



UNIVERSIDADE FEDERAL DO PARÁ
FACOMP – FACULDADE DE COMPUTAÇÃO
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

SISTEMA DE EGRESSOS UFPA – CAMPUS CASTANHAL

FILIPE MORAES RODRIGUES

Castanhal-PA
2021



UNIVERSIDADE FEDERAL DO PARÁ
FACOMP – FACULDADE DE COMPUTAÇÃO
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

FILIFE MORAES RODRIGUES

SISTEMA DE EGRESSOS UFPA – CAMPUS CASTANHAL

Trabalho de Conclusão de Curso
submetido ao colegiado da Faculdade de
Computação da Universidade Federal do
Pará, como requisito parcial para a
obtenção do grau de bacharel em
SISTEMAS DE INFORMAÇÃO.

Orientador: Prof. Dr. José Jailton Júnior
Coorientador: Prof. André Felipe

Castanhal-PA
2021

UNIVERSIDADE FEDERAL DO PARÁ
FACOMP – FACULDADE DE COMPUTAÇÃO
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

SISTEMA DE EGRESSOS UFPA – CAMPUS CASTANHAL

Trabalho de Conclusão de Curso apresentado ao Colegiado da Faculdade de Computação (FACOMP) da Universidade Federal do Pará do campus de Castanhal, como requisito parcial para a obtenção do Grau de bacharel em SISTEMAS DE INFORMAÇÃO.

Prof. Dr. Jose Jailton Júnior
Orientador-UFPA/FACOMP

Prof. Felipe André da Costa Brito
Coorientador – Doutorando/UFPA

Prof. Dr. Igor Ruiz Gomes
Membro da Banca – UFPA/FACOMP

Prof. Dr. Tassio Costa de Carvalho
Membro da Banca – UFPA/FACOMP

Prof. Dr. José Jailton Júnior
Diretor (a) da Faculdade de Computação - FACOMP

DEDICATÓRIA

Dedico este Trabalho de Conclusão de Curso a todos que me ajudaram nesta trajetória, em especial aos meus pais, que incansavelmente estiveram comigo depositando total confiança e apoio durante todo esse percurso. Aos meus irmãos, que sempre acreditaram em mim, apoiando-me em todas as minhas decisões e me dando força nos momentos mais difíceis. Aos demais familiares, que tanto admiro, também dedico o resultado do esforço realizado ao longo deste percurso. Aos melhores amigos que ainda tenho, pelo incentivo diário, ajudando-me a superar todas as fases difíceis desta caminhada, até mesmo onde eu achava que poderia desistir. Dedico também ao meu orientador e coorientador, que me deram essa oportunidade e credibilidade. Aos meus colegas e amigos de curso, que assim como eu encerram uma difícil etapa da vida acadêmica. Aos colegas de trabalho, que me apoiaram e compreenderam sempre quando precisei. E por fim, dedico àquelas pessoas que contribuíram de alguma forma, direta ou indiretamente, com a realização deste trabalho.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por sempre estar comigo em todos os momentos, não sendo diferente nas etapas deste trabalho, dando-me força, coragem, concentração e perseverança para que eu pudesse superar todos os desafios encontrados durante este curso. Aos meus Pais, irmãos e familiares, pelo apoio e credibilidade depositados em toda minha trajetória acadêmica até aqui. Agradeço o apoio dos meus colegas de sala, que sempre me ajudaram com relação à troca de conhecimentos e incentivos diários. Ao meu orientador e coorientador, pela confiança e oportunidade concedidas, pelo apoio, contribuição e disponibilização que foram essenciais para a conclusão deste trabalho. Aos professores da facomp, pela contribuição em meus conhecimentos acadêmicos, imprescindíveis para o desenvolvimento deste trabalho. Aos meus companheiros de trabalho, pela compreensão e companheirismo. E finalmente, agradeço a todos de uma forma geral que, diretamente ou indiretamente, contribuíram para que eu pudesse atingir esse objetivo. Muito obrigado a todos!

*“As dificuldades preparam pessoas
comuns para caminhos extraordinários”
(C. S. Lewis)*

SUMÁRIO

LISTA DE ILUSTRAÇÕES	10
LISTA DE TABELAS.....	11
RESUMO	12
ABSTRACT.....	13
1 INTRODUÇÃO	14
1.1 CONTEXTUALIZAÇÃO	14
1.2 JUSTIFICATIVA.....	14
1.3 OBJETIVOS.....	15
1.3.1 <i>Objetivo geral</i>	15
1.3.2 <i>Objetivos Específicos</i>	15
1.4 ESTRUTURA DO TRABALHO	15
2 REFERENCIAL TEÓRICO	14
2.1 ENGENHARIA DE SOFTWARE	17
2.2 PARADIGMA DA ORIENTAÇÃO A OBJETOS	19
2.2 APLICAÇÕES WEB	20
3 TRABALHOS RELACIONADOS	22
4 METODOLOGIA.....	23
5 FERRAMENTAS E MODELO ARQUITETURAL	24
5.1 SPRING FRAMEWORK.....	24
5.2 REACT JS	25
5.3 JWT	25
5.4 VISUAL STUDIO CODE	26
5.5 MYSQL WORKBENCH.....	26
5.6 ASTAH COMMUNITY	27
5.7 MODELO ARQUITETURAL.....	27
6 SISTEMA DE EGRESSOS	30
6.1 LEVANTAMENTO DE REQUISITOS	31
6.2 USUÁRIOS DO SISTEMA	31
6.3 DIAGRAMA DE CASOS DE USO	31
6.3.1 <i>Casos de Uso específicos do ator Egresso</i>	32
6.3.2 <i>Casos de Uso específicos do ator Empresa</i>	33
6.3.3 <i>Casos de Uso específicos do ator Coordenador</i>	33
6.3.4 <i>Casos de Uso específicos do ator Administrador</i>	33
6.3.5 <i>Casos de Usos comuns ao Administrador e Coordenador</i>	34
6.3.6 <i>Casos de Usos comuns a todos</i>	34
6.4 DIAGRAMA DE CLASSES	34
6.5 MODELO ENTIDADE RELACIONAMENTO.....	36
7.2.1 <i>Diagrama ER</i>	36
7 RESULTADOS	38
7.1 TELA PRINCIPAL DO SISTEMA	38
7.2 TELA DE CADASTRO DE EGRESSOS.....	38

7.3	TELA DE PERFIL DO EGRESSO	38
7.4	TELA DE VISUALIZAÇÃO DE OPORTUNIDADES.....	39
7.5	TELA DE CADASTRO DE EMPRESAS	39
7.6	TELA GERENCIAR OPORTUNIDADES	40
7.7	TELA DE VISUALIZAÇÃO DOS EGRESSOS	41
7.8	TELA DE DADOS ESTATÍSTICOS	42
8	CONCLUSÃO	43
8.1	TRABALHOS FUTUROS	43
	REFERÊNCIAS	44

LISTA DE ILUSTRAÇÕES

Figura 2.1- Camadas da Engenharia de Software.....	18
Figura 5.1- Framework Spring Boot	25
Figura 5.2- Biblioteca React JS	25
Figura 5.3- Sistema Json Web Token.....	26
Figura 5.4- Ferramenta VSCode.....	26
Figura 5.5- Ferramenta MySQL Workbench	27
Figura 5.6- Ferramenta Astah Community.....	27
Figura 5.7- Representação esquemática da arquitetura REST	29
Figura 6.1- Diagrama geral de Casos de Uso do Sistema	33
Figura 6.2- Diagrama de Classes do Sistema	36
Figura 6.3- Diagrama ER do Sistema.....	37
Figura 7.1- Tela Principal do Sistema	38
Figura 7.2- Tela de Cadastro de Egressos	39
Figura 7.3- Tela de Perfil do Egresso	39
Figura 7.4- Tela de Visualização de Oportunidades.....	40
Figura 7.5- Tela de Cadastro de Empresas	41
Figura 7.6- Tela Gerenciar Oportunidades	41
Figura 7.7- Tela de Visualização dos Egressos	42
Figura 7.8- Tela de Dados Estatísticos dos Egressos	42

LISTA DE TABELAS

Tabela 6.1- Requisitos Funcionais e não Funcionais do sistema	30
---	----

RESUMO

Neste Trabalho de Conclusão de Curso foram utilizados os conhecimentos adquiridos no decorrer desta formação acadêmica para o desenvolvimento de um sistema de egressos à Universidade Federal do Pará (UFPA), campus universitário de Castanhal, buscando estreitar a relação entre ex-alunos, a instituição e o mercado de trabalho, tendo em vista que essa interação é geralmente prejudicada devido à perda de vínculo com a universidade, sendo também uma realidade na maioria das IES (Instituições de Ensino Superior). Dessa forma, este sistema será fonte de informações essenciais à gestão de ensino da instituição e auxiliará seus egressos no ingresso ao mercado de trabalho, implicando direta e indiretamente no sucesso da mesma. Utilizou-se em seu desenvolvimento recursos da Engenharia de Software para modelagem de sistemas, uso de padrões de projetos, além de um conjunto de ferramentas de grande utilidade para construção de sistemas modernos.

PALAVRAS-CHAVES: Egressos. IES. Gestão de Ensino, Mercado de trabalho. Modelagem de Sistemas.

ABSTRACT

In this End of Course Work, the knowledge acquired during this academic formation was used for the development of a system of egresses to the Federal University of Pará (UFPA), university campus of Castanhal, seeking to strengthen the relationship between former students, the institution and the labor market, in view that this interaction is usually impaired due to the loss of link with the university, being also a reality in most HEIs (Higher Education Institutions). Thus, this system will be a source of essential information for the institution's educational management and will help its graduates to enter the job market, implying directly and indirectly in its success. It was used in its development Software Engineering resources for system modeling, the use of project standards, and a set of very useful tools for building modern systems.

KEYWORDS: Egresses. IES. Teaching Management, Labor Market. Systems Modeling.

1 INTRODUÇÃO

1.1 Contextualização

O atual cenário de alta competitividade do mercado tem exigido cada vez mais capacitação e qualificação por parte dos profissionais atuantes nas diversas organizações que o integram, bem como daqueles que ainda buscam uma oportunidade de ingressar no mesmo. A procura intensificada por profissionais com esse perfil, baseada em critérios comuns e, por vezes, específicos de cada organização, fez com que nos últimos anos houvesse um aumento expressivo de aberturas de Instituições de Ensino Superior com a intenção de atender a essa demanda do mercado. Entretanto, o sucesso das IES depende da adoção de um conjunto de parâmetros que visem a qualidade do processo de ensino-aprendizado, para que tenhamos egressos qualificados e de conhecimentos tácitos. “Formar cidadãos aptos a exercerem atividades produtivas ainda é um desafio em muitos países como o Brasil. Mas é preciso mais que isso. É preciso formar cidadãos capazes para desempenhar atividades que sequer existem atualmente”. (CASTRO apud MEHEDFF, 1999, p. 5).

A missão de uma IES precisa engajar-se no sentido de formar pessoas capazes de empregar o conhecimento adquirido no desenvolvimento de soluções eficientes, que tragam benefícios à sociedade nos aspectos sociais, econômicos, ambientais, entre outros, tornando-os agentes de mudança e transformação da mesma. Segundo Martins (1986), a sociedade possui um caráter dinâmico e capacidade de criação inerente à própria natureza humana, dessa forma, precisamos de Instituições de Ensino preocupadas em motivar e impulsionar pessoas na busca pelo constante aperfeiçoamento, servindo e influenciando de forma positiva o seu meio externo.

1.2 Justificativa

Na busca por profissionais qualificados que atendam os diversos requisitos do mercado atual e que sejam um diferencial para o crescimento de seus negócios, as empresas têm recorrido às instituições de ensino superior (IES) para contratação de seu pessoal. Entretanto, geralmente depois de formados, os ex-alunos perdem o vínculo com a instituição, o que torna difícil a comunicação com os mesmos, e faz com que não se tenham informações relevantes para indicá-los ao mercado e acompanhá-los em suas trajetórias acadêmicas e de atuação profissional. Na UFPA, campus Castanhal, essa realidade também foi observada. Atualmente, a instituição não dispõe de um sistema

específico para indicação e acompanhamento dos seus egressos. Com isso, eles deixam de ser fonte informações essenciais ao processo de avaliação de seus cursos, as quais poderiam refletir em propostas de melhoria ao método de ensino da instituição, trazendo melhores resultados no desempenho de seus discentes e egressos. Dessa forma, esse projeto visa favorecer a relação entre Egressos, Instituição e Empresas, a partir do desenvolvimento do *Sistema de Egressos UFPA – Campus Castanhal*, um sistema web que auxilie e cadastre os egressos da instituição, buscando manter seus dados atualizados, disponibilizando informações para o acompanhamento de suas trajetórias acadêmicas/profissionais e inserindo-os no mercado de trabalho por meio do vínculo com empresas parceiras.

1.3 Objetivos

1.3.1 Objetivo geral

Desenvolvimento de um sistema web para acompanhamento da trajetória acadêmica/profissional dos egressos da UFPA, campus universitário de Castanhal, e indicação destes ao mercado de trabalho.

1.3.2 Objetivos Específicos

- Promover a interação dos egressos com a instituição e mercado de trabalho.
- Obter informações essenciais para a constante melhoria da qualidade de ensino da instituição.
- Manter uma base de dados consistente desses egressos.

1.4 Estrutura do trabalho

Este trabalho está organizando em oito capítulos, sendo que o restante deste trabalho está organizado como segue:

Capítulo 2: Este capítulo apresenta as principais referências teóricas que fundamentaram o desenvolvimento deste projeto.

Capítulo 3: Capítulo de trabalhos relacionados, que seguem a mesma linha de pesquisa sobre egressos e a importância dos mesmos para as Instituições de Ensino Superior.

Capítulo 4: Capítulo destinado à apresentação da metodologia utilizada no desenvolvimento do projeto, a fim de obter melhores resultados no produto final.

Capítulo 5: Capítulo apresenta as ferramentas tecnológicas e o modelo arquitetural empregados ao sistema desenvolvido, buscando apresentar a lógica de funcionamento e o que entregam de recursos ao sistema.

Capítulo 6: Este capítulo discorre sobre o sistema e suas características, explica seu funcionamento, apresenta seus usuários, aborda sobre o processo de levantamento de requisitos, mostrando o que o sistema deve fazer (requisitos funcionais) e suas restrições (requisitos não funcionais), e por fim realiza sua modelagem por meio do uso de diagramas amplamente utilizados na Engenharia de Software.

Capítulo 7: Neste Capítulo são apresentados os resultados obtidos no desenvolvimento do sistema proposto, utilizando as principais telas construídas para interação com seus usuários.

Capítulo 8: Capítulo destinado a apresentar uma breve conclusão sobre o trabalho apresentado e suas propostas de trabalhos futuros.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta as principais referências teóricas que fundamentaram o desenvolvimento deste projeto de sistema e está organizado em 3 seções. A seção 2.1 discorre sobre a Engenharia de Software. A seção 2.2 apresenta o paradigma da orientação a objetos. A seção 2.3 realiza uma abordagem sobre aplicações web.

2.1 Engenharia de Software

A Engenharia de Software é uma área da Ciência da Computação que abrange os processos de desenvolvimento, operação e manutenção de sistemas, utilizando um conjunto de métodos e ferramentas tecnológicas para obtenção de organização, produtividade e qualidade no processo de software. “É uma disciplina que se preocupa com todos os aspectos de produção de software”. Sommerville (2011, pg. 4).

Para melhor compreender a essência da Engenharia de Software é necessário que se tenha em mente uma definição mais ampla do que é um software.

Software consiste em: (1) instruções (programas de computador) que, quando executadas, fornecem características, funções e desempenho desejados; (2) estruturas de dados que possibilitam aos programas manipular informações adequadamente; e (3) informação descritiva, tanto na forma impressa quanto na virtual, descrevendo a operação e o uso dos programas. (PRESSMAN, 2016, p. 4)

Segundo Pressman (2016), a Engenharia de Software envolve um processo, métodos e ferramentas de desenvolvimento e gerenciamento de software, possibilitando a obtenção de melhor qualidade no produto final. “De fato, a qualidade dos produtos de software depende fortemente da qualidade dos processos usados para desenvolvê-los e mantê-los”. (FALBO, 2014, p. 2).

Pressman (2016), conceitua a Engenharia de Software como uma tecnologia em camadas, onde todas as suas abordagens devem ter como pilar o foco na qualidade. A figura 2.1 faz a apresentação dessa perspectiva.

Figura 2.1 – Camadas da Engenharia de Software

Fonte: Pressman (2016)

A camada de *processo* é a base da engenharia de software, estabelecendo o contexto para a aplicação dos métodos e utilização das ferramentas. “O processo de software define uma metodologia que deve ser estabelecida para a entrega efetiva de tecnologia de engenharia de software”. Pressman (2016).

Os *métodos* da engenharia de software fornecem os detalhes para o desenvolvimento de software, envolvendo um amplo conjunto de atividades: comunicação, análise de requisitos, modelagem de projeto, design, implementação, testes, implantação e gerenciamento.

As *ferramentas* da engenharia de software apoiam o processo e os métodos, de maneira automatizada ou semi-automatizada. Quando essas ferramentas se integram e interagem umas com as outras, formam um sistema de engenharia de software auxiliada por computador. Pressman (2016).

2.1.1 Processo de Software

“Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software. Essas atividades podem envolver o desenvolvimento de software a partir do zero em uma linguagem padrão de programação como Java ou C.” Sommerville (2011, p. 18).

Segundo Falbo (2014), um processo de software, envolve um conjunto de atividades que podem ser classificadas quanto ao seu propósito da seguinte forma:

Atividades de Desenvolvimento: são as atividades voltadas ao processo de desenvolvimento do software, envolvendo as seguintes atividades: análise e especificação de requisitos, projeto, implementação, testes, entrega e implantação do Sistema.

Atividades de Gerência: envolvem um conjunto de atividades de gerenciamento do projeto de forma abrangente, incluindo processo de gerência de projetos, processo de gerência de configuração, processo de gerência de reutilização, entre outros.

Atividades de Controle da Qualidade: estão relacionadas com o procedimento de avaliação da qualidade do produto em desenvolvimento e com o processo de evolução de software.

2.2 Paradigma da orientação a objetos

No paradigma da orientação a objetos é realizado o gerenciamento da complexidade do mundo real através da análise detalhada dos problemas abordados, para assim possibilitar o desenvolvimento de propostas de soluções que incorporem uma abstração do conhecimento necessário ao domínio da aplicação, encapsulando-o dentro de objetos. Segundo Correia (2006), pode-se entender um objeto como uma “coisa” física, por exemplo, um carro, uma casa ou uma moto, ou algo abstrato, como uma equação ou uma conta bancária, caso puramente mental, onde não existe uma “coisa” física que possa impressionar nossos sentidos para que estes o percebam como um objeto físico.

Para o entendimento da proposta da orientação a objetos faz-se necessário a representação das situações do mundo real de forma mais clara possível nos sistemas computacionais. O mundo é composto por vários objetos que, em sua dinâmica, interagem uns com os outros. Esse mesmo conceito é aplicado aos sistemas computacionais.

O paradigma da orientação a objetos visualiza um sistema de software como uma coleção de agentes interconectados chamados objetos. Cada objeto é responsável por realizar tarefas específicas. Para cumprir com algumas das tarefas sob sua responsabilidade, um objeto pode ter que interagir com outros objetos. É pela interação entre objetos que uma tarefa computacional é realizada”. (Bezerra, 2015, p. 18).

Na perspectiva da modelagem de sistemas, cada objeto é uma entidade que incorpora uma abstração importante para o funcionamento de uma aplicação. Para Bezerra (2015, p. 19) “Um sistema de software orientado a objetos consiste em objetos em colaboração com o objetivo de realizar as funcionalidades desse sistema. Cada objeto

é responsável por tarefas específicas. É graças à cooperação entre objetos que a computação do sistema se desenvolve”.

2.2.1 Análise orientada a objetos

O processo de Análise Orientada a Objetos consiste em realizar a identificação de quais objetos existem no contexto do sistema a ser desenvolvido e como eles se comportam em sua dinâmica de interação entre si. São especificados os atributos (características) e ações desses objetos. Desta forma, o analista responsável pelo projeto poderá repassar o entendimento que possui acerca do funcionamento do sistema para o usuário leigo (CORREIA, 2006).

Segundo Bezerra (2016), um dos resultados da análise dentro do processo de desenvolvimento orientado a objetos é a construção do modelo de objetos, representados por meio das classes componentes do sistema. Dessa análise também resulta um modelo funcional do sistema de software a ser desenvolvido. Com isso, é possível ampliar a compreensão da complexidade do problema a ser solucionado e obter um sistema que, por fim, atenda às necessidades de seus utilizadores.

2.3 Aplicações Web

Com o aumento em larga escala dos recursos disponíveis na internet e o seu uso amplamente difundido pelo mundo, nota-se a constante expansão e aperfeiçoamento da área de desenvolvimento web e suas tecnologias, ganhando força e atraindo muitos profissionais com a finalidade de atender à crescente demanda por sistemas que estejam disponíveis na rede global. A observação da diferença existente entre a internet de hoje e aquela há dez anos, mostra o nível de complexidade atingido pelas aplicações web e a rapidez com que ocorreram as mudanças. (Loudon, 2010).

Em termos gerais, os recursos mais utilizados no contexto do desenvolvimento web, ou seja, no desenvolvimento de softwares que utilizam os mecanismos da web, são o HTML, CSS, JavaScript e outras linguagens de programação usadas no lado backend da aplicação (lado servidor). Tais recursos já estão consolidados no mercado para a construção dos sistemas web modernos.

Pressman (2010), lista um conjunto atributos presentes na maior parte das aplicações Web, dentre os quais destacam-se os seguintes:

Desempenho: Essencial para que a aplicação apresente resultado satisfatório com relação ao tempo em que o usuário precisa aguardar para ter acesso e o retorno dos dados requeridos ao servidor.

Evolução contínua: É uma característica apresentada por aplicativos que evoluem de maneira contínua, ou seja, não dependem de um planejamento prévio ou de novas versões.

Segurança: É difícil limitar o acesso a uma aplicação web pelos usuários finais, devido a mesma estar disponível na rede. Logo, é necessário proteger o conteúdo sensível e fornecer mecanismos de segurança à aplicação.

Atualmente pode-se encontrar mais de trinta tipos de aplicações web no mercado, como e-commerce, mecanismos de busca, Web Sites corporativos, portais, entre outros. Segundo Gonçalves *et al.* (2005), na medida em que formas completas de negócio são migradas para o ambiente Web, surgem novos tipos de aplicações web constantemente. E isso continua elevando o grau de complexidade dessas aplicações.

Um dos principais fatores para atribuição de complexidade aos sistemas web são os mecanismos de integração entre os mesmos. No decorrer da evolução desses sistemas, surgiu a necessidade de implementar recursos de comunicação e troca de dados entre eles, realizando procedimentos de consultas, transações, validação, entre outros. E com a alta disponibilidade existem nos serviços web, essa interoperabilidade acontece praticamente a qualquer momento.

2.3.1 Modularização

Para mitigar a complexidade dos sistemas web pode-se utilizar o recurso de modularização, ou seja, “quebrar” esses sistemas em componentes menores (módulos) de forma estruturada. Segundo Loudon (2010, p. 19), “Em uma grande aplicação web, para tratar da complexidade valendo-se de componentes modulares, ou módulos, você deve encapsular tudo de que o componente precisa dentro de partes pequenas e bem definidas da aplicação”.

Talvez a modularidade não seja algo difícil de se atingir quando pensamos em sistema pequenos, mas se tratando de sistemas onde se tem um conjunto extenso de funcionalidade destinadas a diversos e variados tipos de usuários, pode se tornar um processo mais trabalhoso. “A modularidade de grandes aplicações web pode ser atingida, assim como em outros tipos de software, por meio do encapsulamento, da abstração e da

manutenção do maior número possível de acoplamentos fracos entre os módulos de uma aplicação”. Loudon (2010, p. 20).

3 TRABALHOS RELACIONADOS

Diversos trabalhos com estudos sobre egressos de IES apontam a importância do acompanhamento dos mesmos, destacando-os como fonte de informações imprescindíveis a essas instituições. Alguns incluem propostas de desenvolvimento de sistemas para realização desse acompanhamento e para auxiliar na inserção desses egressos no mercado de trabalho. Lousada e Martins (2005), realizaram um estudo sobre os egressos particularmente dos cursos de ciências contábeis, tendo-os como importante fonte de informação à gestão desses cursos. Esse estudo parte de análises empíricas no que diz respeito às percepções e motivações dos dirigentes das IES e especificamente dos cursos referidos, para obter a compreensão do problema abordado. Como resultado, tem-se o reconhecimento da necessidade de institucionalização e prática do acompanhamento dos egressos nas IES, destacando a importância do planejamento e desenvolvimento de sistemas para tal acompanhamento, os quais são tidos como um mecanismo fundamental para o contínuo aperfeiçoamento do planejamento e operações das instituições de ensino superior, resultando assim em serviços educacionais de melhor qualidade aos seus usuários.

Nascimento (2016), em seu trabalho de conclusão de curso, desenvolveu um sistema para o departamento de informática da Universidade Federal do Espírito Santo, que visa estimular o interesse dos alunos do ensino médio pela área de informática, através do acompanhamento dos egressos e seus depoimentos sobre os cursos da referida área. Para isso, são fornecidos dados como área de atuação, faixa salarial, curso de pós-graduação realizados, possibilitando assim obter informações de perfil desses egressos, gerar relatórios e disponibilizá-los na Internet. Nesse trabalho, ele aponta o acompanhamento dos egressos como um instrumento fundamental para o conhecimento do perfil dos graduados.

Costa e Fontana (2009), em monografia desenvolvida durante disciplina de conclusão de curso, mostram que as empresas têm buscado cada vez mais por profissionais que atendam aos requisitos de qualificação esperados para o desempenho de atividades importantes dentro delas, e com isso, têm recorridos às Instituições de Ensino

Superior para alcançar profissionais com esse perfil. Entretanto, após a conclusão da graduação, o vínculo do egresso com a IES pode ser comprometido, devido à falta de interação entre estes, inviabilizando sua indicação pela instituição. Dessa forma, é apontada a necessidade de desenvolver sistemas que realizem essa interação dos egressos com as instituições e o mercado de trabalho, auxiliando esses egressos na construção de suas carreiras profissionais, visto as dificuldades encontradas por eles. Como solução para o problema abordado, foi construído, utilizando recursos da Engenharia de Software e ferramentas de desenvolvimento sistemas, um protótipo de um sistema de egressos, visando ao mesmo poder ser utilizado em qualquer IES.

4 METODOLOGIA

A metodologia aplicada segue um roteiro de boas práticas de desenvolvimento de software, dentre elas, a observação dos princípios da engenharia de software, a definição de arquitetura e padrões de projetos e o uso de ferramentas atuais de mercado que oferecem desempenho e segurança ao sistema, além de maior produtividade no seu desenvolvimento. Para isso foram realizadas as seguintes atividades:

- 1. Revisão Bibliográfica:** Estudo das boas práticas de desenvolvimento de sistemas atuais, engenharia de requisitos, modelagem de banco de dados, programação orientada a objetos, padrões de projeto de sistemas e arquiteturas escaláveis, entre outros. Etapa imprescindível para a qualidade do software.
- 2. Estudo do negócio e levantamento de requisitos:** Etapa onde foi definido o escopo da aplicação, descrição geral do minimundo do sistema, abordando suas principais funcionalidades e regras de negócio, especificando requisitos funcionais e não funcionais, atores do sistema e suas interações com o mesmo por meio dos casos de uso, incluindo detalhes de cada um dos seus componentes.
- 3. Estudo das Tecnologias:** Etapa destinada ao aprofundamento e escolha das tecnologias utilizadas para o desenvolvimento do sistema, tais como: linguagem de programação; ambiente de desenvolvimento; banco de dados e ferramentas de modelagem de sistema; tecnologias de segurança e autenticação; frameworks front-end e back-end, dentre outras.

4. Modelagem do sistema: Nesta etapa foram contruídos diagramas que detalham as características e o funcionamento do sistema, de forma a auxiliar no seu entendimento, utilizando-se dos recursos da UML (em português, Linguagem de Modelagem Unificada).

5. Implementação e testes: Etapa onde o sistema foi implementado e submetido a testes. Todas as funcionalidades descritas foram implementadas buscando seguir as boas práticas abordadas nas etapas anteriores.

5 FERRAMENTAS E MODELO ARQUITETURAL

Para o desenvolvimento otimizado deste projeto foram utilizados softwares de modelagem, visando auxiliar no entendimento do sistema através do uso de elementos visuais; frameworks de programação front-end e back-end, buscando otimizar a construção, organização e estruturação do seu código-fonte; framework para modelagem e criação automatizada do banco de dados; sistema de autenticação baseado em token; software para controle de versão, entre outras ferramentas, as quais são bastante úteis na construção de sistemas modernos, onde o foco é aumentar a qualidade, reduzir custos e o tempo empregado no processo de desenvolvimento de software.

5.1 Spring Framework

O Spring Framework é uma ferramenta desenvolvida para a plataforma Java, amplamente utilizada no desenvolvimento de sistemas modernos. Baseia-se nos padrões de projeto inversão de controle e injeção de dependência, dispondo de um conjunto de tecnologias robustas que simplificam muitas tarefas realizadas pelos programadores, aumentando assim sua produtividade e a fluência de seus códigos.

Com o Spring, pode-se construir aplicações de alto desempenho, utilizando os diversos módulos que o integram de acordo com as necessidades do projeto. Podemos destacar dentro desse ecossistema os módulos Spring Web, Spring MVC, Spring Security, Spring Data JPA, Spring Cloud, entre outros. E para facilitar ainda mais a criação das aplicações, foi desenvolvido o módulo Spring Boot, o qual permite que desenvolvedor se preocupe menos com as configurações para executar o projeto, realizando a abstração de toda a sua complexidade.

A tecnologia Spring é muito utilizada para desenvolver aplicações com estilo arquitetural REST, por meio de seu módulo web, que concentra os recursos necessários para criação e exposição de serviços remotos. Dessa forma, é uma ferramenta de extrema importância para desenvolver esse projeto.

Figura 5.1 – Framework Spring Boot



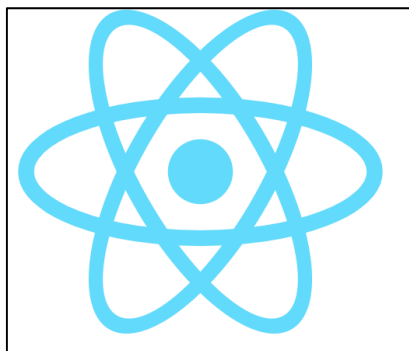
Fonte: <https://spring.io/>

5.2 React JS

O React JS é uma biblioteca JavaScript open-source, mantida pelo Facebook e utilizada no desenvolvimento de interfaces de usuário em aplicações web. Dispõe de um conjunto de módulos poderosos que trabalham com o conceito de componentes e seus estados. Com essa biblioteca é possível atualizar e renderizar de forma eficiente apenas os componentes necessários, na medida em que os dados mudam (alteração de estado).

Atualmente, o React JS é uma das bibliotecas javascript mais utilizadas no mundo, trazendo performance, eficiência, flexibilidade, entre outros benefícios, além de possibilitar a criação de interações complexas. É uma ferramenta chave para o desenvolvimento deste projeto.

Figura 5.2 – Biblioteca React JS



Fonte: <https://pt-br.reactjs.org/>

5.3 JWT

O Json Web Token é um padrão da indústria, implementado por diversas linguagens de programação, para representar reivindicações com segurança entre duas

partes, ou seja, é um mecanismo utilizado para realizar autenticação no sistema, por meio de um token. Utilizando o JWT, um cliente recebe do servidor um token válido ao autenticar-se na aplicação, e esse token deve ser enviado em todas as requisições enviadas por este cliente a fim de validá-lo, para posterior liberação de acesso. É um padrão confiável e já consolidado no mercado, e por isso, usado como um dos mecanismos de segurança do sistema desenvolvido neste projeto.

Figura 5.3 – Json Web Token

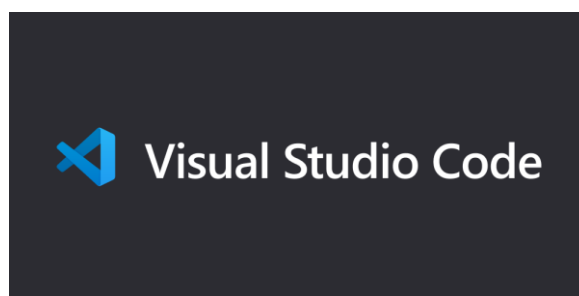


Fonte: <https://jwt.io/>

5.4 Visual Studio Code

O Visual Studio Code é um editor de código open-source, desenvolvido pela Microsoft para os sistemas operacionais Windows, Linux e macOS. Com esse editor é possível desenvolver em diversas linguagens, contando com recursos de depuração, realce de sintaxe, complementação inteligente de código, snippets e refatoração de código, e ainda possui o software Git incorporado, o qual foi utilizado para realizar o controle de versão do sistema. O VSCode foi escolhido como ferramenta principal para escrita do código-fonte.

Figura 5.4 – Ferramenta VSCode



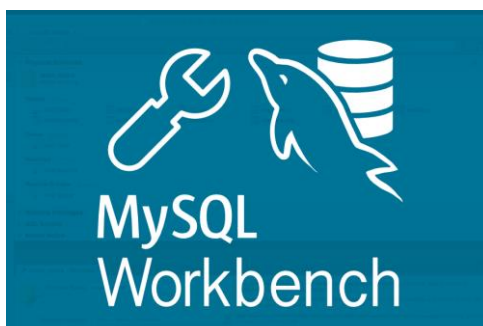
Fonte: <https://code.visualstudio.com/>

5.5 MySQL WorkBench

O MySQL Workbench é uma ferramenta visual unificada, desenvolvida para o sistema gerenciador de banco de dados MySQL, e destinada aos sistemas operacionais

Windows, Linux e Mac OS X, permitindo aos DBAs, desenvolvedores ou arquitetos de dados projetar, modelar, gerar e gerenciar bancos de dados visualmente, além de possuir ferramentas de administração abrangentes para configuração de servidor. Inclui muitos recursos úteis, entre os quais, a possibilidade de criação de modelos ER complexos e o recurso de engenharia direta e reversa, por meio dos quais é possível criar bases de dados a partir do ER e vice-verso. Dessa forma, é uma ferramenta de extrema utilidade para modelagem e criação do banco de dados deste projeto.

Figura 5.5 – Ferramenta MySQL Workbench



Fonte: <https://medium.com/>

5.6 Astah Community

Astah Community é um software utilizado para modelagem de sistemas utilizando a UML (em português, Linguagem de Modelagem Unificada), permitindo representá-los de forma padronizada através da construção de diagramas dinâmicos. É uma ferramenta que possui muitos recursos, sua utilização é relativamente fácil e já está bastante consolidada no mercado, especificamente na área de Engenharia de Software, e por isso, utilizada neste projeto como ferramenta fundamental ao seu desenvolvimento.

Figura 5.6 – Ferramenta Astah Community



Fonte: <https://astah.net>

5.7 Modelo Arquitetural

Uma das principais etapas do processo de desenvolvimento de software é a escolha de seu modelo arquitetural, também chamado de padrão ou estilo arquitetural, sendo este importante para a construção de um sistema compreensível e logicamente

estruturado. Definir um modelo é pensar no melhor caminho para estruturar um sistema de acordo com seu domínio de aplicação. Dessa forma, dentre os modelos arquiteturais mais utilizados para a criação de soluções em softwares, definiu-se a utilização do modelo REST para este projeto.

Por ser baseado nos fundamentos de funcionamento da web, que utiliza o recurso de chamadas de serviços remotos, esse padrão possibilita a escalabilidade e independência entre os módulos do sistema, até mesmo a flexibilização do uso de diferentes linguagens de programação.

5.7.1 Padrão REST

O REST, apresentado pela primeira vez em 2000 por Roy Fielding na Universidade da Califórnia, Irvine, em sua dissertação acadêmica, tem como pilares o uso de serviços web, requisições HTTP e o conceito de recurso. Os recursos são abstrações que nos permitem mapear por meio de URLs um determinado tipo de informação que uma aplicação gerencia para depois acessá-las e/ou manipulá-las.

É um padrão arquitetural simplista, poderoso e suficientemente maduro, sendo capaz de atender requisitos complexos de integração sem que ocorra agregação de alta complexibilidade e alto acoplamento.

Atualmente, no estilo arquitetural REST, esses recursos são disponibilizados nos chamados Web Services ou APIs REST e podem ser consumidos por aplicações clientes através de requisições HTTPs, fazendo o uso das URLs por meio das quais foram mapeados. Além disso, é necessário especificar os verbos HTTPs utilizados nas requisições, que serão explicados logo mais. Como retorno a essas requisições, temos cargas úteis, geralmente formatadas em HTML, XML ou JSON.

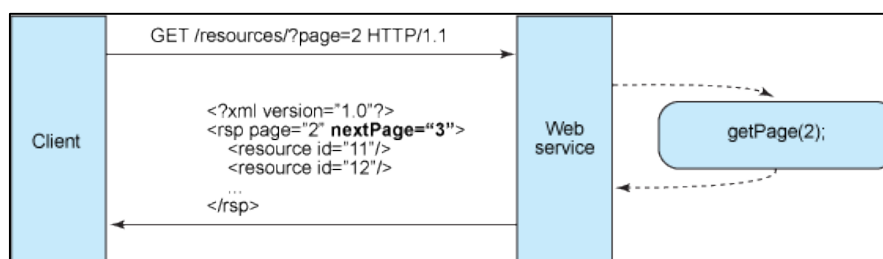
Para a comunicação com serviços baseados em REST, são utilizadas requisições HTTPs, especificando os verbos utilizados em cada uma delas, os quais representam uma convenção geral para a manipulação de serviços. Cada verbo HTTP tem sua forma distinta de manipular um serviço, normalmente criando, visualizando, atualizando e excluindo recursos da aplicação. Estão listados a seguir os mais utilizados.

- POST: utilizado para criação de um novo recurso;
- GET: utilizado para obtenção de um ou mais recursos;
- PUT: utilizado para alteração do conteúdo ou estado de um recurso;

- DELETE: remoção/exclusão de um recurso.

Na figura 5.1, temos um esquema representativo da arquitetura REST, abordando de um lado, o cliente, o qual realiza as requisições, e do outro, o Web Service, que disponibiliza o serviço.

Figura 5.7 – Representação esquemática da arquitetura REST



Fonte: <https://developer.ibm.com/articles/ws-restful/>

6 SISTEMA DE EGRESSOS

O Sistema de Egressos UFPA – Campus Castanhal é uma ferramenta para fomentar a interação entre Egressos, Instituição e Empresas. O mesmo manterá uma base de dados com informações pertinentes aos egressos da instituição e suas interações com o mercado de trabalho, podendo as empresas parceiras incluírem oportunidades de emprego/estágio que serão disponibilizadas a esses egressos.

Utilizando o sistema, o usuário Egresso poderá visualizar e se cadastrar nessas oportunidades, manter seus dados acadêmicos e profissionais atualizados e ainda preencher questionários disponibilizados pelos coordenadores. O sistema mostrará às empresas o perfil de cada egresso que se cadastrar em suas oportunidades, para que possam avaliar a eventual contratação do profissional e enviar um feedback, o qual será essencial para a instituição, informando sobre eventuais contratações de egressos, ou os motivos para que estes não tenham sido contratados nas oportunidades disponibilizadas.

A geração de dashboards com dados estatísticos dos egressos e ainda a emissão de relatórios são umas das funcionalidades mais importantes do sistema, propiciando à instituição recursos para análise de seu sistema de ensino e tomada de decisões. Com isso, as coordenações poderão propor melhorias para seus cursos, visando a adequação dos

mesmos aos requisitos do mercado, melhorando assim a capacidade de formação de profissionais aptos ao desempenho de suas funções dentro das organizações.

6.1 Levantamento de Requisitos

Essa é fase que dá início ao processo de desenvolvimento de softwares, pois é nela que se obtém o entendimento do que efetivamente deve ser feito, a partir da análise do domínio da aplicação. O analista deve se preocupar em entender de forma clara as necessidades apresentadas pelo cliente, buscando sempre se comunicar da melhor forma possível com mesmo. “O principal objetivo do levantamento de requisitos é que usuários e desenvolvedores tenham a mesma visão do problema a ser resolvido”. (Bezerra, 2015, p. 31).

Para obtenção de resultados satisfatório com relação à qualidade do produto final (software), faz-se necessário um estudo detalhado do problema a ser solucionado, identificando todos os requisitos que deverão ser atendidos para pelo software desenvolvido. Segundo Bezerra (2015), o requisito é entendido como uma condição ou capacidade que o sistema desenvolvido ou seu componente precisa alcançar ou possuir, a fim de satisfazer um contrato, padrão, especificação ou outros documentos formalmente impostos. É uma ação (requisitos funcionais) que pode ser realizada na utilização do sistema, ou propriedade (requisitos não-funcionais) do mesmo.

6.1.1 Requisitos funcionais e não funcionais

Na etapa de levantamento dos requisitos do sistema foi utilizada a técnica de questionário, enviado para alguns membros da instituição, onde, a partir das respostas obtidas, foi possível a elaboração desses requisitos. A tabela a seguir, contém a relação dos requisitos funcionais e não funcionais levantados para o desenvolvimento deste projeto.

Tabela 6.1- Requisitos funcionais e não funcionais do sistema

(Requisito)	Descrição do Requisito	Prioridade
RF001	O sistema deve oferecer opção de cadastro de egressos.	Alta
RF002	O sistema deve permitir a atualização dos dados dos Egressos.	Média
RF003	O sistema deve permitir que os egressos visualizem as oportunidades de emprego/estágio	Alta

	cadastradas pelas empresas e validadas pelo coordenador.	
RF004	O sistema deve permitir que o Egresso indique sua participação do processo de seleção da empresa para preenchimento das vagas ofertadas.	Alta
RF005	O sistema deve permite que o Egresso responda aos questionários gerados pelos coordenadores e pelo administrador.	Média
RF006	O sistema deve permitir o cadastro de empresas	Alta
RF007	O sistema deve permitir que as empresas atualizem suas informações quando necessário, assim como desativem seu cadastro no sistema.	Média
RF008	O sistema deve permitir que as empresas insiram, alterem, ou finalizem oportunidades de emprego/estágio no sistema, as quais serão disponibilizadas aos egressos.	Alta
RF009	O sistema deve permitir que a empresa gere um feedback sobre as contratações ou não contratações de egressos da instituição	Média
RF10	O sistema deve permitir que coordenador atualize os dados do curso(s) ao qual está vinculado.	Média
RF11	O sistema deve permitir que o coordenador poste dicas de emprego/estágio que serão mostradas aos Egressos.	Baixa
RF12	O sistema deve permitir que um coordenador valide as oportunidades cadastradas pelas empresas no sistema para serem mostradas aos Egressos.	Alta
RF13	O sistema deve permitir a geração de questionários pelo coordenador e administrador.	Alta
RF14	O sistema deve permitir ao coordenador ou administrador, a geração de relatórios personalizados dos dados contidos no sistema, inclusive com gráficos.	Alta
RF15	O sistema deve permitir ao administrador a validação do cadastro de empresas.	Alta
RF16	O sistema deve permitir que administrador inclua, altere ou exclua cursos no sistema.	Alta
RNF01	O sistema terá uma interface amigável ao usuário primário sem se tornar cansativa aos usuários mais experientes.	Alta
RNF02	O sistema deve eferecer bom desempenho, considerado por corresponder a um fator de qualidade de software.	Média
RNF03	Deve ser adotada como linguagem principal de desenvolvimento, o Java, seguindo	Média

	cuidadosamente as técnicas de orientação a objetos.	
--	---	--

Fonte: O autor

6.2 Usuários do sistema

Administrador: Ator com privilégios irrestritos no sistema, podendo gerenciar os dados referentes aos cursos da instituição, coordenadores, egressos, validação das empresas cadastradas, geração de relatórios, questionários e dashboards, entre outras funcionalidades no sistema.

Coordenador: O ator coordenador é um usuário fundamental do sistema, pois é o mesmo que analisará as informações dos egressos para assim realizar ajustes, mudanças, e outras ações no (s) curso (s) que coordena. Funções no sistema: validação das vagas disponibilizadas pelas empresas parceiras, geração de relatórios, gerar questionários, entre outras.

Empresa: O ator empresa é peça chave para disponibilizar no sistema oportunidades aos egressos da instituição, tendo depois a função de enviar feedbacks à mesma, com as contratações realizadas ou motivos pelos quais não houve as contratações. Com isso, essas informações poderão ser utilizadas para a construção de relatórios e auxiliarão nas decisões da instituição de ensino.

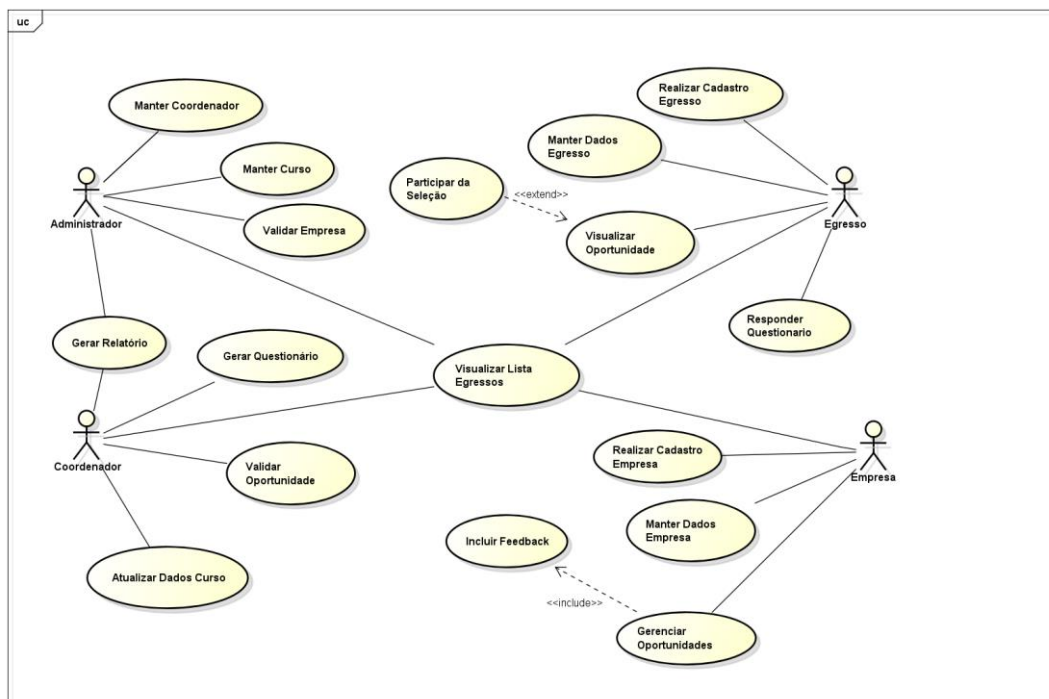
Egresso: O egresso, por uma vez, é o usuário mais importante do sistema, o qual tem a função de manter seus dados pessoais, acadêmicos e profissionais atualizados. Poderá analisar as oportunidades de emprego/estágio disponibilizadas pelas empresas cadastradas e participar da seleção. Além disso, ele poderá preencher os questionários disponibilizados pelo coordenador do seu curso para aumentar a eficiência do acompanhamento de sua trajetória.

6.3 Diagrama de Casos de Uso

Os diagramas de Casos de Uso são artefatos de grande utilidade na engenharia de software, ajudando muito no entendimento de um sistema. Segundo Bezerra (2015), o modelo de caso de uso está associado aos requisitos funcionais do sistema, através dos quais um observador consegue obter o entendimento das funcionalidades fornecidas pelo sistema e quais são seus resultados produzidos.

Na figura 6.1, temos o diagrama de Casos de Uso deste projeto, considerando todos os atores descritos anteriormente e suas interações com o sistema.

Figura 6.1-Diagrama geral de Casos de Uso do sistema



Fonte: O autor

Pode-se notar que existem casos de usos incluídos (include) e estendidos (extends), que indicam, respectivamente, casos de uso que devem ser executados após os casos de uso que o incluem, e casos de uso que podem ou não ser executados após os casos de uso aos quais estendem. A diante serão detalhados os casos de uso apresentados neste diagrama.

6.3.1 Casos de Uso específicos do Ator Egresso

O ator egresso é peça chave na aplicação. Por meio do caso de uso “Realizar Cadastro Egresso” o mesmo poderá realizar seu cadastro no sistema, sendo para isso efetuada sua validação como egresso da instituição. Depois disso, ele poderá realizar inclusões e atualizações de dados pessoais, acadêmicos e profissionais no sistema, por meio do caso de uso “Manter Dados Egresso”.

Será disponibilizada ao egresso a visualização de oportunidades de emprego/estágio incluídas por empresas parceiras, onde ele poderá estar indicando no sistema seu interesse em participar da seleção a essas oportunidades. Seu perfil será

mostrado à empresa que disponibilizou uma oportunidade na qual indicou interesse. Os casos de uso para essas funcionalidades no sistema são “Visualizar Oportunidades” e “Participar da seleção”.

Por fim, por meio do caso de uso “Responder Questionário”, o egresso poderá responder a questionários gerados e disponibilizados no sistema pelo coordenador de seu curso de formação ou pelo administrador.

6.3.2 Casos de Uso específicos do Ator Empresa

O ator empresa, por meio do caso de “Realizar Cadastro Empresa”, poderá se cadastrar no sistema informando um CNPJ válido e seus dados. Esse cadastro será validado posteriormente para liberar o acesso da empresa ao sistema. Através do caso uso “Manter Dados Empresa” o usuário poderá atualizar seus dados ou desativar seu cadastro no sistema.

O caso de uso “Gerenciar Oportunidade” permite que a empresa inclua, atualize ou exclua uma oportunidade de emprego/estágio no sistema, especificando os cursos requisitados para esta, a qual será disponibilizada aos Egressos. Após a exclusão de uma oportunidade, por meio do caso de uso “Incluir FeedBack”, ela deverá incluir um feedback sobre eventual contratação de egresso da instituição. Caso não tenha contratado nenhum egresso, incluirá um dos motivos pré-cadastrados no sistema.

6.3.3 Casos de Uso específicos do Ator Coordenador

O ator coordenador, por meio do caso de uso “Atualizar Dados Curso” tem a função de manter os dados do curso, ou cursos aos quais está vinculado. As informações desses cursos podem ser vistas no perfil dos egressos da instituição. Cabe também ao coordenador, por meio do caso de “Validar Oportunidades”, analisar os detalhes das oportunidades inseridas pelas empresas, destinadas ao (s) curso (s) que coordena, e decidir se está de acordo com o campo de atuação desses cursos, validando-as ou não no sistema, para assim, aparecerem aos egressos.

6.3.4 Casos de Usos específicos do Ator Administrador

Dentre os papéis mais importantes desse ator no sistema, tem-se a função de incluir, alterar e excluir os coordenadores, bem como os cursos da instituição, por meio dos casos de uso “Manter Coordenador” e “Manter Curso”, respectivamente. Além disso, ele é responsável por validar o cadastro de empresas, por meio do caso de uso “Validar

Empresa”, para que estas possam utilizar as funcionalidades desse usuário no sistema.

6.3.5 Casos de Usos comuns ao Administrador e Coordenador

Temos dois casos de usos comuns aos atores Administrador e Coordenador, são estes os casos de uso “Gerar Relatório” e “Gera questionário”. No primeiro, os atores podem gerar relatórios estatísticos sobre os egressos da instituição, podendo filtrar por meio de parâmetros especificados, por exemplo, “por curso, “tempo de formação”, “idade”, entre outros. No segundo, os atores podem criar questionários no próprio sistema e disparar para os egressos responderem.

6.3.6 Casos de Usos comuns a todos

Por fim, temos um caso de uso comum a todos os atores, sendo este um dos mais importantes do sistema, por meio do qual é possível visualizar o perfil dos egressos da instituição (caso de uso “Visualizar Lista Egressos”).

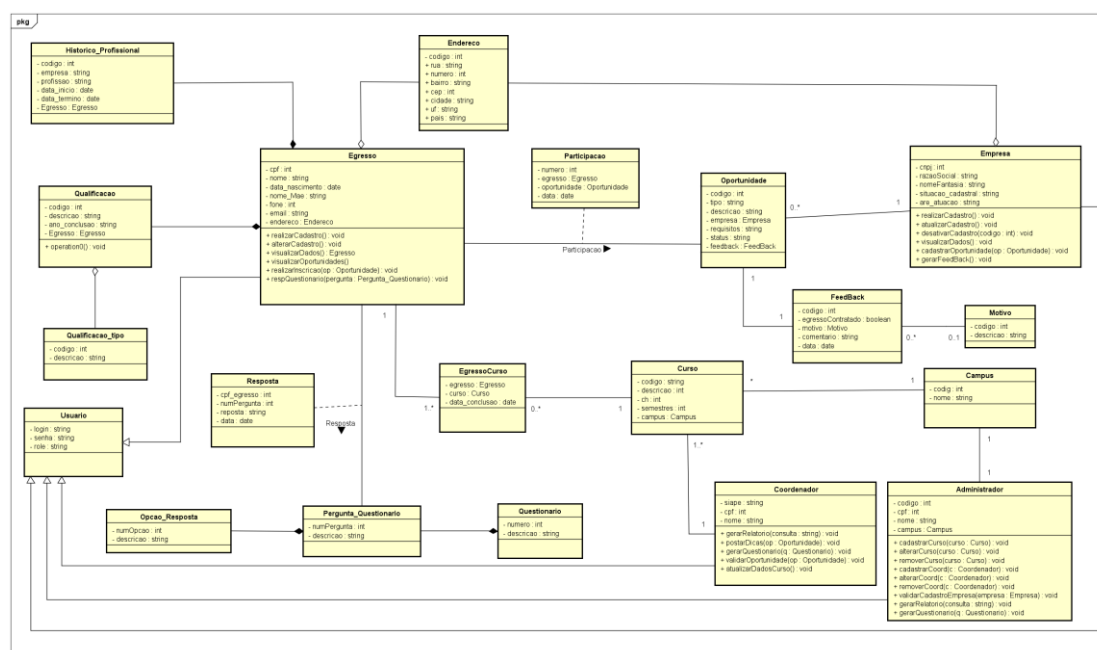
6.4 Diagrama de Classes

O Diagrama de Classes é um artefato da engenharia de software que representa uma abstração do mundo real, por meio do qual podemos descrever a estrutura e relação entre objetos. Cada classe descreve os objetos por meio de seus atributos e operações, que correspondem, respectivamente, às informações que eles armazenam e o conjunto de ações que sabem realizar. Segundo Fowler (2000), um diagrama de classes realiza a descrição dos tipos de objetos presentes no sistema e os vários tipos de relacionamentos estáticos entre eles existentes.

Classe é a representação de um conjunto de coisas reais ou abstratas que são reconhecidas como sendo do mesmo tipo por compartilhar as mesmas características de atributos, operações, relações e semântica. (FURLAN, 1998).

Na imagem a seguir, temos o diagrama de classes deste projeto, representando os objetos que integram o sistema, com seus atributos, métodos (ações) e relacionamentos.

Figura 6.2 - Diagrama de Classes do sistema



Fonte: O autor

Na construção deste diagrama, tem-se uma relação de generalização/especialização das classes “Egresso”, “Coordenador”, “Empresa” e “Administrador” com a classe “Usuario”, a qual representa todos esses usuários do sistema de forma genérica. Existem também classes que surgiram por meio da relação de associação entre outras, que são chamadas de classes associativas, sendo estas as classes “Participacao” e “Resposta”, representando, respectivamente, a participação de um egresso no processo de seleção à uma oportunidade disponibilizada por uma empresa, e a resposta de um egresso à uma pergunta de um questionário criado no sistema.

Uma relação a ser destacada neste diagrama é a qual um ou mais objetos compõem um outro e dependem dele para existir. Logo, as classes desses objetos têm uma relação de composição entre as que compõem e a que é composta. É o caso da classe “Egresso”, que tem como composição as classes “Historico_Profissional” e “Qualificacao”, e a classe “Questionario”, tendo como composição as classes “Pergunta_Questionario” e “Opcao_Resposta”.

Por fim, destaca-se a relação onde um objeto-parte é atributo de um objeto-todo, fazendo sentido mesmo sem ele existir. Essa relação é chamada de agregação regular. É o caso das classes “Egresso” e “Empresa”, que possuem a classe “Endereco” como agregação.

6.5 Modelo Entidade Relacionamento

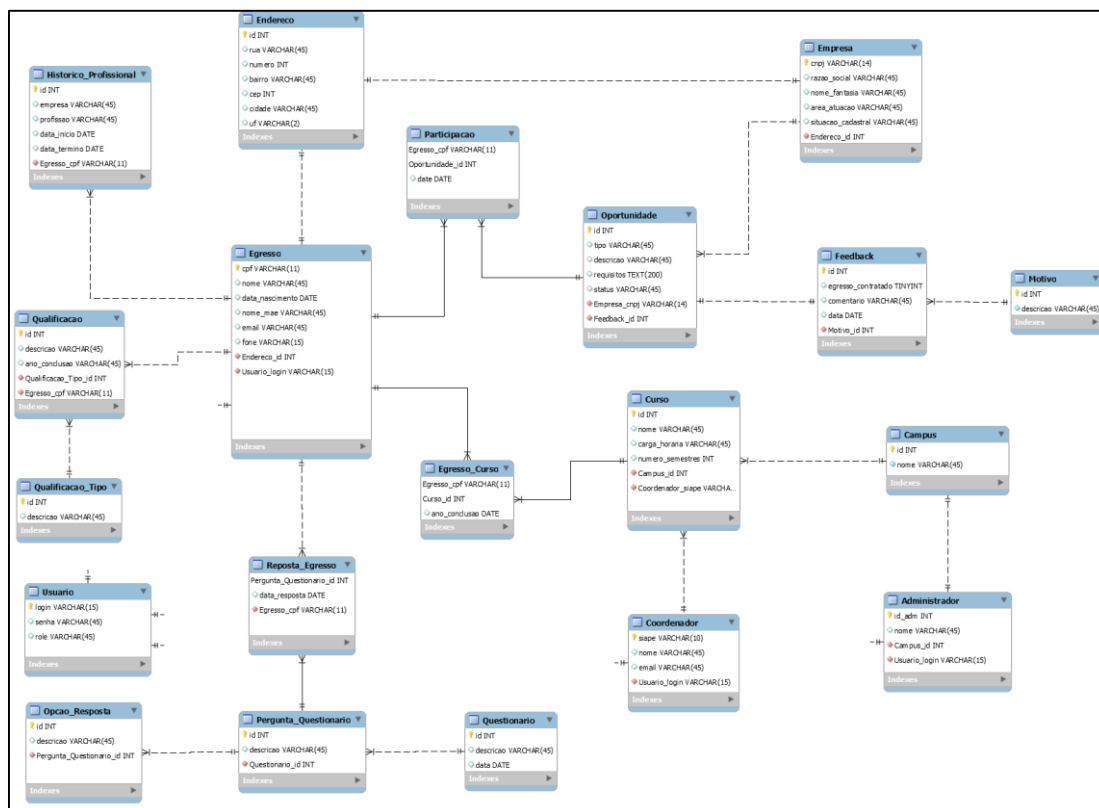
O Modelo ER é responsável por conduzir a modelagem de dados de um sistema, descrevendo dados ou aspectos de informação do seu domínio de negócio ou seus requisitos de processo. Essa modelagem é amplamente utilizada como meio de conhecer os problemas organizacionais e assim poder projetar as soluções.

6.5.1 Diagrama ER

Para descrever as entidade e relacionamentos entre elas num modelo de dados, utiliza-se o Diagrama Entidade Relacionamento, que é a ferramenta gráfica principal desse modelo. Ela é a base para a implementação de bancos de dados relacionais, onde as entidades e seus atributos serão convertidos em tabelas e campos dentro do banco de dados, e os relacionamentos serão realizados por meio de chaves que identificam essas tabelas.

A figura 7.3 mostra o Diagrama ER deste projeto, que é o espelho da base de dados final construída para o sistema de egressos proposto.

Figura 6.3 - Diagrama ER do sistema



Fonte: O autor

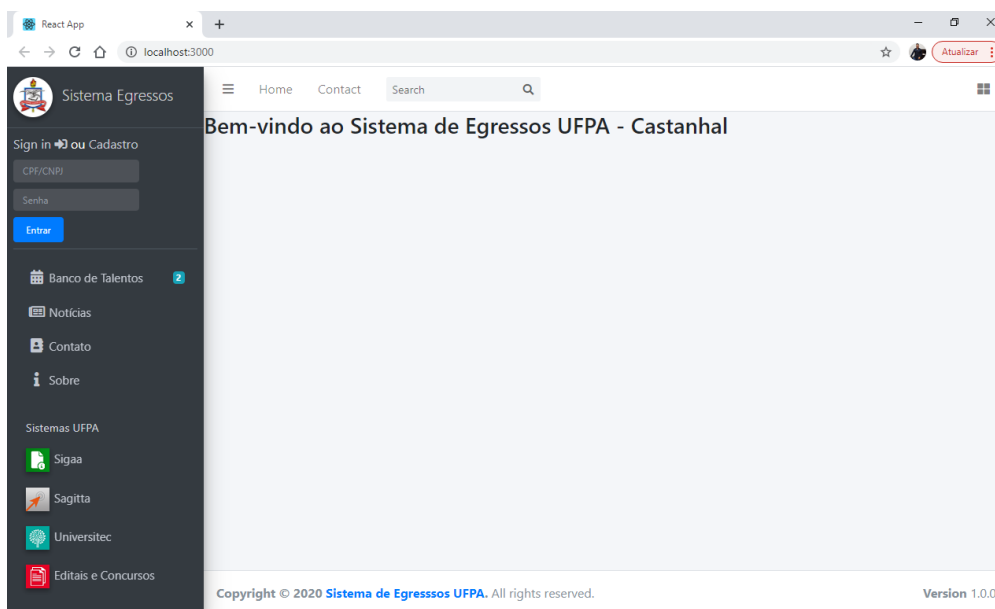
7 RESULTADOS

Para a apresentação do sistema desenvolvido, foram consideradas as principais telas utilizadas na interação com seus usuários.

7.1 Tela Principal

A figura 7.1 apresenta a tela principal do sistema desenvolvido. Nesta, o usuário poderá visualizar um menu vertical que dá acesso ao banco de talentos (egressos que se destacam), notícias, contatos e informações sobre o sistema. Também é possível acessar outros sistemas da UFPA disponíveis no menu. Por fim, o usuário poderá se cadastrar ou entrar no sistema, se caso já tenha o feito.

Figura 7.1 – Tela Principal do Sistema



Fonte: O autor

7.2 Tela de Cadastro de Egressos

A figura 7.2 apresenta a tela de cadastro de egressos, precisando para tal cadastro ser validado como egresso da instituição.

Figura 7.2 – Tela de Cadastro de Egressos

Fonte: O autor

7.3 Tela de Perfil do Egresso

A figura 7.3 apresenta a tela de perfil do egresso, mostrando sua imagem, seus dados pessoais, qualificações, experiências profissionais e seu curso de formação na instituição. É possível alterar alguns dados pessoais, e incluir, alterar ou excluir qualificações e experiências profissionais. Além disso, pode-se acessar seu perfil no Linked-in e, em uma versão futura, ver como foi avaliado pelas empresas parceiras.

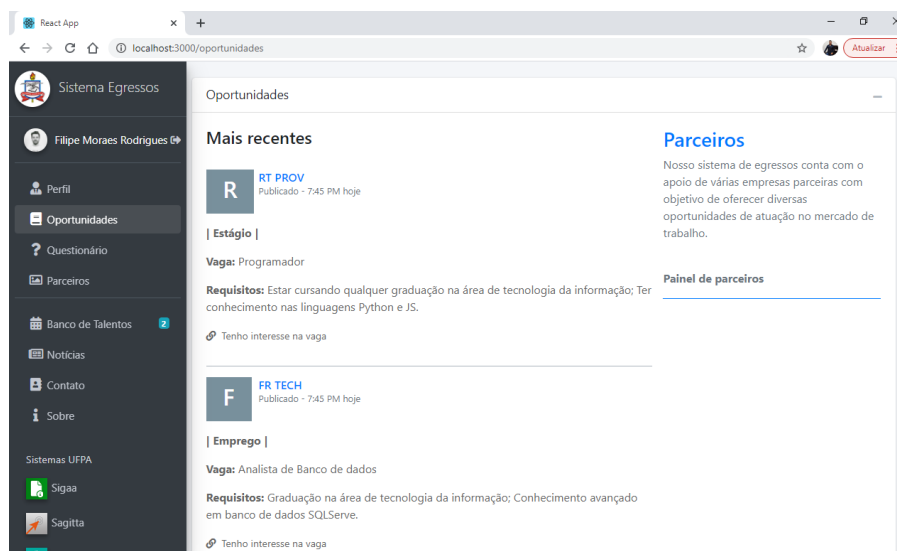
Figura 7.3 – Tela de Perfil do Egresso

Fonte: O autor

7.4 Tela de Visualização de Oportunidades

A figura 7.4 apresenta a tela de oportunidade visualizada pelos egressos, mostrando as oportunidades de emprego e estágio disponibilizadas pelas empresas parceiras. Nesta tela, o egresso poderá indicar o interesse nas vagas e o seu perfil será enviado à empresa para participação na seleção.

Figura 7.4 – Tela de Visualização de Oportunidades



Fonte: O autor

7.5 Tela de Cadastro de Empresas

A figura 7.5 apresenta a tela de cadastro de empresas, precisando para tal cadastro informar um CNPJ válido.

Figura 7.5 – Tela de cadastro de empresas parceiras

Fonte: O autor

7.6 Tela Gerenciar Oportunidades

A figura 7.6 apresenta a tela para as empresas incluírem, atualizarem, excluïrem suas oportunidades, podendo enviar um feedback à instituição no momento da exclusão das mesmas.

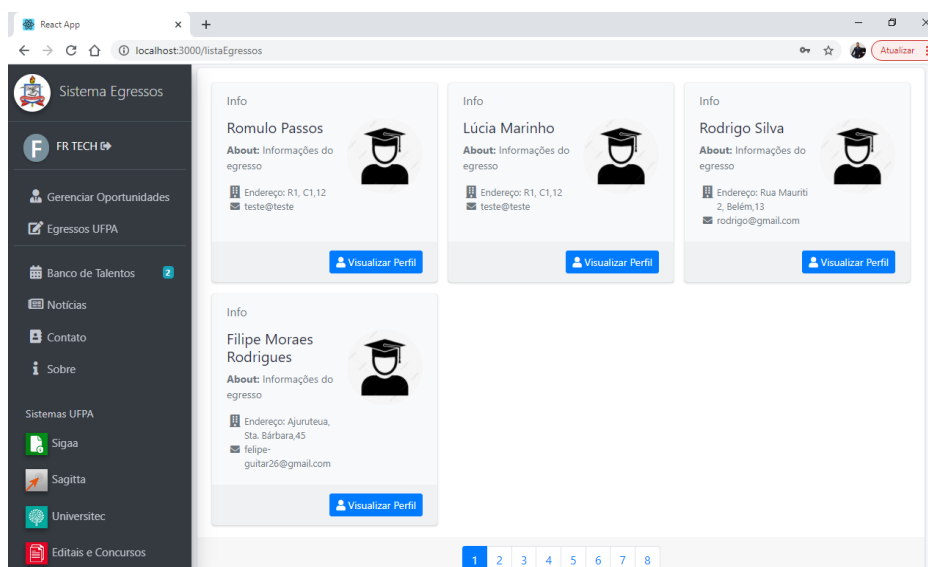
Figura 7.6 – Tela Gerenciar Oportunidades

Fonte: O autor

7.7 Tela de Visualização dos Egressos

A figura 7.7 apresenta a tela para a visualização das informações sobre os egressos da instituição.

Figura 7.7 – Tela de Visualização dos Egressos da Instituição

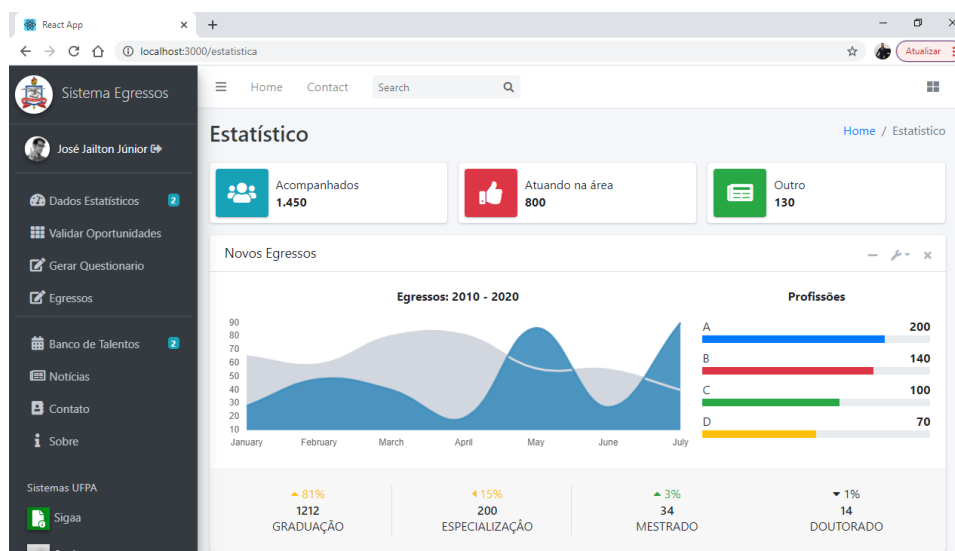


Fonte: O autor

7.8 Tela de Dados Estatísticos

A figura 7.8 apresenta uma das telas mais importantes do sistema. Ela é destinada aos coordenadores dos cursos da instituição e auxiliará os mesmos na tomada de decisões a respeito dos cursos e identificação do perfil de seus egressos, por meio de recursos como dashboards com os dados estatísticos, entre outros. A análise e atualização constante desses dados é imprescindível para a obtenção do resultado esperado do sistema.

Figura 7.8 – Tela de Dados Estatísticos dos Egressos



Fonte: O autor

8 CONCLUSÃO

Neste projeto foi possível ampliar a percepção da figura do egresso como fonte de informações imprescindíveis à gestão das instituições de ensino superior, tendo como base a realidade observada na UFPA, campus Castanhal. A implantação do sistema desenvolvido tende a favorecer a aproximação do egresso com a instituição e mercado de trabalho, assim como o motiva e o impulsiona à busca pelo contínuo aperfeiçoamento pessoal e profissional. Isso reflete direta e indiretamente no modo como ambos (egresso e instituição) são vistos pelo meio externo. De modo geral, este sistema dá subsídios para que a instituição permaneça numa escala ascendente de crescimento no que diz respeito ao seu ensino e planejamento estratégico. Entretanto, não é um caminho fácil, sendo necessário empenho e compromisso por parte de todos que direta ou indiretamente desempenham um papel na instituição.

8.1 Trabalhos Futuros

As propostas futuras a seguir, visam para este projeto a constante melhoria do sistema desenvolvido, por meio da integração de novas funcionalidades e ampliando de sua forma de acesso, buscando aumentar o vínculo com os egressos, e assim, alcançar melhores resultados ao mesmos e à gestão da instituição.

- Desenvolvimento da aplicação mobile.
- Implementação do módulo de avaliação individual dos egressos.
- Implantação em todos os campi da UFPA.

REFERÊNCIAS

- BEZERRA, Eduardo. **“Princípios de Análise e Projeto de Sistemas com UML”**. 3ª ed. Rio de Janeiro: Elsevier, 2015.
- CORREIA, Carlos Henrique; TAFNER, Malcon Anderson. **Análise Orientada a Objetos**. Florianópolis: Visual Books, 2006.
- COSTA, Bruno Martins; FONTANA, Cayo M. da Cruz. **“Sistema de Egressos”**. Monografia (Bacharelado em Ciencia da Computação) – Centro Universitário de Vila Velha. Vila Velha, 2008.
- FOWLER, Martin; SCOTT, Kendal. **“UML Essencial”**. Porto Alegre: Bookman, 2000.
- FALBO, R. A. **“Engenharia de Software”**. Nota de aula. Disponível em: http://www.inf.ufes.br/~jssalamon/wp-content/uploads/disciplinas/engsoft/Notas_Aula_Engenharia_Software_Falbo_2014.pdf. Acesso em: 18 de fevereiro de 2021.
- FURLAN, José Davi. **“Modelagem de Objetos através da UML: Análise e Desenho Orientados a Objetos”**. São Paulo: Makron Books, 1998.
- GOLÇALVES, R. *et al.* **“Uma proposta de processo de produção de aplicações Web”**. Revista Produção. São Paulo, setembro, 2005. Disponível em: https://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-65132005000300008&lng=pt&tlng=pt. Acesso em: 15 de fevereiro de 2021.
- LOUSADA, Ana Cristina Zenha; MARTINS, Gilberto de Andadre. **“Egressos como fonte de informação à gestão dos cursos de Ciências Contábeis”**. Revista Contabilidade & Finanças. São Paulo, abril, 2005. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1519-70772005000100006. Acesso em: 28 de janeiro de 2021.
- LOUDON, Kyle. **“Desenvolvimento de grandes aplicações web”**. 1ª ed. São Paulo: Novatec, 2010.
- MARTINS, Gilberto de Andrade. **“As atividades de marketing nas instituições de ensino superior”**. 1986. 200p. Tese (Doutorado) – Faculdade de Economia, Administração e Contabilidade, Universidade de São Paulo, São Paulo, 1986.
- MEHEDFF, Nassim Gabriel. **A avaliação da educação e a inserção dos egressos do ensino médio no mercado de trabalho**. Brasília: Instituto Nacional de Estudos e Pesquisas Educacionais, 1999.
- NASCIMENTO, Bruno Manzoli. **“Sistema de Acompanhamento de Egressos”**. Monografia (Bacharelado em Ciencia da Computação) – Departamento de Informática da Universidade Federal do Espírito Santo. Vitória, 2016.
- PRESSMAN, Roger S. **“Software Engineering: A Practioner’s Approach”**. 7ª ed. New York: McGraw-Hill, 2010.

PRESSMAN, Roger S. “**Software Engineering: A Practitioner’s Approach**”. 8^a ed. New York: McGraw-Hill, 2016.

SOMMERVILLE, Ian. “**Engenharia de Software**”. 9^a ed. São Paulo: Pearson Prentice Hall, 2011.