

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO  
GRANDE DO NORTE  
CAMPUS AVANÇADO LAJES  
CURSO TÉCNICO INTEGRADO EM INFORMÁTICA

MARIA LUIZA DA SILVA MATIAS

**ITARETIF - UM PORTAL PARA DIVULGAÇÃO DE INFORMAÇÕES DO CAMPUS  
LAJES**

LAJES/RN  
2021

MARIA LUIZA DA SILVA MATIAS

**ITARETIF - UM PORTAL PARA DIVULGAÇÃO DE INFORMAÇÕES DO CAMPUS  
LAJES**

Relatório de Prática Profissional apresentado ao Curso Técnico Integrado em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Avançado Lajes, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em Informática.

Orientador (a): Prof. Me. Fernando Helton Linhares Soares

## **AGRADECIMENTOS**

Agradeço, primeiramente, a minha mãe, que sempre me apoiou, incentivou, nunca me deixou faltar nada e cuidou para que eu continuasse estudando.

Agradeço aos integrantes do grupo, pois cada um deles ajudou a tornar o projeto uma realidade.

Agradeço ao nosso brilhante orientador e co-orientador pela paciência e ensinamentos.

E, por fim, agradeço aos professores do IF pelo carinho e aprendizados.

## RESUMO

Após a aprovação no processo seletivo do Instituto Federal de Educação, Ciência e Tecnologia, os estudantes ingressam na instituição sem conhecer seu funcionamento e suas instalações, quais são as normas de conduta exigidas para eles e, tanto aos novos ingressantes quanto ao público externo, existe um déficit de comunicação no que toca novos projetos e informes da instituição. A maioria dessas informações podem ser encontradas no portal oficial da instituição, porém, a linguagem utilizada nos documentos e a quantidade de dados contidos nele, dificultam o entendimento e a pesquisa das normas, fazendo com que a página da instituição não seja atrativa para o público. Neste sentido, este projeto propõe a criação de um portal capaz de abrigar as informações mais solicitadas pelos estudantes do Campus Avançado Lajes, com o objetivo de proporcionar aos alunos, assim como à população externa, informações sobre a instituição utilizando-se de uma linguagem mais empregada no cotidiano das pessoas, tornando o portal atrativo e de fácil entendimento.

Palavras-chave: informática, informação, campus, Lajes, instituição, portal.

## **ABSTRACT**

After approval in the selection process of the Instituto Federal de Educação, Ciência e Tecnologia, students enter the institution without knowing its operation and facilities, what are the standards of conduct required for them and, both for new entrants and the external public, there is a communication deficit regarding new projects and reports from the institution. Most of this information can be found on the institution's official website, however, the language used in the documents and the amount of data contained on that website make it difficult to understand and search the rules, making the institution's website unattractive to the public. In this sense, this project proposes the creation of a portal capable of housing the information most requested by the students entering the Campus Avançado Lajes, with the objective of providing students, as well as the external population, with information about the institution using a language that is more used in people's daily lives, making the portal attractive and easy to understand.

Keyword: Computing, information, Campus, Lajes, institution, Portal.

## LISTA DE ABREVIATURAS E SIGLAS

|      |  |
|------|--|
| HTML | <i>Hypertext Markup Language</i> (Linguagem de marcação de hipertexto) |
| CSS  | <i>Cascading Style Sheets</i> (Folhas de Estilo em Cascata)            |
| SI   | Sistema de Informação  |
| UML  | <i>Unified Modeling Language</i> (Linguagem Unificada de Modelagem)    |
| RN   | Rio Grande do Norte  |

## SUMÁRIO

|  |    |
|--|----|
| <b>1 INTRODUÇÃO</b>                                | 8  |
| 1.1 OBJETIVO                                       | 9  |
| 1.2 JUSTIFICATIVA                                  | 9  |
| <b>2 DADOS GERAIS DA PESQUISA/EXTENSÃO/ESTÁGIO</b> | 10 |
| <b>3 FUNDAMENTAÇÃO TEÓRICA</b>                     | 11 |
| 3.1 HTML e CSS                                     | 11 |
| 3.2 Bootstrap 7                                    | 12 |
| 3.3 Eclipse  | 12 |
| 3.4 Spring   | 13 |
| 3.5 Github   | 13 |
| 3.6 MySQL  | 13 |
| 3.7 UML  | 14 |
| <b>4 METODOLOGIA</b>                               | 14 |
| <b>5 ANÁLISE E PROJETO DO SISTEMA PROPOSTO</b>     | 16 |
| 5.1 APLICAÇÃO DO QUESTIONÁRIO                      | 17 |
| 5.2 REQUISITOS E CASOS DE USO                      | 20 |
| 5.3 DESENVOLVENDO DIAGRAMA DE CLASSES              | 21 |
| <b>6 PORTAL ITARETIF</b>                           | 23 |
| 6.1 FUNCIONALIDADES DESENVOLVIDAS                  | 24 |
| 6.2 IMPLEMENTAÇÃO                                  | 34 |
| <b>7 ANÁLISE E DISCUSSÃO DOS RESULTADOS</b>        | 41 |
| <b>8 CONSIDERAÇÕES FINAIS</b>                      | 42 |
| <b>REFERÊNCIAS</b>                                 | 44 |
| ANEXO A – FORMULÁRIO DE IDENTIFICAÇÃO              | 47 |

## 1 INTRODUÇÃO

O Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, assim como os demais institutos do país, conta com um portal que tem por finalidade publicar informações, como notícias sobre determinados assuntos das instituições, comunicados mais pertinentes sobre os campi, processos seletivos, concursos públicos, projetos e a difusão de determinados temas e eventos.

Dentro desse portal, cada Campus tem a sua página específica, para que as publicações sejam organizadas de forma mais particular em relação às postagens gerais. Porém, tais publicações acabam não recebendo o devido destaque, ou apresentando certas dificuldades de acesso para quem está visitando o *site*, o que limita bastante a transmissão de informações básicas - como, por exemplo, normas, estrutura física, localidade e setores do Campus - para o público externo e para os calouros que irão ingressar nas suas respectivas instituições.

Além disso, trata-se de uma página que segue fielmente todas as normas gramaticais da língua portuguesa, tornando a formalidade uma característica visível na estrutura do Portal, seja na escrita dos textos, seja na disponibilidade dos conteúdos, o que resulta em pouca atratividade e, apesar de não ser um grande obstáculo, torna o processo de interação entre o visitante e o portal demasiadamente cansativo.

Pensando nessa problemática, surgiu a ideia inicial do ItaretIF, que tem como objetivo disponibilizar um portal exclusivo para o público externo e interno do Campus Avançado Lajes, principalmente os calouros e visitantes interessados em conhecer o lugar. O portal, além de todas as informações básicas do Campus em questão, contém detalhes sobre o desenvolvimento e o funcionamento da instituição de ensino, que veio a ser umas das maiores conquistas para a cidade de Lajes e região. Como já citado neste tópico, as informações publicadas no portal serão exclusivamente sobre o Campus, apresentando uma linguagem mais utilizada no cotidiano da população lajense e, portanto, menos formal, oferecendo conforto e facilidade de navegação aos usuários.

Esta prática tem como finalidade o crescimento do Campus como instituição, a disponibilidade de recursos mais acessíveis para conhecimento sobre o que



acontece na comunidade escolar e, acima de todos os pontos, o foco principal: disponibilizar aos alunos ingressantes no Campus Lajes todas as informações necessárias sobre o polo.

Ao decorrer deste relatório serão apresentadas a caracterização da prática profissional realizada, sua fundamentação teórica, a metodologia utilizada no trabalho, o detalhamento das atividades desenvolvidas, a análise dos resultados alcançados e, por fim, as considerações finais.

## 1.1 OBJETIVO

Disponibilizar e facilitar o acesso às informações sobre o Campus Avançado Lajes com uma linguagem familiar aos usuários, os ingressantes do Campus em questão e o público externo. Para isso, o Portal ItaretIF trará uma linguagem que considera um contexto mais jovem e regional, de forma clara e objetiva, além de apresentar leveza em sua configuração visual. Esse portal irá disponibilizar, também, as normas e regulamentos, informações sobre os setores e servidores, sobre o município e sobre a instituição em geral.

## 1.2 JUSTIFICATIVA

Ao ingressar no Instituto Federal - Campus Lajes, os estudantes não sabem como funciona a instituição e sentem dificuldade em encontrar essas informações, como, por exemplo, quais são as regras do IFRN, quais as normas de fardamento, quais são os setores do prédio principal e quem são os profissionais que atuam em cada um deles, e qual caminho fazer para chegar até um determinado setor quando necessitar de algum tipo de atendimento. O desconhecimento dessas informações pode fazer com que o aluno cometa faltas disciplinares, e, também, perca prazos de requerimento, por exemplo, o prazo para justificativa de ausências.

## **2 DADOS GERAIS DA PESQUISA/EXTENSÃO/ESTÁGIO**

Título do projeto: ItaretIF - Um portal para divulgação de informações do campus

Lajes

Período de realização: de 02/12/2019 a 30/01/2021

Total de horas: mínimo de 550 horas.

Orientador: Fernando Helton Linhares Soares

Nome do co-orientador: Dannylo Johnathan Bernardino Egidio

Função: Aluno Voluntário

Formação profissional: Técnico em Informática

Abaixo seguem as atividades desenvolvidas ao longo do período de realização do projeto destacando a carga horária empenhada e os resultados alcançados em cada uma das atividades.

Quadro 1 – Síntese das Atividades do Aluno no Projeto.

| <b>CARGA HORÁRIA</b> | <b>ATIVIDADES DESENVOLVIDAS</b>    | <b>RESULTADOS ALCANÇADOS</b>  |
|----------------------|------------------------------------|---|
| 30                   | Levantamento de Requisitos         | Todos os requisitos funcionais do portal foram levantados e o diagrama de caso de uso foi desenvolvido.       |
| 60                   | Projeto de arquitetura do sistema  | Desenvolvimento do diagrama de classe e do diagrama de sequência;   |
| 400                  | Implementação do sistema           | Finalização da implementação do portal proposto com todos os requisitos funcionais levantados anteriormente ; |
| 30                   | Escrita do relatório de atividades | Relatório final concluído   |

|    |                                       |  |
|----|---------------------------------------|--|
| 30 | Reuniões do acompanhamento do projeto | As reuniões ocorriam nas salas da própria instituição, mas, devido a pandemia, passaram a ser realizadas por vídeo chamadas através do <i>Google meets</i> . |
|----|---------------------------------------|--|

Fonte: autoria própria (ano).

### 3 FUNDAMENTAÇÃO TEÓRICA

Nesta seção serão apresentadas as tecnologias utilizadas para o desenvolvimento do portal.

#### 3.1 HTML e CSS

O HTML é um conjunto de tags responsáveis pela definição da estrutura do conteúdo de uma página no navegador. As tags são marcações que definem os elementos que compõem as páginas.

[...] HTML é usada para criar páginas web. Veremos que você começa escrevendo as palavras que quer que apareçam na página. Então adicione tags ou elementos às palavras de modo que o navegador saiba o que é um título, onde um parágrafo começa e termina etc. (DUCKETT, 2011, p. 6).

O *Cascading Style Sheets* (CSS) é a linguagem de estilos, utilizada para a implementação de páginas, em que desempenha alteração de fonte, cor, layout e outros.

*Cascading Style Sheets* (ou simplesmente CSS) é uma linguagem de estilo utilizada para definir a apresentação de documentos escritos em uma linguagem de marcação, como HTML ou XML. Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento. Ao invés de colocar a formatação dentro do documento, o desenvolvedor cria um link (ligação) para uma página que contém os estilos, procedendo de forma idêntica para todas as páginas de um portal. Quando quiser alterar a aparência do portal basta portanto modificar apenas um arquivo. Cabem a CSS todas as funções de apresentação de um documento e ao HTML todas as funções de marcação e estruturação de conteúdos. Uma página web é composta então de CSS + HTML. (MEIRA, 2013, p. 4).

#### 3.2 BOOTSTRAP 7

Segundo Alvim (2010, p. 12), “O *framework* é um conjunto de classes que colaboram entre si, proporcionando melhores práticas de desenvolvimento e diminuição da repetição de tarefas.” O Bootstrap é um *framework* que ajuda no desenvolvimento

rápido e seguro de aplicações que utilizam HTML, CSS e JavaScript. Essa ferramenta é gratuita e possui um conjunto de componentes pré-construídos que podem ser reutilizados na aplicação. Com ele é possível criar páginas responsivas de maneira simples e padronizada. Dessa forma, Alvim (2010, p.12) discorre que o *framework* evita variações de soluções diferentes para um mesmo tipo de problema. O que facilita a reutilização e customização dos códigos.

### 3.3 ECLIPSE

O Eclipse é um Ambiente de Desenvolvimento Integrado (IDE) que promove um ambiente de desenvolvimento Java, capaz de integrar diversas tecnologias e facilitar o desenvolvimento de aplicações. Existem diversos *plugins* que podem estender as funcionalidades da ferramenta adaptando esta para as necessidades específicas de cada projeto.

A plataforma do Eclipse fornece vários pacotes de desenvolvimento, tais como Eclipse JDT, que é a base para qualquer plug-in na linguagem java, o Eclipse SDK, que é o pacote de distribuição da IDE java, o Eclipse WTP (*Web Tools Platform*), que é usado para desenvolvimento de linguagem para web, e o compilador do JDT, que é seu próprio compilador java, que é mais rápido e de código aberto. (FARIA; LIMA; DIAS; SILVA; COSTA; BITTAR, 2010, p. 4).

### 3.4 SPRING

O Spring é um *framework* que abarca vários Projetos Springs que facilitam a criação e configuração de aplicações java. Alguns projetos springs utilizados para o desenvolvimento deste projeto foram o Spring Boot, que auxilia na configuração da aplicação, o Spring MVC, responsável pelas classes de controle e injeção de dependências, o Spring *Security*, encarregado pelos processos de autenticação e autorização, e o Spring *Data*, que fornece o acesso aos dados existentes em um banco de dados. Segundo Moreira e Afonso (2017, p. 12), “O Spring não é um *framework* apenas, mas um conjunto de projetos que resolvem várias situações do cotidiano de um programador, ajudando a criar aplicações Java com simplicidade e flexibilidade”.

### 3.5 GITHUB

O GitHub é uma plataforma de hospedagem de código e colaboração que utiliza o Git, um sistema de controle de versão gratuito e de código aberto. Essa plataforma permite que seus usuários construam um projeto e acompanhe as mudanças feitas pelos demais sem a necessidade de estar no mesmo ambiente. O GitHub fornece um caminho direto para a criação de *sites* para seus projetos, o que o torna uma opção natural para publicar e manter a documentação. No GitHub é possível criar vários repositórios públicos, desde que estes projetos tenham licença *open-source*. Dentro desse repositório podem estar: pastas e arquivos, imagens, vídeos, planilhas e conjuntos de dados, tudo o que for necessário para o desenvolvimento do projeto. Em suma, o Github é um serviço de *web* que porta várias funcionalidades, em que “o Git é um sistema de controle de versão de arquivos. Através deles podemos desenvolver projetos na qual diversas pessoas podem contribuir simultaneamente no mesmo, editando e criando novos arquivos.” (SCHMITZ, 2015).

### 3.6 MySQL

O MySQL é um gerenciador de banco de dados de código aberto, sendo este facilmente configurável. Essa aplicação utiliza a Linguagem de Consulta Estruturada (SQL), através da qual o conteúdo que está incluso no armazenamento do banco de dados é administrado. Para utilizar o MySQL, é preciso que seja efetuada a instalação de um servidor e de uma aplicação cliente. O servidor será o responsável pelo armazenamento de dados, como também, por responder às requisições. Já a comunicação do cliente com o servidor irá acontecer por meio da SQL.

### 3.7 UML

A UML, Linguagem Unificada de Modelagem, é uma linguagem gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de *software* (BOOCH; RUMBAUGH; JACOBSON, 2006). A Linguagem Unificada de Modelagem (UML) é uma linguagem de notação que permite aos

programadores modelar e documentar de forma detalhada a estrutura e o comportamento de seus projetos de sistema. Essa modelagem é feita por meio de diagramas e cada diagrama possui seus próprios elementos gráficos que, por sua vez, possuem relação entre si. O objetivo dessa padronização é ajudar na comunicação entre os profissionais envolvidos na produção de um mesmo *software*.

## 4 METODOLOGIA

O principal método que utilizamos neste projeto foi o da pesquisa-ação que, de acordo com André Fontenelle (2020), fundamenta-se em identificar um problema prático de uma comunidade e, em seguida, desenvolver um projeto com ações para solucionar esse problema; tendo como principal característica, a proatividade e o envolvimento ativo do pesquisador para a mudança dos fenômenos.

Ademais, para comprovar a existência da problemática e auxiliar no desenvolvimento do projeto - definir quais informações seriam realmente pertinentes fazer parte do portal ItaretIF, bem como o que seria necessário à implementação de tais informações - fizemos uma pesquisa estruturada com o público alvo, os alunos do Campus Avançado Lajes. A pesquisa consistiu em um breve questionário, criado através do *google forms*, contendo perguntas de múltipla escolha obrigatórias e contabilizou 131 respostas (devido à pandemia do COVID-19, as aulas dos IFRNs estão sendo remotas, portanto, o questionário foi enviado aos grupos das respectivas salas por meio de *links* no *Whatsapp*). Esse método de pesquisa é categorizado como pesquisa quantitativa que, segundo Naína Tumelero (2019), consiste em traduzir os números e opiniões coletadas em informações que serão utilizadas para a sua classificação e posterior análise.

“A pesquisa quantitativa, ou quanti, é uma modalidade de pesquisa baseada na aplicação de questionários estruturados a um grupo de pessoas. [...] Quando falamos que a aplicação é realizada com auxílio de questionários estruturados, quer dizer que as mesmas perguntas são feitas a todos os participantes do levantamento” (DIAS, 2018).

Utilizamos os computadores dos laboratórios de informática, da biblioteca e nossos próprios *notebooks* para fazer as pesquisas, implementar o código e escrever o relatório.

Após o levantamento dos requisitos, dividimos as atividades entre os



integrantes do grupo. Ficamos responsáveis por elaborar os diagramas, de caso de uso e de classe, e cada um ficou responsável pelo seu próprio diagrama de sequência - utilizamos a UML (Linguagem Unificada de Modelagem) para modelar e documentar os diagramas. Além disso, cada um ficou encarregado de implementar, no mínimo, 2 partes e/ou funcionalidades do portal.

Para a implementação do projeto, a linguagem de programação utilizada foi Java, tanto por ser compatível com projetos *web*, rodar em qualquer sistema operacional, quanto por fazer parte da ementa acadêmica e ter sido estudada durante o curso Técnico Integrado em informática que o Campus oferece. O banco de dados ao qual optamos foi o MySQL, visto que é um gerenciador de banco de dados de código aberto, seguro, bastante fácil de configurar, manusear, amplamente utilizado por desenvolvedores, além de também ter sido estudado durante o curso. Para a definição da estrutura do conteúdo da página, utilizamos a linguagem de marcação HTML aliado ao CSS e bootstrap, para um visual mais agradável e elegante. Para a criação e configuração da nossa aplicação, utilizamos o *framework* Spring, que abarca vários projetos Springs, como o spring boot, que auxilia na configuração da aplicação, o spring *web*, que já vem com o servidor Apache TOMCAT embutido, o Spring MVC, responsável pelas classes de controle e injeção de dependências, o Spring *Security*, encarregado pelos processos de autenticação e autorização, e o Spring *Data*, que fornece o acesso aos dados existentes em um banco de dados. A IDE que utilizamos para o desenvolvimento do *software* foi o Eclipse, plataforma para desenvolvimento java e amplamente difundida.

Para que as mudanças no projeto pudessem ser feitas simultaneamente pelos integrantes do grupo, sem a necessidade de estarmos no mesmo local, criamos um repositório público no GitHub, plataforma de hospedagem de código e colaboração que utiliza o Git, um sistema de controle de versão gratuito e de código aberto.

Na efetivação do projeto, utilizamos o modelo de implementação separado por camadas, o spring MVC (*Model, View, Controller*), no qual separamos o código nas camadas modelo (*Model*), classe java que retrata as entidades do sistema e é responsável pelo acesso e manipulação dos dados; visão (*View*), responsável pela interface que mostra as informações presentes no modelo e fica entre o usuário e o

sistema; e controle (*Controller*), classe java responsável por implementar as funcionalidades e que faz a ligação entre o modelo e a visão.

## 5 ANÁLISE E PROJETO DO SISTEMA PROPOSTO

É sabido que informações importantes sobre o Campus Avançado Lajes podem ser encontradas no portal oficial do IFRN, como, por exemplo, notícias, eventos, editais, comunicação social e atos administrativos. Porém, percebemos que, além das informações sobre os Campi estarem em segundo plano e oferecer certas dificuldades de acesso para quem está visitando o *site* pela primeira vez, todo o portal possui uma estrutura estática e textos formais, sendo que alguns deste estão em PDF, tendo que, por vezes, ser feito *download* para poder visualizá-los. Todos esses fatores colaboram para a pouca atratividade, tornando o processo de interação entre o visitante e o portal bastante cansativo. Tal pressuposto foi comprovado com a aplicação do questionário, pois 88,5% dos entrevistados acharam necessário um portal exclusivo do Campus Avançado Lajes com características próprias, componentes mais intuitivos, com um conteúdo mais limpo e fácil de encontrar.

Sendo assim, o portal proposto - ItaretIF - vem com uma abordagem diferente, disponibilizando não somente as normas, regulamentos e informações sobre os setores e servidores, como, também, sobre a instituição em geral e projetos que serão desenvolvidos pelos professores e/ou mestres do Campus. Tudo isso com dinamismo, maior leveza visual e uma linguagem mais próxima do contexto regional em que o Campus está inserido.

### 5.1 APLICAÇÃO DO QUESTIONÁRIO

Como forma de validar e agregar à ideia do portal, foi aplicado um questionário estruturado às turmas do IFRN - Campus Avançado Lajes. Abaixo, seguem as imagens com as perguntas do questionário, os gráficos e as respostas obtidas:

Você conhece o portal IFRN?

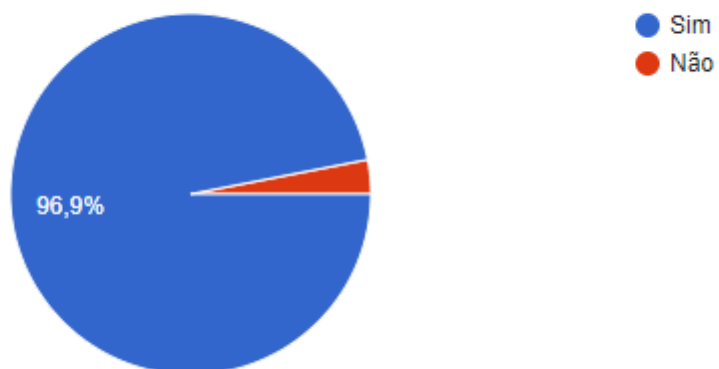


Figura 1 - Gráfico I

Com que frequência você acessa o portal IFRN?

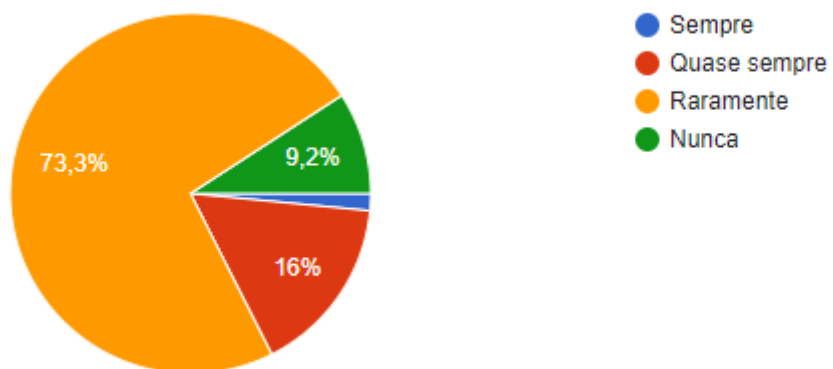


Figura 2 - Gráfico II

Consegue localizar as informações que procura com facilidade?

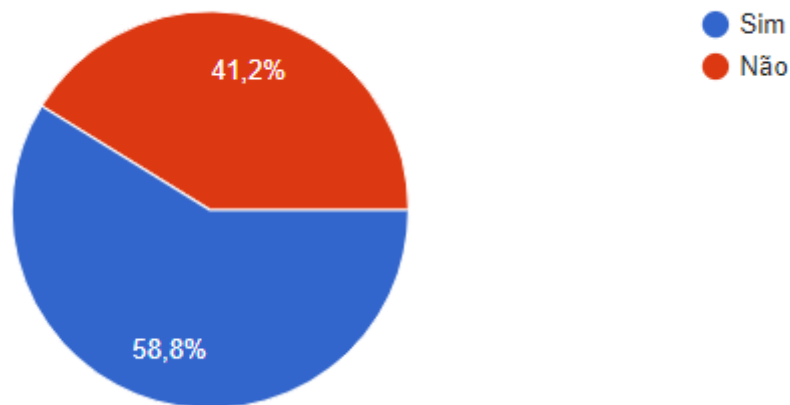


Figura 3 - Gráfico III

O que você acha de um portal que traga, em seu conteúdo, informações específicas do Campus Lajes? (informações sobre normas de conduta de alunos e servidores, sobre o Campus em geral, etc.)

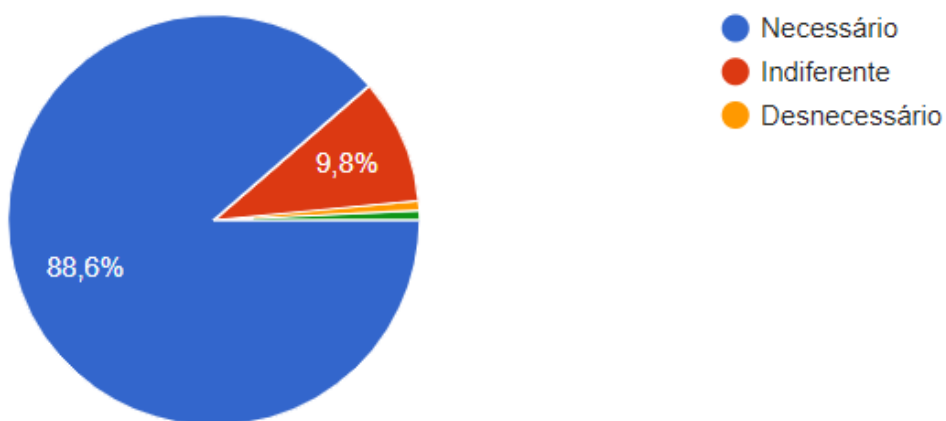


Figura 4 - Gráfico IV

Diante dos gráficos, concluímos que, apesar de 96,9% dos alunos terem conhecimento do portal oficial do IFRN (figura 1 - Gráfico I) e pouco mais da metade (58,8%) conseguir encontrar as informações que procuram com facilidade (figura 3 - Gráfico III), 88,6% dos alunos entrevistados acham necessário a existência de um

portal exclusivo para o Campus Avançado Lajes (figura 4 - Gráfico IV) que traga informações a respeito da instituição de ensino. Portanto, conclui-se que a maioria das respostas obtidas acerca desta temática considera válido um Portal para o Campus Avançado Lajes.

Além disso, analisando as respostas sobre o tópico que abria espaço para que os alunos fornecessem ideias a respeito das informações a serem divulgadas no portal, foram obtidas respostas como: “Acho que um *site* voltado a informações do Campus Lajes seria ótimo, principalmente se tivesse uma interface fácil de acessar”; “As informações precisam ser mais visíveis”; “Um *designer* legal”; “Um portal específico para o campus de Lajes é essencial, tanto para facilitar a encontrar informações, quanto para ficarmos atualizados das notícias. Um portal mais objetivo”; “Acredito que um portal com fácil acesso seja de grande ajuda, pois por mais que o sistema que utilizamos seja prático, ainda assim tem algumas peculiaridades que podem melhorar em relação ao acesso, então acho que a iniciativa de um projeto que visa essa melhoria é bastante efetiva.”; “Deixar o portal mais chamativo, para que os telespectadores que vão visitar o *site* sintam-se chamados e interessados a ler o conteúdo da página”; “Uma interface mais intuitiva!”.

## 5.2 REQUISITOS E CASOS DE USO

Inicialmente, fizemos o levantamento dos requisitos que o *software* deveria atender ao final do seu desenvolvimento e, então, após análise, começamos a decidir as estruturas que formariam o *software*, quais as funcionalidades de cada elemento, o acesso a dados, sincronização, desempenho, entre outros. Segue abaixo a lista com a definição dos requisitos do portal:

- Cadastrar usuários: Administradores, professores(as) e alunos(as);
- Permitir que usuários já cadastrados façam *login* no portal (autenticação);
- Permitir que diferentes usuários tenham acesso a diferentes páginas (autorização);

- Cadastrar servidor(a);
- Cadastrar setor;
- Cadastrar comentário;
- Emitir relatório dos comentários;
- Cadastro de projetos (apenas por professores já cadastrados no portal).

Após o levantamento dos requisitos, criamos o diagrama de caso de uso, de classe e os diagramas de sequência.

O diagrama de caso de uso é uma forma gráfica de demonstrar as funcionalidades do projeto, quem são os usuários que podem - e devem - participar da execução dessas funcionalidades e quais delas podem ou não ser realizadas por cada um deles. Na figura 5 podemos ver quais tarefas podem ser realizadas e quem são os usuários responsáveis por realizar cada uma delas.

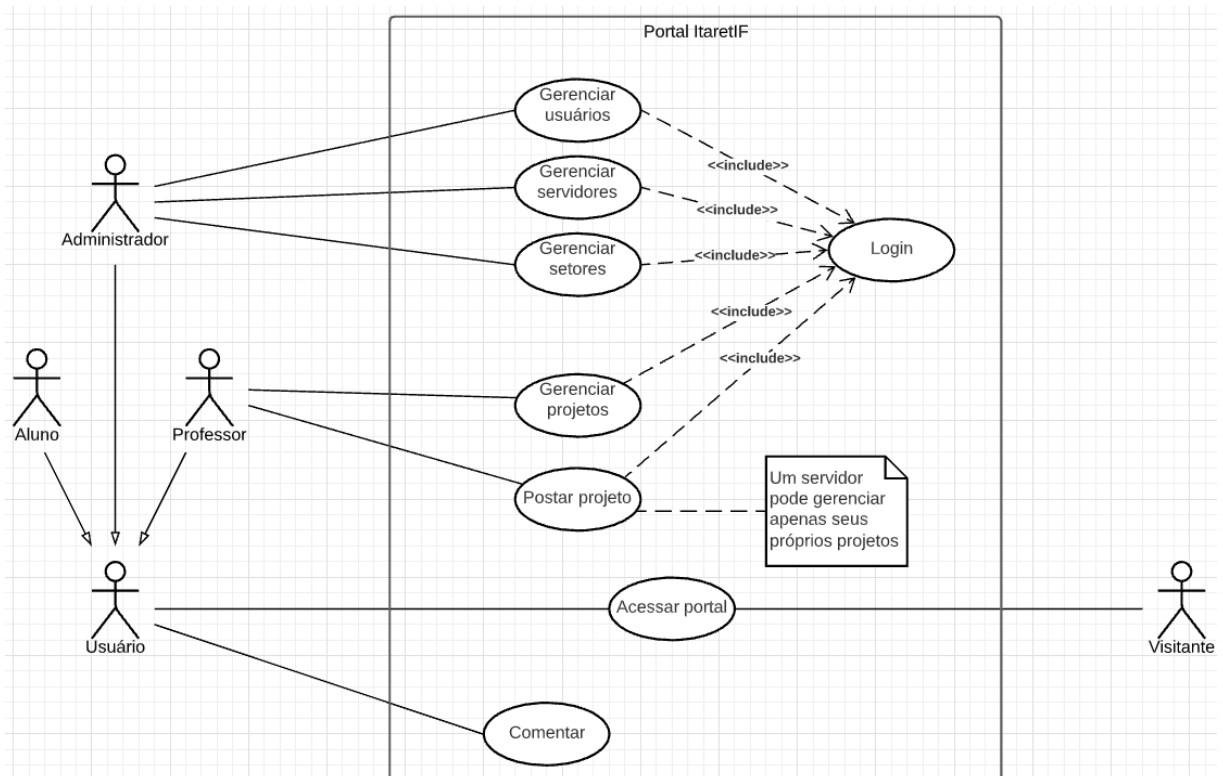


Figura 5 - Diagrama de Casos de uso

O visitante - que também pode ser um(a) aluno(a), servidor(a) ou administrador(a) - pode acessar o portal e navegar por ele sem a necessidade de

fazer *login*. O aluno pode comentar; o(a) servidor(a) pode comentar, postar e gerenciar projetos; e o(a) administrador(a) pode comentar, cadastrar um novo servidor(a), setor ou administrador(a), editar o conteúdo do portal e acessar todas as listas de usuários, setores e servidores já cadastrados. No entanto, como representado no diagrama acima, determinadas ações só podem ser realizadas se o usuário fizer *login* (utilizando matrícula e senha).

Um visitante que não se encaixe em nenhum desses 3 usuários, não pode fazer *login* e poderá apenas navegar pelo portal.

### 5.3 DIAGRAMA DE CLASSES

Na figura abaixo (figura 6) vemos o diagrama de classe. Este descreve a estrutura estática de um *software* por meio de métodos, atributos e relações entre as classes, definindo assim a sua estrutura lógica. Em resumo, o diagrama de classes descreve o que estará presente no sistema modelado.

Para darmos início ao nosso diagrama foi necessário estudarmos um pouco sobre a definição de cada componente, sendo assim temos: A classe, que é uma forma de definir um tipo de dado. Para definir os dados utilizamos atributos, que apresentam o conjunto de propriedades da classe. E por último vem o método que basicamente irá ditar o comportamento e as funcionalidades de uma classe. Na figura abaixo mostramos especificamente a atuação de cada um desses componentes:



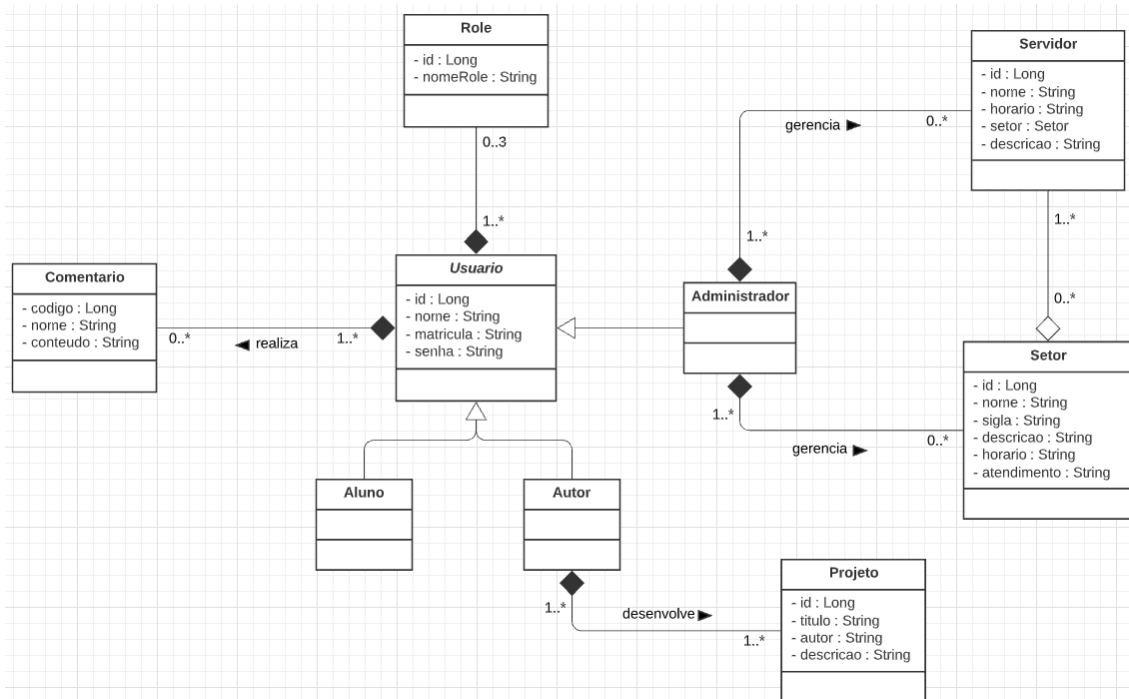


Figura 6 - Diagrama de classes

A classe *Role* é a classe responsável por definir os papéis dos usuários e depende da classe primária *Usuario* para existir. A classe *Usuario* possui as classes *Aluno*, *Autor* e *Administrador* como suas classes secundárias, logo, essas subclasses herdam todos os atributos da superclasse *Usuario*. Isso é necessário, pois, além de evitar repetição de código, o usuário pode ter diferentes papéis no nosso sistema. A classe *Comentario* é uma classe que depende da classe *Usuario*, pois sem a existência dessa classe, também não existiria a classe *Comentario*. Da mesma forma, a classe *Projeto* depende da classe *Autor*, pois a classe projeto não pode existir por conta própria, ela precisa da classe *Autor*. As classes *Setor* e *Servidor* também possuem uma relação de composição com a classe *Administrador* já que esta é a classe responsável pelos cadastros e gerenciamento dos servidores e setores. Por fim, as classes *Servidor* e *Setor* possuem uma relação de agregação, visto que, um servidor não necessariamente precisa de um setor. Sendo assim, a classe *Servidor* não depende da classe *Setor*.

## 6 PORTAL ITARETIF

O portal ItaretIF tem como proposta disponibilizar e facilitar o acesso às informações mais solicitadas pelos alunos do Campus Avançado Lajes, bem como o conhecimento básico sobre o município no qual o polo está localizado. Além disso, ainda no portal, é possível cadastrar e gerenciar projetos, cadastrar setores, servidores, alunos, administradores, listar cada um desses cadastros, acessar os dados de cada usuário e editar ou excluir cadastros.

A estrutura do portal conta com páginas de fácil manuseio, bastante intuitivas e limpas, contendo as funções que podem ser executadas pelos seus respectivos usuários e uma linguagem mais próxima dos jovens e do vocabulário local. Ademais, todas as páginas contam com uma barra de navegação (ou menu) para acesso rápido a outras páginas e funções. A seguir, apresentaremos as funcionalidades desenvolvidas.

A página inicial do portal (figura 7) possui uma breve apresentação, para que quem esteja visitando o portal pela primeira vez possa se situar. No menu, temos o *link* para as páginas: *O Campus, Regras, Lajes/RN, Mapeamento, Projetos, Sobre nós, Login e Cadastre-se!*. Com exceção da página de *login* e de cadastro, todas as outras páginas base do portal contam com esse mesmo menu e uma imagem do Campus como plano de fundo.



Figura 7 - página inicial do portal

## 6.1 FUNCIONALIDADES DESENVOLVIDAS

Nessa seção, vamos apresentar todas as funcionalidades que foram desenvolvidas de acordo com os requisitos levantados e mencionados anteriormente.

A página *O Campus*, além de conter informações sobre a história do Campus, sua estrutura e localização, conta com uma lista de servidores - cadastrados anteriormente - que trabalham no Campus, e a lista de setores, como podemos ver nas figuras 8 e 9. Além disso, a página conta com um pequeno menu do lado direito da tela com um *link* para a lista de servidores e outro para a de setores, levando o usuário rapidamente para os tópicos em questão sem a necessidade de rolar a tela.

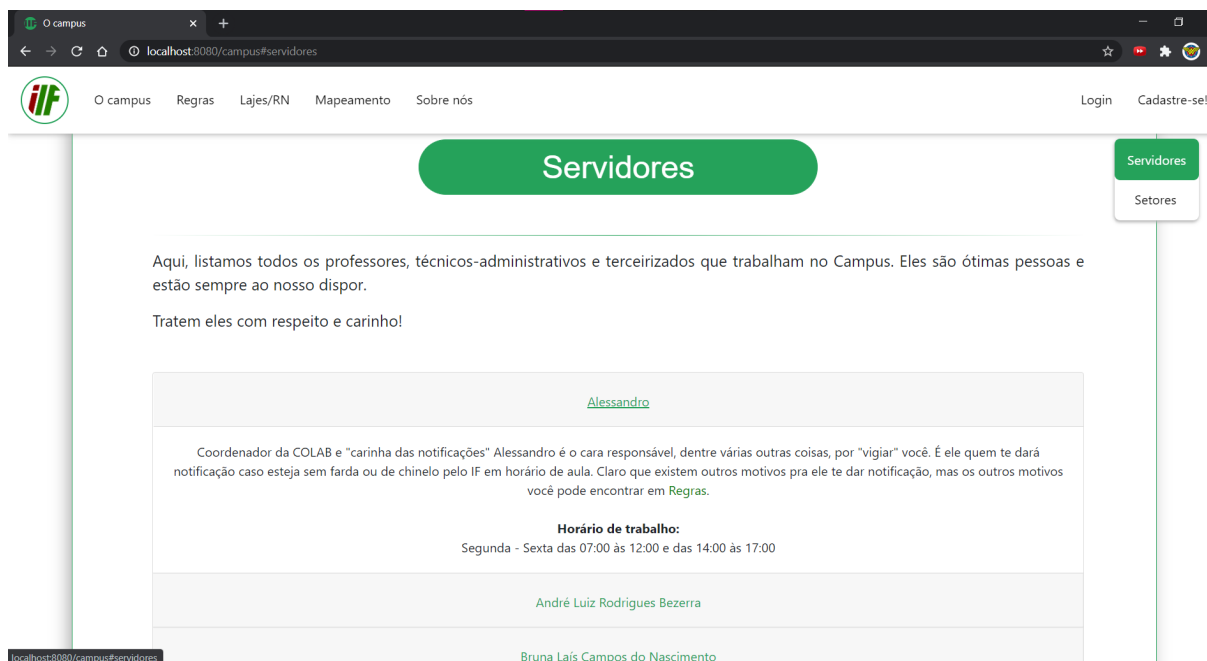


Figura 8 - Página *O Campus*, lista de servidores

| Sector                 | Sigla | Descrição                             | Horário de funcionamento                               | Atendimento                        |
|------------------------|-------|---------------------------------------|--|------------------------------------|
| Direção Acadêmica      | DIAC  | Setor voltado aos assuntos acadêmicos | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | Osailton                           |
| Direção Administrativa | DIAD  | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | A amável Chris (Christiane Maria). |
| Direção Geral          | DG    | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | Helaine                            |
| Secretaria Acadêmica   | SEAC  | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | -                                  |

Figura 9 - Tabela com todos os setores do Campus

Na página *Sobre nós* encontra-se, dentre informações sobre os desenvolvedores, o espaço para comentários e sugestões (figura 10).

Comentários e sugestões

Para que o nosso site tenha sua cara, precisamos da sua ajuda! Para isso, basta apenas nos dar o seu feedback sobre o portal e/ou trazer sugestões sobre o que achar necessário.

Informe seu nome.

Deixa um comentário ou uma sugestão pra gente :)

Opinar

Figura 10 - página *Sobre nós*, campo para comentários

A página de *login* (figura 11), que pode ser acessada pelo aluno, professor, mestre e administrador, possui dois campos para preenchimento: matrícula e senha. Apenas ao fazer o *login* é possível ter acesso às páginas ou a funcionalidades

determinadas ao respectivo uso desses usuários.



Faça login

Matricula:

Senha:

[Entrar](#)

Novo por aqui?! [Crie logo sua conta!](#)

Figura 11 - página de *login*

Caso o usuário não esteja cadastrado no sistema, ele pode se cadastrar indo no *link Cadastre-se!* (que encontra-se no *menu* de todas as páginas irrestritas), ou clicando no *link* logo abaixo do formulário de *login*: *Crie logo sua conta!*. Ao fazer isso, inicialmente, ele terá que informar se é aluno ou servidor - como mostra a figura 12 - para só então ter acesso a página de cadastro (figura 13).

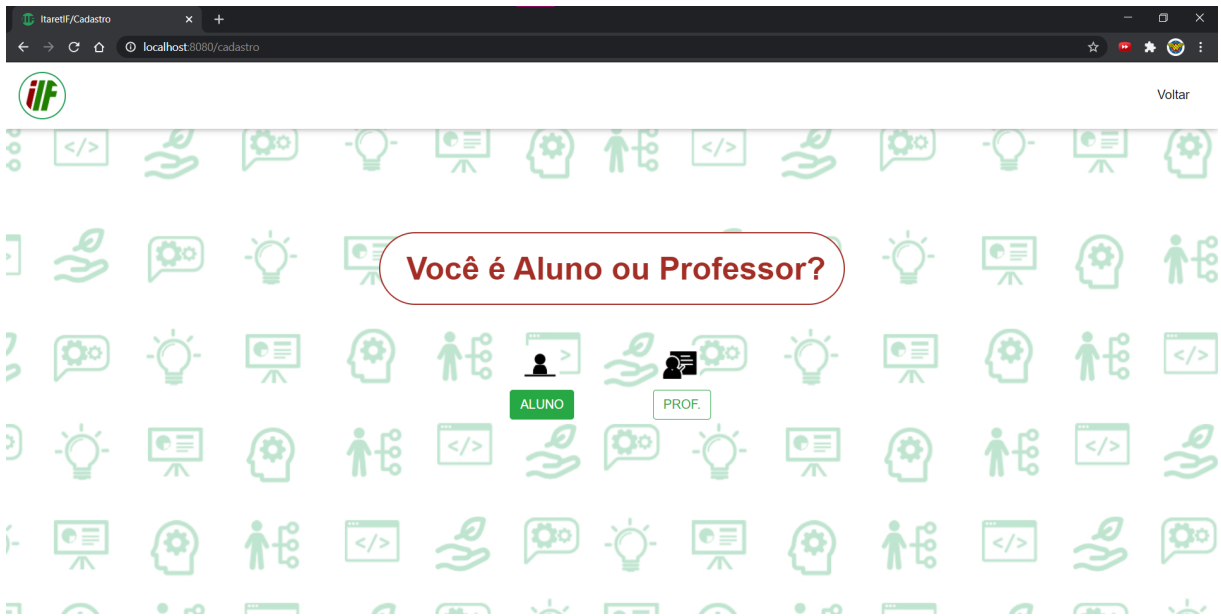


Figura 12 - página de pré-cadastro

O cadastro só será efetivado se o usuário preencher todos os campos, e uma mensagem de sucesso aparecerá para o usuário. Caso contrário, aparecerá uma mensagem de erro na tela.

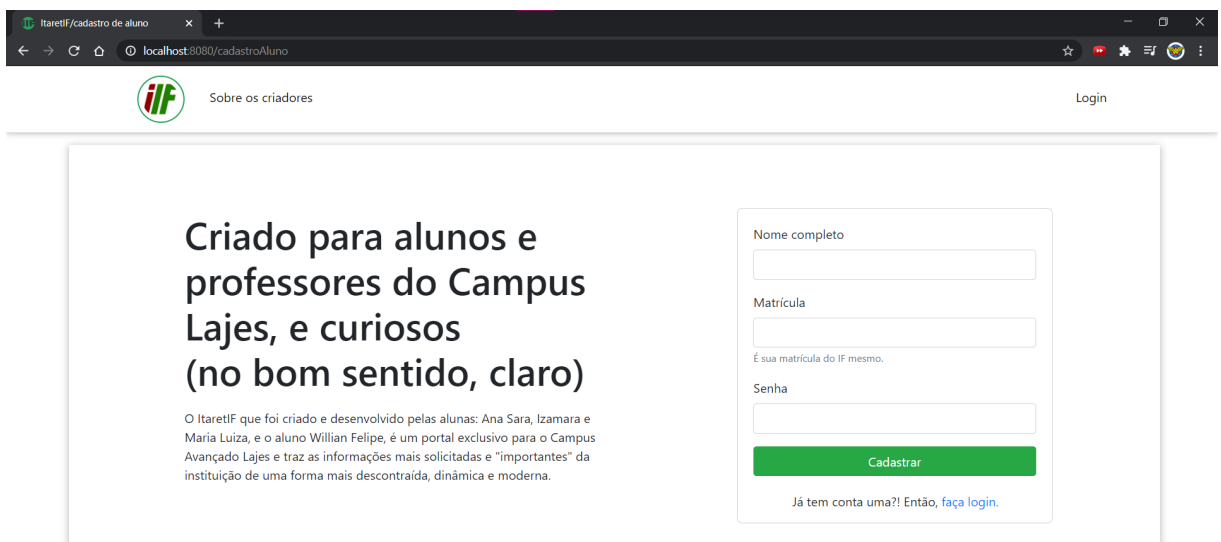


Figura 13 - página com o formulário para cadastro

Na página inicial do(a) administrador(a) há uma barra de navegação com dois *links*: Um para acessar um *menu* com *links* para as listas de cadastros e outro para fazer *logout* (figura 14). No corpo da página temos três formulários: Um para o

cadastro de administradores(as), outro para o cadastro de servidores(as) e o último para o cadastro de setores. Ao finalizar algum desses cadastros, aparecerá uma mensagem de sucesso indicando que o usuário, servidor ou setor foi cadastrado corretamente.

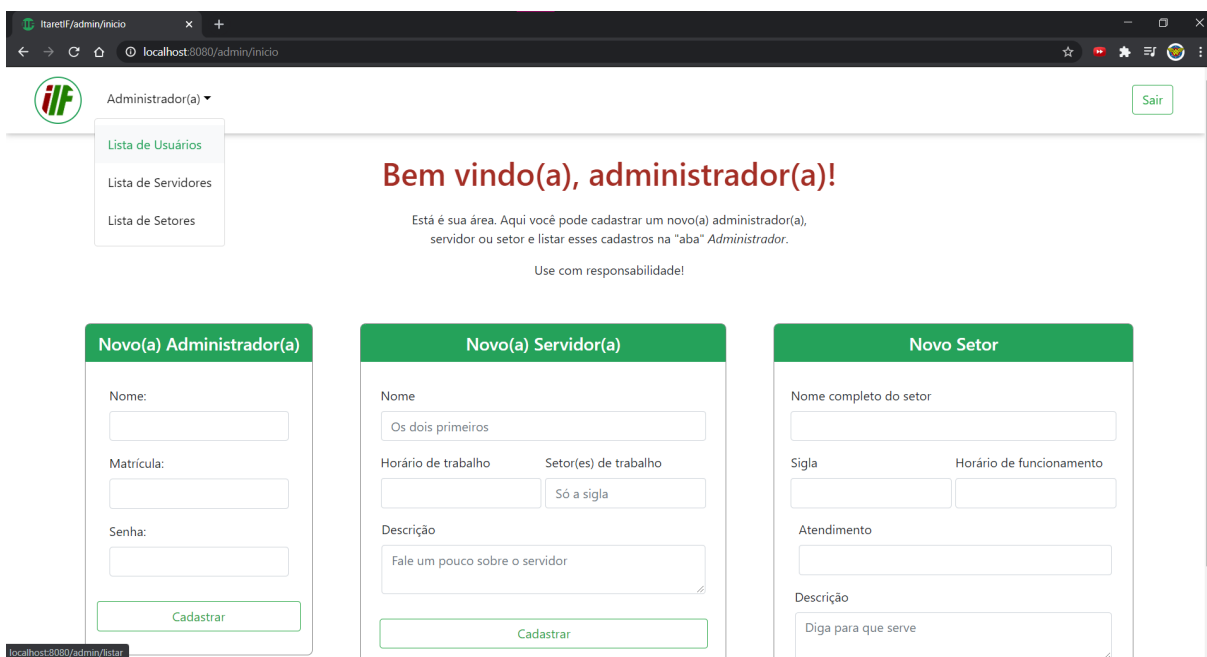


Figura 14 - página inicial do(a) administrador(a)

Ao clicar no *link Lista de Usuários*, o usuário é redirecionado à página com todos os usuários cadastrados que possuem um ou mais papéis (Aluno, Servidor ou Administrador), como demonstra a figura 15. Ao clicar nos nomes dos usuários, somos redirecionados para uma página com mais detalhes sobre o usuário em questão, como podemos ver na figura 16. Também é possível editar (figura 17) ou excluir os usuários.

Administrador(a) Voltar à página inicial

### Lista de usuários cadastrados

| ID: | Nome:                      | Matrícula:    | Ações  |
|-----|----------------------------|---------------|--|
| 1   | Ana Sara da Silva          | 001           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 2   | Izamara Faustino           | 002           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 3   | Nanikka Silva              | 003           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 4   | Willian                    | 004           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 8   | Alexander                  | 0101          | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 12  | Fernando Helton Soares     | 005           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 13  | José Inácio Gomez da Silva | 2019120500064 | <a href="#">Editar</a> <a href="#">Deletar</a> |
| 14  | Fernando Helton Soares     | 003004        | <a href="#">Editar</a> <a href="#">Deletar</a> |

Figura 15 - Lista de usuários cadastrados

Voltar

### Detalhes do Usuário

| ID: | Nome:             | Matrícula: | Papel |
|-----|-------------------|------------|-------|
| 1   | Ana Sara da Silva | 001        |       |

Figura 16 - Página com os detalhes de um usuário



Nome

Matrícula

Senha

Figura 17 - formulário de edição dos dados do usuário

Ao clicar no *link Lista de Servidores*, o usuário é direcionado à página que contém todos os servidores que fazem parte do Campus Lajes, contendo informações como: nome, horário de trabalho, nome do setor em que o servidor(a) fica e a descrição deste (figura 18). Essa lista também aparece na página *O Campus*, página já apresentada anteriormente na figura 8. Também é possível editar (figura 19) ou excluir servidores.

| Nome                              | Horário de Trabalho                                    | Setor | Descrição  | Ações  |
|-----------------------------------|--|-------|--|--|
| Alessandro Nascimento dos Santos  | Segunda - Sexta das 7:00 às 12:00 e das 13:00 às 17:30 | APAC  | Esse é "o cara das notificações". É ele quem vai te dar uma bronca caso você faça algo errado ou não esteja com o fardamento completo. Mas ele é super legal, boe! Serião                        | <input type="button" value="Editar"/> <input type="button" value="Deletar"/> |
| André Luiz Rodrigues Bezerra      |  | DIAC  | Deus magnifico, senhor de tudo, mestre de Arte. Esse é o cara ao qual o Google consulta antes de publicar uma pesquisa. skfdkkk Brincadeiras a parte, o mestre é uma ótima pessoa... aproveitem. | <input type="button" value="Editar"/> <input type="button" value="Deletar"/> |
| Bruna Laís Campos do Nascimento   |  | ASBIB | Ela trabalha na biblioteca, muito simpática e educada (amo). Mas não irrite ela ou ela vai parecer gentilmente assustadora.  | <input type="button" value="Editar"/> <input type="button" value="Deletar"/> |
| Cassia de Fatima Matos dos Santos |  | DIAC  | A melhor professora de literatura e português que já tive na vida. Sério, boe! E ela é tão animadinha e apaixonada pela disciplina... chega a pessoa se empolga junto com ela. Linda perfeita    | <input type="button" value="Editar"/> <input type="button" value="Deletar"/> |

Figura 18 - lista de servidores do Campus

Nome  
Alessandro Nascimento dos Santos

Horário de trabalho  
Segunda - Sexta das 7:00 às 12:00 e das 13:00 às 17:30

Setor de trabalho  
APAC

Descrição  
Esse é "o cara das notificações". É ele quem vai te dar uma bronca caso você faça algo errado ou não esteja com o

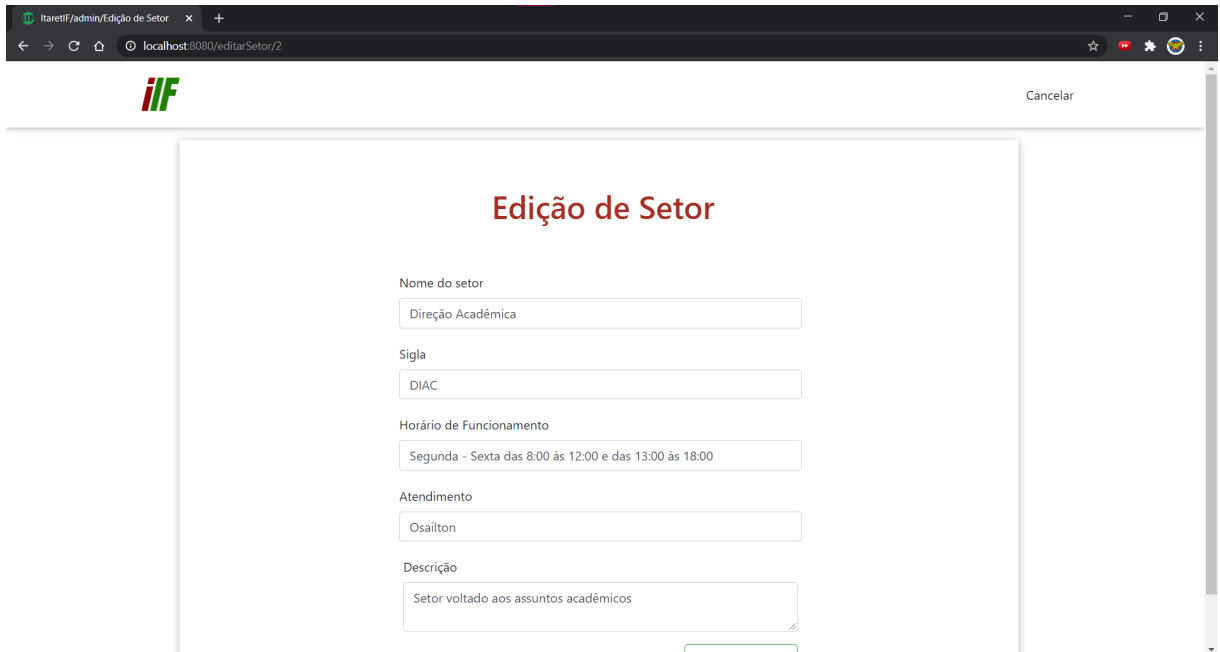
Salvar alterações

Figura 19 - formulário de edição dos dados do servidor

Por fim, ao clicar no *link Lista de Setores*, o usuário será redirecionado à página contendo uma lista com todos os setores da instituição, contendo informações como: o nomes dos setores, a sigla, uma breve descrição sobre o setor, o horário de funcionamento deste e o nome dos servidores que trabalham em cada um deles (figura 20). Também é possível editar (figura 21) ou excluir os setores.

| Nome do Setor          | Sigla | Descrição                             | Horário  | Atendimento                        | Ações  |
|------------------------|-------|---------------------------------------|--|------------------------------------|--|
| Direção Acadêmica      | DIAC  | Setor voltado aos assuntos acadêmicos | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | Osailton                           | <a href="#">Editar</a> <a href="#">Deletar</a> |
| Direção Administrativa | DIAD  | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | A amável Chris (Christiane Maria). | <a href="#">Editar</a> <a href="#">Deletar</a> |
| Direção Geral          | DG    | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | Helaine                            | <a href="#">Editar</a> <a href="#">Deletar</a> |
| Secretaria Acadêmica   | SEAC  | -                                     | Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00 | -                                  | <a href="#">Editar</a> <a href="#">Deletar</a> |

Figura 20 - lista de setores cadastrados



The image shows a web browser window with the URL 'localhost:8080/editarSetor/2'. The page title is 'Itareif/admin/Edição de Setor'. The main content area is titled 'Edição de Setor' and contains a form with the following fields:

- Nome do setor: Direção Acadêmica
- Sigla: DIAC
- Horário de Funcionamento: Segunda - Sexta das 8:00 às 12:00 e das 13:00 às 18:00
- Atendimento: Osailton
- Descrição: Setor voltado aos assuntos acadêmicos

A 'Cancelar' button is visible in the top right corner of the page.

Figura 21 - formulário de edição dos dados do setor

Ao estudarmos o Portal IFRN, percebemos que os editais e projetos desenvolvidos pelo Campus não apresentam grande visibilidade, dificultando a divulgação de tais informações necessárias tanto para a comunidade interna quanto para a comunidade externa do instituto. Pensando nesta problemática, resolvemos implementar outra funcionalidade no nosso portal, uma página específica para comportar tais informações (figura 22).



Figura 22 - Página de projetos

Como vemos na figura 22, ao clicar no botão “Novo projeto”, aparecerá os campos “Título”, “Autor”, e “Descrição” (figura 23). Obrigatoriamente o coordenador do projeto irá preencher todos para que o projeto seja devidamente cadastrado. Caso um campo não receba o devido preenchimento, aparecerá uma mensagem de alerta exigindo o complemento.

Figura 23 - Formulário para adicionar um novo projeto

Ao preencher a descrição, ele também poderá adicionar um estilo, podendo

escolher a fonte, a adição de tabelas, imagens e vídeos e entre outras opções de edição. Ao cadastrar, o projeto se juntará à lista de projetos, como podemos ver na figura 22, onde poderemos detalhar cada um e ler por completo. Na página onde ficará o detalhamento, colocamos um botão para deletar o projeto quando ele chegar ao fim e outro para editar, caso precise alterar alguma informação (figura 24). Mas essas opções só podem ser executadas pelo servidor ou administrador.



Figura 24 - Detalhamento do projeto

Essas funcionalidades não podem ser executadas e administradas por qualquer pessoa, logo, durante o processo de idealização do nosso sistema, percebemos a necessidade de que determinadas páginas e funcionalidades tivessem acesso restrito, podendo ser acessadas somente após a autenticação e autorização do usuário por meio de uma matrícula e senha. À vista disso, criamos a funcionalidade e a página para o *login* do usuário.

## 6.2 IMPLEMENTAÇÃO

O foco deste trabalho está no cadastro de alunos, professores/mestres e servidores, pela página dos administradores e pelo *login*. Porém, iremos discorrer apenas sobre a funcionalidade de *login*.

Inicialmente, o diagrama de sequência dessa funcionalidade foi desenvolvido, a fim de explicar tal aplicabilidade com um maior dinamismo e de uma forma mais detalhada. O diagrama de sequência é a forma gráfica que demonstra a ordem com a qual um determinado processo entre objetos é executado pelo *software*. O principal objetivo do diagrama de sequência é definir quais objetos irão interagir entre si durante determinado processo, como esses objetos irão de fato interagir, a ordem em que os eventos devem ocorrer, os métodos que serão invocados e as mensagens que serão enviadas e/ou trocadas.

Como o diagrama de sequência é baseado no diagrama de caso de uso, para cada caso de uso de relação direta com o ator, existe um diagrama de sequência. Logo, já que a funcionalidade em questão tem relação direta com o ator (representação de um agente externo que interage com o sistema), foi necessário desenvolver seu diagrama de sequência, apresentado na figura abaixo.

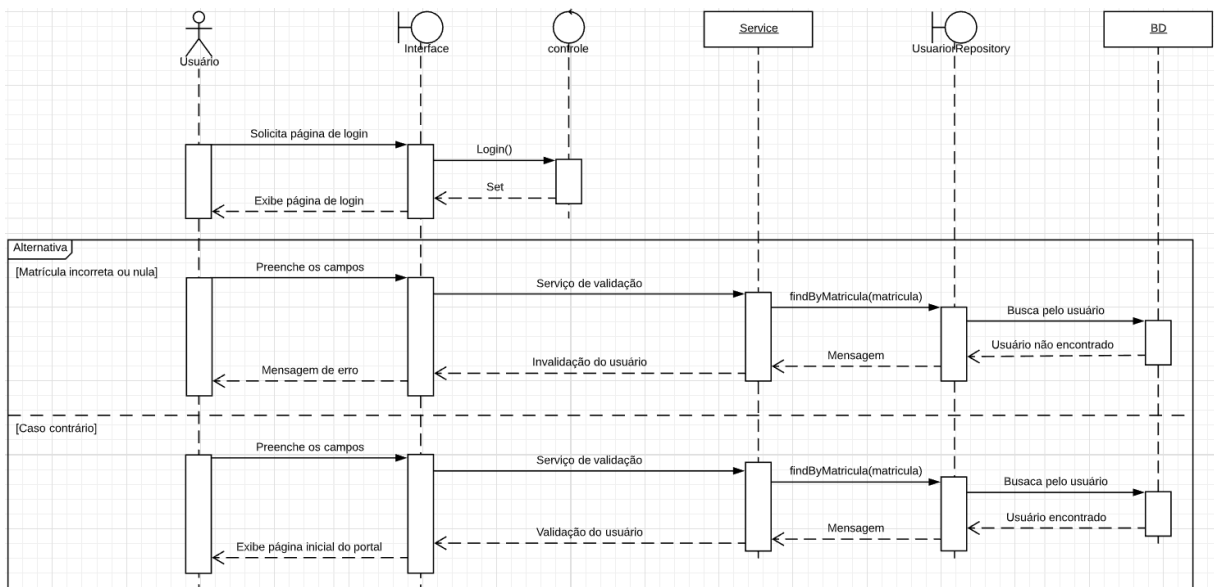


Figura 25 - Diagrama de Sequência da função de *Login*

Quando o usuário solicita a página de login, a interface envia uma mensagem para o *controller* “ControllerLogin” e este faz a invocação do método *set* responsável por retornar a página de *login*. A interface exibe a página de login para o usuário e este insere sua matrícula e senha. Em seguida, a interface envia os dados para o *service* do *Spring* que tratará de fazer o serviço de validação. O *service* vai chamar o repositório do usuário para invocar o método *findByMatricula* e, então, o repositório

enviará uma mensagem ao banco de dados para que este faça uma busca pelo usuário. Neste momento, temos duas condições: se o usuário não for encontrado - matrícula nula ou incorreta - o banco retornará ao repositório que o usuário não foi encontrado e o repositório passará essa mensagem ao *Service*. Dessa forma, o usuário não será autorizado e uma mensagem de erro será retornada à página de *login*; se o usuário for encontrado, o banco passará essa informação ao repositório que, por sua vez, retornará o usuário ao *Service*, de modo que o usuário seja autorizado. Por fim, a interface exibirá a página inicial do portal ao usuário.

Para implementação desta funcionalidade, utilizei o *spring security* (um dos projetos *spring* que faz parte do ambiente *spring*), pois é um projeto que simplifica e facilita a criação e configuração do processo de autenticação e autorização do usuário.

Inicialmente, criou-se uma classe chamada “WebSecurityConfig” (figura 26), anotada com o *@Configuration*, porquanto é uma classe de configuração, responsável pela configuração de segurança, que, por padrão, estende uma outra classe do próprio *spring*, a *WebSecurityConfigurerAdapter*. Nessa classe, foram informadas as URI’s que vão estar envolvidas no processo de autenticação e autorização, tanto as que precisam de autorização quanto as que não precisam, além de informar qual é o(s) usuário(s) que pode acessar determinada URL.

Na linha 13, sobrescrevemos o método *configure*, utilizando a anotação *@Override*, que, por sua vez, recebe como parâmetro um *HttpSecurity*, ferramenta de segurança da *Web*. Na linha 17, com a utilização do método *antMatchers*, foi determinado quais são as URI’s que podem ser acessadas sem a necessidade de *login*, e nas linhas 20 e 21, quais as URI’s que implica fazer *login*.

Como representado na figura 5 (diagrama de caso de uso), para poder acessar o portal e navegar por ele, não é necessário fazer login, sendo assim, a URI de acesso permitido a todos é a */\** (linha 17). Esse parâmetro (*/\**), significa dizer que qualquer URL que use apenas uma barra ou uma barra e um ou mais termos (representado(s) pelo asterisco), como, por exemplo, “/lajes”, poderá ser acessada por qualquer usuário.

Como também já foi abordado, para poder comentar, postar e gerenciar um

projeto, cadastrar novos usuários, setores, servidores, ter acesso às listas desses cadastros, editá-los ou excluí-los, é necessário, antes, fazer *login* com a matrícula e a senha. No entanto, como vimos nos casos de uso (figura 5), o usuário pode ser aluno, servidor ou administrador e cada um desses papéis possibilita o acesso a diferentes páginas e funcionalidades. Quem pode postar e gerenciar projetos são somente os professores/mestres do Campus e quem pode cadastrar setores, servidores, novos usuários, administradores e ver as listas de cadastros são somente os administradores. Dessa forma, nas linhas 20 e 21 foi determinado que as URI's que comecem com */serv* ou */admin* só poderão ser acessadas pelos usuários cujos papéis sejam SERVIDOR e ADMIN, respectivamente. Ou seja, caso alguém queira acessar, por exemplo, a URL “*/admin/inicio*” (que leva para a página inicial dos administradores) terá que, além de fazer *login* com sua matrícula e senha, ter sido cadastrado no portal com o papel ADMIN. Nas linhas 22 e 23 foi determinado que os métodos cujo mapeamento seja “*/deletarProjeto/{id}*” e “*/editarProjeto/{id}*” só poderão ser efetivados por usuários cujos papéis sejam SERVIDOR e ADMIN e SERVIDOR, nesta ordem, como mostra a figura 26.

Na linha 28, por meio do método *loginPage*, foi determinado que a página de *login* (permitida a todos) seja retornada ao usuário quando ele solicitar alguma página de acesso restrito. Na linha 32, por meio do método *logoutSuccessUrl*, indicamos a página a ser retornada no momento do *logout*, que é a página de *login*.

O segundo método sobrescrito (linha 38), que recebe como parâmetro o *WebSecurity*, é referente ao acesso às páginas estáticas do projeto (css, de imagens, etc) pelas páginas *html*. A ordem impede, por meio do método *ignoring*, que as pastas em que os estilos e as imagens das páginas *html* se encontram sejam bloqueadas, evitando, assim, que as páginas não consigam acessar os arquivos css e de imagem e fiquem sem configuração visual. Na linha 39 foi indicado o caminho das pastas que devem ser ignoradas pelo *spring security*.



```

WebSecurityConfig.java
1 package edu.ifrn.itaretif.security;
2
3 import org.springframework.context.annotation.Configuration;
7 //import org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder;
8
9 @Configuration
10 public class WebSecurityConfig extends WebSecurityConfigurerAdapter {
11
12     @Override
13     protected void configure(HttpSecurity http) throws Exception {
14
15         http.
16             authorizeRequests()
17                 .antMatchers("/*").permitAll()
18                 .antMatchers("/projeto/{id}").permitAll()
19                 /*.antMatchers("/sobre/comentar").hasRole("ALUNO") */
20                 .antMatchers("/serv/**").hasRole("SERVIDOR")
21                 .antMatchers("/admin/**").hasRole("ADMIN")
22                 .antMatchers("/deletarProjeto/{id}").hasAnyRole("SERVIDOR", "ADMIN")
23                 .antMatchers("/editarProjeto/{id}").hasRole("SERVIDOR")
24                 .anyRequest()
25                 .authenticated()
26             .and()
27             .formLogin()
28                 .loginPage("/login")
29                 .permitAll()
30             .and()
31             .logout()
32                 .logoutSuccessUrl("/login?logout")
33                 .permitAll();
34
35     }
36
37     @Override
38     public void configure(WebSecurity web) throws Exception {
39         web.ignoring().antMatchers("/estilo/**", "/img/**");
40     }
41
42 }

```

Figura 26 - Classe de configuração de segurança (autenticação e autorização)

Em seguida, criou-se a classe “ImplementsUserDetailsService”, responsável por buscar os usuários na base de dados, como mostra a figura 27. Essa classe foi anotada com `@Component`, pois será gerenciada, e estende a interface “UserDetailsService”. Na linha 19, sobrescrevemos o método `loadUserByUsername` e passamos a matrícula do usuário como parâmetro. Na linha 20, fazemos a busca pelo usuário no banco de dados por meio da matrícula, invocando o método da interface “UsuarioRepository”, `findByMatricula` (por isso referenciamos a interface “UsuarioRepository”, para que fosse feito a injeção de dependência desta, como podemos ver na linha 16). Na linha 22 usamos um `sysout` para que a matrícula do usuário fosse impressa no console e testar se o usuário estava, de fato, sendo encontrado. E, por fim, da linha 24 à 26 utilizamos um `if`, para que a mensagem

“Este usuário não foi encontrado!” seja retornada caso o usuário que for passado seja nulo. Caso ele seja encontrado, isto é, não seja nulo, o método nos retornará o usuário (linha 27). Ao retornar o usuário, estamos retornando o tipo “UserDetails”, interface estendida no *model* “Usuario” (O *model* “Usuario” precisou importar a interface “UserDetails” porque essa interface possui alguns métodos necessários à autenticação do *spring security*).

```
1 package edu.ifrn.itaretif.security;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4
5
6
7
8
9
10
11
12 @Component
13 public class ImplementsUserDetailsService implements UserDetailsService {
14
15     @Autowired
16     private UsuarioRepository ur;
17
18     @Override
19     public UserDetails loadUserByUsername(String matricula) throws UsernameNotFoundException {
20         Usuario usuario = ur.findByMatricula(matricula);
21
22         System.out.println(usuario);
23
24         if (usuario == null) {
25             throw new UsernameNotFoundException("Este usuário não foi encontrado!");
26         }
27         return usuario;
28     }
29 }
30
31 }
```

Figura 27 - Classe responsável pela busca do usuário no banco de dados

Quando o usuário solicita a página de *login*, o *controller* “ControllerLogin” retorna a ele a página de *login* para que seja efetuada a autenticação por meio da sua matrícula e senha e a página solicitada liberada. A figura 28 mostra o arquivo *html* que exibe o formulário de *login*. Nas linhas 29 e 34 temos os *inputs* que recebem, respectivamente, a matrícula e a senha do usuário e, por padrão do *spring security*, têm por nomes: *username* e *password*. Na linha 37 temos o botão (do tipo *submit*) responsável por ativar a ação do formulário, para que, então, seja exibida ao usuário a página solicitada. Logo abaixo, na linha 38, colocamos um *input* oculto com um *token*, a fim de que o *spring* valide nosso formulário e possibilite o seu envio. E, por último, na linha 43 temos uma *tag* de parágrafo com uma pergunta e um *link* que redireciona o usuário à página de pré-cadastro (figura 12) caso ele ainda não tenha um.

```

login.html x
10 <link rel="stylesheet" type="text/css" href="estilo/img.css" />
11 <link rel="stylesheet" type="text/css" href="estilo/cadastro.css">
12 </head>
13 <body>
14
15 <div class="titulo">
16 <a href="/"></img></a>
17
18 <h1>Faça login</h1>
19 </div>
20
21 <div class="cont">
22 <th:block th:include="/login/mv"></th:block>
23
24 <div class="box">
25 <form action="" method="post">
26 <div class="form-group">
27
28 <label for="Matricula">Matricula:</label>
29 <input type="text" class="form-control" id="username" aria-describedby="username" name="username">
30
31 </div>
32 <div class="form-group">
33 <label for="Password">Senha:</label>
34 <input type="password" class="form-control" id="password" name="password">
35 </div>
36
37 <button type="submit" class="btn btn-success dimensoes">Entrar</button>
38 <input type="hidden" th:name="${_csrf.parameterName}" th:value="${_csrf.token}" />
39
40 </form>
41
42 <div class="caixa">
43 <p>Novo por aqui?! <a href="/cadastro">Crie logo sua conta!</a></p>
44 </div>
45
46 </div>
47 <div class="limpar"></div>
48 </div>
49 </body>
50 </html>

```

Figura 28 - Página html com o formulário para *login*

Como não é seguro deixar as senhas dos usuários serem gravadas limpas na base de dados, elas precisam ser criptografadas na hora do cadastro. No entanto, isso fará com que a senha tenha uma forma extensa e o *spring*, por si só, não consiga fazer a relação da senha salva no banco com a senha que o usuário digitou. Por isso, criou-se uma classe auxiliar chamada “SecurityConfiguration” (figura 29), a qual anotamos com *@Component*, para implementar o método de criptografia de senha. Da linha 12 à 13 fizemos a injeção de dependências da classe “ImplementesUserDetailsService”. Da linha 15 à 19 temos uma injeção do método *configureGlobal*, que tem como parâmetro o “AuthenticationManagerBuilder” e o objeto a ser configurado, que é o *builder*, para que o *spring* consiga transformar a senha que o usuário digitar em uma *hash* que seja equivalente a senha que está no salva no banco de dados. E, por último, da linha 21 à 24, temos a bean com o método *passwordEncoder*, responsável por, de fato, criptografar as senhas dos usuários no momento do cadastro.

```
SecurityConfiguration.java x
1 package edu.ifrn.itaretif.security;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4
5
6
7
8
9 @Component
10 public class SecurityConfiguration {
11
12     @Autowired
13     private ImplementsUserDetailsService detailsService;
14
15     @Autowired
16     public void configureGlobal(AuthenticationManagerBuilder builder) throws Exception {
17
18         builder.userDetailsService(detailsService).passwordEncoder(new BCryptPasswordEncoder());
19     }
20
21     @Bean
22     public BCryptPasswordEncoder passwordEncoder() {
23         return new BCryptPasswordEncoder();
24     }
25 }
```

Figura 29 - Classe auxiliar para a criptografia das senhas

## 7 ANÁLISE E DISCUSSÃO DOS RESULTADOS

O portal ItaretIF foi implementado de acordo com o planejado e atende a todos os requisitos que foram levantados previamente à etapa de desenvolvimento dos casos de uso, desempenhando de forma satisfatória as funcionalidades de cadastro e gerenciamento de projeto, cadastro de aluno(a), servidor(a), administrador(a), setor, fazer *login* e comentar. Além disso, as informações sobre o Campus e o município de Lajes/RN, sobre as regras e normas de fardamento da instituição - bem como as punições para o seu descumprimento - as plantas do prédio principal e do anexo também foram incorporadas ao portal de acordo com o previsto, com dinamismo, de forma intuitiva e com um dialeto menos formal.

Alguns ajustes ainda precisam ser feitos quanto a funcionalidade de cadastro de alunos(as) e servidores(as), visto que, antes do usuário ser direcionado(a) à página de cadastro, faz-se necessário que ele(a) informe ao sistema a qual condição ele(a) pertence - Aluno(a) ou Servidor(a) - o que não torna o processo tão eficaz. Seria mais interessante que o indivíduo pudesse informar sua categoria na mesma página em que a matrícula e a senha são informadas, isto é, na página de cadastro propriamente dita, com o intuito de potencializar tal aplicabilidade. Entretanto, esse detalhe não interfere no desempenho da funcionalidade em questão.

Diante dos resultados obtidos, apesar das dificuldades durante o desenvolvimento do projeto e da aplicação, percebe-se que os objetivos propostos foram atingidos. O portal atende às especificações e expectativas apresentadas no decorrer deste relatório, cumprindo com o seu objetivo principal: disponibilizar e facilitar o acesso às informações sobre o Campus Avançado Lajes com uma linguagem familiar aos usuários, os ingressantes do Campus em questão e o público externo.

A criação do ItaretIF contribui não apenas para a disponibilidade e facilidade do acesso à informação, no que tange o Campus Avançado Lajes/RN, como

também para o crescimento da instituição e melhoria da integração do público interno e externo com o polo. Ademais, tal prática proporcionou o aperfeiçoamento das nossas habilidades e dos conhecimentos adquiridos ao longo do curso, uma vez que teoria e prática operaram juntas.

## 8 CONSIDERAÇÕES FINAIS

Baseado na dificuldade que alguns alunos do IFRN - Campus Lajes sentiam em encontrar certas informações no portal oficial do IFRN e sabendo que os novatos não conhecem a instituição ao ingressarem nela, sejam eles de outras cidades ou da cidade do polo, foi criado um portal exclusivo para o Campus Avançado Lajes, para melhorar a integração do público que quer obter uma maior percepção sobre o Campus, sua importância e organização, facilitando o acesso a essas informações, além de possibilitar isso de uma forma mais leve e descontraída. Além disso, é de fácil acesso, pois pode ser acessado através da *internet*, basta ter um dispositivo e uma conexão com a *internet*.

Se levarmos em consideração o objetivo proposto, o esperado é que o portal realmente ajude a dar aos alunos, servidores e comunidade lajense, uma ampla visão acerca da estrutura, atividades e funcionamento do Campus e que agrade no visual e na forma de abordar os assuntos. A nível de desempenho, esperamos que o sistema cadastre os usuários e os relacione com seus respectivos papéis, execute o processo de autenticação e autorização do usuário de forma correta, exiba as listas de cadastro aos administradores para possíveis consultas, alterações ou exclusões, realize o cadastro e postagem dos projetos dos servidores e os disponibilize para que todos os usuários possam acessar através da *internet*.

Ao longo do desenvolvimento do projeto foram aplicados os conhecimentos adquiridos no decorrer do curso técnico em informática do IFRN, em que a aplicação desses conhecimentos pode ser utilizada para solucionar e/ou amenizar um “problema” real da comunidade do Campus.

Averiguando as instâncias dissertadas no projeto, seus objetivos e o tempo dado, podemos concluir que todas as funcionalidades propostas foram implementadas. Como já mencionado neste relatório, o código-fonte deste trabalho está disponível no Github no seguinte repositório: <https://github.com/izamara/itaretif>.

Mesmo com o projeto finalizado, podemos sugerir algumas propostas de

implementações futuras ao modelo desenvolvido neste trabalho, para a melhoria e aperfeiçoamento das funcionalidades implementadas, tais como: a ampla utilização da linguagem de programação JavaScript, possibilitando uma gama maior de funcionalidades relacionadas a dinâmica do portal; a otimização na página de autocadastro e nos métodos envolvidos no processo, para que tanto o aluno quanto o servidor possam se cadastrar através da mesma página, e a implantação do portal no IFRN – Campus Avançado Lajes; Ao integrar essas sugestões ao sistema já desenvolvido, pode-se ajudar ainda mais o portal a cumprir com seu objetivo e ainda aprimorar os conhecimentos dos alunos técnicos em informática.



## REFERÊNCIAS

DUCKETT, Jon. HTML & CSS projete e construa websites, Rio de Janeiro, 2016. Disponível em: <<https://www.passeidireto.com/arquivo/75371077/html-and-css-projete-e-construa-websites>> Acesso em: 20 de Fevereiro de 2020.

MEIRA, Regilan. Apostila de programação para web: HTML e CSS, Bahia, 2013. Disponível em: <[http://www.regilan.com.br/wp-content/uploads/2013/10/Apostila-Programacao-WEB\\_HTML\\_CSS.pdf](http://www.regilan.com.br/wp-content/uploads/2013/10/Apostila-Programacao-WEB_HTML_CSS.pdf)> Acesso em: 08 Nov. 2019.

Desenvolvimento Web com HTML, CSS e JavaScript. Caelum. Disponível em: <<https://www.caelum.com.br/apostila-html-css-javascript/>> Acesso em: 08 Nov. 2019.

ALVIM, Paulo. Tirando o Máximo do Java EE 6 Open Source com jCompany® Developer Suite. 3. Ed. Belo Horizonte: Powerlogic Publishing, 2010.

FARIA, Fernanda. *et al.* Evolução e Principais Características do IDE Eclipse, Goiás, São Paulo, 2010. Disponível em <[https://www.enacomp.com.br/2010/anais/artigos/completos/enacomp2010\\_23.pdf](https://www.enacomp.com.br/2010/anais/artigos/completos/enacomp2010_23.pdf)> Acesso em: 20 de Fevereiro de 2020.

MOREIRA JR, Normandes José; AFONSO, Alexandre. Produtividade no Desenvolvimento de Aplicações Web com Spring Boot. 2. ed. Uberlândia: AlgaWorks Softwares, Treinamentos e Serviços Ltda, 2017.

SCHMITZ, Daniel. Tudo que você queria saber sobre Git e GitHub, mas tinha vergonha de perguntar. TABLELESS, 2015. Disponível em: <<https://tableless.com.br/tudo-que-voce-queria-saber-sobre-git-e-github-mas-tinha-vergonha-de-perguntar/>>. Acesso em: 17 de Março de 2020.

DEVMEDIA, Bruno dos Santos Nascimento. MySQL. Disponível em: <<https://www.devmedia.com.br/mysql-tutorial/33309>> Acesso em: 20 jul. 2019.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. UML: Guia do Usuário. Rio de Janeiro: Elsevier Brasil, 2006.

Pesquisa-Ação – André Fontenelle | Como fazer um TCC. Andrefontenelle.com.br. Disponível em: <<https://novo.andrefontenelle.com.br/tag/pesquisa-acao/#site-header>>. Acesso em: 25 Outubro 2020.

NAÍNA TUMELERO. Metodologia de pesquisa: guia rápido de possibilidades. Disponível em: <<https://blog.mettzer.com/metodologia-de-pesquisa/>>. Acesso em: 20 Outubro de 2020.

MATHEUS DIAS. Pesquisa Quantitativa | Metodologia, Questionário e Coleta de Dados. Instituto OPUS | Pesquisa Eleitoral, de Mercado e Opinião Pública. Disponível em: <<https://www.opuspesquisa.com/blog/tecnicas/pesquisa-quantitativa/>>. Acesso em: 20 Outubro de 2020.

MySQL, Introdução ao MySQL. Disponível em: <<https://dev.mysql.com/doc/mysql-getting-started/en/#mysql-getting-started-basic-ops>>. Acesso em: 12 novembro de 2019.

HTML & CSS projete e construa websistes. Disponível em: <[https://www.academia.edu/36660331/HTML\\_and\\_CSS\\_projete\\_e\\_construa\\_websit es?auto=download](https://www.academia.edu/36660331/HTML_and_CSS_projete_e_construa_websit_es?auto=download)>. Acesso em: 11 de Março de 2020.

Getting Started, O que é um Framework? Entendendo o conceito. Disponível em: <<https://tableless.github.io/iniciantes/manual/js/o-que-framework.html>> Acesso em: 08 de novembro de 2019.

IBM, Chris Aniszczyk e David Gallardo. Introdução à Plataforma Eclipse. Disponível em: <<https://www.ibm.com/developerworks/br/library/os-eclipse-platform/index.html>> Acesso em: 02 ago. 2019.

PRODUTIVIDADE NO DESENVOLVIMENTO DE APLICAÇÕES WEB COM SPRING BOOT. Disponível em: <<http://s3.amazonaws.com/algaworks-assets/ebooks/algaworks-livro-spring-boot-v2.0.0>> Acesso em: 17 de Março de 2020.

DEVMEDIA, Eduardo. Desenvolvendo uma Aplicação Web com Spring Boot e Spring MVC. Disponível em: <<https://www.devmedia.com.br/developendo-uma-aplicacao-web-com-spring-boot-e-spring-mvc/34122>> Acesso em: 08 ago. 2019.

DEVLOG, Hélio Márcio Filho. Web Service RESTful Spring Boot 2.0 com múltiplos banco de dados. Disponível em: <<https://medium.com/dev-log/web-service-restful-spring-boot-2-0-com-m%C3%BAltiplos-bancos-de-dados-ff1676b412cf>> Acesso em: 12 ago. 2019

Spring By Pivotal, Apresentando o Spring. Disponível em: <<https://docs.spring.io/spring-boot/docs/2.2.1.RELEASE/reference/html/getting-started.html#getting-started>> Acesso em: 08 de novembro de 2019.

Introdução à Plataforma Eclipse. IBM, 2012. Disponível em: <<https://www.ibm.com/developerworks/br/library/os-eclipse-platform/index.html>> Acesso em: 10 set.2019.

Evolução e Principais Características do IDE Eclipse. Disponível em: <[https://www.enacomp.com.br/2010/anais/artigos/completos/enacomp2010\\_23.pdf](https://www.enacomp.com.br/2010/anais/artigos/completos/enacomp2010_23.pdf)>. Acesso em: 02 de Março de 2020.

Eclipse documentation - Current Release, Help - Eclipse Platform. Disponível em: <<https://help.eclipse.org/2019-09/index.jsp>> Acesso em: 08 de novembro de 2019.

GitHubGuides, O que é o GitHub?. Disponível em: <<https://guides.github.com/activities/hello-world/>> Acesso em: 14 de novembro de 2019.

GitHubGuides, Documenting your projects on GitHub. Disponível em: <<https://guides.github.com/features/wikis/>> Acesso em: 14 novembro de 2019.

Desenvolvimento de Sistemas de Informação: Da Construção de Sistemas Informáticos à Reengenharia Organizacional. Disponível em: <<http://piano.dsi.uminho.pt/~jac/SI/zdocumentos/DSI.pdf>> Acesso em: 09 de Março de 2020.

MARYLENE GUEDES. O que é MVC? - Blog da TreinaWeb. Blog da TreinaWeb. Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-mvc/>>. Acesso em: 14 Novembro de 2020.

O que é Spring Security? Blog da AlgaWorks. Disponível em: <<https://blog.algaworks.com/spring-security/>>. Acesso em: 14 Novembro de 2020.

GILBERTO CALEGARI BRANDÃO. Top 10 Spring Framework Annotations para desenvolvedores Java. Medium. Disponível em: <<https://medium.com/@gcbrandao/top-10-spring-framework-annotations-para-desenvolvedores-java-ecc1c3b11dae>>. Acesso em: 14 Novembro de 2020.

## ANEXO A – FORMULÁRIO DE IDENTIFICAÇÃO

| <b>Dados do Relatório Científico</b>  |                 |
|---|-----------------|
| Título e subtítulo: ItaretIF - Um portal para divulgação de informações no campus Lajes.  |                 |
| Tipo de relatório: Técnico-Científico.  | Data:           |
| Título do projeto/ programa/ plano: ItaretIF.   |                 |
| Autor(es): Ana Sara da Silva Clementino; Maria Izamara Faustino; Maria Luiza da Silva Matias; Willian Felipe da S. dos Santos.  |                 |
| Instituição e endereço completo: Instituto Federal de Educação, Ciência e Tecnologia - Campus Lajes. BR-304, Km 120, s/n - Centro, Lajes - RN, 59535-000.   |                 |
| Resumo: Após a aprovação no processo seletivo do IFRN – Lajes os estudantes ingressam na instituição sem saber como ela funciona, quais são as normas de conduta exigidas para eles e quais são os setores existentes e seus responsáveis. A maioria dessas informações podem ser encontradas no portal oficial da instituição, porém, a linguagem utilizada nos documentos e a quantidade de informações contidas neste portal, dificulta o entendimento e a pesquisa dessas normas, fazendo com que esse portal não seja atrativo para os alunos. Além disso, há também uma dificuldade em encontrar onde estão localizadas as salas, laboratórios e setores. Neste sentido, este projeto propõe a criação de um portal capaz de abrigar as informações mais solicitadas pelos estudantes ingressantes no Campus Avançado Lajes. Com o objetivo de proporcionar aos alunos, e também a população externa, informações sobre a instituição com a linguagem utilizada no cotidiano desses alunos, tornando o portal atrativo e de fácil entendimento. |                 |
| Palavras-chave/descriptores: informática, informação, campus, Lajes, instituição, portal.   |                 |
| Nº de páginas:  |                 |
| Jornada de trabalho:  | Horas semanais: |

|                    |
|--------------------|
| Total de horas:    |
| Observações/notas: |

Os anexos consistem em um texto ou documento não elaborado pelo autor, que serve de fundamentação, comprovação e ilustração. Os anexos são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos.