

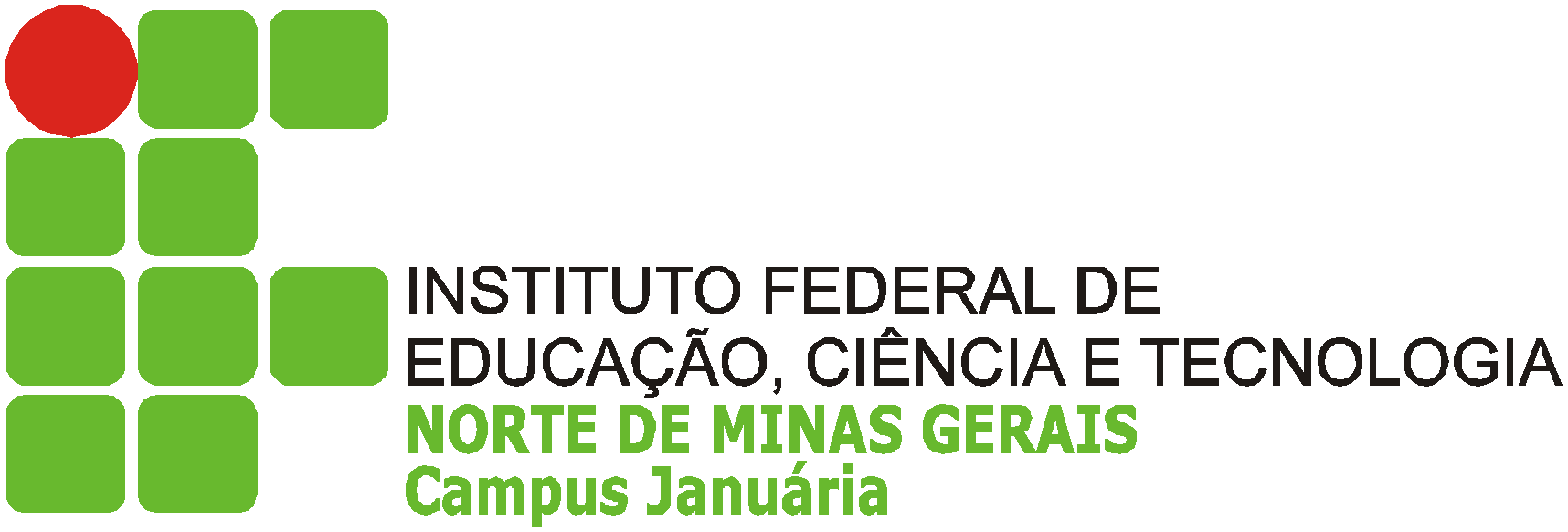
**Tecnologia em Análise e Desenvolvimento de Sistemas**

**Trabalho de Conclusão de Curso**

**Sistema de Gerenciamento de Estágios**

**THALES VINICIUS ALVES DE BRITO**

**Januária – MG, Março de 2016.**



**Tecnologia em Análise e Desenvolvimento de Sistemas**

**Trabalho de Conclusão de Curso**

**Sistema de Gerenciamento de Estágios**

**THALES VINICIUS ALVES DE BRITO**

**Projeto apresentado à Coordenação de TCC do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, como requisito para iniciação do Trabalho de Conclusão de Curso.**

**Januária – MG, Março de 2016**

**ABSTRACT.** This work consists of an academic project in order to facilitate and increase the efficiency of stage management process performed by the stage of coordination of IFNMG - Campus Januária. The idea arose because of the difficulty in providing reports and statements, since all the data were being stored in a spreadsheet without the guarantee of safety or practicality. Added to this, there is likelihood data volume grow rapidly becoming the current process slow and sometimes confused. The Management of the stages is extremely important for total control of the activities carried out in this process. However, this process is done manually which may result in occurrence of errors and data loss. The system developed is a better solution, current and appropriate to provide greater flexibility in the management stages, decreasing significantly, and sometimes even eliminating the inconveniences caused by the management of traditional stages.

**RESUMO.** Este trabalho consiste em um projeto acadêmico com o intuito de facilitar e aumentar a eficiência do processo de gerenciamento de estágios, realizado pela coordenação de estágio do IFNMG – Campus Januária. A idéia surgiu devido à dificuldade no fornecimento de relatórios e declarações, visto que todos os dados estavam sendo armazenados em uma planilha eletrônica, sem a garantia de uma segurança ou praticidade. Somado a isso, existe a probabilidade do volume de dados crescer rapidamente, tornando o processo atual lento e, por vezes, confuso. A Gerência dos Estágios é extremamente importante para um total controle das atividades realizadas neste processo. Entretanto, esse processo é realizado de forma manual que pode ocasionar em ocorrência de erros e perda de dados. O sistema desenvolvido é uma melhor solução, atual e adequada que proporcionará maior agilidade no gerenciamento de estágios, diminuindo sensivelmente, e às vezes até eliminando os inconvenientes causados pela gestão de estágios tradicional.

**PALAVRA-CHAVE**: SGE - Sistema de Gerenciamento de Estágios, Gestão de Estágios, Software, Instituição de Ensino, Sistema WEB.

Sumário

[**1.** **INTRODUÇÃO** 5](#_Toc445134405)

[**2.** **OBJETIVOS** 7](#_Toc445134406)

[**2.1** **OBJETIVOS GERAIS** 7](#_Toc445134407)

[**2.2** **OBJETIVOS ESPECÍFICOS** 8](#_Toc445134408)

[**3.** **FUNDAMENTAÇÃO TEÓRICA** 9](#_Toc445134409)

[**3.1** **Estágio** 9](#_Toc445134410)

[**3.2** **Engenharia de Software** 9](#_Toc445134411)

[**3.3** **Metodologias Ágeis** 10](#_Toc445134412)

[**3.4** **Software Livre** 12](#_Toc445134413)

[**3.5** **Qualidade de Software** 13](#_Toc445134426)

[**3.6** **Linguagem Java** 13](#_Toc445134427)

[**3.7** **Banco de Dados** 18](#_Toc445134430)

[**4.** **Metodologia** 20](#_Toc445134431)

[**4.1** **Descrição do ambiente ALM** 21](#_Toc445134432)

[**5.** **Resultados** 24](#_Toc445134433)

[**5.1** **Requisitos e Funcionalidades** 24](#_Toc445134434)

[**5.2** **Principais Telas** 28](#_Toc445134435)

[**5.3** **Modelagem UML** 32](#_Toc445134436)

[**5.4** **Modelagem Conceitual e Lógica** 35](#_Toc445134438)

[**7.** **Conclusão** 38](#_Toc445134439)

[**8.** **Referências** 39](#_Toc445134440)

1. **INTRODUÇÃO**

O estágio supervisionado é uma atividade curricular do IFNMG – CAMPUS JANUÁRIA e propicia ao aluno a oportunidade de qualificação prática, pela experiência no exercício profissional ou social, acompanhado e supervisionado por profissionais da empresa concedente. O estágio supervisionado normalmente acaba se tornando uma excelente oportunidade para a obtenção de um emprego e, ainda, proporciona melhor aproveitamento e desempenho escolar dos alunos. Em cidades consideradas pequenas, o estágio supervisionado acaba sendo uma das únicas oportunidades de efetiva qualificação profissional dos alunos do IFNMG para o trabalho. Outro benefício a ser garantido pela atividade de estágio, para o aluno, é a da identificação mais clara das opções para a escolha profissional e para a organização de seu perfil de profissionalização.

Como agente de integração com a comunidade e orientação no processo de Estágio, o IFNMG – Campus Januária dispõe de um setor denominado Coordenação de Estágios que realiza os procedimentos e controles legais relativos ao estágio, elabora e analisa os convênios e termos de compromisso entre Instituição/Empresa e mantém controles e fiscalizações externas junto às entidades concedentes.

Apesar da grande quantidade de tecnologias disponíveis no mercado atualmente a coordenação de estágio realiza todo o seu trabalho praticamente de forma manual, o único recurso que o coordenador de estágios utiliza para auxiliar nas suas atividades são planilhas, porém atendem as necessidades parcialmente.

Existe uma grande dificuldade por parte do setor de estágio em manter o arquivo físico, devido à grande quantidade de documentos e relatórios armazenados, além disso, é necessário organizar os arquivos de forma que facilite o trabalho do coordenador. O trabalho manual mesmo feito de forma organizada pode trazer algumas desvantagens, como: Dificuldade em extrair informações dos estágios realizados; Risco de perda de informações e Perda de tempo com organização do arquivo físico.

Outro problema enfrentado pela coordenação de estágio é a falta de um meio que possa aumentar a interação entre todos os envolvidos no processo de estágio: Alunos, Empresas concedentes, Orientadores e Supervisores. As empresas concedentes são responsáveis por disponibilizar vagas de estágios e conseqüentemente receber alunos do IFNMG - Campus Januária para realização dos estágios. Os alunos são responsáveis por realizar atividades determinadas no plano de atividades. É responsabilidade do professor orientador aprovar o Plano de Atividades caso exista pertinência entre as descrições das atividades a serem desempenhadas pelo estagiário conforme o Projeto Pedagógico do Curso. Aos supervisores cabe acompanhar as atividades desenvolvidas pelos estagiários nas Unidades Concedentes.

Com uma maior participação dos demais envolvidos na atividade de estágio conseqüentemente haveria uma descentralização de informações e tarefas somente no coordenador facilitando o trabalho da coordenação de estágio, uma vez que tarefas simples, como: Conferir a situação de determinado aluno em relação ao estágio poderia ser realizada pelo próprio aluno através do sistema de Gerenciamento de Estágios.

Kenneth Laudon e Jane Laudon (2011) definem um sistema de informação, como:

***“****Um sistema de informação (SI) pode ser definido tecnicamente como um conjunto de componentes inter-relacionados que coletam (ou recuperam), processam , armazenam e distribuem informações destinadas a apoiar a tomada de decisões, a coordenação e ao controle de uma organização. Os sistemas de informação contêm informações sobre pessoas, locais e itens significativos para a organização.****”***

Os sistemas de informação são compostos por elementos humanos, organizacionais e tecnológicos que atuam em conjunto na criação de uma solução de sistema de informação para alcançar objetivos organizacionais.

Um sistema de informação gerencial proporciona os seguintes benefícios para a uma organização: Redução de tempo nas operações; Melhoria no acesso às informações, propiciando relatórios mais precisos e rápidos, com menor esforço; Melhoria na produtividade; Melhorias nos serviços realizados e oferecidos; Melhoria na tomada de decisões, por meio do fornecimento de informações mais rápidas e precisas; Melhoria na estrutura organizacional, para facilitar o fluxo de informações; Redução do grau de centralização das informações e tarefas na coordenação, entre outros.

Com os recursos e avanços dos sistemas de informação pode-se fornecer uma melhor solução, atual e adequada que proporcionará maior agilidade no gerenciamento de estágios, diminuindo sensivelmente, e às vezes até eliminando os inconvenientes causados pela gestão de estágios tradicional.

Analisando esse contexto, foi desenvolvido um sistema web com o intuito de informatizar todo o processo de gestão dos Estágios da Coordenação do IFNMG. Com este sistema em funcionamento, é possível o cadastro de todos os elementos necessários para o total controle da gestão de estágios, bem como suas alterações que se fizerem necessárias. Todas as informações inseridas no sistema ainda serão transformadas para gerar relatórios que fazem parte das funcionalidades da Coordenação de Estágios. Esses relatórios contribuem para o total controle do andamento do Estágio. Este sistema não só minimiza os inconvenientes do registro manual, tornando-o com um melhor desempenho, como também estará sempre apto á registrar qualquer atualização necessária. A informação estará sempre atualizada em formato digital e disponível para a Coordenação de Estágios do IFNMG que poderá utilizá-las a qualquer momento, evitando assim, a perda de informações e aumentando a robustez do processo.

1. **OBJETIVOS**
   1. **OBJETIVOS GERAIS**

O objetivo geral deste projeto é implantar um sistema Web de Gerenciamento de Estágios que permita a coordenação de estágios realizar o controle e a administração dos estágios curriculares dos alunos do IFNMG - Campus Januária.

* 1. **OBJETIVOS ESPECÍFICOS**

Os objetivos específicos deste trabalho são:

* Gerenciar dados dos estágios supervisionados;
* Gerenciar alunos, empresas concedentes, orientadores e supervisores;
* Gerar relatórios de estágios, declarações e outros arquivos pertinentes a coordenação de estágios.
* Gerenciar o processo de oferta de vagas de estágios.

1. **FUNDAMENTAÇÃO TEÓRICA**
   1. **Estágio**

"O estágio é um ato educativo"; essa é a primeira frase da lei que rege a atividade no Brasil (Lei 11.788 de 25/09/2008); significa dizer que o estagiário ainda não é um profissional; é um estudante que está em processo de formação, e preparação para o mundo do trabalho.

O estágio supervisionado propicia ao aluno a oportunidade de qualificação prática, pela experiência no exercício profissional ou social, acompanhado e supervisionado por profissionais da empresa concedente. Entretanto, a prática do estágio é extremamente importante não só para os alunos, mas para todos os envolvidos:

* Para as empresas que recebem os estudantes com formação e informações atualizadas e em consonância com o que está acontecendo no mundo, em termos de conhecimento;
* Para o estudante que tem a oportunidade de demonstrar suas habilidades técnicas e comportamentais, adquirindo experiência e diferenciais competitivos;
* Para as instituições de ensino que tem seus alunos em atividades de estágio, os quais podem trazer para discussão em classe, os problemas e situações reais do mundo do trabalho, usando essa interação para melhorar o nível de qualidade da formação que está provendo para seus alunos, futuros profissionais.
  1. **Engenharia de Software**

A Engenharia de Software tem como objetivo apoiar o desenvolvimento de software que atenda todas as necessidades do cliente e para alcançar esse objetivo é necessário seguir algumas etapas importantes, desde documentação até a construção e implantação do sistema proposto. Dentre essas etapas, a engenharia de software tem como objetivo definir e exercitar processos, métodos, ferramentas e ambientes para construção de software que satisfaça necessidades de cliente e usuário dentro de prazos e custos previsíveis, conforme SOMMERVILLE (2011).

As principais atividades da engenharia de software envolvem a **especificação** (onde as funcionalidades e restrições são definidas), o **desenvolvimento** (onde este é produzido), a **validação** (onde é verificado se a solução resolveu o problema) e a **evolução** do software (onde ele evolui para atender as novas necessidades do cliente).

A Engenharia de Software surgiu visando melhorar a qualidade dos produtos de software e aumentar a produtividade no processo de desenvolvimento. Ela trata de aspectos relacionados ao estabelecimento de processos, métodos, técnicas, ferramentas e ambientes de suporte ao desenvolvimento de software.

* 1. **Metodologias Ágeis**

As metodologias ágeis são uma nova forma de gestão e desenvolvimento de Software que usa uma abordagem de planejamento e execução iterativa e incremental. Os métodos ágeis visam sempre reduzir o ciclo de vida do software desenvolvendo e entregando versões mínimas e obtendo feedback constante dos clientes e usuários do software.

Os métodos ágeis são adequados para situações em que a mudança de requisitos é freqüente. Para ser realmente considerada ágil, a metodologia deve aceitar a mudança em vez de tentar prever o futuro.

Embora tenham sido criadas várias metodologias para o desenvolvimento de software, existem atividades fundamentais comuns a todas elas, segundo SOMMERVILLE (2011), essas atividades são:

* **Especificação:** definição das funcionalidades e demais características do produto.
* **Projeto e implementação:** o software é produzido de acordo com as especificações. Nesta fase são propostos modelos por meio de diagramas que serão implementados em alguma linguagem de programação.
* **Validação:** atividades de revisão e testes visando a assegurar que os requisitos sejam cumpridos.
* **Evolução:** atividades de manutenção para adaptar o software a novas necessidades do cliente.
  + 1. **Scrum**

O Scrum é um processo de desenvolvimento iterativo e incremental para gerenciamento de projetos e desenvolvimento ágil de software. Conforme KOSCIANSKI e SOARES (2007) o foco da metodologia é encontrar uma forma de trabalho dos membros da equipe para produzir o software de forma flexível e em um ambiente em constante mudança.

Focando nas atividades necessárias para implementação do software a metodologia Scrum, irá proporcionar maior agilidade neste processo. No Scrum, os projetos são divididos em ciclos (Iterações) denominados de Sprints. O Sprint representa um tempo definido dentro do qual um conjunto de atividades chamado de Sprint Backlog deve ser executado.

As atividades executadas dentro de um Sprint são parte de uma lista de funcionalidades do software denominada Product Backlog. O conjunto de atividades executados no Sprint são escolhidas de acordo um nível de prioridade definido no inicio do projeto pela equipe de desenvolvimento e o cliente.

Na Figura a seguir é apresentada a representação esquemática de um processo baseado na metodologia ágil scrum.

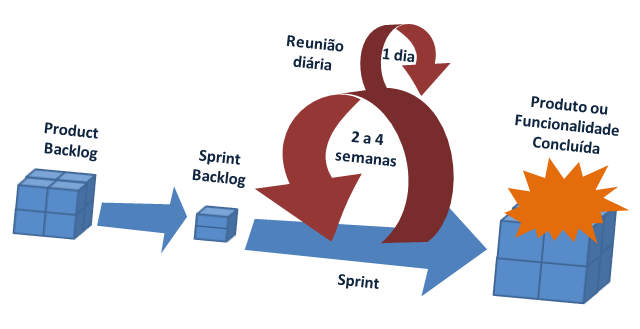


Figura: Estrutura do Scrum

Fonte: http://www.brq.com/metodologia 1

* 1. **Software Livre**

No Software livre os usuários possuem a liberdade de executar, copiar, estudar, mudar, distribuir, e melhorar o software. Segundo A Free Software Foundation (FSF) um programa só é considerado software livre se os usuários possuem as quatro liberdades essenciais:

**Liberdade 0:** A liberdade de executar o programa como você desejar, para qualquer propósito.

**Liberdade 1:** A liberdade de estudar como o programa funciona, e adaptá-lo às suas necessidades.(Acesso ao código fonte).

**Liberdade 2:** A liberdade de redistribuir cópias de modo que você possa ajudar ao próximo.

**Liberdade 3:** A liberdade de distribuir cópias de suas versões modificadas a outros. Desta forma, você pode dar a toda comunidade a chance de beneficiar de suas mudanças. (Acesso ao código fonte).

Por software livre devemos entender aquele software que respeita a liberdade e senso de comunidade dos usuários. Assim sendo, “software livre” é uma questão de liberdade.

4. 5. **Qualidade de Software**

PRESSMAN (2011) define a qualidade de software como a conformidade a requisitos funcionais e de desempenho que foram explicitamente declarados à padrões de desenvolvimento claramente documentados e à características implícitas que são esperadas de todo software desenvolvido por profissionais.

Os requisitos de software são a base a partir da qual a qualidade é medida. A falta de conformidade com os requisitos significa falta de qualidade. As falhas em requisitos estão entre as principais razões para a baixa qualidade de um software. Entre as principais razões destacam-se os requisitos mal expressos, requisitos desnecessários para os clientes e a dificuldade para lidar com requisitos freqüentemente mutáveis. Os requisitos compreendem as funcionalidades presentes no software, portanto, requisitos de um software são as descrições sobre seu comportamento, funções e especificações das operações que deve realizar.

A qualidade de o software estar diretamente ligada aos requisitos. Portanto, realizar corretamente o levantamento e administração de requisitos é extremamente importante para a qualidade do software.

* 1. **Linguagem Java**

A Linguagem Java segue o paradigma Orientado Objetos, o que a torna uma ferramenta extremamente poderosa, pois é possível fazer reuso de código já produzido. Além disso, é Multi-Threaded, o que significa dizer que ela suporta processamento paralelo múltiplo.

A linguagem Java possui portabilidade, onde o mesmo código Java roda em diversas plataformas sem a necessidade de alteração de código e as aplicações podem ser facilmente migradas entre servidores. Tornado desnecessário ficar preso a somente um fabricante.

De acordo com FURGERI (2012), a linguagem Java possui as seguintes características:

* Orientação a objetos: uma prática de programação que torna possível elaborar um software a partir da geração de objetos que se comunicam entre si.
* Portabilidade: Java é uma linguagem multiplataforma, ou seja, uma mesma aplicação pode ser executada em diferentes tipos de plataforma sem a necessidade de adaptação de código.
* Multithreading: threads (linhas de execução) são o meio pelo qual se consegue fazer com que mais de um evento aconteça simultaneamente em um programa.
* Suporte à Comunicação: uma das vantagens do Java é fornecer um grande conjunto de classes com funcionalidades especificas, ou seja, muitos detalhes de programação são encapsulados em classes já prontas. Nesse contexto, a linguagem oferece um conjunto de classes para programação em rede, o que agiliza a implementação de sistemas multiusuários. Tais classes são desenvolvidas para suportar tecnologias avançadas de comunicação.

O Java é bastante flexível por conta da possibilidade de expansão através das bibliotecas, ou APIs, além das extensões do Java. Através dessas extensões, é possível desenvolver praticamente qualquer tipo de aplicação (Desktop, Web, Móbile), em Java, de uma maneira bem mais documentada e específica.

* 2. 1. **Java EE**

O Java EE (Enterprise Edition) é uma plataforma amplamente usada que contém um conjunto de tecnologias coordenadas que reduz significativamente o custo e a complexidade do desenvolvimento, implantação e gerenciamento de aplicações web.  O Java EE é construído sobre a plataforma Java SE e oferece um conjunto de APIs (interfaces de programação de aplicações) para desenvolvimento e execução de aplicações portáteis, robustas, escaláveis confiáveis e seguras.

“O Java EE simplifica o desenvolvimento do aplicativo e diminui a necessidade de programação e de treinamento do programador, criando componentes modulares reutilizáveis padronizados e ativando a camada para tratar de muitos aspectos da programação de forma automática.” ORACLE (2010).

* + 1. **Java EJB**

Enterprise JavaBeans (EJB): é uma arquitetura gerenciada de componente do lado do servidor utilizada para encapsular a lógica de negócios de uma aplicação. Segundo RAHMAN (2009) EJB oferece inúmeras soluções para servidor, incluindo persistência, envio de mensagens, agendamento simplificado, comunicação distribuída, entre outros.

A arquitetura EJB permite ainda que os desenvolvedores de aplicações se foquem na implementação da regra de negocio sem ter que perder tempo na construção do código de infra-estrutura, além disso, a tecnologia EJB permite o desenvolvimento rápido e simplificado de aplicações distribuídas, transacionais, seguras e portáteis baseadas na tecnologia Java.

* + 1. **Primefaces**

PrimeFaces é uma biblioteca de componentes de código aberto para o JSF e possui centenas de componentes, utilizados para criar interfaces ricas para aplicações web de forma simplificada e eficiente. Ele é considerado a melhor biblioteca de componentes para o JSF, tendo em vista que é uma biblioteca enxuta e com um único arquivo (. Jar), não sendo necessária nenhuma configuração e não tem nenhuma dependência.

Segundo FERNANDES (2012), o PrimeFaces, além de possuir uma gama de componentes maior que outras bibliotecas semelhantes, como RichFaces e ICEfaces, também incorpora recursos importantes do novo padrão HTML 5. Desta maneira, além de utilizar componentes com visual mais rico, também consegue ter melhor desempenho, principalmente devido ao uso de componentes leves baseados *jQuery*, que utiliza puramente o HTML5.

* + 1. **JSF**

JavaServerFaces (JSF) é um framework da especificação [Java](https://pt.wikipedia.org/wiki/Java_(linguagem_de_programa%C3%A7%C3%A3o)) para a construção de [interfaces de usuário](https://pt.wikipedia.org/wiki/Interface_de_usu%C3%A1rio) baseadas em [componentes](https://pt.wikipedia.org/wiki/Engenharia_de_software_baseada_em_componentes) para [aplicações web](https://pt.wikipedia.org/wiki/Aplica%C3%A7%C3%B5es_web). Segundo GODOY (2011) o JSF permite o desenvolvimento de aplicações web possuindo um excelente conjunto de funcionalidades para o cenário de desenvolvimento em que é inserido. O framework JSF integra o modelo MVC (Model-View-Controller), um padrão de desenvolvimento que divide a aplicação em três camadas:

* Model - Responsável pela Lógica, como se diz ela é responsável pela regra do negocio, essa camada é o software em si. A camada model representa e manipula seguindo as regras do negocio.
* View - Essa camada é responsável pela apresentação, é a interface de representação do modelo, ou seja, trata-se da fronteira entre usuário e o sistema em si.
* Controller - Essa camada é responsável por controlar a comunicação entre o modelo e a visão. Pode ser considerado como a fronteira entre os outros dois componentes do MVC e sua finalidade é controlar interações que ocorram a partir do usuário que trabalha sobre elementos na camada de visão.

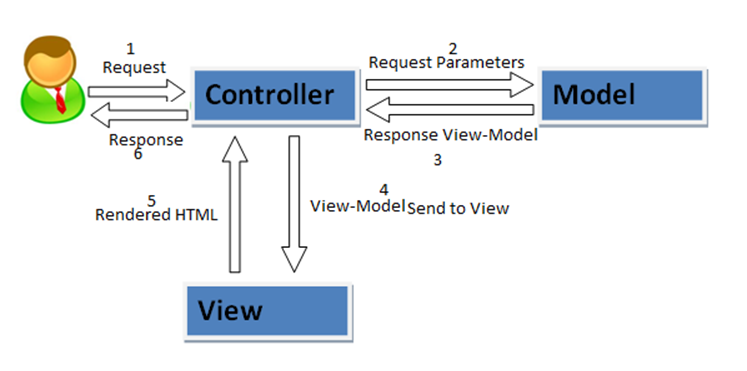


Figura: Modelo MVC 1

Fonte: http://aspilham.blogspot.com.br/2011/01/better-aspnet-mvc-architectural.html

O JSF Possui um modelo de programação dirigido a [eventos](https://pt.wikipedia.org/wiki/Evento_(computa%C3%A7%C3%A3o)), abstraindo os detalhes da manipulação dos eventos e organização dos componentes, permitindo que o programador se concentre na lógica da aplicação, deixando as tarefas básicas e trabalhosas por conta do framework.

* + 1. **JPA**

Java Persistence API (JPA) é uma API padrão da linguagem Java que descreve uma interface comum para frameworks de persistência de dados. A JPA define um meio de mapeamento objeto-relacional para objetos Java simples e comuns, denominados beans de entidade. Segundo Oliveira (2011) Uma entidade corresponde a um objeto que pode ser gravado na base de dados através de um mecanismo de persistência. Estes objetos são POJOs (Plain Old Java Objects) e se utilizam de ORM (Mapeamento objecto-relacional) para tal. A ORM permite ao programador criar mapeamentos das classes Java e seus relacionamentos para as tabelas do banco de dados. Dessa forma frameworks que possuem suporte JPA podem realizar inserções , buscas , exclusões e alterações nos dados da aplicação. Com ORM, as tabelas do banco de dados são representadas através de classes e os registros de cada tabela são representados como as instâncias destas classes. Portanto, o código da aplicação fica independente de frameworks de persistência, pois todo o código utilizado para manipular o banco de dados passa a ser o JPA.

* 1. **Banco de Dados**

Um banco de dados é uma estrutura para manipulação eficiente de um grande volume de informações estruturadas e armazenadas de forma organizada e integrada. A estrutura de um banco de dados é constituída por:

* **Tabela**. Armazena um conjunto de informações referente a um determinado objeto. Assim como seu nome sugere, sua estrutura é igual a uma tabela.
* **Linha**. É um elemento da tabela. Cada linha é formada por um conjunto de colunas e representam um registro. Os registros não precisam, necessariamente, possuir valores em todas as colunas, sendo possível existir colunas nulas (sem valor atribuído). Os registros na tabela são desordenados e sua localização se faz por meio um campo que assume o papel de chave primária da tabela, com isso o banco de dados se torna mais fácil de manipular e entender.
* **Coluna**. São os atributos de uma tabela, ou seja, cada coluna armazena um tipo de dado (inteiros, números reais, strings de caracteres, data, dentre outros).

Essa visão de dados organizados em tabelas oferece um conceito simples e familiar para a estruturação dos dados, sendo um dos motivos do sucesso de sistemas relacionais de dados. RICARTE (2002). Para manipulação dos bancos de dados relacionais existem Sistemas Gerenciadores de Banco de Dados, que são softwares que tem por objetivo principal gerenciar o acesso e a correta manutenção dos dados em um banco de dados, tornando o banco mais seguro, confiável e mais fácil de manter. Existem diversos SGBD’s, dentre eles:

* [PostgreSQL](http://www.postgresql.org/)
* [HSQLDB](http://hsqldb.org/)
* [IBM DB2](http://www-01.ibm.com/software/data/db2/)
* [MySQL](http://www.mysql.com/)
  + 1. **SQL**

O SQL foi criado pela IBM em 1974. Rapidamente fez sucesso como forma de consulta e manipulação de dados dentro de um banco de dados. Tanto que vários SGBD‟s utilizam-se da SQL como linguagem padrão para acesso às suas bases de dados. E, em 1982, o ANSI (American National Standard Institute) tornou-o linguagem padrão para tal.

Sua principal vantagem está em ser uma linguagem padronizada para bancos. Isso quer dizer que, pelo menos em teoria, é possível mudar de SGBD sem a necessidade de alterar o SQL. Entretanto, esta não é sua única vantagem. Ele também é independente, ou seja, não depende do tipo de sistema operacional, usa sentenças em inglês permitindo um entendimento imediato do comando e permite alterar, expandir ou incluir estruturas dos dados armazenados.

* + 1. **MySQL**

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. Assim como o Java, MySQL possui suporte para qualquer plataforma atual e é um software livre com excelente desempenho, estabilidade e facilidade de uso. É um software de banco de dados com uma versão gratuita, desenvolvido pela empresa Oracle Corporation. Incluso no pacote, além do banco de dados, vem uma ferramenta (MySQL Workbench) para modelagem do Diagrama Relacional.

1. **Metodologia**

Para a construção do sistema de gerenciamento de estágios foi preciso passar por algumas etapas, desde a documentação até a construção e implantação do sistema. O sistema foi desenvolvido para suprir as necessidades da coordenação de estágios, portanto, foi de suma importância a participação dos funcionários deste setor durante todo o processo de construção do sistema.

Foram utilizados dois tipos de pesquisa: A pesquisa aplicada, pois objetiva gerar conhecimento para aplicação prática, com intuito de solucionar problemas específicos; e a pesquisa exploratória que proporciona maior familiaridade com o problema, além de envolver: levantamento bibliográfico; Entrevistas; e análise de trabalhos similares.

O Levantamento e Análise de Requisitos têm como principal objetivo identificar, definir e compreender o problema para qual será desenvolvida a solução. Para tanto, foram realizadas entrevistas com a coordenação para explicação do processo para qual seria desenvolvido o sistema.

Após realização de entrevistas e uma breve discussão com a coordenação de estágios sobre os problemas enfrentados no controle dos estágios e conseqüentemente a realização do levantamento de requisitos, foi feita análise de artigos de sistemas similares ao sistema de gerenciamento de estágios pesquisados na internet. A pesquisa foi feita com a busca de artigos científicos que referenciavam o desenvolvimento de produtos semelhantes. Na análise das informações encontradas foi considerada a procedência das mesmas, sua quantidade, relevância, características desejáveis e limitações relacionadas à aplicação desejada. As características das soluções similares identificadas nessa atividade foram confrontadas com os requisitos obtidos nas entrevistas objetivando estabelecer as funcionalidades mais importantes para o projeto.

A metodologia ágil foi utilizada durante desenvolvimento e implementação do sistema de gerenciamento de Estágios, pois proporciona melhor controle nas etapas de planejamento, consistindo em reuniões freqüentes com a coordenação de estágios em que eram definidas as funcionalidades prioritárias do sistema conforme os requisitos obtidos através de entrevistas na Coordenação de Estágios do IFNMG Campus Januária.

No desenvolvimento do software foi utilizada à arquitetura em três camadas: Padrão arquitetural que define que o sistema deve ser dividido em três camadas: A Lógica de Aplicação, que contém as regras do negócio, a camada de Acesso à Dados, que contém toda a lógica de interação com o SGBD e a camada de Apresentação, que contém toda a lógica de interface com o usuário. Dessa forma é possível manter um isolamento entre os componentes de cada camada. Esse isolamento possibilita a reutilização do código, facilita a manutenção, além de permitir que novas funcionalidades sejam adicionadas sem grande impacto nas já existentes.

* 1. **Descrição do ambiente ALM**
     1. **Linguagem**

A linguagem utilizada na programação foi Java por ser “uma linguagem computacional completa, adequada para o desenvolvimento de aplicações baseadas na rede Internet, redes fechadas ou ainda programas stand-alone”. (Indrusiak, 1996). Além de ser […] a força propulsora por trás de alguns dos maiores avanços da computação mundial, como:

- Acesso remoto a bancos de dados

- Bancos de dados distribuídos

- Comércio eletrônico no WWW

- Network CAD

- Interatividade em páginas WWW

- Interatividade em ambientes de Realidade Virtual distribuídos

- Gerência de Documentos

- Integração entre dados e forma de visualização

- Network Computer

- Ensino à distância

-Jogos e entretenimento.” (Indrusiak, 1996).

* + 1. **IDE**

Segundo o site oficial do *NetBeans* ele: é a IDE oficial do Java 8. Ele possui editores, conversores e analisadores de código, que possibilitam uma rápida atualização de aplicativos para usar as novas construções de linguagem Java 8, como lambdas, operações funcionais, e referências de método. Analisadores e conversores são fornecidos para pesquisa através de vários aplicativos ao mesmo tempo, os padrões para a conversão correspondentes a novas construções de linguagem Java 8. Com o seu constante aperfeiçoamento Java Editor, muitas características ricas e uma ampla gama de ferramentas, modelos e exemplos, NetBeans IDE define o padrão para o desenvolvimento com tecnologias de ponta para fora da caixa.

* + 1. **Controle de versão**

A ferramenta de controle de versão utilizada foi o Git, por que segundo Chacon (2009) citado por Mello (2013, p. 11).

[…] tem sido uma das mais aplicadas nos últimos anos. Este apresenta todas capacidades esperadas de um sistema com este propósito como controle de versões de um arquivo e capacidades de fusão de arquivos. Além disso possui uma implantação que facilita a criação de ramificações e funciona de forma distribuída. (CHACON, 2009) citado por (MELLO, 2013, p. 11).

Além de ser uma das ferramentas mais utilizadas para controle de versão, assim como relatado por Peter (2009) e citado por Cavalcanti (2013, p. 4), “Dentre os sistemas de controle de versões, um dos mais populares e difundidos da atualidade é o git, principalmente entre a comunidade *open source*.”

* + 1. **SGBD**

O Sistema Gerenciador de Banco de Dados (SGBD) implementado foi o MySQL por sua velocidade, robustez e facilidade de uso além de outras qualidades citadas por Gonzaga e Birckan (2000, p. 03).

MySQL é um servidor de banco de dados SQL multi-usuário e multi-threaded. SQL é a linguagem de banco de dados mais popular no mundo. MySQL é uma implementação cliente-servidor que consiste de um servidor e diferentes programas clientes e bibliotecas. SQL é uma linguagem padronizada que torna fácil o armazenamento e acesso de informações. Por exemplo, pode-se usar SQL para recuperar informações de produtos e armazenar informações de clientes para um site Web. O servidor MySQL é também rápido e flexível o suficiente para permitir armazenar logs e figuras nele. (GONZAGA; BIRCKAN, 2000, p. 03).

As qualidades desse SGBD são também citadas por Niederauer e Prates (2006, p. 04).

É o mais popular entre os bancos de dados com código-fonte aberto. Há mais de cinco milhões de instalações do MySQL no mundo todo, inclusive em sites com alto volume de dados e de tráfego, como Associated Press, Google, NASA, Sabre Holdings e Suzuki. […] Mesmo possuindo uma tecnologia complexa de banco de dados, seu custo é bastante baixo. Tem como destaque suas características de velocidade, escalabilidade e confiabilidade, o que vem fazendo com que ele seja adotado por departamentos de TI (Tecnologia da Informação), desenvolvedores Web e vendedores de pacotes de softwares. (NIEDERAUER; PRATES, 2006, p. 04).

* + 1. **Criação de diagramas UML**

Para criação dos diagramas de UML, será utilizada a ferramenta Dia por sua facilidade de uso, rapidez de criação e suporte ao processo de design, assim como relatado por Gilliam (2001).

Dia é um aplicativo que independente da plataforma para modelação baseada nas especificações UML 1.2. O aplicativo foi originalmente desenvolvido como uma alternativa Linux em relação ao Microsoft Visio e fornece amplo suporte para todo o processo de design de sistema. A ferramenta não é formalmente um aplicativo UML e pode ser usada apenas para representar diagramas de Actividade, Use Case, Colaboração e Componentes. (GILLIAM, 2001).

1. **Resultados**
   1. **Requisitos e Funcionalidades**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **F1 - Controle de acesso de usuários** | | | |  |
| **Descrição:** Autenticação dos usuários que terão acesso ao sistema SGE. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF1. 1** – Campo Login. | O campo Login do usuário deve ter no mínimo quatro caracteres. | () | (x) |
| **NF1. 2** – Campos Senha | O campo Login do usuário deve ter no mínimo seis caracteres. | () | (x) |
| **NF1. 3** – Dígitos da Senha. | O campo senha deve ser formatado para não mostrar a senha digitada e sim caracteres especiais. | (x) | (x) |
| **NF1. 4** - Mensagem de erro. | O sistema deve mostrar uma mensagem de erro ao usuário caso o ele não permita que o usuário acesse o mesmo.  Mostrando qual tipo de erro ocorreu (Login ou a senha estejam incorretas, campos em branco ou quantidade mínima de caracteres). | (x) | (x) |
| **NF1. 5** – Mensagem de Boas Vindas. | O sistema deve mostrar uma mensagem de boas vindas ao usuário caso o mesmo consiga acessar o sistema. Mostrado o nome do usuário. | () | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F2 - Consulta de ofertas de estágios** | | | |
| **Descrição:** Listagem de todas as oportunidades de estágios. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF2. 1 –** Filtro de Busca | O sistema deve mostrar todas as oportunidades de estágios e deve provê meios de o usuário buscar oportunidades de forma eficaz através de filtro de busca (Por: Empresa, Área, Curso, etc.) | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F3 - Cadastro de ofertas de estágios** | | | |
| **Descrição:** Manter o controle das ofertas de estágios de determinada empresa. Provê inclusão, exclusão e alteração, além de uma opção para finalizar a mesma. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF3.1–** Campo obrigatório | Os campos considerados obrigatórios devem ser indicados ao usuário. | (x) | (x) |
| **NF3. 2 –** Mensagem | O sistema deve mostrar uma mensagem de erro ao usuário caso ele não preencha os campos corretamente.  Mostrando qual tipo de erro ocorreu (Ex: campos obrigatórios em branco) | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F4 - Candidatura às ofertas de estágios** | | | |
| **Descrição:** Todos os alunos poderão se candidatar a qualquer vaga de estágio | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF4. 1 –** Currículo | Durante a candidatura de algum aluno o sistema deve provê ao usuário uma forma dele anexar o seu currículo em PDF. | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F5 - Selecionar candidato** | | | |
| **Descrição:** Controle dos candidatos as vagas de estágios. A empresa concedente poderá classificar ou desclassificar da próxima etapa de seleção qualquer candidato. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF5. 1 –** Seleção de candidato | A empresa somente poderá classificar ou desclassificar algum candidato após o termino do período de oferta da vaga, após esse período a empresa terá acesso aos currículos anexados pelos candidatos. | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F6 - Registrar Estágio** | | | |
| **Descrição:** Processamento de recuperação, criação, exclusão e alteração de estágios. Será possível emitir relatórios e documentos referentes aos estágios | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF6. 1 –** Campo data | Todos os campos de datas devem possui mascara | (x) | (x) |
| **NF6. 2 –** Busca de informações complementares | A tela de gerenciamento de estágio deve possuir meios de o usuário buscar informações complementares para o registro do estágio (Alunos, Cursos, Empresas, Orientador) | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F7 – Acompanhar os estágios.** | | | |
| **Descrição:** Os alunos e orientadores poderão verificar a situação dos estágios que estão realizando e orientando respectivamente. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF7. 1 –** Busca dos estágios | O sistema deve provê aos alunos uma tela para visualizar a situação dos estágios e uma tela com filtros de busca (Por aluno, curso, ou data) para os orientadores verificar a situação dos estágios que orienta. | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F8 – Geração de boletim de estágio de cada discente** | | | |
| **Descrição:** Coordenador de estágio poderá emitir o boletim de estágio. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF8. 1 –** Boletim de estágio. | O boletim deverá ser gerado em arquivo PDF e poderá ser emitido nas telas de busca de estágios. | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F9 – Geração de relatórios gerenciais para a coordenação de estágios e departamento de extensão.** | | | |
| **Descrição:** Emissão de relatórios das bases de dados do SGE | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF9.1–** Relatórios | Deve ser emitidos relatórios em arquivo PDF: relatórios de estágios, empresas concedentes, cursos, discentes, etc. | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F12 – Gerenciamento de Discentes (Cadastros, Edição, Remoção)** | | | |
| **Descrição:** Manter o controle dos discentes. Provê a inclusão, exclusão ou alteração de discentes. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF12. 1 –** Telas | Os usuários devem ter acesso às funções de gerenciamento de discentes em uma mesma tela. Funções (Inclusão, Exclusão, Alteração) | (x) | (x) |
| **NF12. 2 –** Campos | Todos os campos necessários devem possuir mascaras (DATA, CPF, RG). O campo nome não pode ficar em branco e deve possuir mais de dois caracteres. | (x) | (x) |
| **NF12. 3 –** Login | Cada cadastro de discente deve estar vinculado á algum Login. | (x) | (x) |

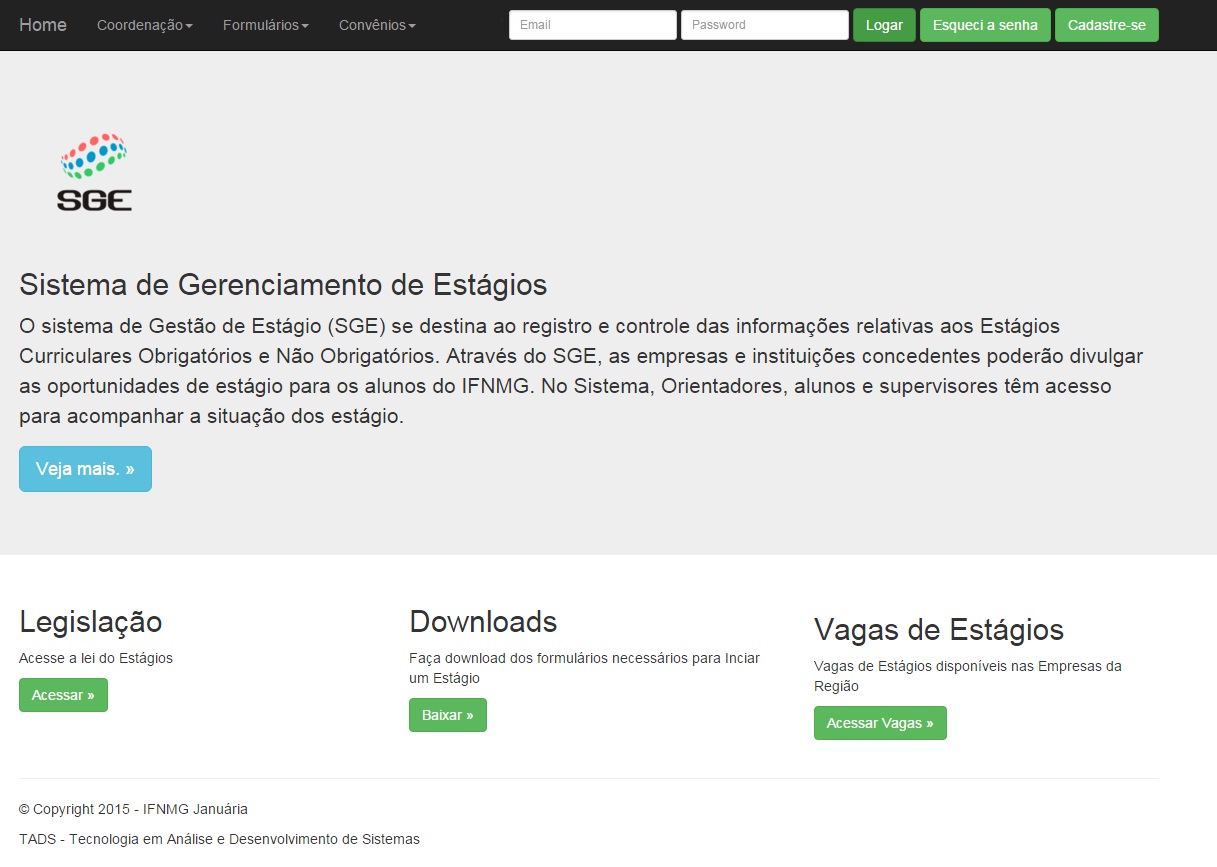
|  |  |  |  |
| --- | --- | --- | --- |
| **F13 – Gerenciamento de Empresas (Cadastros, Edição, Remoção)** | | | |
| **Descrição:** Manter o controle das empresas concedentes. Provê a inclusão, exclusão ou alteração de empresas. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF13. 1 –** Campos | Todos os campos necessários devem possuir mascaras (CNPJ, RG, Telefone). O campo nome não pode ficar em branco e deve possuir mais de dois caracteres. | (x) | (x) |
| **NF13. 2 –** Login | Cada cadastro de empresa deve estar vinculado á algum Login. | (x) | (x) |
| **NF13. 3**Responsável | Para cada empresa deve ser cadastrada uma pessoa responsável pelos estágios da empresa | (x) | (x) |

|  |  |  |  |
| --- | --- | --- | --- |
| **F14 – Gerenciamento de Cursos (Cadastros, Edição, Remoção)** | | | |
| **Descrição:** Manter o controle dos cursos. Provê a inclusão, exclusão ou alteração de cursos do IFNMG. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF14. 1 –** Telas | Os usuários devem ter acesso às funções de gerenciamento de cursos em uma mesma tela. Funções (Inclusão, Exclusão, Alteração) | (x) | (x) |

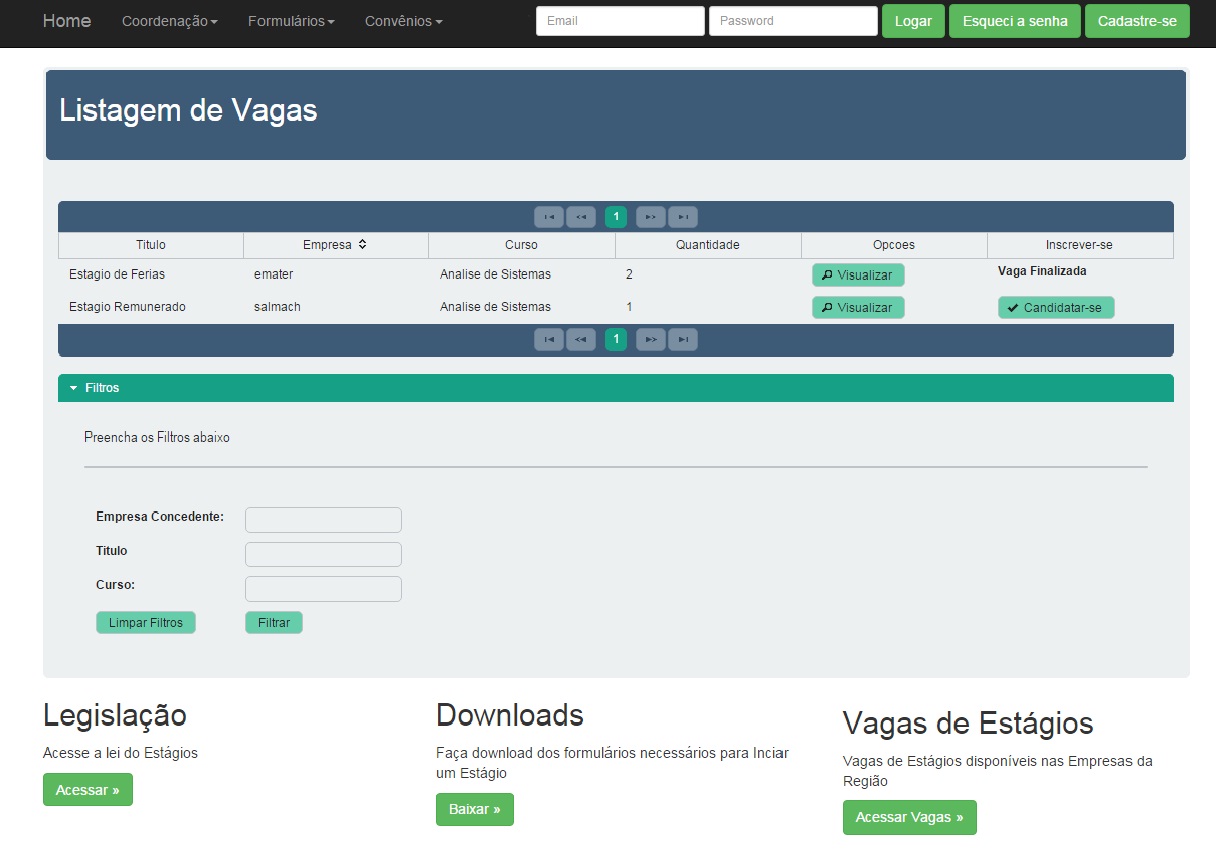
|  |  |  |  |
| --- | --- | --- | --- |
| **F15 – Acompanhamento das vagas de estágios** | | | |
| **Descrição:** Os candidatos devem ter acesso às informações sobre suas vagas. Provê meios de o candidato consultar resultado da sua candidatura em vagas. | | | |
| ***Requisitos Não Funcionais (NF)*** | | | |
| **Nome** | **Restrição** | **Desejável** | **Permanente** |
| **NF15. 1 –** Telas | Os usuários devem ter acesso ao resultado das vagas. | (x) | (x) |

* 1. **Principais Telas**

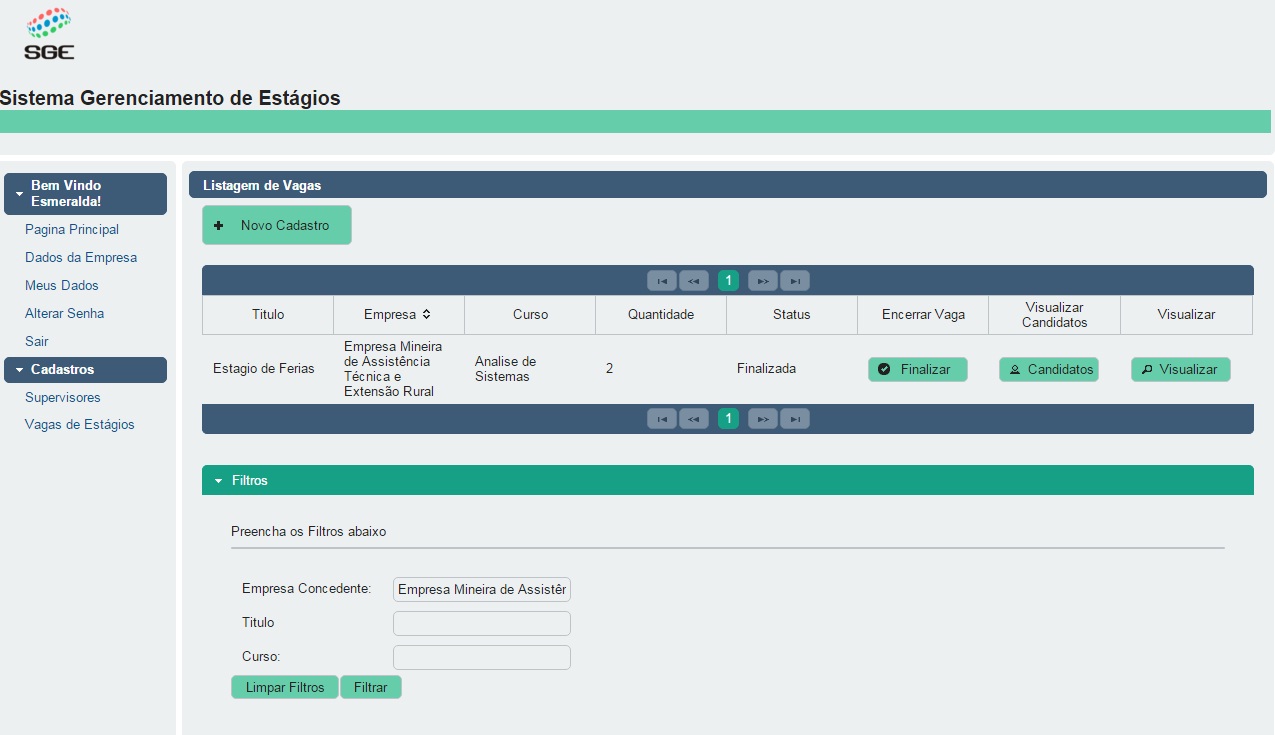
Tela de Login

****

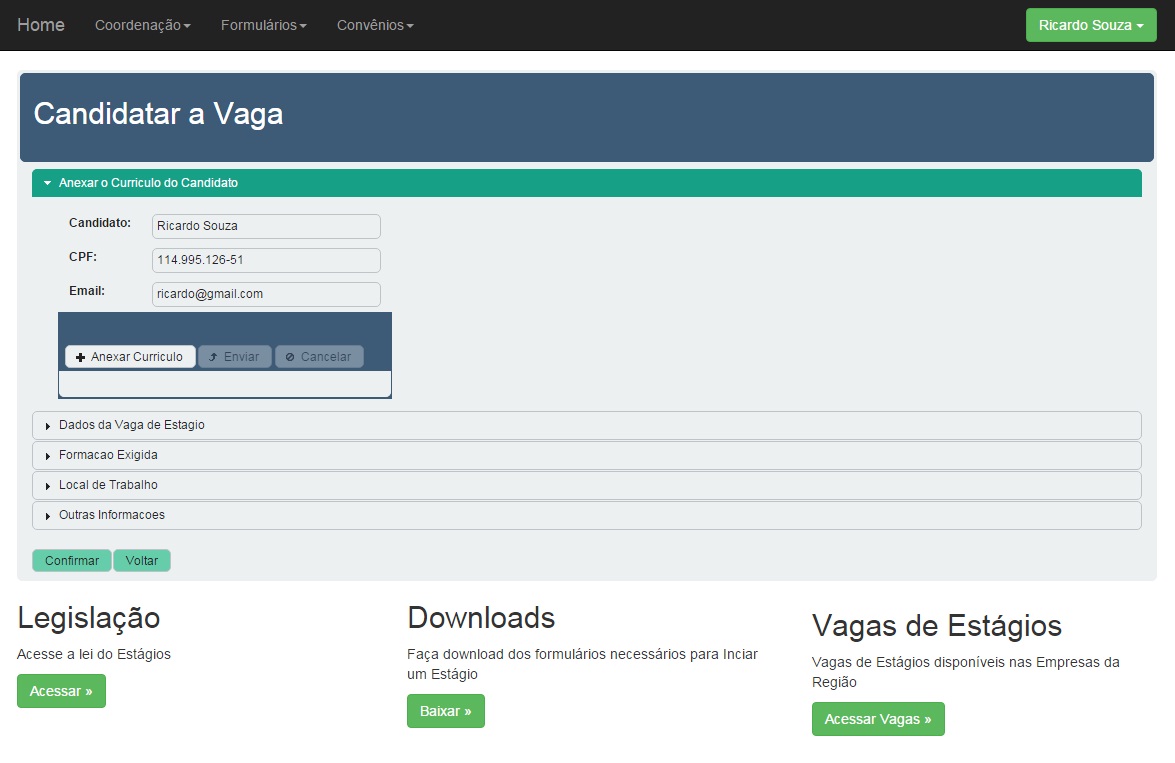
Tela de Listagem de Vagas de Estágios

****

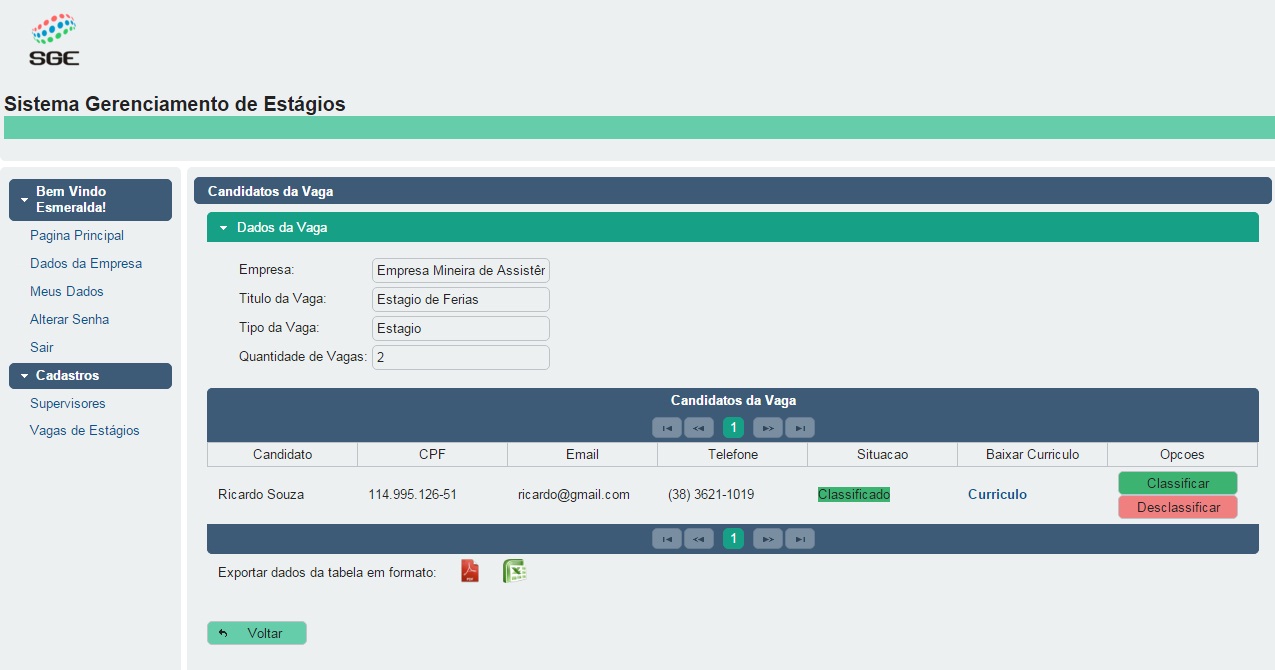
Tela de Gerenciamento de Vagas



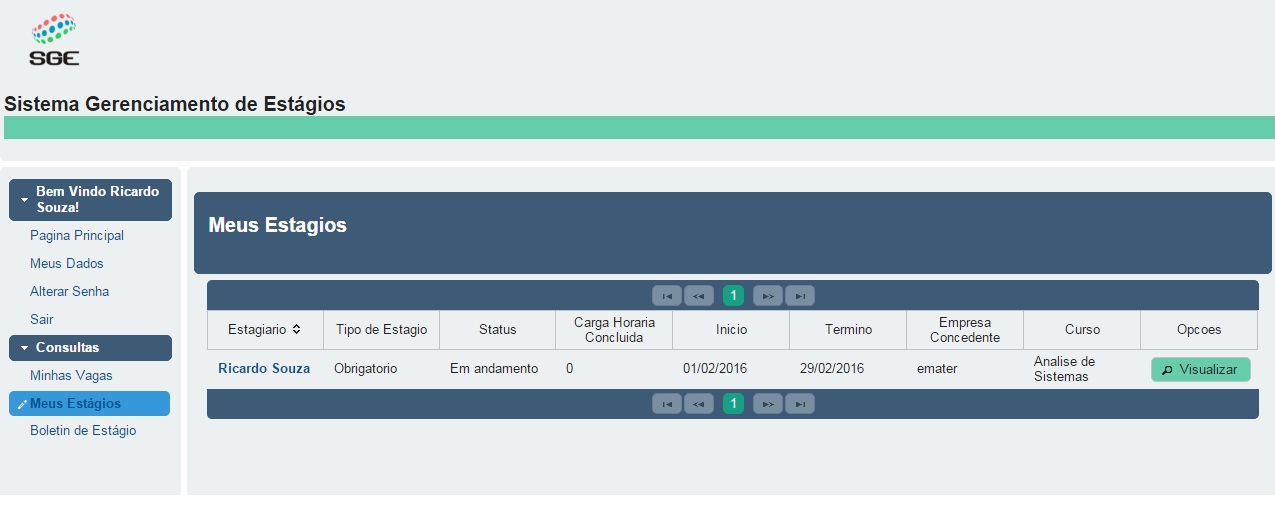
Tela de Candidatura a vagas

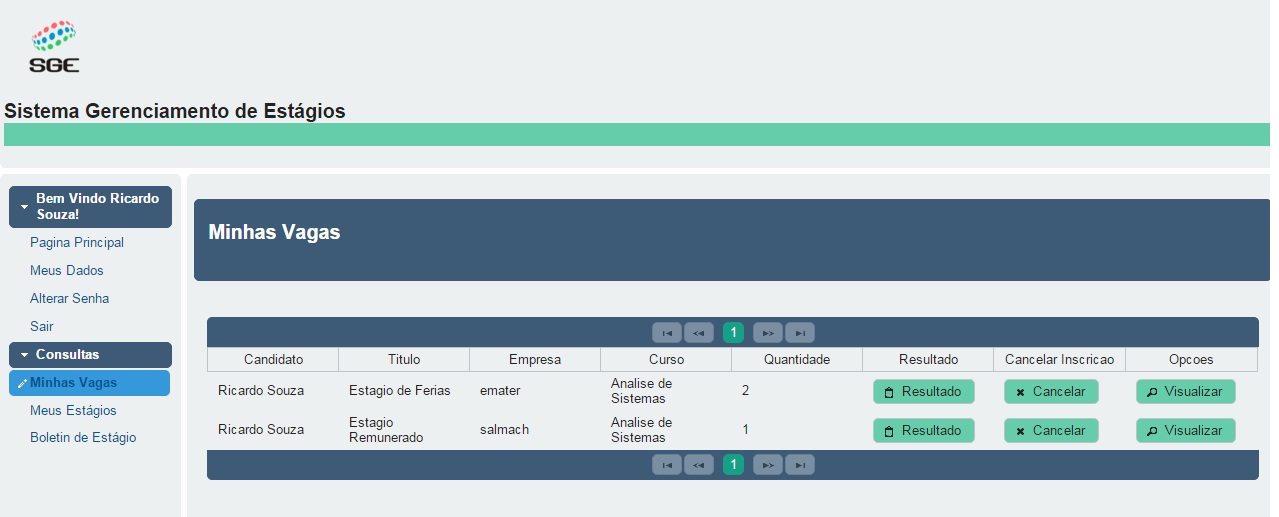
****

Tela de Seleção de Candidatos a Vagas

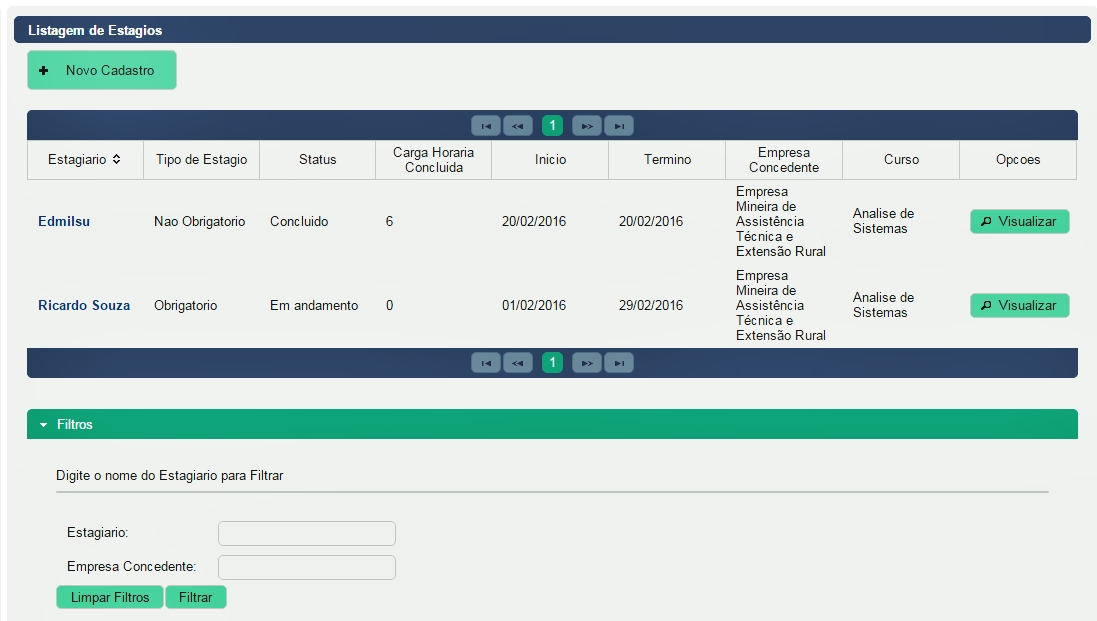


Telas de Acompanhamento de Estágios e Vagas

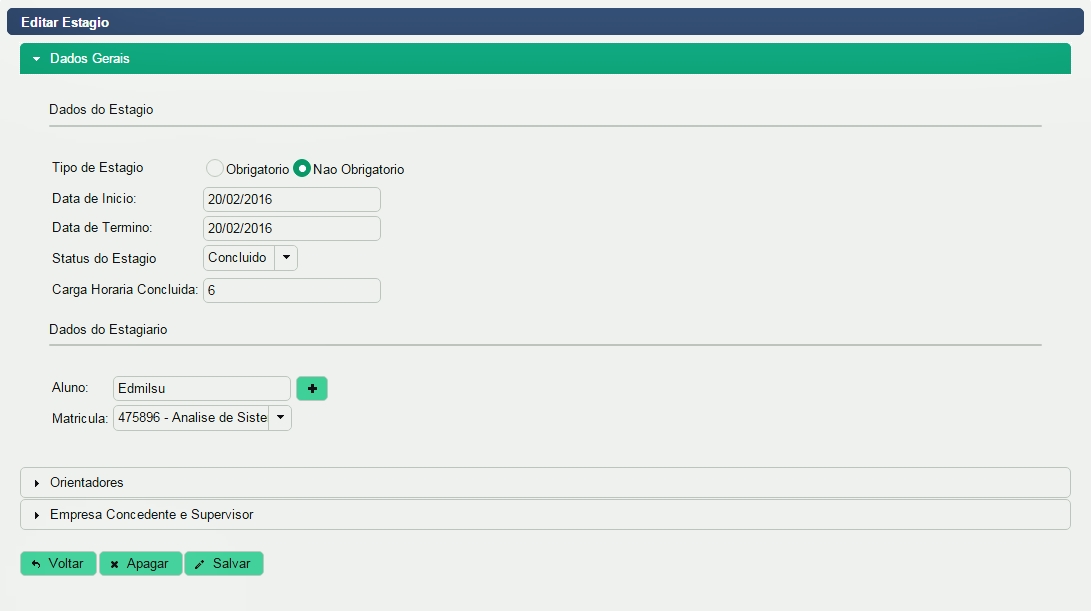




Telas de Listagem de Estágios

****

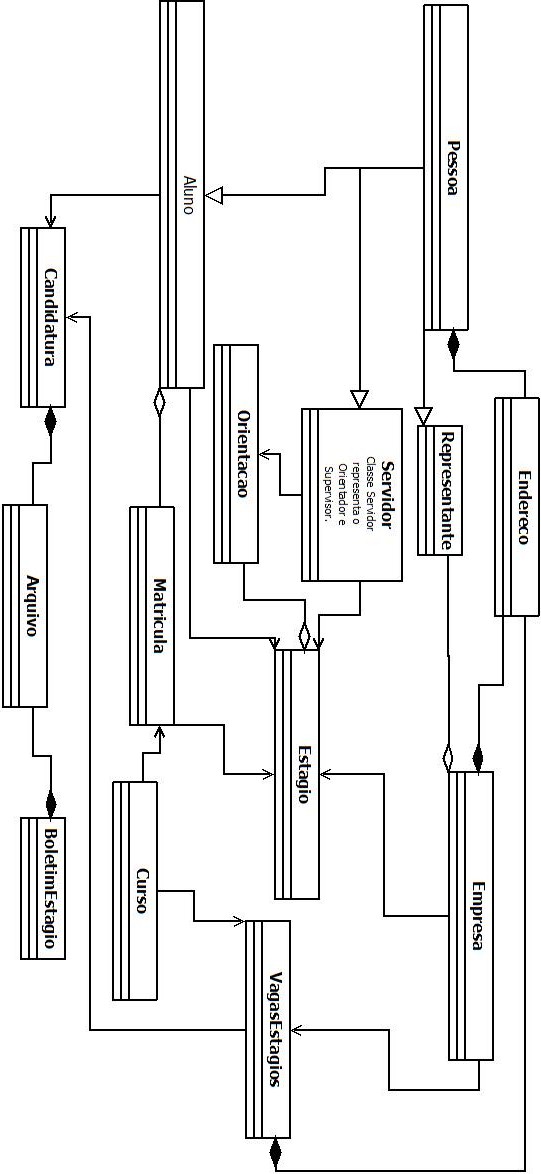
Telas de Gerenciamento de Estágios

****

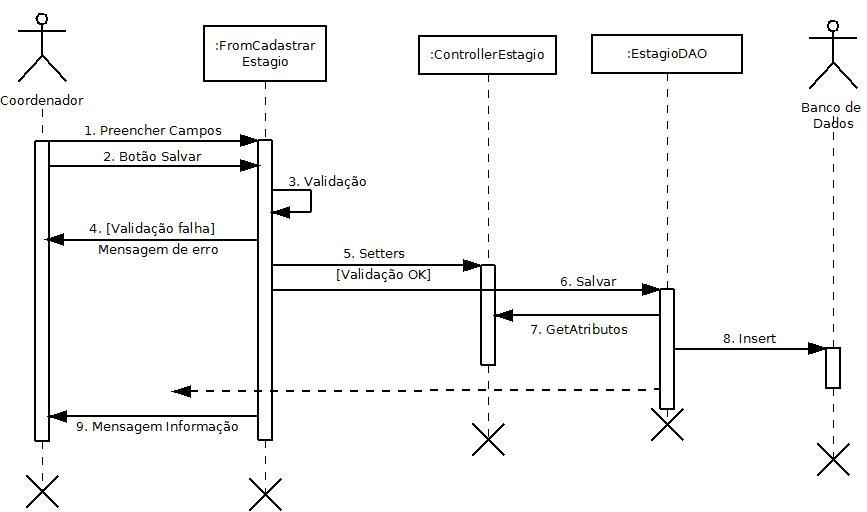
* 1. **Modelagem UML**

O UML (Unified Modeling Language ou Linguagem de Modelagem Unificada) é uma linguagem padrão para modelagem orientada a objetos. Seus objetivos são a especificação, documentação e estruturação do sistema/software e, para atingi-los, conta com vários diagramas, os quais permitem uma pré-visualização das funções do sistema. Uma de suas razões reside no fato de que sistemas definidos como “pequenos” tem a tendência natural de “crescer”, isto é, de aumentar de tamanho, complexidade e abrangência. Tendo esses diagramas atualizados, proporcionar futuras mudanças tornam-se mais fáceis.

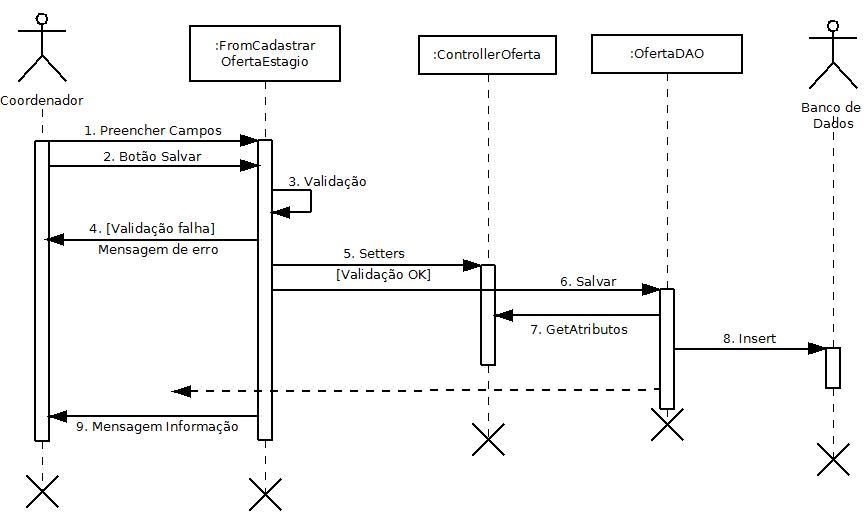
* + 1. **Diagrama simplificado de Classes: Modelo de Domínio do SGE**

****

* + 1. **Diagrama de seqüência - Cadastro de Estágio**

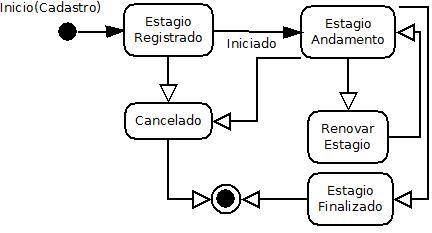
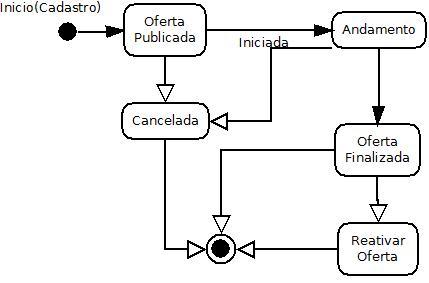


* + 1. **Diagrama de seqüência: Cadastro de Oferta de Estágios**

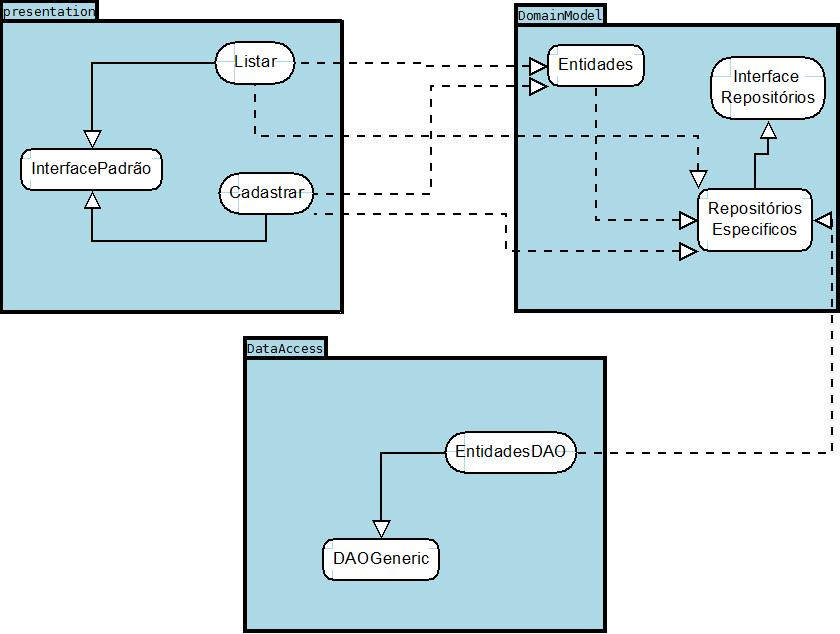
1. 

**3**

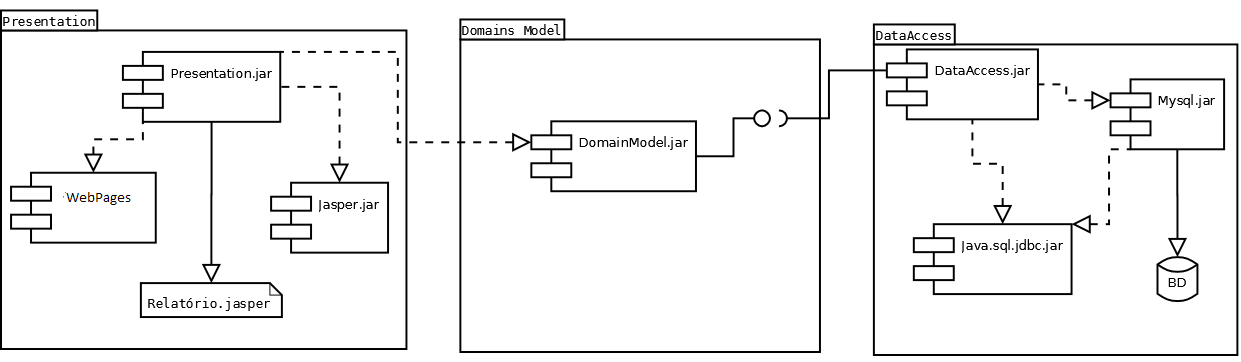
* + 1. **Diagramas de Estado**

**Diagrama de Estado: Classe Estágio           Diagrama de Estado: Classe Oferta Estágio**

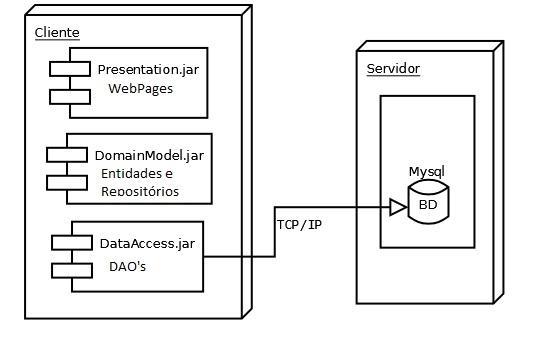
* + 1. **Diagrama de Pacotes**



* + 1. **Diagrama de Componentes**



* + 1. **Diagrama de Implantação**

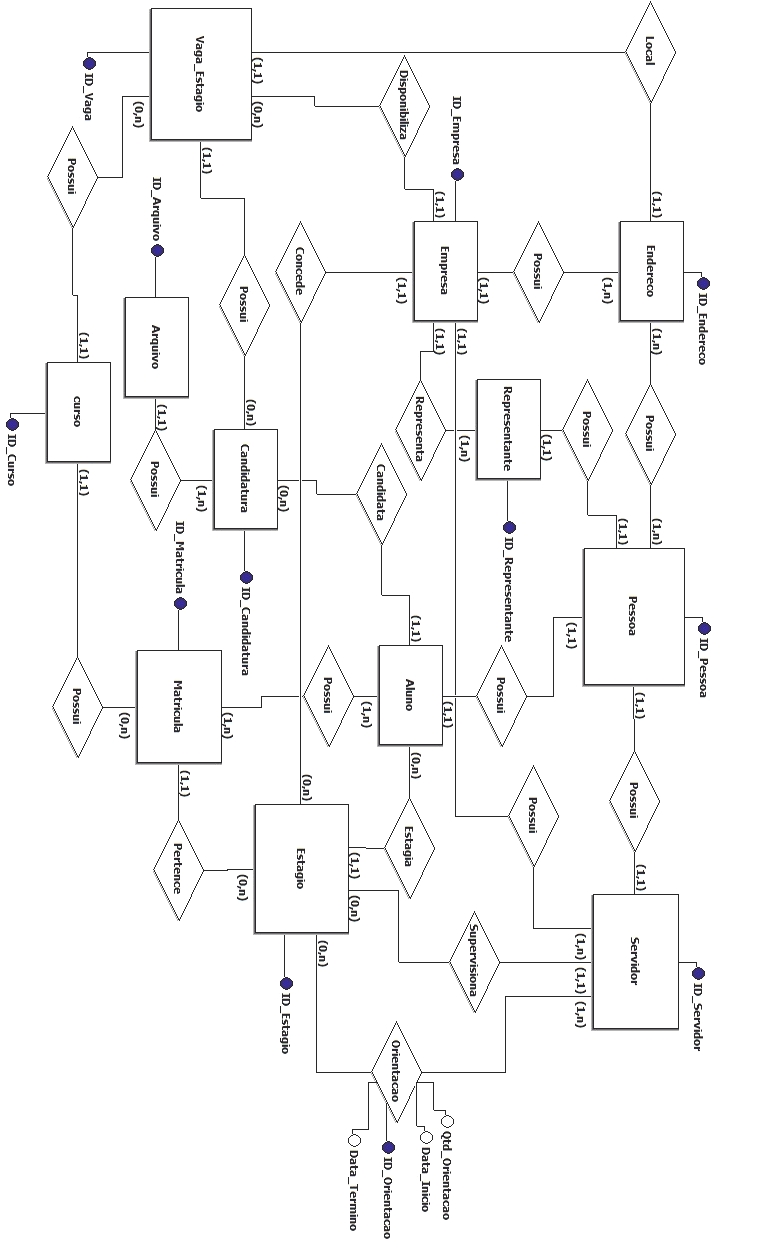


* 1. **Modelagem Conceitual e Lógica**

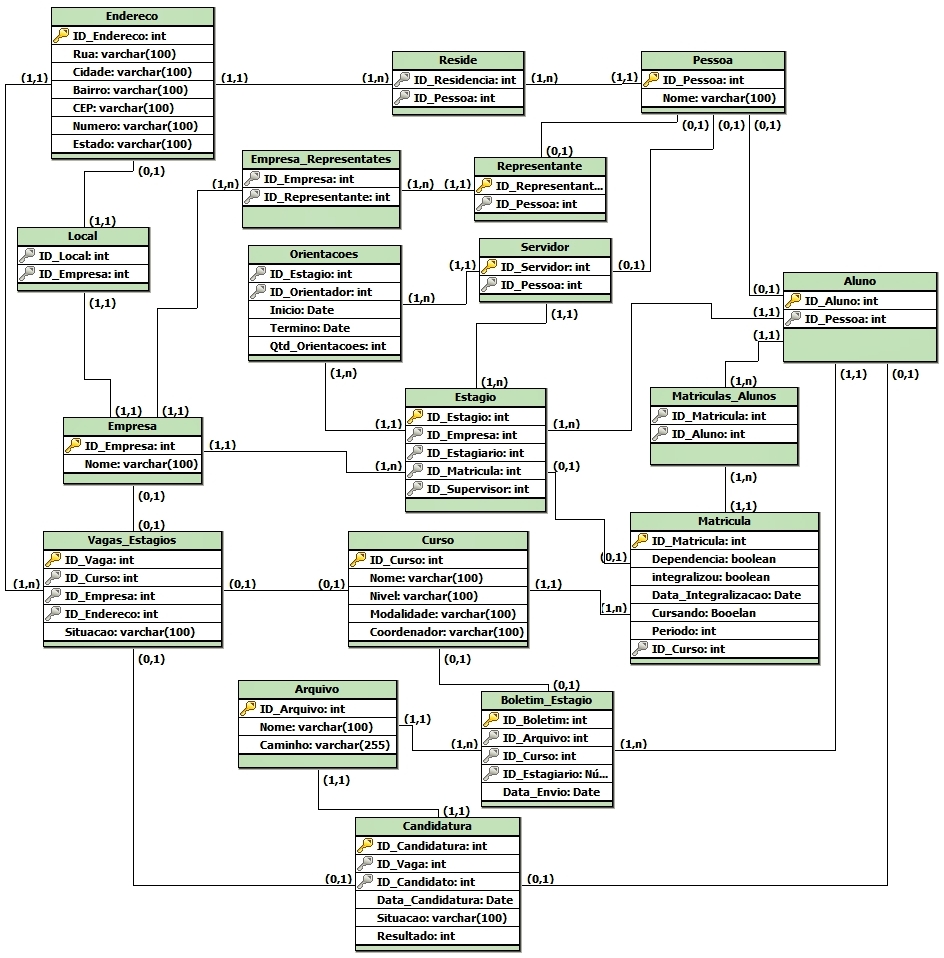
A modelagem de dados é uma técnica que faz a especificação das regras de negócios e as estruturas de dados de um banco de dados. Ela é de vital importância para o bom resultado do projeto. Modelar dado consiste em desenhar o sistema de informações, concentrando-se nas entidades lógicas e nas dependências lógicas entre essas entidades. Segundo MACEDO (2011), para criar um banco de dados, o primeiro passo é a modelagem de dados, é nele que criamos um modelo de dados específico para um domínio. Sua função é dar auxílio no entendimento da complexidade do mundo real, e a partir dele, tem-se mais segurança para o seu projeto dar certo, atendendo as necessidades do usuário final.

* + 1. **Modelo Entidade-Relacionamento**

O Modelo Entidade-Relacionamento, ou Diagrama Entidade-Relacionamento (DER), é um modelo em formato de diagrama que descreve o banco de dados de um sistema. Sua finalidade é identificar entidades, atributos e seus relacionamentos. Uma Entidade representa um conjunto de informações referentes a um determinado objeto. Enquanto que, um Relacionamento, descreve sua interação com as demais. A entidade é composta por Atributos, que representam suas características. Este último é classificado em três tipos:

* **Determinante** - Representa um valor único na entidade e deve ser sublinhado;
* **Composto** - Necessita ser dividido em sub-atributos para que seu significado seja compreendido;
*  **Multi-valorado** - Pode assumir mais de um valor para cada registro da entidade.
  + 1. **Modelo Relacional**

O Modelo Relacional (MR) é um modelo para diagramação que mais se assemelha à estrutura de um banco de dados, pois seu principio é de que todos os dados são armazenados em tabelas. Ele é desenvolvido usando, como base, o DER (Diagrama Entidade Relacionamento). Este modelo é constituído por tabelas e suas relações são definidas por passagem de valores de determinados campos. Estes recebem o nome de foreing key. Entretanto, para que um campo possa ser passado, é necessário que ele possua valores únicos, que não se repetem internamente, chamado de primary key (chave amarela).



1. **Conclusão**

A coordenação de estágios não disponibilizava de ferramentas eficientes para o correto gerenciamento dos estágios realizados pelos alunos do IFNMG – Campus Januária, sendo assim realizado através de uma planilha no Excel que atende as necessidades parcialmente.

A Gestão de Estágios exercida na Coordenação de Estágios do IFNMG Campus Januária é uma prática realizada pelos funcionários deste setor em um processo manual e burocrático. Tal processo é suscetível à ocorrência de erros e extravios, dificultando a segurança das informações referentes à gestão dos estágios e acesso automatizado à informação. Analisando esse contexto, foi desenvolvido o Software de Gestão de Estágios para a Coordenação de Estágios do IFNMG Campus Januária com a finalidade de qualificar o processo vivenciado neste setor.

O Sistema de Gerenciamento de Estágios utiliza componentes abertos e gratuitos de software e possibilita a sua distribuição e uso nos termos da licença GNU.

O Sistema de Gerenciamento de Estágios teve seus objetivos alcançados, mas ainda há melhorias que podem ser realizadas, visto que o sistema tende a crescer em novas funcionalidades, com o decorrer do tempo. Assim sendo, ele foi projetado para receber facilmente modificações.

1. **Referências**

Medeiros. Higor. **Introdução à JPA - Java Persistence API (2009),** Disponível em: < (http://www.devmedia.com.br/introducao-a-jpa-java-persistence-api/28173) >

**Oliveira. Erick C M.** Persistência em Java com API JPA **(2011),** Disponível em:http://www.linhadecodigo.com.br/artigo/25**25/persistencia-em-java-com-api**jpa.aspx

Kenneth Laudon e Jane Laudon. **Sistema de Informação Gerenciais**, 9 ed. São Paulo: Editora PEARSON, 2011.

SOMMERVILLE, Ian. **Engenharia de Software**, 9 ed. São Paulo: Editora PEARSON, 2011.

PRESSMAN, Roger S. **Engenharia de Software**, 3 ed. São Paulo: Editora MAKRON, 1995.

GUEDES, Gilleanes T. A. **UML Uma Abordagem Prática**, 2 ed. São Paulo: Editora NOVATEC, 2004.

LARMAN, Craig. **Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**, 3 ed. São Paulo: Editora BOOKMAN, 2005.

KOSCIANSKI, André; SOARES, Michel dos Santos. **Qualidade de Software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. 2. ed. São Paulo : Nova Tec Editora, 2007.

FURGERI, Sérgio. **Java 7: ensino didático.** 2° ed. São Paulo: editora Erica, 2012.

ORACLE - JAVA. **JEE - Java Enterprise Edition,** 2010. Disponível em: <https://www.java.com/pt\_BR/download/faq/techinfo.xml >.

PANDA, Debu; RAHMAN Reza; LANE, Derek. **EJB 3 em ação.** 2° edição - Rio de Janeiro: Alta Books, 2009.

GODOY. Fernando. **O que é JSF?**, 2011. Disponível em: < http://fernandogodoy.wordpress.com/2011/02/12/o-que-e-jsf/>.

FERNANDES, Michel Pereira. **Desenvolvendo Aplicações com Primefaces**, 2012. Disponível em: <http://www.devmedia.com.br/desenvolvendo-aplicacoes- ria-com-primefaces-revista-easy-java-magazine-24-parte-2/26400>.

RICARTE, Ivan Luiz Marques. **Bancos de Dados Relacionais,** 2002. Disponível em: <http://www.dca.fee.unicamp.br/cursos/PooJava/javadb/bdrel.html>.

MACEDO, Diego. **Modelagem Conceitual, Lógica e Física de Dados,** 2012. Disponível em: <http://www.diegomacedo.com.br/modelagem-conceitual-logica-e-fisica-de-dados/>.

CAVALCANTI, Guilherme José Carvalho. **Um Estudo sobre a Abordagem Semiestruturada para Resolução de Conflitos no Git.** Recife, 2013. Disponível em: <http://www.cin.ufpe.br/~tg/2013-2/gjcc-proposta.pdf>..

GILLIAM, James O. Optimizando o Modelo Open Source Software com Ferramentas Case UML. Tradução por BENNETT, Bill. **Gazeta do Linux… tornando o Linux um pouco mais divertido!**, 2001. Disponível em: <http://www.gacetadelinux.com/pr/lg/issue67/gilliam.html>.

GONZAGA, Flávio S; BIRCKAN, Guilherme. **Curso de PHP e MySQL**. **Florianópolis – SC**, 2000. Disponível em: <http://bars.paginas.sapo.pt/php2.pdf>.

INDRUSIAK, Leandro Soares. Linguagem Java. **Grupo JavaRS JUG Rio Grande do Sul**, 1996. Disponível em: <http://www.cin.ufpe.br/~arfs/introjava.pdf>.

MELLO, Eduardo Mattos da Silva**. Gitabs: Uma extensão ao sistema Git para gestão de projetos**. **Porto Alegre**, 2013. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/109005/000909876.pdf?sequence=1>.

**NIEDERAUER, Juliano; PRATES, Rubens. Guia de Consulta Rápida MySQL 5. Novatec Editora,** 2006. Disponível em: <https://1ce45944-a-62cb3a1a-s-sites.googlegroups.com/site/inedweb/Home/estrutura-de-dados/MySQL-GuiadeConsultaR%C3%A1pida.pdf?attachauth=ANoY7cqjXx43u-9z4EgaJ4mgca0S-HYPERLINK "https://1ce45944-a-62cb3a1a-s-sites.googlegroups.com/site/inedweb/Home/estrutura-de-dados/MySQL-GuiadeConsultaRápida.pdf?attachauth=ANoY7cqjXx43u-9z4EgaJ4mgca0S-f2VJfHo3vNLYnE3N2pkJ\_E-NgrMUQ6S9zliFHviRQ0FNL-RJtpRDR58yfo-LU004BFVB9xfU8\_4AUBFyJuOpRwLjOD8xFJzEXTnB52Y7nQONlGQja99IXqTN1hj79HClNPfY02GoV7DQZX8\_gXv66eLb7\_Uq2OrpFkXLQYRamrXYGWRZvl2ZaUkbuInCewUGFN3wVS2vDOpJIM\_uqcOSxToXNIYtPVdwjEOcKB3lBUOGmTdgtJ9vZ79TN5OLbIgBdMwWA==&attredirects=0"f2VJfHo3vNLYnE3N2pkJ\_E-NgrMUQ6S9zliFHviRQ0FNL-RJtpRDR58yfo-LU004BFVB9xfU8\_4AUBFyJuOpRwLjOD8xFJzEXTnB52Y7nQONlGQja99IXqTN1hj79HClNPfY02GoV7DQZX8\_gXv66eLb7\_Uq2OrpFkXLQYRamrXYGWRZvl2ZaUkbuInCewUGFN3wVS2vDOpJIM\_uqcOSxToXNIYtPVdwjEOcKB3lBUOGmTdgtJ9vZ79TN5OLbIgBdM-wWA%3D%3DHYPERLINK "https://1ce45944-a-62cb3a1a-s-sites.googlegroups.com/site/inedweb/Home/estrutura-de-dados/MySQL-GuiadeConsultaRápida.pdf?attachauth=ANoY7cqjXx43u-9z4EgaJ4mgca0S-f2VJfHo3vNLYnE3N2pkJ\_E-NgrMUQ6S9zliFHviRQ0FNL-RJtpRDR58yfo-LU004BFVB9xfU8\_4AUBFyJuOpRwLjOD8xFJzEXTnB52Y7nQONlGQja99IXqTN1hj79HClNPfY02GoV7DQZX8\_gXv66eLb7\_Uq2OrpFkXLQYRamrXYGWRZvl2ZaUkbuInCewUGFN3wVS2vDOpJIM\_uqcOSxToXNIYtPVdwjEOcKB3lBUOGmTdgtJ9vZ79TN5OLbIgBdMwWA==&attredirects=0"&HYPERLINK "https://1ce45944-a-62cb3a1a-s-sites.googlegroups.com/site/inedweb/Home/estrutura-de-dados/MySQL-GuiadeConsultaRápida.pdf?attachauth=ANoY7cqjXx43u-9z4EgaJ4mgca0S-f2VJfHo3vNLYnE3N2pkJ\_E-NgrMUQ6S9zliFHviRQ0FNL-RJtpRDR58yfo-LU004BFVB9xfU8\_4AUBFyJuOpRwLjOD8xFJzEXTnB52Y7nQONlGQja99IXqTN1hj79HClNPfY02GoV7DQZX8\_gXv66eLb7\_Uq2OrpFkXLQYRamrXYGWRZvl2ZaUkbuInCewUGFN3wVS2vDOpJIM\_uqcOSxToXNIYtPVdwjEOcKB3lBUOGmTdgtJ9vZ79TN5OLbIgBdMwWA==&attredirects=0"attredirects=0>.

**NETBEANS.ORG**, Disponível em: <https://netbeans.org/features/index.html>. Acesso em: 07 Março. 2016.