

LIGA DE ENSINO DO RIO GRANDE DO NORTE
CENTRO UNIVERSITÁRIO DO RIO GRANDE DO NORTE
ESPECIALIZAÇÃO EM DESENVOLVIMENTO DE SISTEMAS CORPORATIVOS

RODRIGO REINALDO FERREIRA

**APLICABILIDADE DO GRAILS EM UM AMBIENTE DE DESENVOLVIMENTO
MULTI-PLATAFORMA**

NATAL/RN

2015

RODRIGO REINALDO FERREIRA

**APLICABILIDADE DO GRAILS EM UM AMBIENTE DE DESENVOLVIMENTO
MULTI-PLATAFORMA**

Trabalho de Conclusão de Curso
apresentado ao Centro Universitário do RN
como requisito final para obtenção do título
de Pós-Graduado em Desenvolvimento de
Sistemas Corporativos.

Orientador: Profº (Esp.) Romulo Fagundes
Cantanhede

NATAL/RN

2015

Catálogo na Publicação – Biblioteca do UNI-RN
Setor de Processos Técnicos

Ferreira, Rodrigo Reinaldo.

Aplicabilidade do grails em um ambiente de desenvolvimento multi-plataforma / Rodrigo Reinaldo Ferreira. – Natal, 2015.
54 f.

Orientador: Prof. Esp. Romulo Fagundes Cantanhede.

Relatório Técnico (Pós-Graduação em Desenvolvimento de Sistemas Corporativos) – Centro Universitário do Rio Grande do Norte.

1. Groovy and Grails - Monografia. 2. Linguagem Ágil - Monografia. 3. Migração de Tecnologias - Monografia. I. Cantanhede, Romulo Fagundes II. Título.

RN/UNIRN/BC

CDU 004

Dedico este trabalho à minha mãe, minha esposa, minha família que não mediram esforços para que eu realizasse mais um sonho. Aos meus amigos que me deram apoio, me incentivaram durante essa jornada e ao meu orientador que me acompanhou durante esse processo, me ajudando bastante a concluir esse trabalho.

AGRADECIMENTO

Agradeço em primeiro lugar a Deus, por permitir que eu chegasse até aqui, agradeço minha família, mãe, esposa, primos, tios a todos que estiveram presentes comigo em toda essa trajetória.

Agradeço aos meus amigos e companheiros de trabalho que foram essenciais para a realização desse projeto, ao Instituto Metrópole Digital que confiou no meu trabalho e me dando a oportunidade de trabalhar no Instituto, que está sendo de fundamental importância para o meu crescimento pessoal e profissional.

Aos educadores que foram de fundamental importância para a minha formação, em especial quero agradecer ao meu orientador o Professor Especialista Romulo Fagundes Cantanhede por todo o suporte e incentivo.

A todos os colegas de classe pela amizade e companheirismo que recebi e a todos que diretamente e indiretamente fizeram parte de minha formação, muito obrigados a todos vocês.

RESUMO

Este trabalho tem como objetivo explicar todo um processo de migração entre tecnologias, mostrando todos os pontos positivos e pontos negativos das tecnologias utilizadas no desenvolvimento de sistemas do instituto. Foi escolhido o Groovy and Grails como linguagem padrão do setor, então será mostrado aqui às vantagens de se trabalhar com uma linguagem ágil dinâmica para a plataforma Java, será apresentado as dificuldades de trabalhar com diversas linguagens em um setor desenvolvimento, e as dificuldade de encontrar profissionais para esse cenário e quais as vantagens de padronizar todos os sistemas para uma única linguagem. Para exemplificar todo esse processo, serão reportadas todas as etapas do desenvolvimento de um novo sistema, já utilizando o novo padrão definido para o setor, a linguagem ágil Groovy and Grails.

Palavras-Chaves: Groovy and Grails. Linguagem Ágil. Migração de Tecnologias.

ABSTRACT

This technical report has purpose to explain the whole process of migration between technologies, showing all the positives and negatives of each one technologies used in the development of the institute's systems. Groovy and Grails was chosen as default programming language of development sector, then will be shown here the advantages of working with an agile dynamic language for the Java platform, will be show the difficulties of working with several programming languages in a development sector and the difficulties to find professionals for this reality and what are the advantages to standardize all systems to a single programming language. To illustrate this whole process, will be reported all steps of development of a new system, already using the new programming language defined to development sector, the agile language Groovy and Grails.

Keywords: Groovy and Grails. Agile language. Technology migration.

LISTA DE FIGURAS

Figura 1 - Tela de acesso para o Moodle	21
Figura 2 - Tela inicial para usuários externos	23
Figura 3 - Tela de login do OPENRES.....	24
Figura 4 - Tela de login do LimeSurvey	25
Figura 5 - Tela para registro de ponto.....	26
Figura 6 - Tela de acesso a área administrativa	28
Figura 7 - Lista com os sistemas do Instituto.....	30
Figura 8 - Digrama de Casos de Uso	36
Figura 9 - Digrama de Classes	37
Figura 10 - Tela de login do sistema.....	39
Figura 11 - Tela inicial do sistema.....	41
Figura 12 - Tela cadastro de curso	42
Figura 13 - Tela cadastro da Disciplina.....	43
Figura 14 - Tela cadastro do tipo de atividade	44
Figura 15 - Tela cadastro para o servidor com perfil Coordenador de Curso	45
Figura 16 - Tela cadastro para o Nível de Ensino	46
Figura 17 - Tela cadastro para o Regime de Trabalho.....	47
Figura 18 - Tela cadastro para a Produção Intelectual	48
Figura 19 - Tela cadastro para criar tutoria.....	49
Figura 20 - Tela cadastro para criar Orientação	50
Figura 21 - Tela cadastro para criar a Colaboração em Projetos.....	51

LISTA DE TABELAS

Tabela 1 – Requisitos do Sistema.....	32
Tabela 2 – Requisitos suplementares	33
Tabela 3 – Papéis presentes no sistema	34
Tabela 4 – Funcionalidades presentes no sistema	34

LISTA DE SIGLAS E ABREVIACES

MVC	Model – View – Controller
UML	Unified Modeling Language
CRUD	Create, Read, Update e Delete
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SGBD	Data Base Management System
SQL	Structured Query Language
IMD	Instituto Metrpole Digital
SIGAA	Sistema integrado de Gesto de Atividades Acadmicas
UFRN	Universidade Federal do Rio Grande do Norte
LDAP	Lightweight Directory Access Protocol

SUMÁRIO

1 INTRODUÇÃO	12
2 OBJETIVOS	13
2.1 GERAL	13
2.2 ESPECIFICO	13
3 REFERENCIAL TEÓRICO	14
3.1 FRAMEWORK	14
3.2 MVC	14
3.3 UML	15
3.4 SCRUM	15
3.5 GRAILS	16
3.6 FRAMEWORK FRONT-END BOOTSTRAP	16
3.7 POSTGRESQL	17
3.8 PROGRAMAÇÃO POR CONVENÇÃO	18
4 CENÁRIOS	19
4.1 CENÁRIO ATUAL	19
4.1.1 Moodle	19
4.1.1.1 Bloco Genome	20
4.1.1.2 Bloco Matuto	20
4.1.2 Palestre	20
4.1.2.1 Criar evento	21
4.1.2.2 Editar evento	21
4.1.2.3 Deletar evento	22
4.1.2.4 Gerir presentes	22
4.1.2.5 Enviar E-mail inscritos presentes	22
4.1.2.6 Inscrição no evento	23
4.1.2.7 Download do certificado	23
4.1.2.8 Validar certificados	23
4.1.3 Openres	24
4.1.3.1 Reservas simples	24
4.1.3.2 Reservas múltiplas	24
4.1.4 Limesurvey	25

4.1.5 Pontoimd	26
4.1.5.1 Registrar ponto	27
4.1.5.2 Cadastrar usuários	27
4.1.5.3 Listar registros de ponto	27
4.1.5.4 Cadastrar setores	27
4.1.6 Seletor	28
4.1.6.1 Criar Edital	28
4.1.6.2 Verificar Inscrições	28
4.1.6.3 Validar Documentos	29
4.1.6.4 Realizar Inscrição	29
4.2 MOTIVAÇÃO	29
4.3 CENÁRIO PROPOSTO	30
5 ESTUDO DE CASO	32
5.1 LEVANTAMENTOS DE REQUISITOS	32
5.2 CASOS DE USO	34
5.3 DIAGRAMA DE CLASSES	36
6 RESULTADO	38
6.1 SISTEMA WEB	38
6.1.1 Tela de autenticação	39
6.1.2 Tela inicial do sistema	39
6.1.2.1 Topo	40
6.1.2.2 Menu lateral	40
6.1.2.3 Conteúdo principal	40
6.1.2.4 Rodapé	40
6.2 CASOS DE USO	41
6.2.1 UC01 - Manter Cursos	41
6.2.2 UC02 - Manter Disciplinas	42
6.2.3 UC03 - Manter Atividade Administrativa	42
6.2.4 UC04 - Manter Servidor	44
6.2.5 UC05 - Manter Nível de Ensino	45
6.2.6 UC06 - Manter Regime	46
6.2.7 UC07 - Manter Produção Intelectual	47
6.2.8 UC08 - Manter Tutoria	48

6.2.9 UC11 - Manter Orientação.....	49
6.2.10 UC12 - Manter Colaboração em Projeto	50
7 CONCLUSÃO	52
REFERÊNCIAS.....	53

1 INTRODUÇÃO

Os sistemas de informação são de suma importância para o desenvolvimento e sustentabilidade das corporações, são eles que asseguram a Confidencialidade, Integridade e Disponibilidade das informações, essas informações são de grande valia na administração.

Devido à grande demanda de informações foi exigido dos gestores uma melhor administração dessas informações, crescendo então a demanda dos sistemas de informação.

Tendo em vista todo esse cenário o Instituto Metrópole Digital criou o setor de desenvolvimento, que visa criar mecanismos para ter o total controle dessas informações, desenvolvendo sistemas de informação para gerir todas essas informações.

O setor de desenvolvimento do IMD foi fundado em 2013, surgiu com a necessidade de oferecer o suporte ao sistema Moodle utilizado pela instituição em seus cursos técnicos e também ao módulo do IMD localizado no SIGAA. Atualmente o setor é responsável pelo suporte de outros sistemas utilizados pela instituição e o desenvolvimento de novos sistemas, tendo essa como sua principal responsabilidade dentro do instituto.

Hoje o setor de desenvolvimento é responsável por vários sistemas, alguns foram desenvolvidos pelo setor, outros desenvolvidos por terceiros e utilizados pelo instituto, já que se trata de sistemas OpenSource, porém ambos estão sempre sendo atualizados com novas funcionalidades desenvolvidas pelo setor e também atualizado com soluções desenvolvidas para corrigir possíveis bugs.

2 OBJETIVOS

2.1 GERAL

Explicar a principal causa da mudança de tecnologia entre os sistemas do instituto e apontar o porquê da escolha da nova tecnologia a ser trabalhada, mostrando as vantagens e desvantagens da tecnologia atual e da tecnologia futura.

2.2 ESPECIFICO

- Descrever o processo de migração de tecnologias.
- Apontar as dificuldades encontradas durante o processo.
- Explicar como funciona e qual a utilidade de cada sistema.
- Mostrar os pontos fortes e fracos do cenário atual e do futuro cenário.

3 REFERENCIAL TEÓRICO

Alguns itens abordados a nesse trabalho faz-se necessário do leitor entender alguns conceitos para o melhor entendimento do trabalho e do desenvolvimento deste sistema. Para auxiliar o leitor esse capítulo tem como objetivo explicar alguns conceitos encontrados no desenvolvimento do Sistema de Estágio Probatório como também um maior entendimento a respeito de todos os sistemas encontrados no instituto.

3.1 FRAMEWORK

Framework de software pode ser definido como um conjunto de classes desenvolvidas em uma linguagem específica, com o objetivo de auxiliar no desenvolvimento de software.

A grande vantagem de trabalhar com framework é a agilidade no processo de desenvolvimento das aplicações, a possibilidade de reutilização de códigos poupará muito tempo e trabalho.

3.2 MVC

O MVC (Model – View – Controller) é um padrão de arquitetura que tem como objetivo separar a informação da interface em que o usuário interage. Esse modelo que arquitetura separa a aplicação em três partes distintas: o controle da informação (Model), Interface de interação com o usuário (View) e as regras de negócio (Controller).

Controle da informação (Model) - Camada que trabalha diretamente com as informações da aplicação, realiza a leitura e escrita de dados

Interface de interação com o usuário (View) - Essa camada é responsável pela interação com o usuário, essa camada apenas realiza a função de exibir as informações para o usuário, não realiza nenhuma persistência na base de dados.

Regras de negócio (Controller) – Essa camada tem como objetivo a unir as camadas Model e View, ela apenas define quais informações o sistema deverá resgatar da base de dados para se exibida na camada view, quais as informações serão persistidas, lembrando que ela não irar persistir apenas irar controlar essas

informações.

3.3 UML

UML (*Unified Modeling Language*) é uma linguagem de modelagem que permite representar a aplicação orientada a objeto que será desenvolvida, com o objetivo de auxiliar a tarefa de modelar, visualizar, documentar e especificar a aplicação, vale lembrar que a UML não é uma linguagem de desenvolvimento.

A UML possui nove diagramas que auxiliam nesse processo de documentar e modelar os diversos itens da aplicação, cada um desses diagramas tem sua importância no processo de desenvolvimento do software.

São diversas as vantagens que a modelagem proporciona para a aplicação a ser desenvolvida, podemos citar que com a modelagem desenvolvemos aplicações de forma mais rápida, eficiente e efetiva, permite a visualização e controle da arquitetura da aplicação, melhorando o entendimento do sistema que será desenvolvido. Mais a frente alguns dessas modelagens será exemplificada para a aplicação em questão.

3.4 SCRUM

O Scrum é um framework que tem como objetivo organizar e gerenciar os projetos a serem desenvolvidos, muitas vezes uma prática relacionada com o desenvolvimento ágil de um software, porém pode ser utilizado nos mais diversos tipos de projetos e nas mais diversas áreas já que sua principal função é a gestão dos projetos.

O Scrum oferece diversas vantagens para os seus utilizadores, com a adoção dele podemos reduzir os riscos de insucessos, aumentar a qualidade do produto a ser entregue, melhorar a produtividade da equipe e a visibilidade do progresso do projeto. A sua adoção não significa dizer que os problemas serão resolvidos, o

Scrum é apenas uma ferramenta que serve como uma referencia para a realização das atividades, entretanto o sucesso da ferramenta apenas será comprovada se bem utilizada.

3.5 GRAILS

Grails se trata de um framework para criação de aplicações web com a utilização da linguagem de programação Groovy e outros frameworks consagrados como o Hibernate, Spring e Sitemesh. O Framework desenvolvido em código aberto (open-source), e visa à alta produtividade graças ao paradigma de programação por convenção introduzida pelo Ruby on Rails. O Grails trás a proposta de trabalhar com a produtividade do Ruby on Rails, porém, na plataforma Java.

Trabalhando com o Grails o desenvolvedor se ocupa mais na implementação da aplicação do que na configuração, poupando um grande tempo, essa é uma das vantagens em relação ao Java, que ao iniciarmos um projeto Java é necessário realizarmos diversas configurações, no Grails após a realização do download, o ambiente já vem todo configurado.

A linguagem Groovy é uma linguagem Orientada a Objeto para a plataforma Java, aceita código Java, mas possui as mesmas características das linguagens Python, Ruby entre outras, pois se trata de uma linguagem ágil e dinâmica, porém fornece varias simplificações comparadas ao padrão Java, permite que classes e métodos sejam alterados em tempo de execução, não necessita de ponto e virgula para separar comandos, e algumas outras diferenças em relação ao Java.

O Grails trabalha com o recurso de Scaffolding, que é uma técnica que permite ao desenvolvedor criar as funcionalidades básicas como a inserção, exclusão, seleção e atualização de objetos (CRUD) de forma ágil e muito simples, em questão de minutos essas operações estão funcionando perfeitamente.

3.6 FRAMEWORK FRONT-END BOOTSTRAP

O Bootstrap pode ser considerado uma coleção de elementos, funções personalizadas para as aplicações web, esses elementos estão agrupados em única ferramenta, logo os desenvolvedores podem escolher quais são os elementos que farão parte da sua aplicação.

Foi desenvolvido em 2011, surgiu de uma necessidade da equipe de engenharia do Twitter em criar um padrão para as aplicações desenvolvidas por eles, já que não havia nenhuma definição de estrutura no momento, e cada desenvolvedor tinha seu estilo de codificação diferente, então para evitar possíveis

problemas no futuro esse framework foi desenvolvido, permitindo que a equipe trabalhasse com maior rapidez, eficiência e seguindo o mesmo princípio de codificação. O framework foi tão bem-sucedido que foi lançado a estrutura Bootstrap como projeto de software livre, e com o passar do tempo milhares de desenvolvedores passaram a utilizar esse framework e contribuir com aprimoramentos.

Os elementos contidos no Bootstrap foram criados utilizando o HTML, CSS e Javascript, esses elementos estão em constante aprimoramento, hoje o bootstrap está na versão 3.0, contendo diversas funções que permitem sua utilização em dispositivos móveis, e também possui todo o suporte para o desenvolvedor criar sua aplicação com o layout responsivo.

O Bootstrap possui diversas outras vantagens que fez dele um dos frameworks de front-end mais utilizado atualmente, ela possui uma interface amigável e moderna, possui diversos temas, uma grande variedade de plugins e possui em seu site uma ampla documentação.

3.7 POSTGRESQL

O PostgreSQL é um sistema de gerenciamento de banco de dados objeto relacional (SGBDOR), tem seu código fonte aberto, pode ser utilizado, modificado e distribuído por qualquer pessoa para qualquer finalidade, roda em todos os grandes sistemas operacionais, incluindo GNU/Linux, Unix, e MS Windows, suporta grande parte do padrão SQL, oferecendo diversos recursos:

- ✓ Consultas complexas.
- ✓ Chaves estrangeiras.
- ✓ Gatilhos.
- ✓ Visões.
- ✓ Indexação por texto.
- ✓ Integridade transacional.
- ✓ Controle de concorrência

O PostgreSQL surgiu em 1995 e derivou de um outro projeto que iniciou aproximadamente em 1976. Atualmente o PostgreSQL tem se ampliado para além da comunidade de software livre, já existe diversas corporações entre os usuários.

O PostgreSQL tem como principal concorrente o MySQL, ambos tem

desempenho similar, porém o PostgreSQL se comporta com melhor desempenho quando se precisa escalar em hardware multiprocessador.

3.8 PROGRAMAÇÃO POR CONVENÇÃO

A programação por convenção visa o desenvolvimento de software que busca diminuir o número de decisões que os desenvolvedores precisam tomar, com o objetivo de dar simplicidade e agilidade sem perder a flexibilidade.

Esse paradigma de programação aumenta a produtividade devido o fato do programador não se preocupar com configurações do sistema, apenas com a lógica de negócio, caso o desenvolvedor não queria configurar nada, basta seguir a convenção do framework, no Grails ele assume diversas configurações como, por exemplo, as de banco de dados, as de log de erro entre outras.

4 CENÁRIOS

4.1 CENÁRIO ATUAL

Logo mais você encontrará uma breve descrição dos sistemas que fazem parte do Setor de Desenvolvimento do IMD

4.1.1 Moodle

O Moodle (Modular Object-Oriented Dynamic Learning Environment, que significa Ambiente de aprendizado Modular Orientado ao Objeto), trata-se de um Ambiente Virtual de Aprendizagem, onde professores podem criar salas de estudos, inserir material didático para estudos, conteúdos em vídeo, animações, textos, abrir fóruns de discussões, criar wikis, chats, atividades para avaliar o aprendizado dos alunos e mais uma serie de outras funcionalidades.

Conforme mostrado na **figura 01** você visualizará uma tela encontrada no sistema.

Figura 1 – Tela de acesso para o Moodle



Fonte: Resultado da pesquisa realizada pelo próprio autor.

O Moodle é distribuído gratuitamente como software OpenSource, o download

do aplicação pode ser realizada no site <https://moodle.org>.

Foi desenvolvido na linguagem PHP, tendo sua persistência de dados em PostgreSQL, mais podemos trabalhar com outros SGBDs como o MySQL, SQLite e Oracle, roda em servidores APache e IIS e nos sistemas operacionais Linux, Windos e MAC OS X, a versão utilizada pela instituição é a 2.8 versão mais recente do Moodle.

A instalação padrão trás uma serie de blocos e plugins, mais nada impede que possamos baixar novos plugins ou blocos com funcionalidades que não existem na instalação padrão, podemos também criar nossos blocos e plugins com nossas funcionalidades. A equipe de desenvolvimento além de prestar todo o suporte necessário para o Moodle ainda desenvolve alguns blocos quando necessário. Dois blocos foram desenvolvidos no instituto, o bloco Genome (Gestor de Notas Metr pole Digital) e o bloco Matuto (Modulo de avalia  es dos tutores).

4.1.1.1 Bloco Genome

Esse bloco tem como principal objetivo, gerenciar as notas dos alunos do curso t cnico e enviar para o SIGAA, atrav s de uma sincroniza  o feita entre esses dois sistemas.

Para o gerenciamento de notas o Moodle divide todo o semestre por semanas, e atividades s o inseridas nessas semanas, dessa forma o Genome realiza a busca dessas atividades contabilizando a nota dos alunos, e tamb m realiza  o uma pesquisa pelas a  es dos alunos no ambiente virtual para gerar uma nota de participa  o virtual.

4.1.1.2 Bloco Matuto

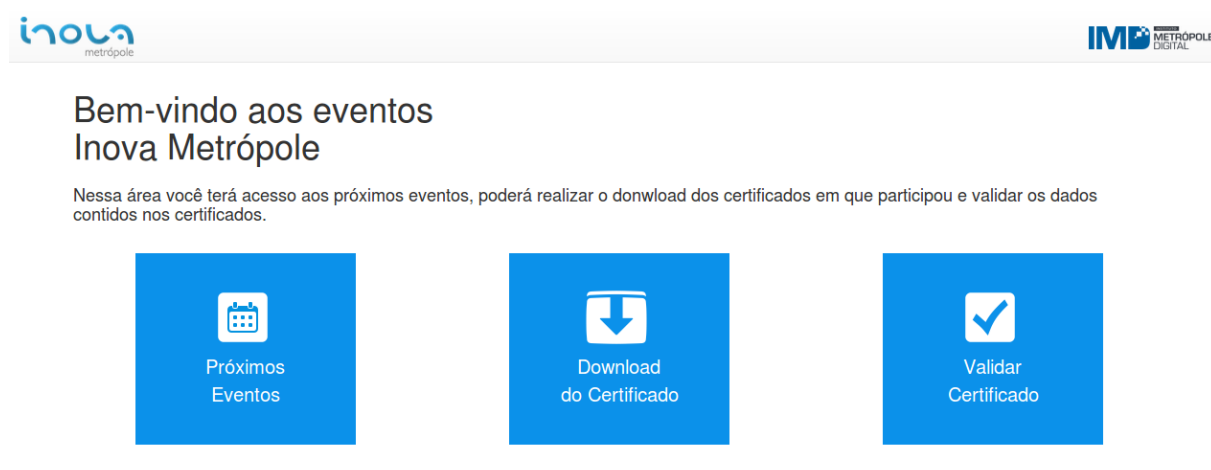
O bloco Matuto tem como principal objetivo verificar a participa  o do tutor com a sua turma, atrav s de a  es realizadas pelo tutor no Moodle, alguns dados sobre o tutor s o armazenadas para poss veis consultas, o bloco gerencia esses dados onde s o exibidos na tela em forma de gr ficos para o coordenados do curso, os dados s o exibidos na tela por per odos a escolha do coordenador.

4.1.2 Palestre

O sistema PALESTRE foi desenvolvido pelo instituto, tem como principal objetivo o gerenciamento dos eventos que são realizados pela incubada do instituto, o sistema surgiu para otimizar todo um processo que anteriormente era totalmente manual.

Conforme mostrado na **figura 02** você visualizará uma tela encontrada no sistema.

Figura 2 – Tela inicial para usuários externos



Fonte: Resultado da pesquisa realizada pelo próprio autor.

O Sistema foi desenvolvido em PHP e tem sua base de dados em PostgreSQL, roda em um servidor Apache, logo abaixo você encontrará uma breve descrição de algumas funcionalidades contidas no sistema.

4.1.2.1 Criar evento

Essa funcionalidade permite ao usuário administrador criar os eventos, depois de criados eles estarão disponíveis ao usuário comum para realizarem a inscrição, para a criar um evento é necessário informar algumas informações:

- ✓ Título do evento.
- ✓ Palestrante do evento.
- ✓ Descrição sobre a palestra.
- ✓ Dia de realização.

- ✓ Horário do evento.
- ✓ Valor do evento

4.1.2.2 Editar evento

Essa funcionalidade permite o usuário administrador editar um evento, acessando a área exclusiva do evento e escolhendo a opção editar evento, essa funcionalidade está disponível até a realização do evento.

4.1.2.3 Deletar evento

Essa funcionalidade permite o usuário administrador apagar um evento, acessando a área exclusiva do evento e escolhendo a opção deletar evento. Essa funcionalidade entra no mesmo critério do editar evento.

4.1.2.4 Gerir presentes

Essa funcionalidade permite o usuário administrador gerir os inscritos presentes ou ausentes no evento, pode ser executada de duas formas, a primeira é marcar a presença do inscrito no momento da realização do evento por uma funcionalidade de busca. A segunda opção é após o evento, através da lista com todos os inscritos o administrador deve marcar um a um setando como ausente ou presente. Após gerir todos os usuários o administrador dispara um e-mail para os presentes informando que o download do certificado já está disponível.

4.1.2.5 Enviar e-mail inscritos presentes

Essa funcionalidade permite o envio de e-mail para os inscritos presentes no evento, informando-o que o certificado já está disponível para download, essa opção só estará disponível após o usuário administrador gerir os inscritos.

Para o perfil usuário comum (usuário que está se inscrevendo no evento) as funcionalidades são:

4.1.2.6 Inscrição no evento

Essa funcionalidade permite o usuário se inscrever no evento, ao acessar a área de eventos o usuário pode visualizar uma lista com os próximos eventos, permitindo ao usuário realizar a inscrição, uma vez já inscrito em algum evento, seus dados ficam cadastrados na base de dados do sistema, então em um próximo evento no momento da inscrição o usuário não precisa cadastrar os campos novamente, o sistema realiza uma busca do usuário em questão, se encontrado o sistema retorna todos os campos preenchidos automaticamente.

4.1.2.7 Download do certificado

Essa funcionalidade permite ao usuário realizar o download do certificado, lembrando que o usuário deve ter comparecido ao evento. Acessando a área exclusiva para download o usuário irá visualizar todos os certificados que estão disponíveis para ele, permitindo o download.

4.1.2.8 Validar certificados

Essa funcionalidade permite qualquer pessoa verificar se algum certificado gerado pelo sistema está com os dados corretos, essa validação é realizada por um código de verificação que está impresso no certificado, esse código é único e é gerado pelo sistema. As informações contidas nessa validação são:

- ✓ Título da palestra.
- ✓ Carga horária do evento.
- ✓ Nome do palestrante.
- ✓ Nome do inscrito.

O sistema conta com a responsividade de seu Layout, sendo possível acessá-lo de dispositivo móvel sem deixarmos de ter acesso a informações disponibilizadas no site, desenvolvido com Bootstrap.

4.1.3 Openres

Sistema de Gestão dos Recursos do Instituto, que tem como principal

finalidade permitir uma gestão mais eficiente dos espaços físicos e os recursos do instituto, que possui uma ampla estrutura para abrigar os cursos nele oferecidos.

Conforme mostrado na **figura 03** você visualizará uma tela encontrada no sistema.

Figura 3 – Tela de login do OPENRES

Fonte: Resultado da pesquisa realizada pelo próprio autor.

O sistema foi desenvolvido no instituto, tem seu código aberto (OpenSource), foi desenvolvido em Ruby on Rails e com a persistência de dados em PostgreSQL.

Com o sistema podemos realizar reservas de salas e automóveis, checar as reservas realizadas. A reserva de salas está dividida em dois módulos, reservas simples e reservas múltiplas.

4.1.3.1 Reservas simples

O usuário poderá realizar apenas a reserva de uma determinada sala em um determinado horário, não existindo a possibilidade de realizar duas ou mais reservas de uma única vez.

4.1.3.2 Reservas múltiplas

O usuário pode realizar diversas reservas para uma determinada sala, essas reservas podem ter a duração de alguns dias como duração de meses, podendo

também intercalar apenas os dias que são de interesse do usuário.

A funcionalidade de verificar reservas, também trabalha com a idéia de reservas simples e múltiplas.

O sistema oferece um mapa onde podemos visualizar onde cada sala se encontra no instituto e verificar também as especificações das salas, as reservas de veículos podem ser reserváveis da mesma forma que a reserva de salas.

O sistema trabalha com diversos tipos de perfis, cada perfil possui sua importância no sistema, os perfis mais básicos podem apenas realizar as reservas, um perfil mais avançado pode realizar reservas e aprovar as reservas que já foram realizadas e o perfil completo que pode realizar as duas ações já citadas com outras ações como cadastrar usuários atribuir perfil a eles, criar salas, cadastrar veículos.

4.1.4 Limesurvey

Sistema tem como finalidade criar questionários para alunos, professores e colaboradores da incubação de empresas do instituto. Para criar esses questionários estamos utilizando o software livre LimeSurvey que anteriormente era conhecido como PHPSurveyor,

Conforme mostrado na **figura 04** você visualizará uma tela encontrada no sistema.

Figura 4 - Tela de login do LimeSurvey



Fonte: Resultado da pesquisa realizada pelo próprio autor.

O sistema foi desenvolvido em PHP e para a persistência de dados utiliza PostgreSQL e roda no servidor Apache. Atualmente a equipe de desenvolvimento do LimeSurvey está desenvolvendo o LimeSurvey 2.0, que está tendo sua base de dados totalmente reescrita, para essa versão será utilizado o padrão MVC e como framework web o CakePHP, outra mudança que poderemos perceber quando essa nova versão for lançada é que estará com uma interface gráfica mais acessível e totalmente modular, fatores que não estão presentes nessa atual versão.

A aplicação é bem completa, permite aos usuários administradores uma ampla variedade de recursos, como gerenciar questionários, gerar estatísticas com base nos resultados obtidos nos questionários, questionários que podem ser abertos ao público como também direcionado a um público específico, podemos configurá-lo a desejo do usuário, por exemplo, podemos especificar data para expirar, definir tipos de questões e outras diversas funcionalidades.

4.1.5 Pontoimd

O PontoIMD foi outro sistema que foi desenvolvido pelo setor de desenvolvimento do instituto, esse sistema oferece algumas funcionalidades para a gestão de horas dos bolsistas e tutores do instituto.

Conforme mostrado na **figura 05** você visualizará uma tela encontrada no sistema.

Figura 5 – Tela para registro de ponto



Fonte: Resultado da pesquisa realizada pelo próprio autor.

Foi desenvolvido em PHP e utiliza o PostgreSQL para a persistência de dados e roda no servidor Apache.

O sistema está dividido por módulos, cada um desses módulos tem suas funcionalidades, basicamente os módulos estão divididos em: Público, Comum, Gerenciar Setor, Gerenciar Unidades e Gerenciar Usuários, logo abaixo você poderá visualizar algumas funcionalidades encontradas no sistema.

4.1.5.1 Registrar ponto

Essa funcionalidade permite que usuários cadastrados possam realizar o registro no ponto, essa funcionalidade pertence ao módulo público.

4.1.5.2 Cadastrar usuários

Essa funcionalidade permite que usuários com perfil de coordenador de setor possam cadastrar usuários para o seu setor, essa funcionalidade pertence ao módulo comum.

4.1.5.3 Listar registros de ponto

Essa funcionalidade permite que usuários possam listar todos os registros de pontos de usuários que estão sobre sua jurisdição, essa listagem exibe os horários de entrada e saída desses usuários e podendo também verificar o total de horas trabalhadas por esse usuário. Essa funcionalidade pertence aos módulos Gerenciar Setor e Gerenciar Unidades.

4.1.5.4 Cadastrar setores

Essa funcionalidade permite que outros setores sejam cadastrados no sistema, funcionalidade que está presentes no módulo gerenciador de unidades.

O sistema PontoIMD surgiu com o principal objetivo ter o total controle de horas dos tutores e bolsistas do instituto, evitando assim possíveis conflitos no futuro.

4.1.6 Seletor

Por fim existe o Sistema de Gestão de Processos Seletivos, esse sistema é responsável pela gestão do processo seletivo da incubação de empresas do instituto.

Conforme mostrado na **figura 06** você visualizará uma tela encontrada no sistema.

Figura 6 - Tela de acesso a área administrativa do inscrito no processo seletivo

Fonte: Resultado da pesquisa realizada pelo próprio autor.

O sistema que foi desenvolvido em PHP e utiliza o PostgreSQL para a persistência de dados e roda em um servidor Apache, logo mais você encontrará as funcionalidades que estão disponíveis no sistema.

4.1.6.1 Criar Edital

Essa funcionalidade permite que o usuário administrador possa criar um edital para a realização de um novo processo de incubação.

4.1.6.2 Verificar Inscrições

Essa funcionalidade permite ao usuário administrador do sistema checar

quantas inscrições já foram realizar e quais foram esses inscritos.

4.1.6.3 Validar Documentos

Essa funcionalidade permite ao usuário administrador do sistema realizar a validação de todos os documentos enviados pelo candidato.

4.1.6.4 Realizar inscrição

Essa funcionalidade está voltada para o usuário que participará da seleção, ela permite que usuário interessados a se inscrever na incubação possa realizar sua inscrição e participar do concurso.

O principal objetivo desse sistema é realizar o gerenciamento do processo de incubação de empresas com uma maior eficiência, possibilitando ao usuário administrador uma maior facilidade e controle de todo o processo da seleção, otimizando todo o processo

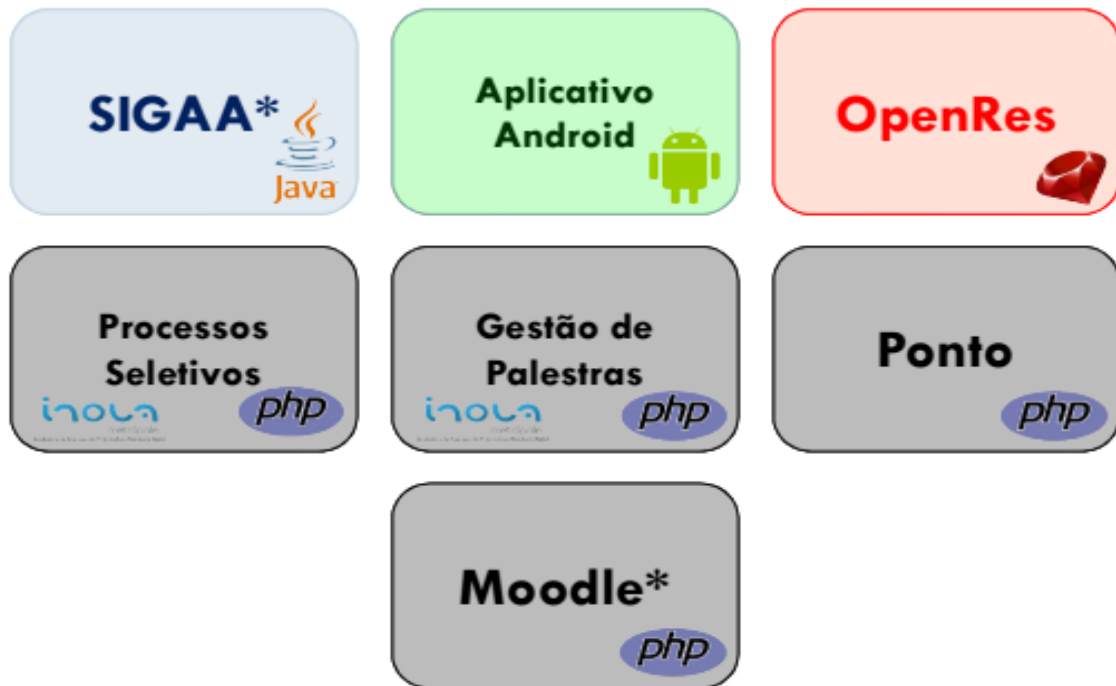
4.2 MOTIVAÇÃO

Dentre os sistemas que estão presentes no dia-a-dia do setor podemos destacar os que foram desenvolvidos pelo instituto, caso do Sistema de Ponto, Seleto, Palestre e OpenRes.

Como foi explicado durante o relatório o esses sistemas estão divididos entre as linguagens PHP, Ruby on Rails e Java, a diversidade de linguagens acaba trazendo alguns pontos negativos para o setor.

Na **figura 07** você poderá visualizar os sistemas encontrados no instituto e com a respectiva linguagem em que foi desenvolvido.

Figura 7 – Panorama atual dos sistemas do Instituto



Fonte: Resultado da pesquisa realizada pelo próprio autor.

Essa diversidade pode gerar uma dependência sobre algum membro da equipe e dificultar o gerenciamento dos sistemas, sendo necessários diversos profissionais com diversos perfis, uns familiarizados a um determinado sistema e se ausentando da implementação ou suporte dos demais sistemas com linguagens que não estão em seu perfil.

4.3 CENARIO PROPOSTO

O cenário proposto para o setor foi realizar uma migração de todos os sistemas para uma só linguagem de programação, que foi definida o Groovy com a utilização do framework Grails, assim criando uma equipe multidisciplinar, onde não existirá dependência sobre nenhum membro da equipe, e todos estarão aptos para trabalhar no desenvolvimento ou suporte de todos os sistemas da instituição.

Um fator decisivo para a escolha do Groovy on Grails foi a linguagem Java que é utilizada para o desenvolvimento de suas aplicações, Java que foi a linguagem adotada pela Universidade Federal do Rio Grande do Norte (UFRN) para seus sistemas.

Os novos projetos já estão sendo desenvolvidos nesse novo padrão. Para

exemplificar a utilização do framework foi escolhido um projeto que está em desenvolvimento no momento, será descrito todo o processo de desenvolvimento, citando as praticidades do framework e também todas as dificuldades encontradas durante o desenvolvimento.

5 ESTUDO DE CASO

Logo abaixo se descrito todos os passos que foram realizados para o desenvolvimento do sistema.

5.1 LEVANTAMENTOS DE REQUISITOS

Para essa fase do desenvolvimento do sistema foi realizada a técnica da entrevista, foram agendadas alguns reuniões com os interessados pelo sistema, para essa entrevista foi realizado um plano de entrevista para que não houvesse mudança de foco e que a entrevista não se torna muito longa e possivelmente cansativa.

Após todo esse processo os requisitos encontrados estão exibidos na **tabela 1.**

Tabela 1 - Requisitos do Sistema

Cód	Descrição
F01	O Sistema deve permitir o cadastro de Cursos
F02	O Sistema deve permitir o cadastro de Disciplinas
F03	O Sistema deve permitir o cadastro da Atividade Administrativa
F04	O Sistema deve permitir o cadastro do Servidor
F05	O Sistema deve permitir o cadastro do Nível de Ensino
F06	O Sistema deve permitir o cadastro do Regime
F07	O Sistema deve permitir o cadastro da Produção Intelectual
F08	O Sistema deve permitir a cadastro das Tutorias
F09	O Sistema deve permitir o cadastro da Ficha de Acompanhamento
F10	O Sistema deve permitir a cadastro do Plano de Trabalho
F11	O Sistema deve permitir o cadastro da Orientação
F12	O Sistema deve permitir o cadastro da Colaboração em projetos
F13	O Sistema deve permitir o envio da Plano de Trabalho
F14	O Sistema deve permitir o envio da Ficha de Acompanhamento
F15	O Sistema deve permitir o envio de relatórios pelos Coordenadores de cursos e pelos Integrantes da comissão avaliativa
F16	O Sistema deve permitir o envio do parecer do Tutor
F17	O Sistema deve permitir o envio do parecer do final, que é o parecer da Chefia
F18	O Sistema deve permitir o cadastro de novos usuários no sistema

Fonte: Resultado da pesquisa realizada pelo próprio autor.

Para os requisitos não funcionais levamos em consideração alguns itens necessários para o sucesso de uma aplicação, tais como desempenho, usabilidade e confiabilidade.

O resultado dessa coleta de requisitos pode ser encontrado na **tabela 02**.

Tabela 2 – Requisitos suplementares

Cód	Nome	Restrição	Categoria	Desejável	Permanente
RNF01	Tipo de Interface	As interfaces da aplicação devem ser desenvolvidas para que o usuário possa acessar pelo browser ou por qualquer dispositivo móvel sem perder nenhuma informação	Interface	()	(x)
RNF02	Armazenamento de dados	A persistência de dados deverá ser implementada utilizando o SGBDOR, PostgreSQL seguindo assim o padrão dos demais sistemas	Persistência	()	(x)
RNF03	Perfil de usuário	Os perfis para acesso ao sistema são: Chefia, Coordenador de curso, Membro da comissão avaliativa, Tutor, Professor em estágio Probatório e Administrador do sistema. Cada perfil contendo suas funcionalidades e acesso a algumas áreas específicas do sistema.	Segurança	()	(x)

Fonte: Resultado da pesquisa realizada pelo próprio autor.

Nessa fase também foram encontrados quais seriam os papéis que irão fazer parte do sistema, e quais as permissões e funcionalidades que serão associadas a determinado papel.

O resultado dessa coleta de requisitos pode ser encontrado na **tabela 03**.

Tabela 3 – Papéis presentes no sistema

Papel	Descrição	Permissões
Chefia	Responsável por emitir o parecer final do servidor	Perfil com permissões gerais no sistema
Coordenador de Curso	Responsável por gerar relatórios com atividades e atribuições do servidor em questão	Perfil com permissões gerais sobre o módulo de Ensino
Membro Comissão Avaliativa	Responsável por gerar relatórios com atividades e atribuições do servidor em questão	Perfil com permissões para visualizar avaliações dos servidores e emitir seus relatórios com informações de atividades executadas e atribuições
Tutor	Responsável por gerar os parecer do servidor em estágio probatório e encaminhar as avaliações para a chefia	Perfil com permissões para visualizar avaliações dos servidores e emitir seus relatórios com informações de atividades executadas e atribuições
Servidor Estágio Probatório	Responsável pelo preenchimento das avaliações periódicas e após isso enviar a avaliação para o tutor	Perfil com permissões básicas, apenas responder as avaliações e enviar para o tutor
Administrador do sistema	Responsável por manter o sistema	Perfil com permissões gerais no sistema

Fonte: Resultado da pesquisa realizada pelo próprio autor.

5.2 CASOS DE USO

Os casos de uso foram desenvolvidos através da coleta de requisitos que foram identificados em reuniões com interessados no sistema, os casos de usos definidos para o sistema estão descritos na **tabela 4**.

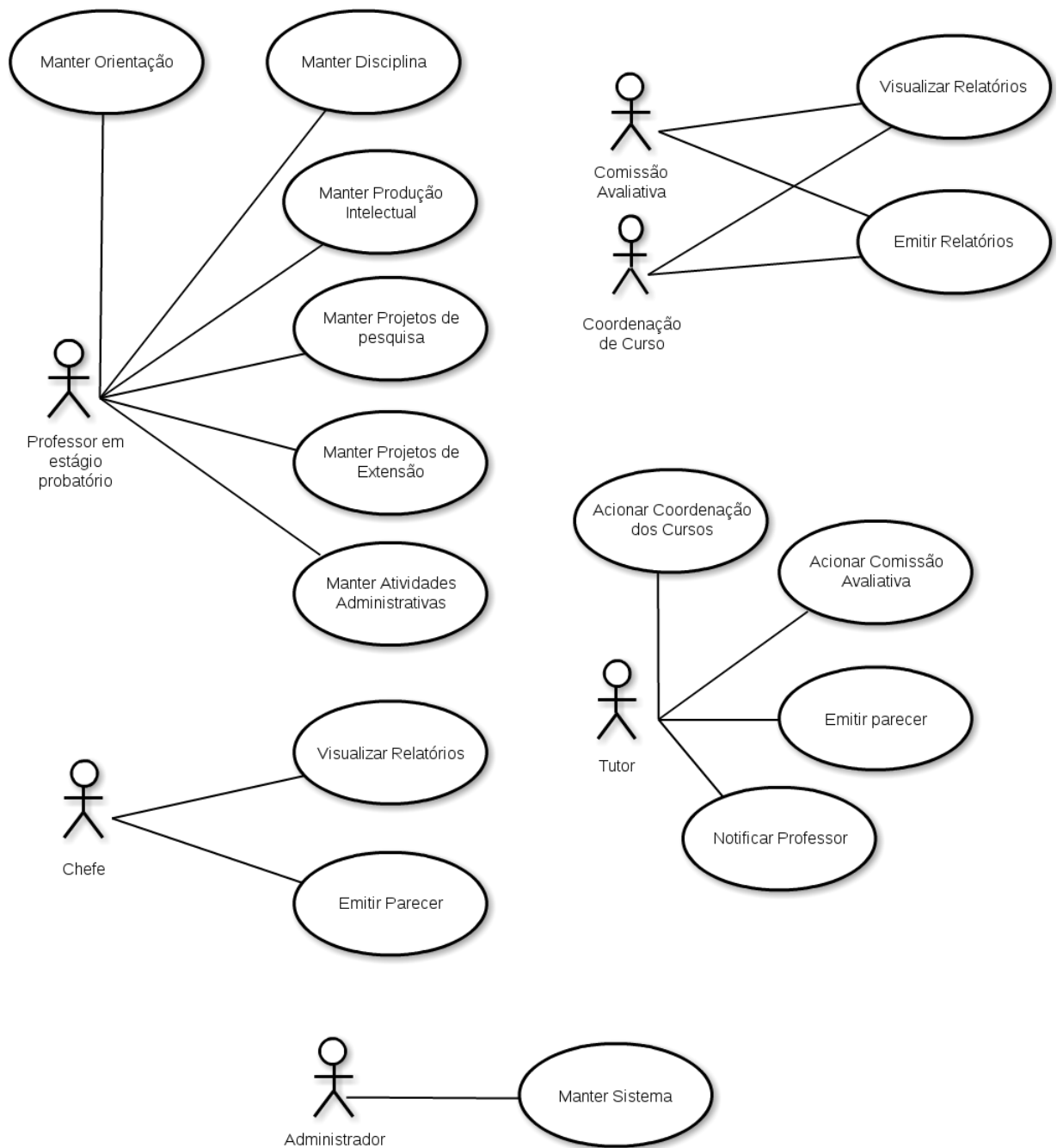
Tabela 4 – Funcionalidades presentes no sistema

Cód	Caso de Uso	Descrição
UC01	Manter Cursos	O Sistema deve permitir o cadastro de Cursos
UC02	Manter Disciplinas	O Sistema deve permitir o cadastro de Disciplinas
UC03	Manter Atividade Administrativa	O Sistema deve permitir o cadastro do Tipo de Atividade Administrativa
UC04	Manter Servidor	O Sistema deve permitir o cadastro do Servidor
UC05	Manter Nível de Ensino	O Sistema deve permitir o cadastro do Nível de Ensino
UC06	Manter Regime	O Sistema deve permitir o cadastro do Regime
UC07	Manter Produção Intelectual	O Sistema deve permitir o cadastro do Tipo de Produção Intelectual

UC08	Manter Tutoria	O Sistema deve permitir a cadastro das Tutorias
UC09	Manter Ficha de Acompanhamento	O Sistema deve permitir o cadastro da Ficha de Acompanhamento
UC10	Manter Plano de Trabalho	O Sistema deve permitir a cadastro do Plano de Trabalho
UC11	Manter Orientação	O Sistema deve permitir o cadastro da Orientação aos Alunos
UC12	Manter Colaboração em Projeto	O Sistema deve permitir o cadastro da Colaboração em projetos de Pesquisa
UC13	Enviar Plano de Trabalho	O Sistema deve permitir o envio da Plano de Trabalho
UC14	Enviar Ficha de Acompanhamento	O Sistema deve permitir o envio da Ficha de Acompanhamento
UC15	Enviar Relatórios	O Sistema deve permitir o envio de relatórios pelos Coordenadores de cursos e pelos Integrantes da comissão avaliativa
UC16	Enviar o Parecer do Tutor	O Sistema deve permitir o envio do parecer do Tutor
UC17	Enviar o Parecer Final	O Sistema deve permitir o envio do parecer do final, que é o parecer da Chefia
UC18	Manter Usuários	O Sistema deve permitir o cadastro de novos usuários no sistema

Fonte: Resultado da pesquisa realizada pelo próprio autor.

Após a finalização da coleta de requisitos, identificamos que o sistema estará dividido em cinco módulos, o Administrador do Sistema, Chefia, Comissão Avaliativa / Coordenadores de Curso, Tutor e o Professor em Estágio Probatório, sendo assim o diagrama de Casos de uso pode ser representado da seguinte forma, descrito na **figura 8**.

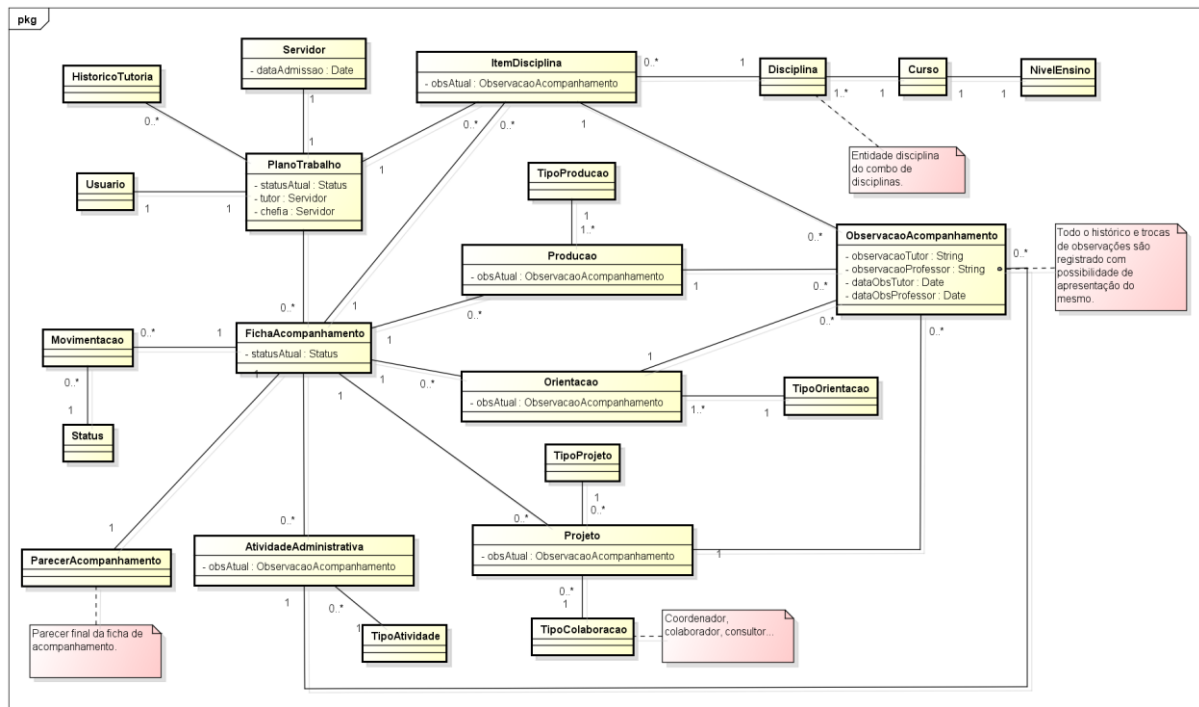
Figura 8 – Digrama de Casos de Uso

Fonte: Resultado da pesquisa realizada pelo próprio autor.

5.3 DIAGRAMA DE CLASSES

Uma das modelagens mais úteis para o sucesso da aplicação tem como objetivo definir todas as classes que a aplicação necessita possuir e serve de modelo para os objetos. Na **Figura 9** exemplifica esse diagrama de classes

Figura 9 – Digrama de Classes



Fonte: Resultado da pesquisa realizada pelo próprio autor.

6 RESULTADO

Com a implantação do sistema em questão, todo o processo de avaliação em que os servidores nomeados se submetem será automatizado, otimizando o fluxo de informações permitindo maior organização e agilidade nos processos, acesso as informações de forma simples e rápida, com maior integridade dessas informações e com mais segurança.

6.1 SISTEMA WEB

O sistema disponibiliza meios para os servidores realizarem as avaliações, que serão realizadas semestralmente, e se estenderá por todo o período do estágio probatório, equivalente há 36 meses e sendo realizadas 05 avaliações.

Após a realização dessas avaliações as informações estarão organizadas no sistema divididas em módulos, esses módulos estão divididos da seguinte forma:

- ✓ Disciplina e atividades desempenhadas
- ✓ Orientação aos alunos
- ✓ Produções Intelectuais
- ✓ Projetos de Pesquisa
- ✓ Projetos de Extensão
- ✓ Atividades administrativas

Todos esses itens fazem parte da ficha de avaliação do servidor, ele terá que cadastrar todos os itens dessa lista, caso tenha realizado alguma ação que faz parte desses itens a serem cadastrados.

Após o preenchimento de todos esses itens a ficha é enviada para o tutor, que por sua vez pode entrar em contato com a Coordenação do Curso e Comissão Avaliativa para solicitar relatórios sobre as ações desse servidor e verificar se os dados cadastrados estão coerentes, se não, o tutor entrará em contato com o servidor através que uma notificação no sistema e solicitará a correção do item.

O servidor em avaliação recebe a notificação e realizará a correção e novamente envia a lista para o tutor, que refaz as ações anteriores para verificação dos dados cadastrados, após essa verificação e a constatação que os dados estão corretos o tutor emite o parecer de aprovação da ficha do servidor, esse processo de verificação pode acontecer quantas vezes for necessário.

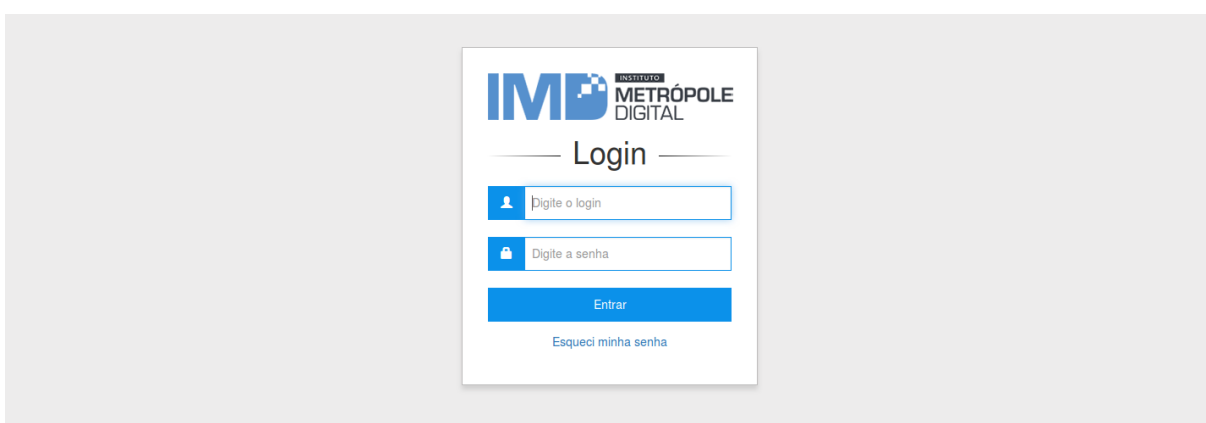
Quando o tutor emitir o parecer a chefia é notificada, a chefia é responsável pelo parecer final, esse parecer final determina a aprovação ou não do servidor em questão.

O Layout do sistema teve como objetivo deixar o sistema mais com uma ótima usabilidade, acessível e intuitivo, com o layout responsivo será possível acessar de dispositivo móvel sem nenhuma dificuldade. Algumas das telas do sistema podem ser visualizadas a seguir.

6.1.1 Tela de autenticação

A **figura 10** exibe a tela para realizar a autenticação no sistema, para ter acesso ao sistema é necessário que o usuário tenha cadastro no LDAP, então o sistema solicita o usuário e senha para o usuário se autenticar, o Usuário e Senha são os mesmos cadastrados inicialmente no LDAP, a tela para autenticação também oferece aos usuários a funcionalidade para recuperar senha, caso o usuário não consiga lembrar quais foram as informações cadastradas no LDA, em casos de login ou senha não estiverem corretos e a autenticação não for bem sucedida o sistema exibe uma mensagem de erro na tela.

Figura 10 – Tela de login do sistema



Fonte: Resultado da pesquisa realizada pelo próprio autor.

6.1.2 Tela inicial do sistema

A tela inicial do sistema apresenta a seguinte estrutura.

6.1.2.1 Topo

Exibe a logo do sistema alinhado a esquerda, alinhado a direita o sistema exibe informações do usuário ligado, essas informações são: Nome do usuário e perfil do usuário. Nessas informações é possível abrir um sub-menu com opções para editar o perfil ou realizar o logout.

6.1.2.2 Menu lateral

Exibe um menu com todas as funcionalidades disponível para o usuário logado, essas funcionalidades podem alterar de acordo com as permissões dos usuários, ou até mesmo não exibir nenhum menu lateral caso o perfil do usuário logado não exista menu, que é o caso do perfil de Professor no estágio probatório, pois todas as suas ações estão distribuídas pela dashborad.

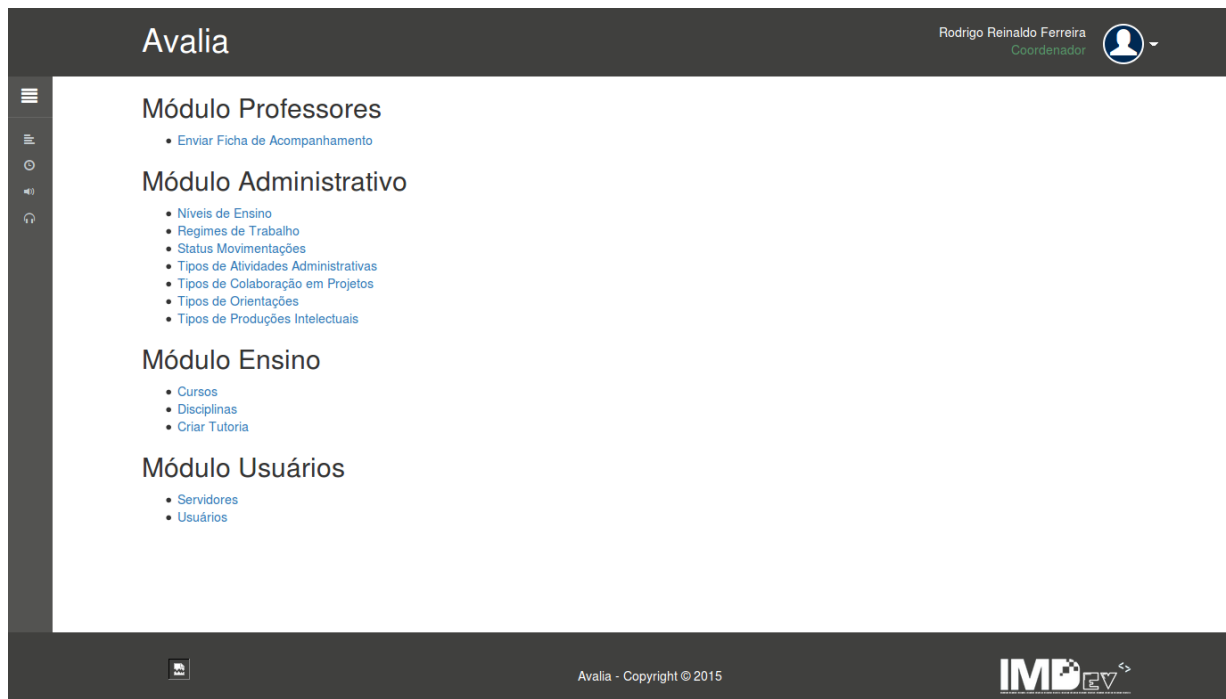
6.1.2.3 Conteúdo principal

Atualmente o sistema está exibindo o um menu com as funcionalidades que o usuário poderá realizar, semelhante ao menu lateral, porém será implementado um dashboard que conterà as informações mais importantes para o perfil de cada usuário, e também servirá como menu para o perfil Professor em estágio probatório.

6.1.2.4 Rodapé

Exibe a logo o instituto alinhado a esquerda, no centro exibe informações do setor que desenvolveu e o ano do desenvolvimento e alinhado a direita é possível visualizar a logo do setor. Na **figura 11** é possível ver a tela inicial.

Figura 11 - Tela inicial do sistema



Fonte: Resultado da pesquisa realizada pelo próprio autor.

6.2 CASOS DE USO

6.2.1 UC01 - Manter Cursos

A imagem a seguir **figura 12** representa a tela onde será possível o usuário realizar o cadastro de um Curso, sendo de preenchimento obrigatório o nome do curso e selecionar um nível de ensino.

Figura 12 - Tela cadastro de curso

The screenshot displays a web interface for course registration. At the top, a dark header bar contains the word 'Avalia' on the left and the user's name 'Rodrigo Reinaldo Ferreira' with the role 'Coordenador' and a profile icon on the right. A vertical sidebar on the left contains several menu icons. The main content area is titled 'Cadastrar Curso' and features two required input fields: 'Curso *' and 'Nível de Ensino *'. Below these fields are two green buttons labeled 'Listagem de Curso' and 'Cadastrar'. The footer of the interface includes a small logo on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMDeV' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor.

6.2.2 UC02 - Manter Disciplinas

A imagem a seguir **figura 13** representa a tela onde é possível o usuário realizar o cadastro da Disciplina, onde será de preenchimento obrigatório o nome da disciplina, código, carga horária e também necessária associar a disciplina a algum curso o campo descrição não é de preenchimento obrigatório.

Figura 13 - Tela cadastro da Disciplina

The screenshot displays the 'Cadastrar Disciplina' (Register Discipline) form within the 'Avalia' application. The interface features a dark header with the application name 'Avalia' and the user's name 'Rodrigo Reinaldo Ferreira' with the role 'Coordenador'. A sidebar on the left contains navigation icons. The main content area is titled 'Cadastrar Disciplina' and contains the following form fields:

- Código ***: A text input field.
- Disciplina ***: A text input field.
- Carga Horária ***: A dropdown menu.
- Descrição**: A text input field.
- Curso ***: A dropdown menu.

At the bottom right of the form, there are two green buttons: 'Listagem de Disciplina' and 'Cadastrar'. The footer of the application shows the text 'Avalia - Copyright © 2015' and the 'IMD' logo.

Fonte: Resultado da pesquisa realizada pelo próprio autor.

6.2.3 UC03 - Manter Atividade Administrativa

A imagem a seguir **figura 14** representa a tela onde será possível o usuário realizar o cadastro do Tipo de Atividade Administrativa, a página exibe apenas um campo que será de preenchimento obrigatório, e possui o botão permitindo o usuário ver uma lista com todas as atividades administrativa cadastradas.

Figura 14 - Tela cadastro do tipo de atividade

The screenshot shows a web application interface for 'Avalia'. At the top, the header bar contains the name 'Avalia' on the left and the user profile 'Rodrigo Reinaldo Ferreira' with the role 'Coordenador' on the right. A sidebar on the left contains a menu with icons for home, list, add, and search. The main content area is titled 'Cadastrar Tipo de Atividade Administrativa'. It features a text input field labeled 'Tipo de Atividade Administrativa *'. Below the input field are two green buttons: 'Listagem de Tipo de Atividade Administrativa' and 'Cadastrar'. The footer bar includes a small logo on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMDeV' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.4 UC04 - Manter Servidor

A imagem a seguir representa a tela onde será possível dar início ao cadastro do servidor, como podemos visualizar o sistema oferece algumas opções de perfil para esse servidor, após a escolha do perfil o sistema exibe a página de cadastro para o perfil selecionado, a página exibe na tela diversos campos para o preenchimento, os campos com asterisco são campos de preenchimento obrigatório, como pode ser visto na **figura 15**.

Figura 15 - Tela cadastro para o servidor com perfil Coordenador de Curso

Avalia Rodrigo Reinaldo Ferreira **Chefe**

Cadastrar Servidor

Nome *

CPF *

Sexo * Seleccione...

E-mail *

Data de Admissão * 13 Abril 2015

Nível Seleccione...

Regime Seleccione...

Login *

Senha *

Listagem de Servidor Cadastrar

Avalia - Copyright © 2015

IMDeV

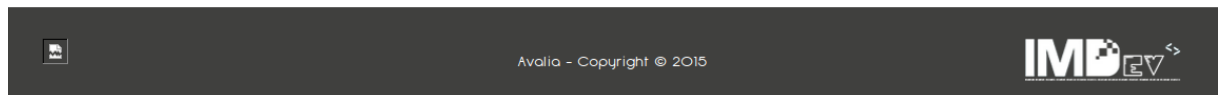
Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.5 UC05 - Manter Nível de Ensino

A imagem a seguir **figura 16** representa a tela onde será possível o usuário cadastrar o Tipo de Colaboração em Projeto, a página exibe apenas um campo que será de preenchimento obrigatório, e possui o botão permitindo o usuário ver uma lista com todos os Tipo de Colaboração em Projeto

Figura 16 - Tela cadastro para o Nível de Ensino

The screenshot shows the 'Avalia' system interface. At the top, there is a dark header bar with the 'Avalia' logo on the left and the user's name 'Rodrigo Reinaldo Ferreira' with the role 'Chefe' and a profile icon on the right. Below the header, the title 'Cadastrar Nível de Ensino' is displayed. The form contains two input fields: 'Nível de Ensino *' and 'Abreviação *'. Below these fields are two green buttons: 'Listagem de Nível de Ensino' and 'Cadastrar'.



Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.6 UC06 - Manter Regime

A imagem a seguir **figura 17** representa a tela onde será possível o usuário realizar o cadastro do Regime de Trabalho, a página exibe apenas dois campos de texto ambos de preenchimento obrigatório, onde o usuário informa o Regime de Trabalho e a Carga Horária, contém também um campo do tipo check onde o usuário informa se terá dedicação exclusiva ou não, a página exibe também um botão permitindo o usuário ver uma lista com todos os Regimes cadastrados.

Figura 17 - Tela cadastro para o Regime de Trabalho

Avalia

Rodrigo Reinaldo Ferreira
Chefe

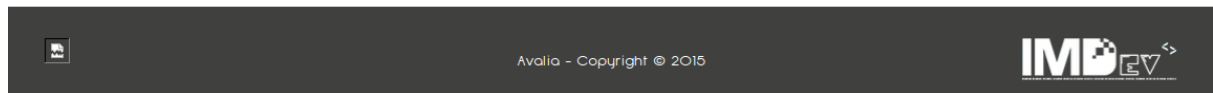
Cadastrar Regime de Trabalho

Regime de Trabalho *

Carga Horária *

Dedicação Exclusiva? ☐

Listagem de Regime de Trabalho Cadastrar



Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.7 UC07 - Manter Produção Intelectual

A imagem a seguir **figura 18** representa a tela onde será possível o usuário realizar o cadastrar o Tipo de Produção Intelectual, a página exibe apenas um campo que será de preenchimento obrigatório, e possui o botão permitindo o usuário ver uma lista com todos os Tipos de Produção Intelectuais cadastrados.

Figura 18 - Tela cadastro para a Produção Intelectual

The screenshot shows a web interface for the 'Avalia' system. At the top, there is a dark header bar with the 'Avalia' logo on the left and the user's name 'Rodrigo Reinaldo Ferreira' with the role 'Chefe' and a profile icon on the right. Below the header, the main title 'Cadastrar Tipo de Produção Intelectual' is displayed. The form consists of a single text input field labeled 'Tipo de Produção Intelectual *'. Below the input field, there are two green buttons: 'Listagem de Tipo de Produção Intelectual' and 'Cadastrar'. At the bottom of the page, there is a dark footer bar containing a small logo on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMB' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.8 UC08 - Manter Tutoria

A imagem a seguir **figura 19** representa a tela onde será possível o usuário criar uma tutoria, a página exibe apenas esses dois campos que são de preenchimento obrigatório, nesses campos selecionamos o Professor e logo abaixo selecionamos o Tutor para o Professor, dessa forma associamos o Tutor ao Professor e a tutoria é criada, e possui o botão permitindo o usuário ver uma lista com as tutorias cadastradas.

Figura 19 Tela cadastro para criar tutoria

The screenshot shows a web interface for a system named 'Avalia'. At the top, there is a dark header bar with the 'Avalia' logo on the left and the user's name 'Rodrigo Reinaldo Ferreira' with the role 'Chefe' and a profile icon on the right. Below the header, the main title 'Cadastrar Tutoria' is displayed. The form contains two dropdown menus: 'Professor *' and 'Tutor *', both with 'Selecione...' as the placeholder text. To the right of these fields are two green buttons: 'Listagem de Tutoria' and 'Cadastrar'. At the bottom of the page, there is a dark footer bar containing a small logo on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMD' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.9 UC11 Manter Orientação

A imagem a seguir **figura 20** representa a tela onde será possível o usuário cadastrar o Tipo de Orientação e a descrição do Tipo de Orientação, a página exhibe apenas esses dois campos que são de preenchimento obrigatório, e possui o botão permitindo o usuário ver uma lista com os Tipos de Orientação cadastrados e suas respectivas descrições.

Figura 20 - Tela cadastro para criar Orientação

The screenshot shows a web application interface for 'Avalia'. At the top, a dark header bar contains the 'Avalia' logo on the left and the user profile 'Rodrigo Reinaldo Ferreira' with the role 'Chefe' and a profile icon on the right. Below the header, the main title 'Cadastrar Tipo de Orientação' is displayed. The form consists of two input fields: 'Tipo de Orientação *' and 'Descrição *'. Below these fields are two green buttons: 'Listagem de Tipo de Orientação' and 'Cadastrar'. At the bottom of the page, a dark footer bar contains a small logo on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMB EV' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor

6.2.10 UC12 - Manter Colaboração em Projeto

A imagem a seguir **figura 21** representa a tela onde será possível o usuário cadastrar o Tipo de Colaboração em Projeto, a página exibe apenas um campo sendo de preenchimento obrigatório, e possui o botão permitindo o usuário ver uma lista com todos os Tipo de Colaboração em Projeto

Figura 21 - Tela cadastro para criar a Colaboração em Projetos

The screenshot displays the 'Avalia' web application interface. At the top, a dark header bar contains the 'Avalia' logo on the left and the user profile 'Rodrigo Reinaldo Ferreira' with the role 'Chefe' and a profile icon on the right. Below the header, the main title 'Cadastrar Tipo de Colaboração em Projeto' is centered. The form area includes a label 'Tipo de Colaboração em Projeto *' followed by a text input field. Below the input field are two green buttons: 'Listagem de Tipo de Colaboração em Projeto' and 'Cadastrar'. At the bottom, a dark footer bar contains a small icon on the left, the text 'Avalia - Copyright © 2015' in the center, and the 'IMDeV' logo on the right.

Fonte: Resultado da pesquisa realizada pelo próprio autor

7 CONCLUSÃO

Após algum tempo trabalhando no sistema que ainda não foi finalizado totalmente, existirão ainda mais algumas funcionalidades que estão sendo desenvolvidas, temos a total certeza que após finalizado o processo de avaliação do estágio probatório se tornará muito mais simples, confinável e seguro, o tempo de cadastro das informações necessárias para a avaliação reduzirá drasticamente em relação ao atual processo, e as informações transitadas na aplicação não poderão ser perdidas e estarão disponíveis a qualquer momento, levando em conta que o layout é totalmente responsivo amplia ainda mais a facilidade de acesso ao sistema.

Com relação à tecnologia utilizada para o desenvolvimento da aplicação, uma das vantagens para o desenvolvimento em Groovy on Grails, por se tratar de uma linguagem dinâmica ela contribuiu para o rápido desenvolvimento da aplicação, o recurso de Scaffolding deu uma grande agilidade ao desenvolvimento dos CRUD's, uma vez que são gerados automaticamente, não sendo necessário o desenvolvedor implementar esses CRUD's e conseqüentemente economizando muito tempo, tempo que pode ser utilizado na implementação de algum recurso do sistema, e evitando também muito trabalho repetitivo.

Outro ponto positivo para a utilização do framework foi na questão de configuração de ambiente, o Grails adota o conceito de convenções sobre configuração já citado anteriormente, sendo desnecessária toda a configuração inicial dos arquivos de configuração, essa que é uma das reclamações constantes no desenvolvimento de aplicações corporativas na plataforma Java.

Para a equipe essa padronização da linguagem trouxe diversos benefícios, depois da padronização será possível trabalhar com uma equipe mais enxuta, mais uniforme, facilitando o atendimento ao cliente e facilitando também o treinamento e suporte, tanto para o cliente como para a própria equipe. Vale destacar também que a padronização trará uma grande consistência nos processos, tornando cada vez mais rápido e confiável o desenvolvimento de novos sistemas.

REFERÊNCIAS

FUENTES, VINÍCIUS B. **Ruby on Rails – Coloque sua aplicação nos trilhos**. São Paulo: Casa do Código, 2012.

BOOTSTRAP. **Bootstrap**. Disponível em: <<http://getbootstrap.com/>>. Acesso em: 1 maio 2015.

COMPUTAÇÃO UFCG. **Framework**. Disponível em: <<http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/frame/oque.htm>>. Acesso em: 2 maio 2015.

COMPUTAÇÃO UFCG. **UML**. Disponível em: <<http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/uml/uml.htm>>. Acesso em: 24 abr. 2015.

DESENVOLVIMENTO AGIL. **Scrum**. Disponível em: <<http://www.desenvolvimentoagil.com.br/scrum/>>. Acesso em: 1 maio 2015.

GRAILS BRASIL. **Grails**. Disponível em: <<http://www.grailsbrasil.com.br/>>. Acesso em: 3 abr. 2015.

Limesurvey. **LimeSurvey**. Disponível em: <<https://www.limesurvey.org/en/>>. Acesso em: 9 maio 2015.

MOODLE. **Moodle**. Disponível em: <https://moodle.org/?lang=pt_br>. Acesso em: 20 mar. 2015.

OFICINA DA NET. **MVC**. Disponível em: <http://www.oficinadanet.com.br/artigo/desenvolvimento/o_que_e_model-view-controller_mvc>. Acesso em: 7 fev. 2015.

POSTGRESQL. **PostgreSQL**. Disponível em: <<http://www.postgresql.org/>>. Acesso em 15 de Maio de 2015.

SABBAGH, Rafael . **Scrum: gestão ágil para projetos de sucesso**. São Paulo: Casa do Código, 2013.

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

SPRING. **Grails**. Disponível em: <<https://spring.io/>>. Acesso em: 3 abr. 2015.

UML. **UML**. Disponível em: <<http://www.uml.org/>>. Acesso em: 24 abr. 2015.

ZEMEL, TÁRCIO. **Web Design Responsivo**: Páginas adaptáveis para todos os dispositivos. São Paulo: Casa do Código, 2012.