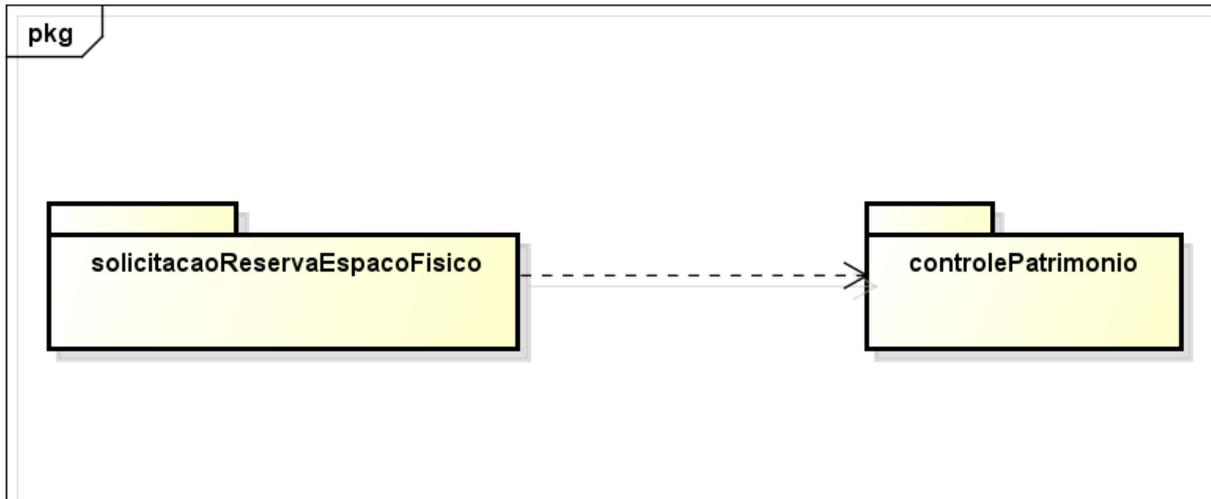


Figura 1 - Diagrama de Pacotes e os Subsistemas Identificados  
Fonte: O autor (2014)

O Quadro 1 mostra descrições mais detalhadas dos dois subsistemas.

Quadro 1 - Subsistemas

Subsistema	Descrição
<b>controlePatrimonio</b>	Envolve toda a funcionalidade relacionada com o gerenciamento de reservas eventuais e fixas, abrangendo controle de blocos, tipos de espaços, espaços, períodos letivos, feriados, recessos e replicação de reservas fixas.
<b>solicitacaoReservaEspacoFisico</b>	Envolve as funcionalidades relacionadas a solicitação e manutenção de reservas eventuais e fixas de espaços físicos.

Para o desenvolvimento do SCAEF, além dos estudos sobre a linguagem de programação *Ruby*, o *framework Ruby on Rails* e o *framework bootstrap* foram realizadas entrevistas para o levantamento dos requisitos e elaboração de diagramas a partir das informações levantadas.

### 3.1 Requisitos do Sistema

Para o levantamento dos requisitos do SCAEF foram realizadas reuniões com o chefe do setor de patrimônio que é o responsável pelos espaços físicos da UFPI campus Senador Helvídio Nunes de Barros. Durante as reuniões foram realizadas entrevistas (APÊNDICE A) e foi analisado o *modus operandi* atual do controle e alocação de espaços físicos. Com isso, foi possível a identificação das necessidades e dificuldades enfrentadas atualmente. Também foram questionadas as possíveis funcionalidades que o SCAEF deveria possuir.

Após as reuniões, foi possível elaborar os requisitos funcionais, não funcionais e regras de negócio do SCAEF. O Quadro 2 mostra os requisitos funcionais cada um com seu identificador, uma descrição e as dependências que esse possui.

Quadro 2- Requisitos funcionais

Identificador	Descrição	Depende de
RF01	O sistema deverá possuir cadastro de usuários associando-os com um tipo. E a uma função quando for o caso.	
RF02	O usuário possuirá <i>login</i> e senha para acessar o sistema.	RF01
RF03	Os níveis de acesso aos recursos do sistema serão de acordo com os tipos e/ou funções dos usuários.	RF02
RF04	O sistema permitirá cadastro básico de: espaços físicos, blocos, tipos de espaços, feriados, recessos e períodos letivos.	RF03
RF05	O sistema permitirá dois tipos de reservas: eventuais e fixas.	RF04
RF06	O sistema permitirá a solicitação de reservas eventuais.	RF05
RF07	O sistema permitirá a solicitação de reservas fixas.	RF05
RF08	O sistema permitirá o gerenciamento de reservas eventuais.	RF06
RF09	O sistema permitirá o gerenciamento de reservas fixas.	RF07
RF10	O sistema permitirá a geração de relatórios sobre a ocupação dos espaços.	RF08, RF09
RF11	O sistema permitirá a cópia de reservas fixas já aprovadas de um período para outro.	RF07
RF12	O usuário solicitante de reserva poderá acompanhar o estado das suas reservas.	RF06, RF07

O Quadro 3 mostra os requisitos não funcionais cada um com seu identificador, uma descrição, a categoria a qual o requisito pertence e suas dependências.

Quadro 3 - Requisitos não funcionais

Identificador	Descrição	Categoria	Depende de
RNF01	O sistema deve controlar o acesso às funcionalidades. As Funcionalidades de solicitação de reservas e as de gerencia (que devem estar disponíveis somente para usuários com permissão).	Segurança de Acesso	RF02
RNF02	O sistema deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado.	Portabilidade	
RNF03	A persistência das informações deve ser implementada em um Sistema Gerenciador de Bancos de Dados Relacionais (SGBDR) livre (Postgres ou MySQL).	Manutenibilidade	

O Quadro 4 mostra as regras de negócios cada uma com seu identificador, uma descrição e suas dependências.

Quadro 4 - Regras de negócio

Identificador	Descrição	Depende de
RN01	Não serão permitidas reservas de um mesmo espaço nos mesmos dias e horários.	
RN02	Os horários das reservas possuem a duração fixa de uma hora.	
RN03	Os horários que podem ser reservados são das 6 às 22 horas.	RN02
RN04	As reservas somente podem ser excluídas pelos usuários solicitantes.	
RN05	O sistema permite que os usuários (que possam realizar reserva fixa) somente possam realizar uma reserva fixa para cada espaço por período.	RN03

Depois do levantamento dos requisitos funcionais, não funcionais e regras de negócio é preciso validá-los junto aos usuários. Assim esses requisitos e regras podem ser utilizados para construção de diagramas de casos de uso que servem para demonstrar aos usuários o que se pretende desenvolver e se está de acordo com as suas necessidades.

### 3.2 Diagramas de Casos de Uso

Guedes (2011, p.52) afirma que:

O diagrama de casos de uso é de grande auxílio para a identificação e compreensão dos requisitos do sistema, ajudando a especificar, visualizar e documentar as características, funções, serviços do sistema desejados pelo usuário. O diagrama de casos de uso tenta identificar os tipos de usuários que irão interagir com o sistema, quais papéis esses usuários irão assumir e quais funções um usuário específico poderá requisitar.

A partir das informações sobre diagrama de casos de uso pode-se entendê-lo como um meio que fornece uma visão geral do funcionamento do sistema e os usuários que estão envolvidos em cada funcionalidade.

O SCAEF diferencia seus usuários pelo seu tipo: externo, funcionário público e estudante. Outra forma de diferenciação é a atribuição de funções aos usuários do tipo funcionário público. As funções são divididas em níveis e são: nível 1 – técnico administrativo e professor, nível 2 – chefe de setor, nível 3 – coordenador administrativo financeiro e nível 4 - diretor.

O Quadro 5 mostra todos os atores identificados no sistema e uma descrição breve de quem são e o que podem fazer.

Quadro 5 - Atores

<b>Ator</b>	<b>Descrição</b>
<b>Usuário não cadastrado</b>	São usuários que podem acessar o sistema e visualizar reservas eventuais aprovadas sem a necessidade de um cadastro.
<b>Usuário externo</b>	São pessoas externas a UFPI, não se tratando de funcionários ou estudantes. Podem realizar solicitações de reservas eventuais.
<b>Usuário estudante</b>	São estudantes matriculados nos curso da UFPI. Podem realizar solicitações de reservas eventuais.
<b>Usuário nível 1</b>	São usuários do tipo funcionário público e que possuem as funções de professor ou técnico administrativo na UFPI. Podem realizar solicitações de reservas eventuais.
<b>Usuário nível 2</b>	São usuários do tipo funcionário público e que possuem a função de chefe de setor/coordenador. Podem realizar solicitações reservas eventuais e fixas.
<b>Usuário nível 2 – chefe de setor de patrimônio</b>	É um usuário do tipo funcionário público e que possui a função de chefe do setor de patrimônio. Realiza a administração do sistema.
<b>Usuário nível 3</b>	É um usuário do tipo funcionário público e que possui a função de coordenador administrativo financeiro. Realiza a administração do sistema.
<b>Usuário nível 4</b>	É um usuário do tipo funcionário público e que possui a função de diretor. Realiza a administração do sistema.

O diagrama de casos de uso da Figura 2 mostra as funcionalidades do subsistema “solicitacaoReservaEspacoFisico” que é a parte do SCAEF acessada pelos usuários solicitantes. Esses usuários representados pelos atores (bonecos magros) são: Usuário não cadastrado, Usuário externo, Usuário estudante e Usuário funcionário público nível 1 e nível 2. As linhas que existem entre os atores e casos de usos (elipses) representam associações entre eles. As linhas que possuem setas significam generalizações/espacializações. A seta indica o ator geral (para qual a seta aponta) e o ator especializado (os que se encontram na outra extremidade da seta). Os atores especializados podem realizar suas funcionalidades e as herdadas do seu ator geral.

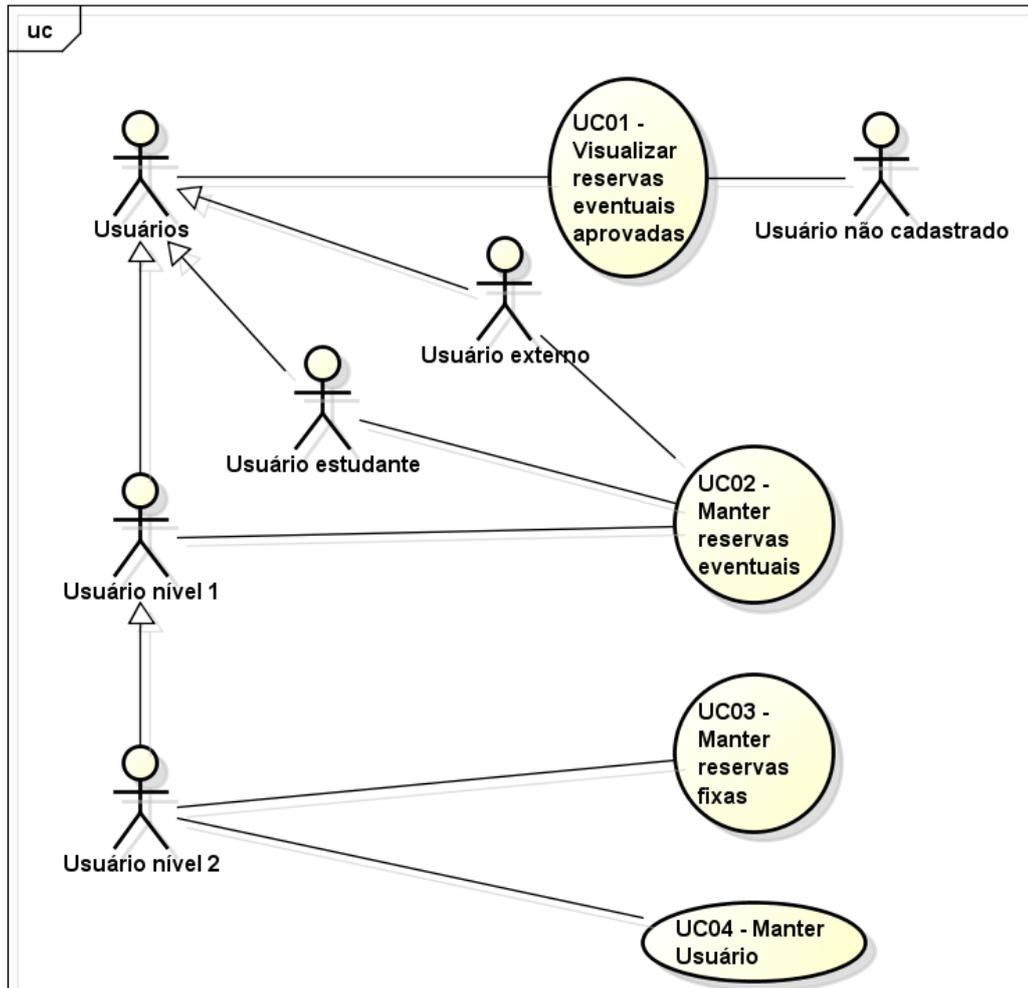


Figura 2 - Diagrama de Casos de Uso do Subsistema solicitacaoReservaEspacoFisico  
Fonte: O autor (2014)

O diagrama de casos de uso da Figura 3 mostra as funcionalidades do subsistema “controlePatrimonio” onde o usuário administrador pode realizar suas tarefas. O Usuário nível 2 e chefe do setor de patrimônio é o usuário administrador que está representado pelo ator. Os Usuários nível 3 e nível 4 realizam as mesmas atividades que o Usuário nível 2 e chefe de setor de patrimônio. Porém uma atividade feita por um usuário de nível maior não pode ser desfeita por um de nível menor. As linhas com setas entre os casos de uso significam generalizações/especializações e seguem a mesma lógica da explicação para atores. Os usuários administradores também podem realizar solicitações de reservas que foram mostradas no diagrama de casos de uso da Figura 2.

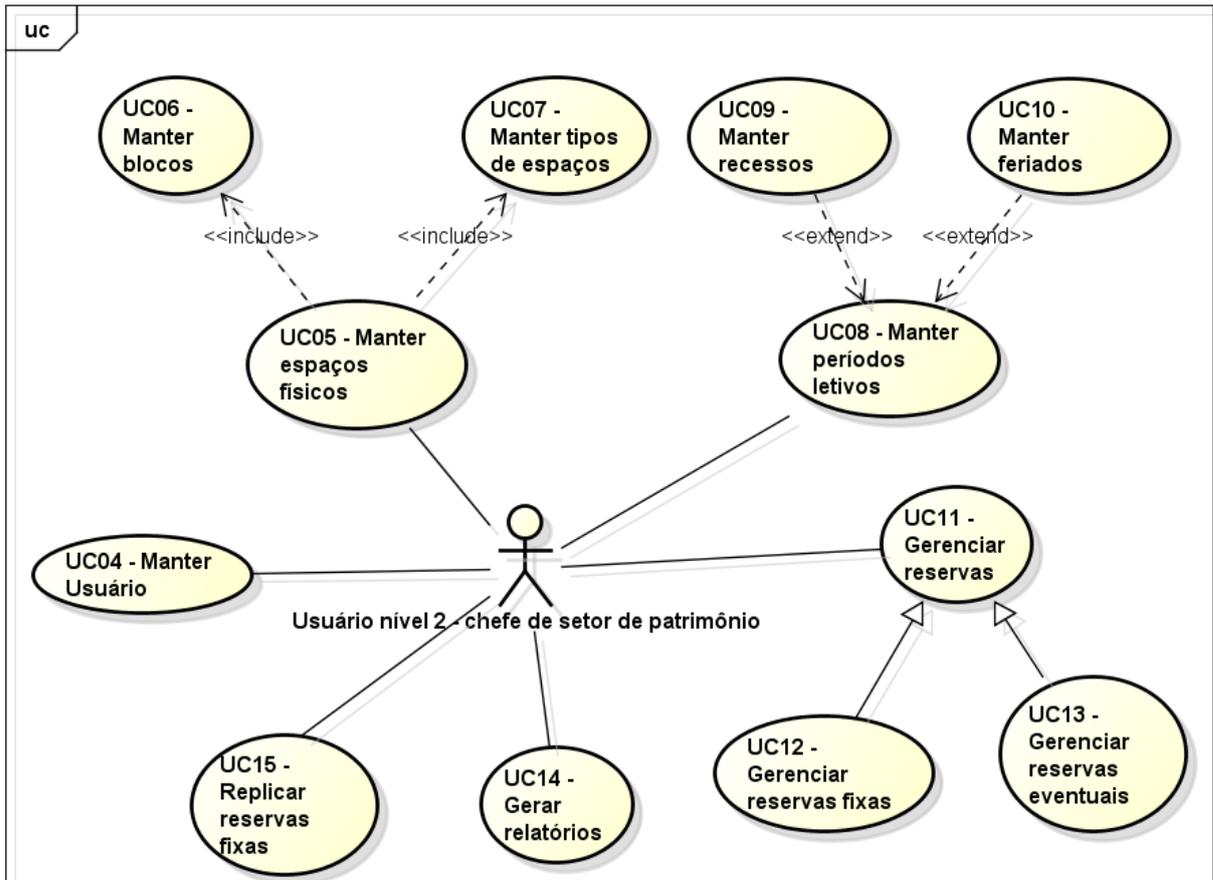


Figura 3 - Diagrama de Casos de Uso do Subsistema controlePatrimonio  
Fonte: O autor (2014)

A seguir são mostrados mais detalhes dos casos de uso das Figuras 2 e 3, explicitando os atores que estão associados ao caso de uso, uma descrição e os requisitos que esses estão relacionados:

- **UC01 - Visualizar reservas eventuais aprovadas:**
  - **Ator:** Usuários, Usuário não cadastrado, Usuário Externo, Usuário estudante, Usuário nível 1, Usuário nível 2;
  - **Descrição:** Pode-se visualizar reservas eventuais aprovadas;
  - **Requisitos:** RF08.
- **UC02 - Manter reservas eventuais:**
  - **Ator:** Usuário externo, Usuário Estudante, Usuário nível 1, Usuário nível 2, Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se solicitar, editar, excluir e listar reservas eventuais feitas pelo usuário;
  - **Requisitos:** RF06.

- **UC03 - Manter reservas fixas:**
  - **Ator:** Usuário nível 2, Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se solicitar, editar, excluir e listar reservas fixas feitas pelo usuário;
  - **Requisitos:** RF07.
- **UC04 – Manter usuário**
  - **Ator:** Usuário nível 2, Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir, listar e atribuir funções a usuários. Um usuário só pode cadastrar outro de nível inferior ao seu. Os usuários nível 3 e 4 podem cadastrar qualquer usuário. O usuário nível 2 pode cadastrar usuários que pertencem ao seu setor. O usuário nível 2 – chefe de setor de patrimônio pode cadastrar usuários do seu setor e usuários do tipo externo;
  - **Requisitos:** RF01.
- **UC05 - Manter espaços físicos:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar espaços físicos. Para cadastrar um espaço físico deve ser informado a qual bloco esse pertence e qual é o seu tipo;
  - **Requisitos:** RF04.
- **UC06 - Manter blocos:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar blocos;
  - **Requisitos:** RF04.
- **UC07 - Manter tipos de espaços físicos:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar tipos de espaços físicos;
  - **Requisitos:** RF04.

- **UC08 - Manter períodos letivos:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar períodos letivos. Pode-se associar a um período cadastrado feriados ou recessos que ocorrem dentro desse;
  - **Requisitos:** RF04.
- **UC09 - Manter recessos:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar recessos;
  - **Requisitos:** RF04.
- **UC10 - Manter feriados:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Pode-se cadastrar, editar, excluir e listar feriados;
  - **Requisitos:** RF04.
- **UC11 - Gerenciar reservas:**

Permite o gerenciamento das reservas onde o usuário pode visualizar, aprovar, desaprovar e editar uma reserva.

  - **UC12 - Gerenciar reservas fixas**
    - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
    - **Descrição:** Pode-se visualizar, aprovar, desaprovar e editar reservas fixas feitas por um usuário solicitante;
    - **Requisitos:** RF09.
  - **UC13 - Gerenciar reservas eventuais**
    - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
    - **Descrição:** Pode-se visualizar, aprovar, desaprovar e editar reservas eventuais feitas por um usuário solicitante;
    - **Requisitos:** RF08.
- **UC14 - Gerar relatórios:**

- **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
- **Descrição:** Permite gerar relatórios de ocupação de espaços tanto para reservas fixas como para reservas eventuais. Nas reservas fixas mostra a ocupação semanal do espaço. Nas reservas eventuais mostra uma lista com a data e horário que essas ocorrem;
- **Requisitos:** RF10.
- **UC15 - Replicar reservas fixas:**
  - **Ator:** Usuário nível 2 – chefe de setor de patrimônio, Usuário nível 3 e Usuário nível 4;
  - **Descrição:** Consiste em copiar uma reserva fixa aprovada de períodos anteriores;
  - **Requisitos:** RF11.

Com a utilização do SCAEF pretende-se deixar a tarefa de alocação de espaços físicos mais fácil, organizada e que elimine a utilização das fichas de papel. Esse Sistema possibilita a geração de relatórios de utilização dos espaços físicos e traz autonomia a seus usuários pelo fato de ser *web*, podendo ser acessível através da internet.

Para iniciar a utilização do SCAEF deve-se primeiro efetuar as atividades dos casos de uso UC04, UC05, UC06, UC07, UC08, UC09 e UC10. Após isso os usuários podem realizar reservas (conforme definição anterior, reservas fixas e eventuais). Reservas fixas, descritas no UC03, são as solicitadas por chefe de setores/coordenadores. Esses escolhem os horários semanais livres de uma sala para serem utilizados durante todo período letivo para aulas de um determinado curso. Reservas Eventuais, descritas no UC02, são aquelas que ocupam os espaços físicos ocasionalmente. Como exemplo desse tipo de reserva tem-se reuniões, grupos de estudos, palestras, minicursos e aulas eventuais. Esse tipo de reserva pode ser solicitada por docentes, discentes, técnicos administrativos e pessoas externas a instituição devidamente cadastradas. Em seguida essas reservas são avaliadas (UC11, UC12 e UC13) por usuários que possuem a função de administração do sistema.

Verificando-se as funcionalidades com os usuários através dos casos de usos é necessário construir o diagrama de classes. Esse diagrama apresenta as informações e suas relações, fornecendo uma visão estática de como estas estão estruturadas no SCAEF.

### 3.3 Diagrama de Classes

Com o diagrama de classes da Figura 4 é possível representar as informações necessárias ao sistema, dispostas nos atributos pertencentes às classes e como essas informações se relacionam através das associações entre as mesmas.

O sistema diferencia seus usuários pelo tipo que são e pela função que desempenham. No diagrama da Figura 4 as classes são representadas pelos retângulos com o seu nome na parte superior, seus atributos na parte central e métodos na parte inferior. As associações entre as classes são representadas pelas linhas que as interligam. A classe “Usuario” apresenta os dados dos usuários que serão mantidos. Possui relação com a classe “TipoUsuario” que é utilizada para diferenciar os usuários (funcionário público, externo, estudante). As classes “Cargo”, “Vigencia”, “Funcao” e “EstadoVigencia” são utilizadas para controlar as funções dos usuários do tipo funcionário público.

A classe “Reserva” foi concebida para controlar dados de reserva de alocação de espaço físico. Em associação com a classe “TipoReserva” é possível diferenciar as reservas em fixas ou eventuais. A classe “StatusReserva” controla os estados das reservas (que podem ser aprovado, em espera, cancelado). As reservas sempre ocorrem relacionadas a um espaço e um período. Por isso existe uma associação entre as classes “Reserva”, “PeriodoLetivo” e “Espaco”. A classe que se encontra no meio dessa associação mantém a data hora que a reserva ocorrerá. Isso é feito através do atributo “horario” que é do tipo *timestamp* (tipo que possui a capacidade de armazenar um valor de data e tempo, armazenando ano, mês, dia e hora da reserva).

Os recessos e feriados são colocados em classes sem associações, pois servem apenas para manter essas informações que podem ser consultados no momento de realizar as reservas.

O diagrama entidade relacionamento (APÊNDICE B) mostra como as informações representadas no diagrama de classes serão armazenadas no banco de dados.

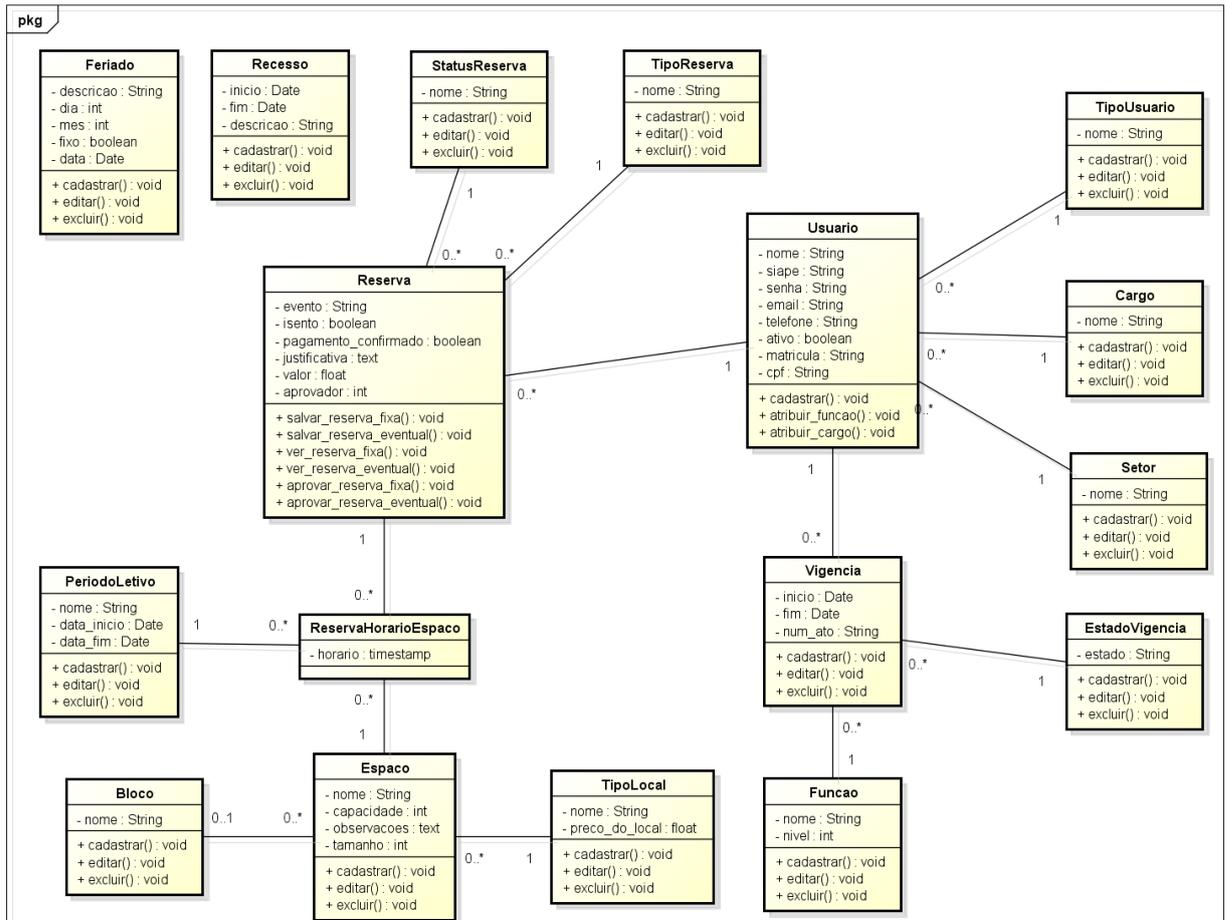


Figura 4 - Diagrama de classes

Fonte: O autor (2014)

Ainda na fase de especificação é possível a construção de mais diagramas da UML que mostrem outros aspectos dos sistemas. Um dos diagramas usado para retratar outro aspecto é o diagrama de atividades.

### 3.4 Diagrama de Atividades

O diagrama de atividades tem a função de mostrar os aspectos dinâmicos dos sistemas. Serão mostrados alguns diagramas de atividades que representam as principais atividades que os usuários podem realizar no SCAEF. Essas atividades são a solicitação de reserva eventual, solicitação de reserva fixa e gerenciamento dessas reservas. Outras atividades realizadas no SCAEF estão relacionadas a cadastramento e manutenção de dados que são utilizados tanto nas solicitações de reservas quanto no seu gerenciamento.

A solicitação de reservas eventuais, descrita no UC02, é mostrada no diagrama de atividades representado na Figura 5. A solicitação de reserva eventual segue os seguintes passos:

1. Inicia-se pela escolha do período que contém as datas que se deseja reservar;
2. Em seguida é escolhido o espaço que a ser reservado;
3. Após isso são mostradas informações sobre a ocupação da sala. São mostradas as reservas eventuais e fixas já aprovadas. E também os feriados e recessos;
4. Por último são preenchidos as datas e horários que se deseja reservar. Também são preenchidos o nome do evento e a justificativa da reserva. Após o fornecimento dessas informações a reserva é cadastrada.

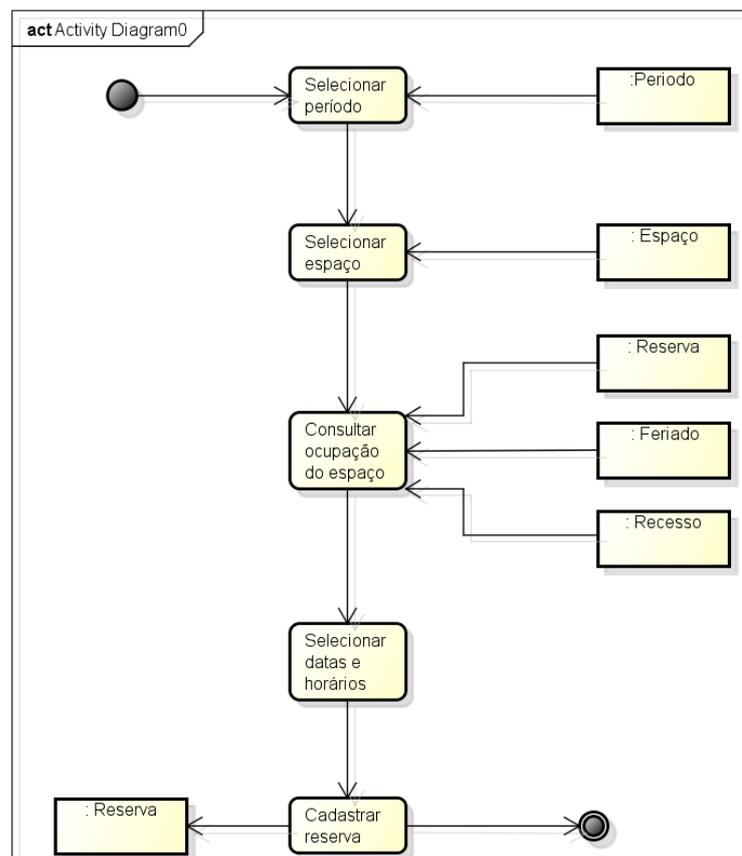


Figura 5 – Diagrama de atividades de solicitação de reserva eventual  
Fonte: O autor (2014)

A solicitação de reservas fixas, descrita no UC03, é mostrada no diagrama de atividades representado na Figura 6. A solicitação de reserva fixa segue os seguintes passos:

1. Inicia-se pela escolha do período ao qual se deseja realizar a reserva fixa;
2. Em seguida é escolhido o espaço a ser reservado;

3. O sistema permite que o usuário nível 2 realize apenas uma solicitação de reserva fixa por espaço em um período. Se o usuário já tiver realizado a solicitação é exibida uma mensagem e esse é obrigado a escolher outro espaço;
4. Após isso são mostrados os horários livres e ocupados que o espaço possui;
5. Por último são preenchidos as datas e horários que se deseja reservar. Após o fornecimento dessas informações a reserva é cadastrada.

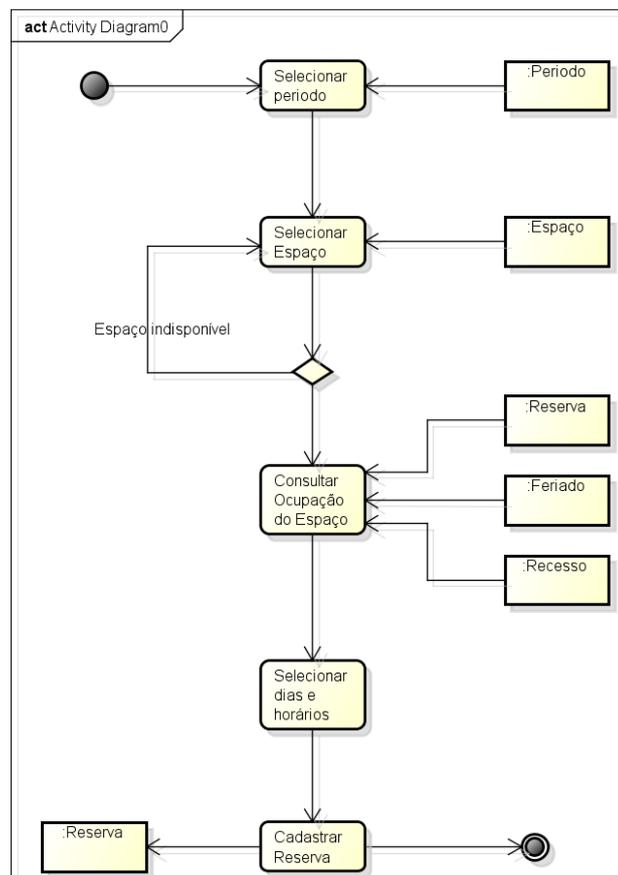


Figura 6 – Diagrama de atividades de solicitação de reserva fixa

Fonte: O autor (2014)

O processo de gerenciamento de reservas eventuais e fixas, descritas nos casos de uso UC11, UC12 e UC13, é mostrado no diagrama de atividades representado na Figura 7. Esses processos possuem os mesmos passos para serem realizados:

1. Inicia-se pela escolha da reserva que se deseja analisar;
2. Após a escolha da reserva sua análise poderá ser feita, verificando quais seus dias e horários (em caso de reserva fixa), ou (em caso de reserva eventual) serão verificadas suas datas e horários, bem como sua justificativa;

3. Caso o administrador desaprove a reserva, esse tem que preencher uma justificativa de desaprovação explicando os motivos dessa ação;
4. Caso o administrador aprove a reserva e houver choque de horário com outra reserva que foi realizada para o mesmo espaço, data e hora então o sistema preenche parte da justificativa de desaprovação com os choques de horários. O administrador pode editar o que o sistema preencheu ou acrescentar mais informações;
5. Caso a reserva não possua nenhum choque de horário com outra, então a reserva é aprovada.

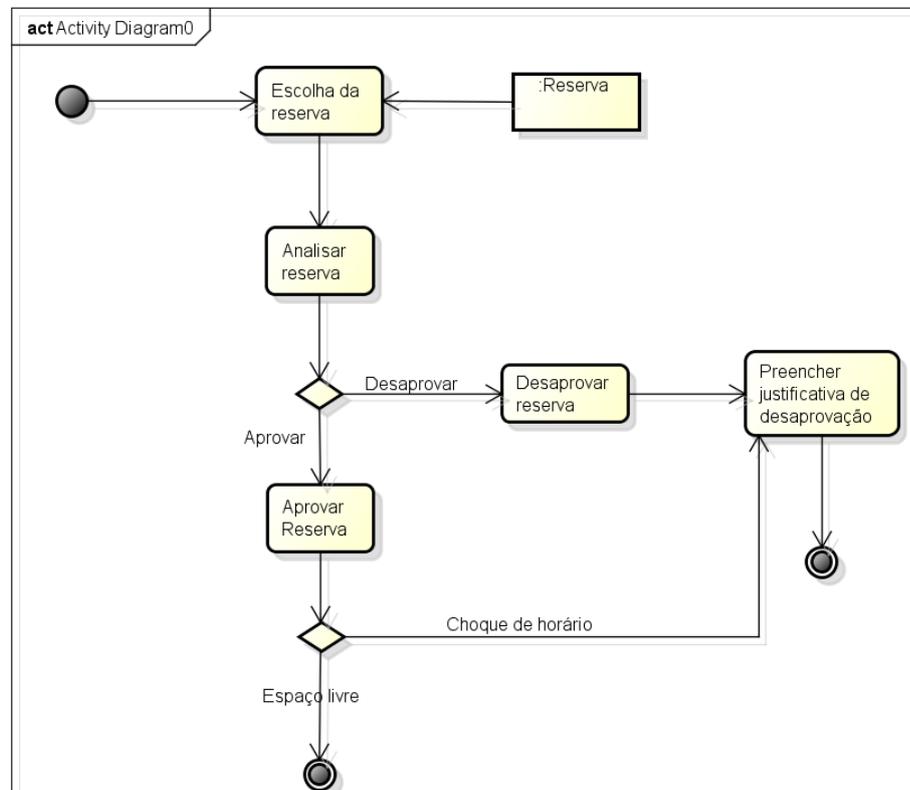


Figura 7 - Diagrama de atividades de gerenciamento de reservas  
 Fonte: O autor (2014)

## 4 FUNCIONAMENTO DO SISTEMA

Esse capítulo explica as principais funcionalidades do SCAEF. Serão mostradas as funcionalidades que os tipos de usuários possuem acesso. Também serão explicados os procedimentos de solicitação de reservas eventuais e fixas. E o procedimento de gerenciamento dessas reservas pelos administradores.

### 4.1 Usuários do sistema

Como mostrado no início do capítulo anterior o SCAEF é um módulo do SCAD. Os usuários cadastrados no SCAD podem ter acesso a três, dois ou a apenas um módulo. Esse acesso depende a qual tipo o usuário pertence. Os usuários do tipo externo possuem acesso apenas ao módulo de reservas de espaços físicos. Os usuários do tipo estudante possuem acesso ao módulo de reservas de espaços físicos e ao de alocação de veículos. Os usuários do tipo funcionário público possuem acesso ao módulo de reservas de espaços físicos, ao de alocação de veículos e ao de diárias.

Os usuários do tipo funcionário público recebem funções que definem quais atividades esses podem realizar (explicado detalhadamente na seção 3.2). Para acesso ao SCAD o usuário possui um *login* e senha. As funcionalidades que o usuário tem acesso aparecem no menu de acordo com seu tipo e a função que desempenha. Os tipos de usuários e os seus respectivos menus são mostrados nas Figuras 8 e 9. As opções dos menus pertencentes ao SCAEF serão explicitadas com o nome do módulo entre parênteses.

Os menus dos usuários solicitantes do SCAEF (que foram explicados na seção 3.2 e apresentados no diagrama de casos de uso da figura 2) são mostrados na Figura 8. O usuário externo, Figura 8(a), possui em seu menu “Minhas solicitações” apenas a opção referente a solicitação de reserva eventual de espaço físico (SCAEF) explicada no UC02 (seção 3.2). O usuário estudante, Figura 8(b), possui as opções de solicitação de reserva eventual de espaço físico (SCAEF) e solicitação de transporte. O usuário funcionário público e nível 1, Figura 8(c), possui um menu onde aparecem as opções de solicitação de reserva eventual de espaço físico (SCAEF), solicitação de transportes e solicitação de diárias. O usuário funcionário público nível 2, Figura 8(d), possui um menu onde aparecem as opções de solicitação de reserva eventual de espaço físico (SCAEF), solicitação de reserva fixa de espaço físico (SCAEF) explicada no UC03 (seção 3.2), solicitação de transportes e solicitação de diárias. Esse menu ainda possui outras opções referentes aos outros módulos do SCAD.



Figura 8 - Menus dos usuários solicitantes: (a) usuário externo, (b) usuário estudante, (c) usuário funcionário público nível 1, (d) usuário funcionário público nível 2

Fonte: O autor (2014)

Os menus dos usuários administradores do SCAEF (que foram explicados na seção 3.2 e apresentados no diagrama de casos de uso da Figura 3) são mostrados na Figura 9.

O usuário funcionário público nível 2 e chefe do setor de patrimônio, Figura 9(a), possui as opções de solicitação dos três módulos do SCAD. Possui o menu “Controle de Patrimônio” que apresenta as opções de administração do SCAEF. Esse menu contém as opções de gerenciamento explicadas nos casos de uso UC05, UC06, UC07, UC08, UC09, UC10, UC11, UC12, UC13 e UC15 (mostrados na seção 3.2).

O usuário funcionário público e nível 3, Figura 9(b), também possui as opções de solicitação dos três módulos do SCAD. Possui o menu “Controle de Patrimônio” além dos menus de controle do sistema de alocação de transporte e solicitação de diárias. O menu do usuário nível 4 é igual ao do usuário nível 3.

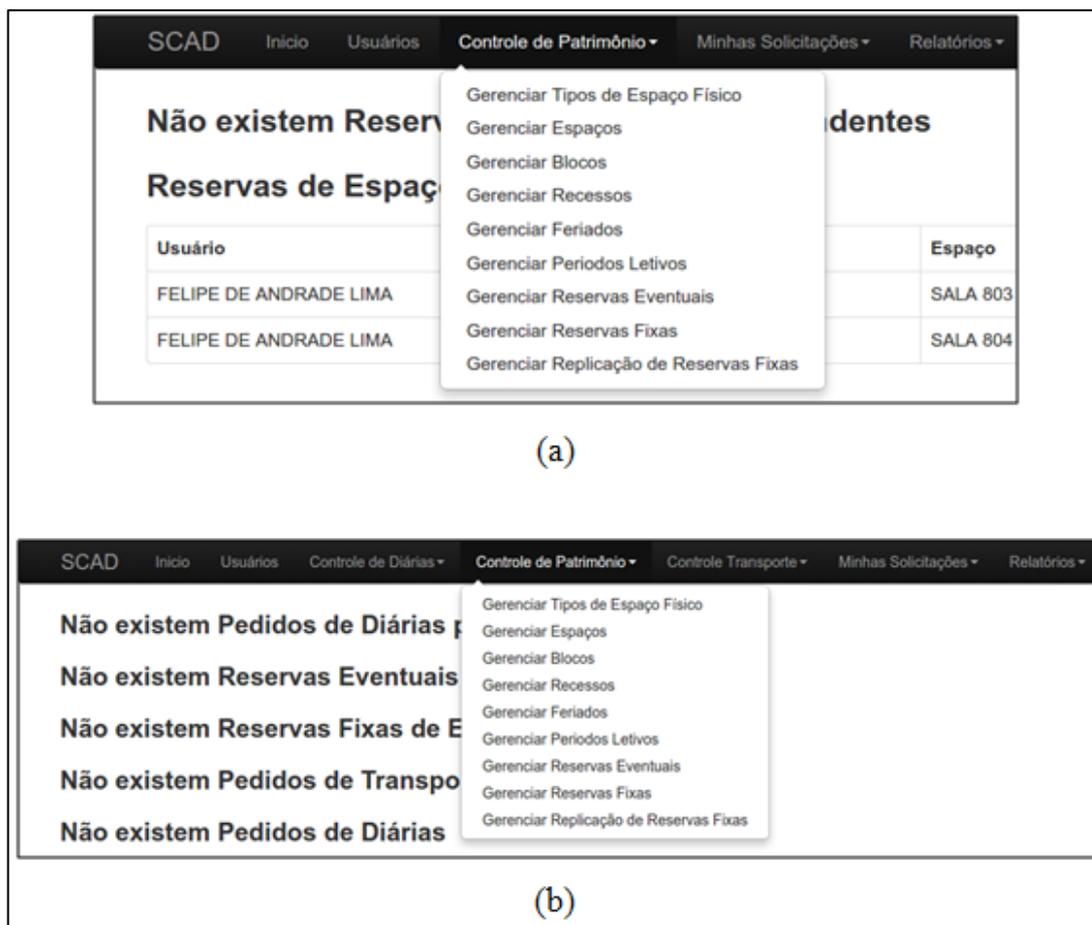


Figura 9 - Menus dos usuários administradores: (a) usuário funcionário público nível 2 e chefe do setor de patrimônio, (b) usuário funcionário público nível 3

Fonte: O autor (2014)

Após observar cada opção que os usuários possuem acesso, deve-se verificar os passos para realizar as principais funcionalidades que o SCAEF fornece (que são as solicitações de reservas eventuais e fixas).

## 4.2 Solicitação de reserva eventual

Existem vários tipos de usuários que podem solicitar reservas eventuais. Porém independente do tipo ou função do usuário o procedimento para se solicitar uma reserva eventual será o mesmo.

Antes de iniciar o processo de solicitação de reserva eventual o usuário tem acesso a uma página igual a da Figura 10 onde são mostradas as reservas já realizadas anteriormente e seus estados diferenciados por cor. Essas reservas podem ser visualizadas, excluídas ou editadas. A ação de editar só pode ser realizada se a reserva não tiver sido aprovada. Como descrito no diagrama de atividades da Figura 5 (seção 3.4) as primeiras ações no processo de

solicitação de uma reserva eventual são a escolha do período que possui a faixa de tempo de quando se deseja realizar a reserva e o espaço que será reservado.

Evento	Espaço	Período	Status da Reserva	Justificativa	Valor	Justificativa de desaprovação	Ações
Aulas extras	804	2014.2	CANCELADA	Exibir		Exibir	☰ ☒ 🗑
Apresentação de projetos	803	2014.2	EM ESPERA	Exibir			☰ ☒ 🗑
Aula de reposição	803	2014.2	APROVADA	Exibir			☰ ☒ 🗑
Reunião	803	2014.2	APROVADA	Exibir			☰ ☒ 🗑

Figura 10 - Listagem das reservas eventuais de um usuário  
Fonte: O autor (2014)

Após a seleção do período e do espaço é possível visualizar informações relativas a reservas fixas aprovadas (cor cinza), reservas eventuais aprovadas (cor amarela) e feriados/recessos (cor laranja) em uma página igual a Figura 11. Essas informações pertencem a um período completo, desde sua data inicial até a sua data final. Essa página tem a função de fornecer informações sobre horários livres e ocupados. E assim nortear o usuário na tomada de decisão de quais datas e horários escolher para reservar.

Dia da semana	Qua	Qui	Sex	Sáb	Dom	Seg	Ter	Qua	Qui	Sex	Sáb	Dom	Seg	Ter	Qua	Qui	Sex	Sáb	Dom	Seg	Ter
Horário	10/9	11/9	12/9	13/9	14/9	15/9	16/9	17/9	18/9	19/9	20/9	21/9	22/9	23/9	24/9	25/9	26/9	27/9	28/9	29/9	30/9
06:00 - 07:00																					
07:00 - 08:00																					
08:00 - 09:00																					
09:00 - 10:00																					
10:00 - 11:00	👤																				
11:00 - 12:00	👤																				
12:00 - 13:00																					
13:00 - 14:00																					
14:00 - 15:00																					
15:00 - 16:00																					
16:00 - 17:00																					
17:00 - 18:00																					
18:00 - 19:00																					
19:00 - 20:00																					
20:00 - 21:00																					
21:00 - 22:00																					

Figura 11 - Página com informações de reservas fixas e eventuais aprovadas, feriados e recessos  
Fonte: O autor (2014)

Quando o usuário decidir quais datas e horários poderão ser reservados, esse é levado a uma página igual a Figura 12. Essa página possui uma estrutura bem parecida com a página da Figura 11. As reservas fixas (cor cinza) e eventuais (cor amarela), já aprovadas, nessa nova página são representadas por *checkboxes* marcados, desabilitados e de cor cinza.

Uma facilidade que o SCAEF possui é poder realizar uma reserva que apresenta datas e horários bem flexíveis. Uma reserva pode ocorrer em uma data específica e em horários contínuos. Por exemplo, uma reserva para reposição de aula de uma determinada disciplina. A reserva também pode ocorrer em várias datas e em diferentes horários (tanto as datas como os horários podem ser contínuos ou não). Por exemplo, uma reserva para uma semana acadêmica de um curso.

Os horários e datas das reservas são escolhidos através dos *checkboxes* que se encontram desmarcados. As colunas que estes estão localizados representam as datas das reservas e as linhas representam os horários. Ao serem preenchidos os checkboxes, que correspondem as datas e horários da reserva, ficam na cor verde. Ainda é necessário preencher o nome do evento e a justificativa para a utilização do espaço. Ao ser salva a reserva é encaminhada para um usuário administrador poder analisá-la podendo aprová-la ou desaprovará-la.

The screenshot shows the SCAEF interface for creating a reservation. At the top, it displays 'SCAD' and 'Minhas Solicitações'. Below that, it shows the current period 'Período Letivo: 2014.2 SALA 803' and buttons for 'Salvar' and 'Voltar'. The main part of the interface is a calendar grid with columns for days of the week (Qua, Qui, Sex, Sáb, Dom, Seg, Ter) and rows for time slots (06:00 - 07:00, 07:00 - 08:00, 08:00 - 09:00, 09:00 - 10:00, 10:00 - 11:00, 11:00 - 12:00, 12:00 - 13:00, 13:00 - 14:00, 14:00 - 15:00, 15:00 - 16:00, 16:00 - 17:00, 17:00 - 18:00, 18:00 - 19:00, 19:00 - 20:00, 20:00 - 21:00, 21:00 - 22:00). Each cell in the grid contains a checkbox. To the right of the grid, there are input fields for 'Evento' and 'Justificativa'.

Figura 12 - Página de escolha dos horários e datas da reserva eventual  
Fonte: O autor (2014)

Uma tarefa complexa para o desenvolvimento dessa funcionalidade foi trabalhar com informações de várias tabelas do banco de dados do SCAEF. Essas informações são

recuperadas e devem ser organizadas para serem apresentadas conforme as Figuras 11 e 12. Quando há o preenchimento dos dados da reserva (como na Figura 12) é necessário tratar a relação existente entre datas e horários (respectivamente colunas e linhas) para assim realizar a inserção das informações, sobre cada horário da reserva, no banco de dados.

### 4.3 Solicitação de reserva fixa

As reservas fixas somente podem ser solicitadas por usuários do tipo funcionário público e que possui a função de nível 2 (chefes de setores/coordenadores). Antes do início do processo de solicitação de reserva fixa o usuário visualiza todas as reservas fixas já realizadas por ele e seus estados (aprovado, em espera e cancelado) em uma página igual a Figura 13. Nessa página assim como nas reservas eventuais o usuário pode visualizar, excluir ou editar reservas fixas. A ação de editar está acessível a partir da visualização da reserva e só pode ser realizada se a reserva não tiver sido aprovada. Conforme o diagrama de atividades da Figura 6 (seção 3.4) as primeiras ações para se realizar uma solicitação de reserva fixa são a escolha do período e em seguida o espaço que se deseja realizar a reserva.

Reserva	Espaço	Período	Status da Reserva	Justificativa de desaprovação	Ações
MATEMÁTICA 2015.1 803	803	2015.1	CANCELADA	Exibir	⋮ 🗑
MATEMÁTICA 2014.2 804	804	2014.2	APROVADA		⋮ 🗑
MATEMÁTICA 2014.2 803	803	2014.2	EM ESPERA		⋮ 🗑

Figura 13 - Listagem das reservas fixas de um chefe de setor  
Fonte: O autor (2014)

Após a seleção do período e do espaço é possível visualizar todos os horários e dias da semana que podem ser reservados. Essas informações são mostradas em uma página igual a Figura 14. As reservas fixas já aprovadas aparecem nessa página com o nome do curso/setor ao qual pertencem. Os horários que não possuem nenhum nome de curso podem ser reservados pelo usuário. Essas informações ajudam o usuário a verificar a disponibilidade do espaço no período e assim ajudar a decidir quais horários escolher.

SCAD Início Usuários Aprovar Solicitações de Diárias Minhas Solicitações - FRANCISCO RONALDO DE AZEVEDO - Sair							
Período Letivo: 2014.2 SALA: 803 Fazer Reserva Voltar							
Horário	Dom	Seg	Ter	Qua	Qui	Sex	Sáb
06:00 - 07:00	---	---	---	---	---	---	---
07:00 - 08:00	---	---	---	---	---	---	---
08:00 - 09:00	---	BIOLOGIA	BIOLOGIA	BIOLOGIA	---	BIOLOGIA	---
09:00 - 10:00	---	BIOLOGIA	BIOLOGIA	BIOLOGIA	---	BIOLOGIA	---
10:00 - 11:00	---	BIOLOGIA	---	---	BIOLOGIA	BIOLOGIA	---
11:00 - 12:00	---	BIOLOGIA	---	---	BIOLOGIA	BIOLOGIA	---
12:00 - 13:00	---	---	---	---	---	---	---
13:00 - 14:00	---	---	---	---	---	---	---
14:00 - 15:00	---	BIOLOGIA	BIOLOGIA	---	BIOLOGIA	BIOLOGIA	---
15:00 - 16:00	---	BIOLOGIA	BIOLOGIA	---	BIOLOGIA	BIOLOGIA	---
16:00 - 17:00	---	BIOLOGIA	---	---	BIOLOGIA	BIOLOGIA	---
17:00 - 18:00	---	BIOLOGIA	---	---	BIOLOGIA	---	---
18:00 - 19:00	---	SISTEMAS DE INFORMAÇÃO	SISTEMAS DE INFORMAÇÃO	SISTEMAS DE INFORMAÇÃO	SISTEMAS DE INFORMAÇÃO	---	---
19:00 - 20:00	---	SISTEMAS DE INFORMAÇÃO	---	---	---	---	---

Figura 14 - Página com reservas fixas aprovadas e horários livres

Fonte: O autor (2014)

Quando o usuário decidir quais dias e horários poderão ser reservados esse é levado a uma página igual a Figura 15. Essa página possui uma estrutura bem parecida com a página da Figura 14. As reservas fixas (cor cinza), já aprovadas, nessa nova página são representadas por *checkboxes* marcados, desabilitados e de cor cinza.

Os horários e dias das reservas são escolhidos através dos *checkboxes* que se encontram desmarcados. As colunas que estes estão localizados representam os dias da semana das reservas e as linhas representam os horários. Ao serem preenchidos os *checkboxes*, que correspondem aos dias da semana e horários da reserva, ficam na cor verde.

SCAD Início Usuários Aprovar Solicitações de Diárias Minhas Solicitações - FRANCISCO RONALDO DE AZEVEDO - Sair							
Período Letivo: 2014.2 SALA: 803 Salvar Voltar							
Horário	Dom	Seg	Ter	Qua	Qui	Sex	Sáb
06:00 - 07:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
07:00 - 08:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
08:00 - 09:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
09:00 - 10:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
10:00 - 11:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
11:00 - 12:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
12:00 - 13:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13:00 - 14:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
14:00 - 15:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
15:00 - 16:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
16:00 - 17:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
17:00 - 18:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18:00 - 19:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19:00 - 20:00	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figura 15 - Página de escolha dos dias e horários da reserva fixa

Fonte: O autor (2014)

Da mesma forma que a reserva eventual, a reserva fixa ao ser salva é encaminhada para um usuário administrador poder analisá-la, podendo aprová-la ou desaprová-la. Outro ponto de semelhança com a reserva eventual é a complexidade dessa funcionalidade em tratar a relação entre colunas e dias da semana, linhas e horários tanto no momento de exibir as informações como na inserção dos dados dos horários das reservas. Após o cadastramento tanto de reservas fixas como eventuais os usuários solicitantes precisam esperar a análise de um dos usuários administradores.

#### **4.4 Gerenciamento de reservas fixas e eventuais**

Os usuários administradores realizam várias atividades para a administração do SCAEF. Esses realizam as atividades descritas nos casos de uso UC04, UC05, UC06, UC07, UC08, UC09, UC10, UC14 e UC15. Uma das atividades principais e mais realizadas por esses usuários é o gerenciamento das reservas descritas nos casos de UC11, UC12 e UC13.

As reservas eventuais e fixas são gerenciadas separadamente. Porém como mostrado no diagrama de atividades da Figura 7 explicado na seção 3.4 possuem a mesma sequência de passos para serem gerenciadas.

A página para gerenciamento de reservas eventuais é igual a Figura 16. Mas ela será utilizada para detalhar tanto o processo de gerenciamento de reservas eventuais como fixas. Essa página possui uma listagem das reservas que estão sendo solicitadas por outros usuários. As mais recentes são sempre colocadas no topo da lista. Para cada reserva o usuário administrador pode realizar um conjunto de ações que são: visualizar a justificativa dada pelo usuário para a utilização da sala (no caso de reservas eventuais), visualizar as datas e horários (em caso de reserva eventual) ou dias da semana e horários (em caso de reserva fixa), aprová-la ou desaprová-la.

Após a análise de uma reserva um usuário administrador pode decidir aprová-la. Caso essa não possua choque de horários com outra reserva (anteriormente aprovada) o seu estado é apenas mudado para aprovado. Porém se houver algum choque de horários não será permitida a mudança do estado para aprovada. Em vez disso o sistema preencherá o formulário de desaprovação (Figura 17) com os horários e datas (no caso de reserva eventual) ou dias da semana e horários (no caso de reserva fixa) que sofreram choques e em seguida é mudado o estado da reserva para cancelado. O formulário, preenchido com os choques de horários, pode ser editado pelo administrador para acrescentar mais alguma informação para o usuário solicitante.

Usuário	Reserva	Espaço	Período	Status da Reserva	Justificativa	Justificativa de desaprovação	Opções
FRANCISCO RONALDO DE AZEVEDO	Reunião do colegiado	SALA 803	2014.2	EM ESPERA	Exibir		☰ ✓ ✕
FRANCISCO RONALDO DE AZEVEDO	Aula prática extra	SALA 803	2014.2	EM ESPERA	Exibir		☰ ✓ ✕
FELIPE DE ANDRADE LIMA	aulas de reposição	SALA 804	2015.1	APROVADA	Exibir		☰ ✓ ✕
MEDYNA	teste março 2	SALA 804	2015.1	APROVADA	Exibir		☰ ✓ ✕
MEDYNA	teste janeiro 2	SALA 804	2014.2	APROVADA	Exibir		☰ ✓ ✕
MARIA	teste janeiro	SALA 804	2014.2	APROVADA	Exibir		☰ ✓ ✕
MARIA	testes 2015.1 804	SALA 804	2015.1	APROVADA	Exibir		☰ ✓ ✕
FELIPE DE ANDRADE LIMA	aulas de reposição de cálculo	SALA 804	2014.2	EM ESPERA	Exibir		☰ ✓ ✕
FELIPE DE ANDRADE LIMA	orientação tcc	SALA 803	2014.2	APROVADA	Exibir		☰ ✓ ✕
FELIPE DE ANDRADE LIMA	Aulas extras	SALA 804	2014.2	CANCELADA	Exibir	Exibir	☰ ✓ ✕
FELIPE DE ANDRADE LIMA	Apresentação de projetos	SALA 803	2014.2	EM ESPERA	Exibir		☰ ✓ ✕

Figura 16 - Página de gerenciamento de reservas eventuais  
Fonte: O autor (2014)

Outra possibilidade é desaprovar uma reserva. Para efetuar a desaprovação é necessário ao usuário administrador preencher a justificativa de desaprovação (Figura 17) explicitando os motivos. Em seguida, o estado da reserva é mudado para cancelado.

SCAD
Início
Usuários
Controle de Patrimônio ▾
Minhas Sol

Esse espaço já está reservado

Justificativa de desaprovação eventual
Salvar
Cancelar

Justificativa

Houve os seguintes choques de horário:

11/9/2014 às 14,

12/9/2014 às 14,

Desenvolvido por GDIE (Grupo de Desenvolvimento de Informática e Eletrônica)

Figura 17 - Exemplo de choque de horário de reservas eventuais  
Fonte: O autor (2014)

Os usuários administradores podem ainda gerar relatórios. Existe um relatório para cada tipo de reserva (fixa ou eventual). O relatório de reserva fixa mostra os horários (das reservas aprovadas) ocupados do espaço em uma tabela igual a Figura 14 da seção 4.3. O relatório de reserva eventual mostra uma lista das reservas eventuais aprovadas (que encontram-se em uma faixa de datas informada pelo usuário) para um determinado espaço.

## 5 CONSIDERAÇÕES FINAIS

Esse trabalho foi desenvolvido com o objetivo de apresentar um sistema de controle de alocação de espaços físicos para Universidade Federal do Piauí – Campus Senador Helvídio Nunes de Barros. Esse sistema foi desenvolvido para a plataforma *web* e por isso permite seus usuários utilizá-lo de qualquer local que possua acesso à *internet*. Outra característica que o sistema possui é proporcionar um maior controle sobre como está a ocupação dos espaços físicos.

Foram demonstrados os passos para o desenvolvimento do sistema, começando pelo estudo do problema, verificando como as atividades relacionadas a alocação de espaços físicos eram realizadas. Em seguida, foram mostrados os métodos e tecnologias utilizadas para o desenvolvimento e, por último, foram mostradas as principais funcionalidades do sistema.

Até o presente momento, o projeto se encontra na fase de início de treinamento de usuários e, por isso, os dados obtidos como resultados são as descrições das funcionalidades que foram desenvolvidas e testadas para atendimento aos requisitos.

Em relação a trabalhos futuros, têm-se como objetivos, tornar o sistema responsivo para que esse possa se adaptar a telas de diferentes tamanhos como *Smartphones* e *tablets*. É necessário ainda implementar uma funcionalidade de envio de *e-mails* para os usuários solicitantes para que esses acompanhem o estado das suas reservas. Também podem ser utilizadas técnicas de *real time* para mudar a forma como o servidor do sistema se comunica e troca informações com as máquinas clientes.

## REFERÊNCIAS

- CAELUM, Apostila curso: **Desenv. Ágil para Web com Ruby on Rails, 2014**. Disponível em: <<http://www.caelum.com.br/apostila-ruby-on-rails/>> Acesso em: 26 de julho de 2014.
- FLANAGAN, David; MATSUMOTO, Yukihiro. **The Ruby Programming Language**. 1 ed. Sebastopol: O'Reilly Media, 2008.
- GARCIA, E; BAZZOTTI, C. **A importância do sistema de informação gerencial na gestão empresarial para tomada de decisões**. Ciências Sociais Aplicadas em Revista (Cascavel), v. 6, p. 33-60, 2006. Disponível em <<http://www.unioeste.br/campi/cascavel/ccsa/VISeminario/Artigos%20apresentados%20em%20Comunica%E7%F5es/ART%203%20%20A%20import%E2ncia%20do%20sistema%20de%20informa%E7%E3o%20gerencial%20para%20tomada%20de%20decis%E7es.pdf>>. Acessado em: 19 de julho de 2014.
- GUEDES, Gilleanes T. A. **UML 2 Uma abordagem prática**. 2 ed. São Paulo: Novatec, 2011.
- HARTL, Michael. **Ruby on Rails Tutorial: Learn Web Development with Rails**. 2 ed. Michigan: Addison-Wesley, 2012.
- LAUDON, Kenneth C; LAUDON, Jane P. **Sistemas de Informação Gerenciais Administrando a empresa digital**. 5 ed. São Paulo: Pearson Prentice Hall, 2004.
- MAGNO, Alexandre. **Mobile First Bootstrap** Develop advanced websites optimized for mobile devices using the Mobile First feature of Bootstrap. 1 ed. Birmingham: Packt Publishing, 2013.
- O'BRIEN, James A. **Sistemas de informação e as decisões gerenciais na era da internet**. 2 ed. São Paulo: Saraiva, 2006.
- PÁDUA, Wilson de. **Engenharia de Software: fundamentos, métodos e padrões**. [S.l]: LTC, 2000.
- PAULI, Josh. **Introdução ao web hacking** Ferramentas e técnicas para invasão de aplicações web. 1 ed. São Paulo: Novatec, 2013.
- PRESSMAN, Roger S. **Engenharia de Software: uma abordagem profissional**. 7 ed. Porto Alegre: AMGH, 2011.
- RODRIGUES, Andréa. **Desenvolvimento para Internet** Curitiba: LT, 2010.
- RUBY, Sam; THOMAS, Dave; HANSSON, David Heinemeier. **Agile Web Development with Rails**. 4 ed. Dallas: Pragmatic Bookshelf, 2011.
- RUBY-LANG.ORG. About Ruby. **Ruby - A Programmer's Best Friend**, julho 2014. Disponível em: <<http://www.ruby-lang.org/pt/about/>>. Acesso em: 06 de julho 2014.
- SABBAGH, Rafael. **SCRUM: Gestão ágil para projetos de sucesso**. São Paulo: Casa do Código, 2013.

SCHMITT, Christopher. **CSS Cookbook**. 3 ed. Sebastopol: O'Reilly Media, 2010.

SILVA, Alberto M. R; VIDEIRA Carlos A. Escaleira. **UML, Metodologias e Ferramentas CASE**. 1 ed. Lisboa: Centro Atlântico, 2001.

SOMMERVILLE, Ian. **Engenharia de Software**. 6 ed. São Paulo: Pearson Prentice Hall, 2005.

\_\_\_\_\_, Ian. **Engenharia de Software**. 9 ed. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, Lucas. **Ruby: Aprenda a programar na linguagem mais divertida**. São Paulo: Casa do Código, 2012.

SPURLOCK, Jake. **Responsive Web Development Bootstrap**. 1 ed. Sebastopol: O'Reilly Media, 2013.

STAIR, Ralph M; REYNOLDS, George W. **Princípios de Sistemas de Informação**. 6 ed. São Paulo: Pioneira Thomson Learning, 2006.

THOMAS, Dave. **Programming Ruby: The Pragmatic Programmers' Guide**. Dallas: Pragmatic Bookshelf, 2009.

VERSIONONE. **State of Agile survey – “the state of Agile development – 8th annual”**. 2013. Disponível em <<http://stateofagile.versionone.com/>>. Acessado em: 19 de julho de 2014.

**APÊNDICES**

## **APÊNDICE A – Entrevista com o Chefe de Setor do Patrimônio**

A seguir são mostradas as perguntas e respostas mais importantes realizadas nas entrevistas feitas com o chefe de setor do patrimônio.

### **1) Como é feito o controle de reserva atualmente?**

O controle de reserva é feito atualmente por meio de dois modelos de formulários onde são anotados os dados das reservas. Um modelo de formulário é para reservas dos auditórios, sendo um formulário para o auditório novo e outro para o auditório velho. Esse modelo deve ser preenchido com a data da reserva, o turno, horário, evento, responsável e status. O outro modelo de formulário é para reservas de salas e laboratório em geral. Nele são colocadas todas as reservas dos outros espaços da UFPI. Possui uma estrutura bem parecida com o modelo do auditório com os campos data, horário, responsável e status.

### **2) Quais as dificuldades de se trabalhar com a maneira de controle atual?**

A dificuldade está em gerenciar todo esse conjunto de formulários. Ter que trabalhar com planilhas separadas por ano. Verificar quais as reservas que foram canceladas ou modificadas pelas pessoas. A maneira de controle atual gera muito trabalho tanto para marcar como para procurar reservas.

### **3) Quem pode realizar solicitação de reserva?**

Todas as pessoas podem fazer solicitação de dentro e de fora da universidade. Discentes, Docentes e Técnicos Administrativos. O público externo pode realizar reserva mediante pagamento de uma taxa.

### **4) Quais os dias e horários que podem ser reservados?**

Os horários que podem ser reservados possuem a duração fixa de uma hora, podendo a pessoa reservar várias horas seguidas. Esses começam de 06:00 da manhã e se estende até 22:00 horas de segunda a sexta. Aos sábados de 06:00 às 18:00. E aos domingos não é possível realizar reservas, pois a UFPI é fechada.

Nos dias letivos, as salas de aula possuem horários que não podem ser reservados. Esses são os horários que elas estão ocupadas com aulas. Esses horários ocupados mudam de período para período, pois dependem das aulas das disciplinas. Esses horários são pedidos pelos coordenadores dos cursos antes do início de cada período e valem pelo período todo.

**5) Quais os espaços que podem ser reservados?**

Qualquer espaço da universidade pode ser reservado como auditório novo, auditório velho, salas, laboratórios, pátio. Porém é necessária a apresentação de uma justificativa informando qual o evento e qual a pessoa responsável.

**6) Como é feita a análise de solicitações de reservas?**

A análise é feita basicamente através da justificativa apresentada pela pessoa. Alguns espaços como salas aulas em horários que não está ocorrendo aula podem ser utilizados com a justificativa de reunião de grupo de estudo. Para os auditórios as reservas estão relacionadas mais a eventos.

**7) Existe um modelo de pedido de reserva?**

Não existe um modelo de pedido de reserva. Atualmente para se reserva um espaço o interessado tem que se dirigir até o setor de patrimônio e então são anotados os dados das reservas nos formulários. A pessoa tem que apresentar as informações básicas para a reserva: horário, data, espaço, responsável, evento e justificativa para então ser marcada a reserva.

**8) Nos feriados e recessos os espaços físicos podem ser utilizados?**

Podem ser utilizados dependendo de autorização da direção do Campus.

**9) Para uma mesma reserva existe uma quantidade máxima de dias e horas permitida?**

Não existe um limite para horas e dias de reservas dentro do bom senso. Existem reservas que ocupam somente 1 hora de uma sala em apenas uma data, como exemplo, uma sala ser utilizada por estudante para fazer trabalhos. Outras reservas ocupam um turno inteiro, como reuniões de conselhos, colegiados. Ainda existem reservas que podem ocupar vários dias e em horários diferentes, como por exemplo, reservas para semanas acadêmicas que se estendem por uma semana toda. O auditório pode ser utilizado todos os dias a noite para palestras e um ou dois dias de manhã ou a tarde ser utilizado para mesas redondas. E todos esses dias e horários pertencendo a uma mesma reserva do espaço.

**10) Quem são os responsáveis para dar acesso aos espaços físicos as pessoas? Como as informações das reservas são repassadas para eles?**

Os responsáveis por abrir e fechar os espaços físicos da universidade são os vigilantes e outros terceirizados que cuidam dos auditórios auxiliando com equipamentos de som e suporte aos eventos.

Para que os responsáveis pelos espaços fiquem sabendo quais as reservas que ocorrerão em um espaço e quem é o responsável por cada reserva, é necessário disponibilizar para eles uma lista com todas as informações das reservas. Por exemplo, para as reservas do sábado é deixada uma lista com os vigilantes na sexta a tarde.

Uma funcionalidade interessante que o sistema poderia ter seria a geração de um relatório que possuísse os dados das reservas dos espaços para então serem impressos e disponibilizados para os responsáveis pelos espaços.





**TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA  
“JOSÉ ALBANO DE MACEDO”**

**Identificação do Tipo de Documento**

- ( ) Tese  
( ) Dissertação  
( X ) Monografia  
( ) Artigo

Eu, **José Denes Lima Araújo**, autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de 02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação **Sistema Web para controle de alocação de espaços físicos da Universidade Federal do Piauí – Campus Senador Helvídio Nunes de Barros** de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título de divulgação da produção científica gerada pela Universidade.

Picos-PI 20 de janeiro de 2015.

  
Assinatura