

SIGESTAGIOS: SISTEMA WEB PARA CONTROLE DE ESTÁGIOS SUPERVISIONADOS NO ENSINO SUPERIOR

Isaac Bruno Siqueira de Souza
isaacwinfor@gmail.com
Universidade Estadual do Piauí - UESPI
Francisco das Chagas Rocha
rocha@phb.uespi.br
Universidade Estadual do Piauí - UESPI
Rodrigo Augusto Rocha Souza Baluz
rodrigo.baluz@phb.uespi.br
Universidade Estadual do Piauí - UESPI

Resumo

O estágio supervisionado é um elemento obrigatório a muitos cursos de graduação servindo como forma de conectar o estudante do ensino superior ao mercado de trabalho. Nele o aluno pode aplicar os conhecimentos adquiridos em sala de aula no exercício prático de sua profissão. O curso de bacharelado em Ciência da Computação da Universidade Estadual do Piauí (UESPI) – Campus Parnaíba, assim como vários cursos de graduação no Brasil, não dispõe de um sistema informatizado para gerenciar os estudantes durante o tempo em prática de estágio. Este trabalho tem como objetivo desenvolver um sistema web que irá sistematizar e automatizar os processos que envolvem os atores, alunos, professores, coordenação de curso e empresas relacionados à atividade acadêmica do estágio supervisionado. O SIGESTAGIOS ficou disponível para testes por parte de atores selecionados do curso em estudo, durante um período de 30 dias, sendo em seguida, avaliado quanto sua usabilidade, alcançando um bom índice de aceitação.

Palavras-chave: Ensino Superior. Estágio Supervisionado. Sistema Web. Informatização de Processos.

Abstract

The supervised internship is a required element of many undergraduate courses serving as a way to connect the higher education student to the job market. In it the student can apply the knowledge acquired in the classroom in the practical exercise of his profession. The Bachelor of Science degree in Computer Science at the State University of Piauí (UESPI) - Campus Parnaíba, as well as several undergraduate courses in Brazil, does not have a computerized system to manage students during their internship time. This work aims to develop a web system that will systematize and automate the processes involving the actors, students, teachers, course coordination and companies related to the academic activity of the supervised internship. SIGESTAGIOS was available for testing by selected actors of the course under study for a period of 30 days, then evaluated for usability, achieving a good acceptance rate.

Keywords: Higher Education. Supervised Internship. Web System. Process Computerization.

1 Introdução

A Tecnologia da Informação (TI) vem auxiliando cada vez mais na tomada de decisões e proporcionando agilidade no andamento de processos da administração pública, fazendo com que os gestores possam ter maior controle sobre suas atribuições (FERREIRA, 2016). Dentre as diferentes áreas que a TI vêm trazendo numerosos impactos, cita-se a educação (OLIVEIRA; MOURA, 2016).

Nesse contexto, o curso de Ciência da Computação da Universidade Estadual do Piauí (UESPI) – Campus Parnaíba, sujeito de estudo deste trabalho, vem buscando melhorar as atividades educacionais com o auxílio da TI, servindo como modelo experimental para possível expansão aos demais *campi* da instituição.

Atualmente, o curso em estudo não dispõe de um sistema informatizado para controle de estágios supervisionados. Ou seja, todo o processo é feito de forma manual pelos atores envolvidos. Assim, tendo como base entrevistas realizadas com a coordenação de curso e de posse de documentos administrativos do estágio curricular, foi possível fazer uma abordagem mais criteriosa na definição dos requisitos funcionais para o desenvolvimento do sistema.

Tomando como base os argumentos mencionados, o principal objetivo deste artigo é sistematizar por meio de um *software* em plataforma *web* o processo de estágios supervisionados obrigatórios do curso de Bacharelado em Ciência da Computação da UESPI – Campus Parnaíba. Este sistema permitirá organizar e armazenar de forma centralizada os documentos referentes ao estágio, colaborando para um melhor controle e garantia da integridade dos dados, evitando assim, sobrecarregar os atores envolvidos com preenchimento de dados duplicados e manuseio de papel, além de disponibilizar uma biblioteca dinâmica de documentos e relatórios.

Para o desenvolvimento deste projeto foi criada uma aplicação *web*, aqui denominada SIGESTAGIOS, utilizando as ferramentas Django para *back-end* e Vue.js para o *front-end*. O *front-end*, que será escrito na forma de uma *Single Page Application* (SPA), acessará o *back-end* por meio de requisições utilizando o protocolo *Hypertext Transfer Protocol* (HTTP) assíncronas via *Asynchronous JavaScript and XML* (AJAX). Os dados serão disponibilizados e enviados de formulários para o *back-end* via uma *Application Programming Interface* (API) de arquitetura de *software Representational State Transfer* (REST), e persistidos em um banco de dados PostgreSQL. Haverá comunicação com serviços de terceiros, em particular com o serviço de dados do Instituto Brasileiro de Geografia e Estatística (IBGE), o que fornece uma API de localidades que será utilizada pela aplicação para exibir estados e cidades em campos

de formulários de dados. Testes foram realizados com alunos do último período do curso de Bacharelado em Ciência da Computação e professores da instituição, além de supervisores das instituições concedentes de estágio. Por fim, foi aplicado um questionário para avaliação do nível de usabilidade do sistema.

Este trabalho está estruturado em seis seções, sendo a primeira esta introdução. A segunda relaciona os trabalhos encontrados na literatura que usam de sistemas informatizados para melhorar a qualidade do serviço em cursos no ensino superior. A terceira aborda a metodologia bem como as ferramentas utilizadas para o desenvolvimento do protótipo. Na quarta seção são exibidas as principais telas do sistema e funcionalidades. Na quinta seção são apresentados os resultados alcançados e suas análises. Por fim, são feitas as considerações finais e apresentação dos possíveis trabalhos futuros.

2 Trabalhos Relacionados

No trabalho de (SILVA, A.; CONCEIÇÃO; SILVA, L., 2016), é proposto o desenvolvimento de um sistema de controle de Atividades Complementares (ACs) para melhorar as condições de gerenciamento de ACs, tanto para os estudantes quanto para o coordenador que avalia as atividades de todos os alunos. Os autores selecionaram e estudaram tecnologias tais como: Ruby On Rails como *framework* de aplicações *web*, MySQL como banco de dados e Git como ferramenta para controle de versão. E como plataforma de desenvolvimento foi utilizado o sistema operacional Linux. Foram efetuadas reuniões durante o período letivo do curso de Bacharelado em Sistemas de Informação do Instituto Federal Fluminense para levantar requisitos do sistema e criar tarefas para um melhor entendimento sobre os módulos que compõem o mesmo. Segundo os autores, a maioria dos usuários percebeu uma melhoria no processo em relação ao método anterior.

Segundo Almeida (2018), as metodologias atuais para controle do número de faltas na Universidade Federal de Ouro Preto (UFOP), não permitem a divulgação por parte dos docentes de maneira apropriada e quando um grande número de discentes são matriculados em determinada turma, o processo se torna custoso. Ainda, o autor propõe o desenvolvimento de um sistema *web* para informatizar e agilizar o processo, além de melhorar a metodologia de controle de assiduidade da UFOP. Durante o desenvolvimento utilizou técnicas de engenharia de software para auxílio em todo o processo. O autor desenvolveu o sistema utilizando a linguagem PHP e o *framework* Laravel em conjunto com o banco de dados MySQL. Segundo ele, o sistema atende o processo de controle de faltas da UFOP e é capaz de substituir as metodologias utilizadas atualmente.

Leoncio (2018), por meio de pesquisas realizadas, constatou a ausência de um sistema para controle de Estágio no Departamento Acadêmico de Informática da Universidade Tecnológica do Paraná, Campus Ponta Grossa, e que o armazenamento das informações é realizado com planilhas eletrônicas, tornando insegura e ineficiente a manipulação dos dados. O trabalho, apresenta o desenvolvimento de um sistema *web*, para controlar o histórico dos estágios obrigatórios e não obrigatórios, baseado em entrevistas com os usuários. Para o desenvolvimento da aplicação foi feita a modelagem do sistema através de diagramas da UML, a fim de abstrair as informações do mundo real. Segundo o autor, através da tecnologia empregada foi possível melhorar o desempenho das operações e a segurança dos dados.

SILVA, M. (2017), afirma que existe uma dificuldade pela maioria dos alunos da Faculdade de Computação da Universidade Federal de Uberlândia na fase de conclusão do curso por não encontrarem um tema adequado, e um orientador para o desenvolvimento do Trabalho de Conclusão de Curso (TCC). Diante disso, propôs o desenvolvimento de um sistema *web*, dedicado ao gerenciamento de TCCs, visando melhorar a comunicação entre alunos e professores, para gerar menos frustrações aos discentes e conflitos de informações. O autor frisa que, devido a alta usabilidade do sistema criado, os usuários não terão mais dificuldades na utilização das funcionalidades.

3 Materiais e Métodos

O objetivo deste trabalho é automatizar o processo de estágio supervisionado obrigatório da UESPI-Parnaíba com o desenvolvimento de um sistema para ambiente *web*, a fim de fornecer aos envolvidos uma solução que controle de forma segura todos os recursos, tais como: controle de instituições conveniadas, orientadores, supervisores, alunos etc. Há também a preocupação em facilitar a geração dos documentos referentes ao processo de maneira que com apenas um clique o usuário possa tê-los disponíveis.

3.1 Ferramentas de desenvolvimento

A escolha destas ferramentas se deu por critérios como produtividade; baixa curva de aprendizagem; documentação atualizada, bem escrita e de fácil e livre acesso; comunidades muito ativas e licença de uso. Em seguida serão listadas as principais ferramentas e/ou tecnologias utilizadas no desenvolvimento do sistema.

- *Back-End*
 - *Framework Django*
 - *Django REST Framework*

- SGBD PostgreSQL
- *Front-End*
 - *Framework* Vue.js
 - *Framework* Vuetify

Todas estas ferramentas citadas são gratuitas para uso pessoal e comercial, ou seja, podem ser utilizadas livremente sem ou quase nenhuma restrição. As subseções seguintes referem-se às definições e características de cada uma delas.

3.1.1 *Framework* Django

Django é um *framework* para desenvolvimento *web* de alto nível, escrito na linguagem de programação Python, que incentiva o rápido desenvolvimento de aplicações com *design* limpo e prático. Criado por desenvolvedores experientes, para que se possa escrever aplicações *web* sem necessidade de “reinventar a roda”. É uma ferramenta gratuita e *open-source* (DJANGO, 2005). Para o desenvolvimento do *software* deste trabalho, foi utilizada a versão 1.11 do *framework*.

3.1.2 Django REST Framework

O Django REST Framework é uma biblioteca Python que permite a criação de APIs REST da *web* de forma simples e flexível em projetos Django, que proporciona produtividade (BRENDON, 2016). Este trabalho fez uso da versão 3.9 do Django REST Framework.

3.1.3 SGBD PostgreSQL

O PostgreSQL é um poderoso Sistema Gerenciador de Banco de Dados (SGBD) objeto-relacional *open-source*, que usa a linguagem SQL para manipulação dos dados (POSTGRESQL, 2019). Tem como função principal, armazenar dados de forma segura, se apoiando a boas práticas. Pode ser utilizado tanto em aplicações de pequeno porte quanto em aplicações de grande porte voltadas para a *internet*, onde vários usuários utilizarão de forma simultânea (DIONISIO, 2015). Este trabalho fez uso da versão 11.0 do PostgreSQL.

3.1.4 *Framework* Vue.js

O Vue.js é um *framework* progressivo para a construção de interfaces de usuário, escrito em JavaScript. Ele faz parte da camada visual (*view layer*) do modelo de arquitetura MVC, sendo fácil de integrar com outras bibliotecas existentes. Ele é distribuído sob a licença MIT e também pode ser usado na criação de sofisticadas *Single Page Applications* (VUE, 2014), como no caso deste trabalho, onde foi usada a versão 2.5 do *framework*.

3.1.5 Framework Vuetify

Vuetify é um *framework* para criar belas aplicações com interface responsiva, escrito em Vue.js, possui uma boa quantidade de componentes reutilizáveis, uma documentação de fácil acesso, disponibilizado sob a licença MIT e é baseado no *Material Design* (VUETIFY, 2016). O *Material Design* é uma metodologia de *design* com foco nas melhores práticas para projetos de interface de usuário, desenvolvido pela Google (RALLO, 2017). Neste trabalho foi utilizada a versão 1.4 do Vuetify.

3.2 Local de Estudo

O Campus Professor Alexandre Alves de Oliveira da Universidade Estadual do Piauí (UESPI) com sede em Parnaíba-PI foi criado em 1991. Atualmente oferta os seguintes cursos de graduação: Bacharelado em Direito; Bacharelado em Ciência da Computação; Bacharelado em Enfermagem; Bacharelado em Odontologia; Bacharelado em Engenharia Agrônômica; Licenciatura em Letras-Ingês; Licenciatura em Ciências Biológicas; Licenciatura em História; Licenciatura em Letras-Português; Licenciatura em Pedagogia; Licenciatura em Ciências Sociais e Licenciatura em Filosofia. A instituição possui como principais objetivos ser referência em ensino superior e oferecer um serviço educacional de qualidade para toda a sociedade, e não apenas parnaibana.

3.3 Requisitos do Sistema

Para chegar à definição dos requisitos necessários para o desenvolvimento da proposta foram realizadas reuniões junto à coordenação do curso de Ciência da Computação da UESPI-Parnaíba, assim como a análise de documentos referentes ao estágio supervisionado. Diante disso, a listagem abaixo exhibe resumidamente os 10 principais requisitos que caracterizam o cenário aqui proposto.

1. O coordenador de curso insere no sistema os alunos aptos a fazerem estágio.
2. Os alunos recebem um *e-mail* e a partir daí entram no sistema e preenchem a ficha de inscrição do estágio.
3. O coordenador de curso pode imprimir um relatório chamado “Mapa de Estágio”.
4. Na ficha de inscrição o aluno pode, caso necessário, cadastrar uma instituição que não possui convênio com a UESPI.
5. A coordenação imprime o termo de compromisso e, solicita se necessário, a confirmação de contrato de convênio entre as instituições.

6. O aluno e seu orientador devem preencher o “Plano de Estágio”, que deverá ser validado pelo supervisor de estágio da empresa/instituição.
7. O aluno deverá a cada dia de estágio fazer o registro de sua permanência na empresa, que deve ser validado pelo orientador e supervisor da empresa/instituição.
8. A cada mês de estágio o aluno preenche a “Ficha de Acompanhamento”, que também deve ser validado pelo orientador e supervisor.
9. Ao final, o aluno deve preencher o “Relatório Final” de estágio, que deve ser validado pelos mesmos envolvidos.
10. O orientador e supervisor, ao final do ciclo, devem preencher a “Ficha de Desempenho”, avaliando o aluno estagiário.

3.4 Modelos do Sistema

Após o processo de levantamento de requisitos junto a coordenação do curso de Ciência da Computação da UESPI em Parnaíba, foram criados diagramas para a representação das funcionalidades requeridas na gestão de estágios supervisionados, utilizando para tal a Linguagem de Modelagem Unificada (*Unified Modeling Language* - UML). A UML é uma linguagem para a modelagem de sistemas de computador, que oferece uma forma gráfica para a visualização dos requisitos do sistema, baseada no paradigma da orientação a objetos (BOOCH; RUMBAUGH; JACOBSON, 2005; FREITAS, 2008).

A fase de modelagem do sistema se deu primeiramente por meio da criação do diagrama de casos de uso. Os casos de uso demonstram uma visão estática das principais funcionalidades e interações dessas funcionalidades com os usuários do sistema sem a necessidade de detalhes técnicos de como o sistema faz (BOOCH; RUMBAUGH; JACOBSON, 2005; RIBEIRO, 2012).

A Figura 1 apresenta as funcionalidades que serão fornecidas dentro do sistema, representadas por casos de uso e os respectivos usuários que poderão executá-las, representados por atores. Neste diagrama estão listados apenas os casos de uso com as principais funções do sistema.

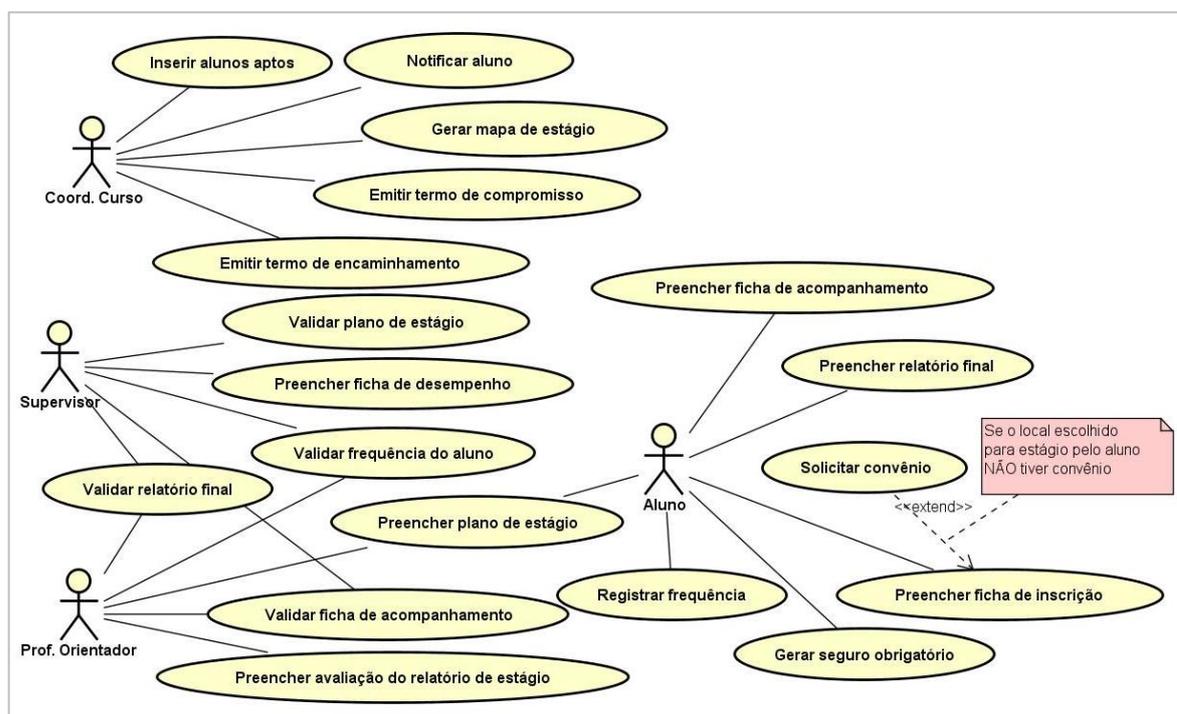
É importante mencionar que antes de tudo, todos os usuários precisarão fornecer suas credenciais de acesso (*e-mail* e senha) para executar as ações dentro do sistema. Neste diagrama não foram representadas as funcionalidades de *login* e *logout*, assim como as operações básicas de CRUD no banco de dados.

A partir do diagrama da Figura 1, pode-se observar diferentes usuários do sistema: Coordenador de Curso, Supervisor de Estágio, Professor Orientador e Aluno. No caso de uso

“Inserir alunos aptos” o coordenador irá cadastrar os alunos aptos a fazerem estágio e poderá imediatamente notificá-los através do caso de uso “Notificar aluno”. No caso de uso “Preencher ficha de inscrição” o aluno poderá, caso o local escolhido para estágio não possua convênio com a instituição, executar o caso de uso “Solicitar convênio”.

Antes do sistema, o aluno deveria preencher uma ficha de seguro obrigatório. Atualmente, com apenas um clique, será executado o caso de uso “Gerar seguro obrigatório”, onde o sistema irá fornecer um documento no formato PDF gerado dinamicamente, recuperando os dados que o aluno preencheu na ficha de inscrição.

Figura 1 – Diagrama de Casos de Uso simplificado.



Fonte: O próprio autor.

Dos casos de uso acima foi gerado o diagrama de sequência (Figura 2). Os diagramas de sequência dão uma visão dinâmica de um sistema e é uma espécie de diagrama de interação, que exhibe a troca de mensagens entre um conjunto de objetos, destacando uma ordenação temporal dessas mensagens (BOOCH; RUMBAUGH; JACOBSON, 2005). O diagrama (Figura 2) está nivelado em dois, o *Front-end*, que é por onde o usuário interage com o sistema, e o *Back-end (REST)*, representando aqui o sistema que fará interações com o *Front-end*.

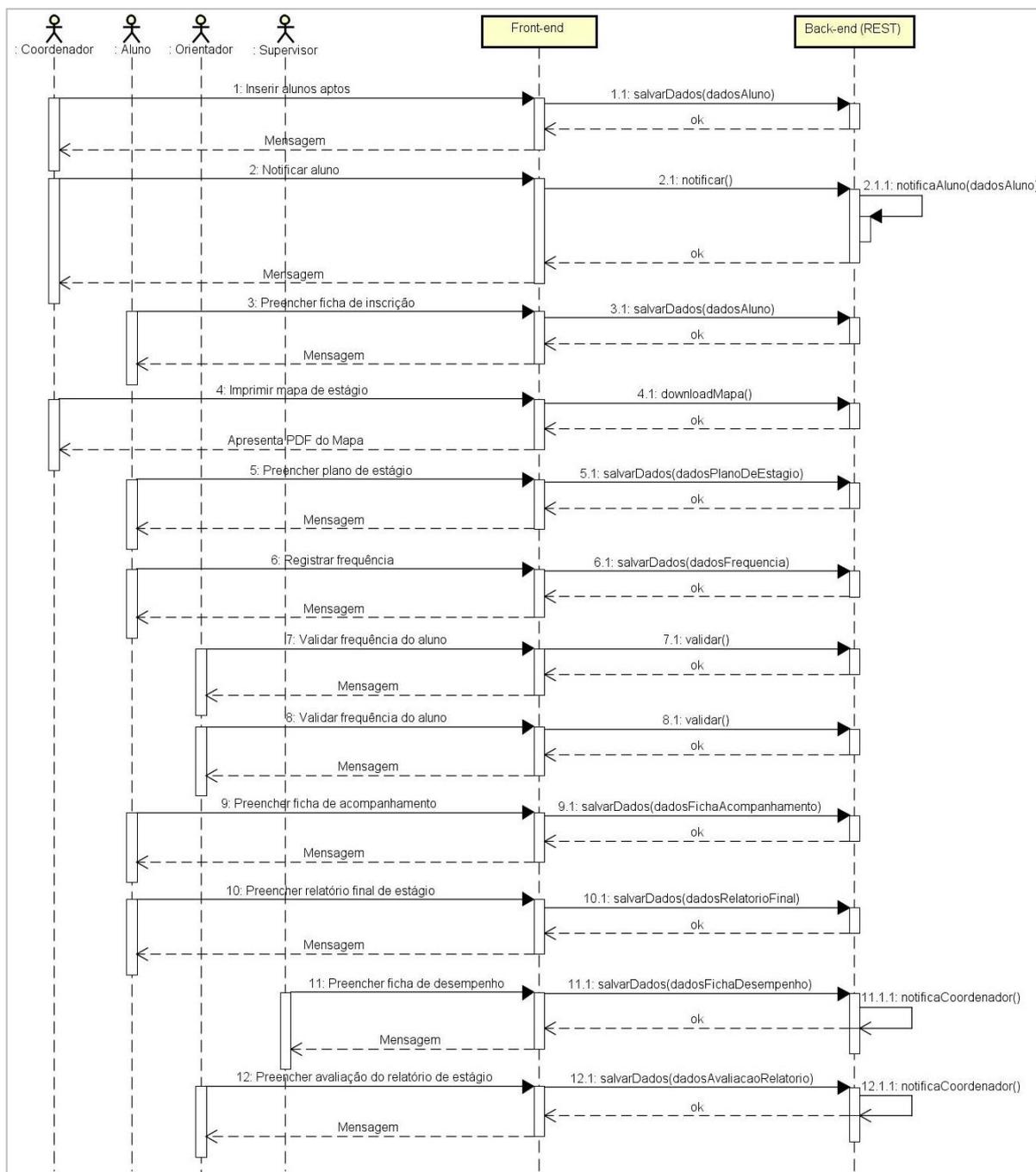
O primeiro passo se dá através do coordenador de curso, que é inserir os alunos aptos a fazerem estágio, que após a inserção deve notificá-los para que eles possam concluir o cadastro. Logo após ser notificado, o aluno pode realizar a conclusão da inscrição. Após todos os alunos terem efetuado suas respectivas inscrições no sistema, o coordenador de curso poderá gerar o

relatório “Mapa de Estágio”, um documento PDF com o nome, CPF, local de estágio e o período do estágio dos alunos, fornecendo para tal o período de realização dos estágios.

Para o preenchimento do plano de estágio, ficha de acompanhamento, relatório final e registro de frequência, o aluno não precisa necessariamente seguir uma ordem, mas após realizar tais tarefas ele deverá notificar os usuários interessados clicando no botão “NOTIFICAR INTERESSADOS”.

O usuário supervisor de estágio irá preencher a ficha de desempenho e submeter avaliando o estagiário. O usuário professor orientador também avaliará o aluno preenchendo e submetendo a avaliação de desempenho do relatório final. Em ambos os casos, após a avaliação ser submetida, o sistema deve notificar os interessados automaticamente por envio de *e-mail*, nesse caso o coordenador de curso, para que ele possa prosseguir no sistema.

Figura 2 – Diagrama de Sequência simplificado.



Fonte: O próprio autor.

Nos diagramas supracitados, não foi mencionado o que diz respeito aos documentos que são anexados no sistema. A cada documento que for anexado, a aplicação irá automaticamente notificar os devidos usuários para que estes possam prosseguir para a próxima etapa, garantindo uma melhor sequência no fluxo do estágio.

Após a definição dos diagramas UML foi criado o modelo lógico do banco de dados. O modelo lógico é uma representação de um banco de dados em um nível de abstração compreendido pelo usuário do SGBD (HEUSER, 2009).

Para testar a plataforma SIGESTAGIOS, foram convidados 13 voluntários, sendo estes com perfis de atores que se relacionam com o estágio supervisionado no ensino superior: 1 coordenador de curso, 7 alunos de graduação, 2 professores orientadores e 3 supervisores ligados às empresas concedentes do estágio. O sistema foi implantado no dia 04 de junho de 2019, data também de início dos testes. O usuário do tipo coordenador de curso foi cadastrado no sistema. Este usuário é o principal administrador do sistema e quem dá início ao processo para que os outros possam ter acesso. Os convidados tiveram um breve treinamento e foram cadastrados no sistema. Também foi disponibilizado um documento para cada usuário, descrevendo o passo a passo de execução das tarefas dentro do sistema. O período de testes terminou no dia 04 de julho de 2019.

3.6 Escala de Usabilidade do Sistema

A Escala de Usabilidade do Sistema - *System Usability Scale* (SUS) é um método simples, porém de fácil avaliação do problema de usabilidade de um sistema, criado em 1986 por John Brooke. Ele pode ser utilizado para avaliar qualquer tipo de interface, produtos, serviços, hardware, software, websites, aplicações (TEIXEIRA, 2015; SAURO, 2011).

Para avaliação da usabilidade do SIGESTAGIOS, foi disponibilizado um questionário com as 10 primeiras perguntas apoiadas no SUS, adaptadas de Sauro (2011) por Teixeira (2015) acrescentado de outras 2 questões, que permitem avaliar o nível funcional de sua aplicação aos cursos de graduação no ensino superior. A seguir, serão listadas as 12 perguntas básicas do questionário que foi aplicado em um formulário *online*, após a finalização dos testes dos usuários.

1. Eu acho que gostaria de usar esse sistema com frequência.
2. Eu acho o sistema desnecessariamente complexo.
3. Eu achei o sistema fácil de usar.
4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
5. Eu acho que as várias funções do sistema estão muito bem integradas.
6. Eu acho que o sistema apresenta muita inconsistência.
7. Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
8. Eu achei o sistema atrapalhado de usar.
9. Eu me senti confiante ao usar o sistema.
10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.

11. Eu usaria o sistema caso fosse implantado no curso.
12. Eu acredito que o sistema pode contribuir com a gestão dos estágios supervisionados em cursos de graduação.

Cada questão assume um formato de resposta (SAURO, 2011), como demonstrado na Figura 4. Onde a escala, fundamentada na escala *Likert*, varia de 1 a 5, com 1 significando “Discordo Completamente” e 5 significando “Concordo Completamente” (TEIXEIRA, 2015). Likert (1932) desenvolveu esta escala para medir até que ponto as pessoas concordam ou discordam de uma série de afirmações sobre um tópico. Aos entrevistados podem ser apresentadas cinco a sete ou até nove opções de respostas pré-codificadas, sendo uma como ponto neutro (MCLEOD, 2008).

Figura 4: Formato de resposta de acordo com a metodologia do SUS.

Eu acho que gostaria de usar esse sistema com frequência.*

1 2 3 4 5

Discordo Completamente Concordo Completamente

Fonte: O próprio autor.

O principal objetivo deste questionário foi a coleta de dados da usabilidade do sistema no contexto do estágio supervisionado do curso de Computação da UESPI-Parnaíba. Ele foi criado e disponibilizado *online* através do aplicativo Formulários Google. Na aquisição das respostas não foi solicitado nenhum tipo de identificação dos voluntários.

4 Análise e Discussão dos Resultados

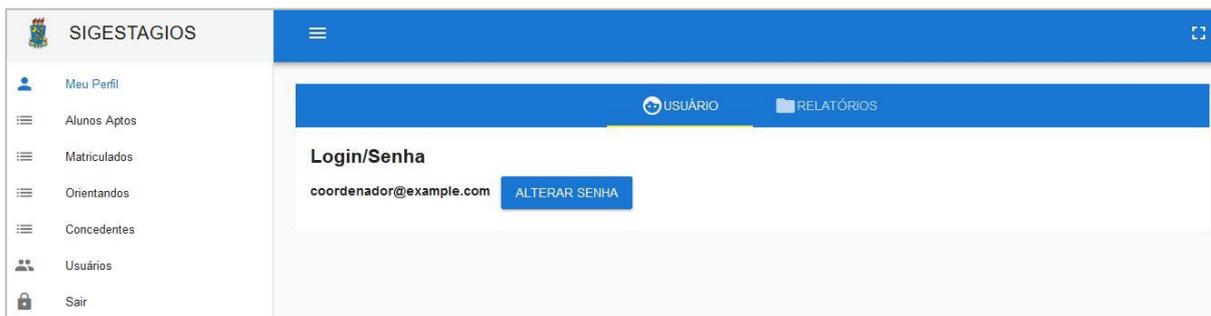
Nesta seção serão apresentadas as principais telas do sistema com suas funcionalidades e, por conseguinte, os resultados dos testes efetuados por meio do questionário SUS. É importante mencionar que os dados apresentados nas figuras de telas do sistema são somente ilustrativos.

4.1 A plataforma SIGESTAGIOS

Na página de perfil (Figura 5), primeira tela do sistema a ser exibida ao usuário após ter sido identificado na tela de *login*, pode-se observar ao lado esquerdo da tela o menu de funcionalidades que, dependerá da função (Aluno, Coordenador de curso, Orientador ou Supervisor) do usuário que fez o *login*. Nesta página, também existirá um painel com uma ou mais abas, que também dependerão do tipo de usuário.

Na Figura 5, pode ser observado o painel de perfil sob a ótica de um coordenador de curso com as opções de alterar senha e gera relatórios específicos deste usuário. As opções do menu para este usuário são: “Meu Perfil”, “Alunos Aptos”, “Matriculados”, “Orientandos”, “Concedentes”, “Usuários” e “Sair”. Clicando em “Alunos Aptos” será exibida a tela de listagem de todos os alunos aptos a fazerem estágio.

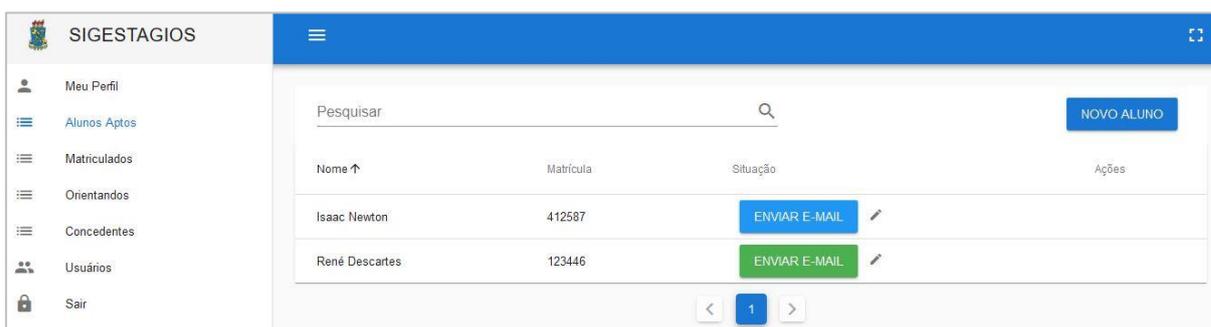
Figura 5 – Tela de perfil de usuário coordenador de curso.



Fonte: O próprio autor.

Na tela de alunos aptos (Figura 6), o coordenador de curso terá as seguintes opções: “ENVIAR E-MAIL” para notificar o aluno que ele deverá preencher suas informações de estágio; “NOVO ALUNO” para adicionar aluno apto; um campo de texto intitulado “Pesquisar” que poderá ser usado para buscar alunos ou pelo nome ou pela matrícula; e um botão com um ícone de lápis que permitirá a alteração dos dados de aluno apto. Além dos botões de ordenação no cabeçalho da tabela e dos botões de paginação vistos abaixo dela.

Figura 6 – Tela de listagem de alunos aptos.

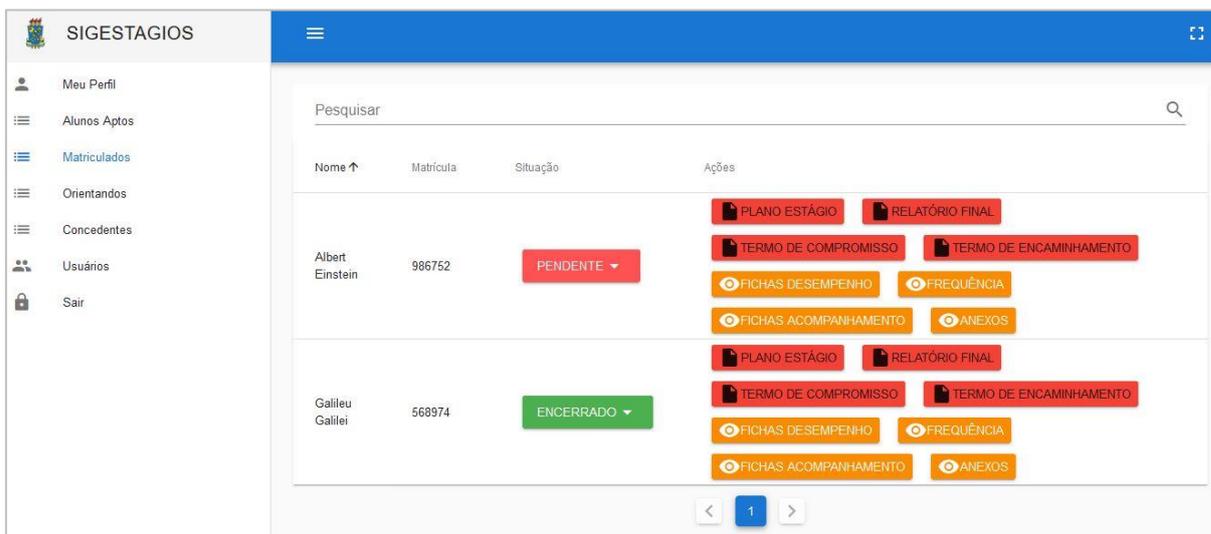


Fonte: O próprio autor.

O coordenador de curso também pode ser orientador. Na fase de implementação caso o usuário logado fosse do tipo coordenador de curso, optou-se por dividir a listagem dos estudantes em duas páginas distintas. A página dos matriculados, que exibirá todos os alunos que já completaram o cadastro de seus estágios e a dos orientandos, que exibirá os alunos que escolheram o coordenador de curso inclusive como orientador. A Figura 7 mostra a página de alunos matriculados, que apresenta as mesmas opções de busca, ordenação e paginação já

mencionados para os alunos aptos, além de algumas outras opções de ações. Os botões da cor vermelha significam que serão gerados documentos PDFs de forma dinâmica com os dados dos respectivos estagiários. Os botões de cor laranja significam acompanhamento por parte do coordenador, nesse caso.

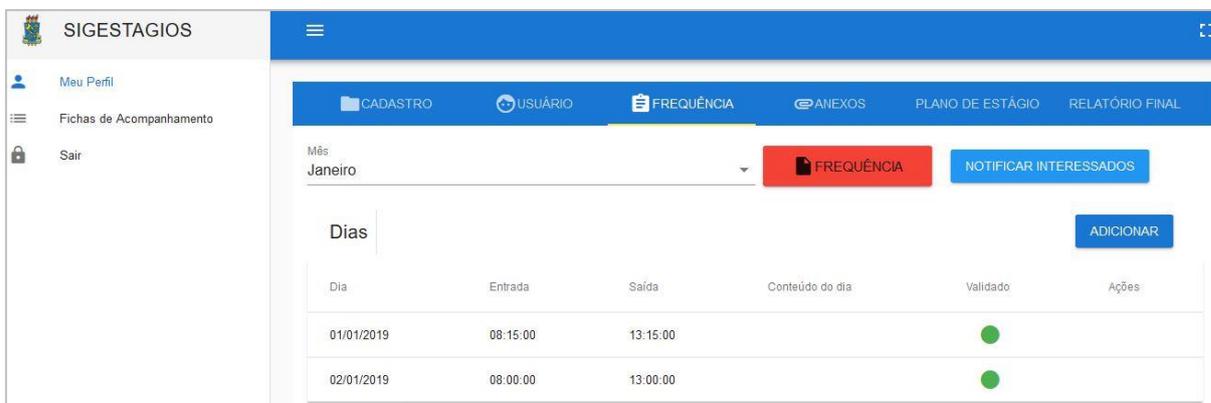
Figura 7 – Tela de listagem de alunos matriculados.



Fonte: O próprio autor.

Na página de perfil do estagiário, mais especificamente na aba de frequência (Figura 8), o aluno estagiário poderá adicionar os dias da frequência para determinado mês. Ao final ele terá a opção de notificar os atores interessados (professor orientador e supervisor de estágio), clicando no botão “NOTIFICAR INTERESSADOS”. Clicando no botão “FREQUÊNCIA”, será gerado um documento PDF com os dados do modelo referentes ao mês selecionado.

Figura 8 – Tela de frequência do aluno estagiário.



Fonte: O próprio autor.

Na página de listagem de supervisionados, assim como na página de listagem de orientandos, terá o botão nomeado “DESEMPENHO”. Nesta página, ao clicar no botão será

exibida tanto ao supervisor de estágio quanto ao orientador uma janela com um formulário para preenchimento da ficha de avaliação de desempenho (no caso do orientador, a ficha de avaliação de desempenho do relatório final) do respectivo estagiário, como é apresentado no exemplo da Figura 9.

Figura 9 – Tela com formulário de avaliação de desempenho do estagiário.

Item	Nota	Comentário
ASSIDUIDADE (1,0 ponto) Normas de horário e permanência durante o expediente de trabalho.	0,8	
DEDICAÇÃO (1,0 ponto) Zelo e interesse pelos trabalhos de sua responsabilidade.	1	
INICIATIVA (1,5 ponto) Capacidade de encaminhamento de determinadas situações e discernimento.	1,1	
CRIATIVIDADE (1,0 ponto) Capacidade de criação, rapidez e habilidades de execução das tarefas.	0,6	

Fonte: O próprio autor.

4.2 Avaliação de Usabilidade

O questionário SUS foi uma maneira quantitativa de medir a satisfação dos usuários na realização das tarefas desempenhadas pelo sistema. Para a obtenção das respostas do questionário foi utilizada a ferramenta Formulários Google, utilizada para recuperar as respostas dos usuários, juntamente com a Planilhas Google, utilizada para manipular os cálculos e obter o *score*.

Pode-se observar através da Tabela 1 as respostas obtidas no questionário. Foram 13 voluntários como mencionado na seção 3. No cabeçalho da tabela nota-se que na primeira coluna, para cada linha representa-se um voluntário. A expressão “Q1” na segunda coluna significa “Questão 1”, e assim sucessivamente para as demais colunas. As questões Q11 e Q12 não foram utilizadas para fins de cálculos do *score* final. Servem para indicar o nível de aceitação referente à gestão administrativa dos estágios no curso em estudo, além de medir a importância do uso por demais cursos de nível superior. Somente 1 dos 13 voluntários que responderam as questões citadas anteriormente, logo após os testes, se mostrou neutro em relação aos outros participantes da pesquisa.

Tabela 1 - Respostas obtidas no questionário.

Voluntário	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
1	5	1	5	2	5	2	5	1	5	1	5	5
2	4	2	5	1	4	3	4	2	4	1	5	5
3	4	1	5	1	4	2	5	2	5	1	5	5
4	5	2	3	2	4	2	5	1	4	1	5	5
5	5	1	5	1	5	1	4	1	5	1	5	5
6	5	3	4	2	3	3	5	1	4	1	5	5
7	5	1	4	1	5	1	5	1	5	1	5	5
8	5	1	5	3	5	1	4	1	5	1	5	5
9	3	4	2	4	3	3	3	3	3	3	5	5
10	3	1	4	1	3	2	5	1	5	1	5	5
11	5	1	5	1	4	2	5	1	5	1	5	5
12	5	1	5	1	4	1	5	1	5	1	5	5
13	5	5	2	5	3	5	2	5	2	1	3	3

Fonte: O próprio autor.

O *score* final é obtido através dos passos a seguir, definidos por Sauro (2011):

1. Para questões ímpares, subtraia 1 do valor da resposta do usuário.
2. Para questões pares, subtraia o valor da resposta de usuário de 5.
3. Então, some todos os valores convertidos das dez questões para cada usuário, e multiplique por 2,5.

O *score* final pode variar de 0 a 100 e é atingido pelo cálculo da média dos resultados de cada voluntário, que no caso deste trabalho foi de aproximadamente 81,92. De acordo com Sauro (2011), a média do *score* é de 68 pontos. Sendo assim, é possível observar que o resultado adquirido está bem acima da média.

5 Considerações Finais

São muitas as dificuldades apresentadas na gestão do estágio curricular supervisionado. No entanto, com o auxílio das Tecnologias da Informação e Comunicação é possível oportunizar novas experiências aos estudantes em campo de estágio, assim como permitir aos gestores do ensino superior e empresas/instituições concedentes uma forma dinâmica e automatizada no controle de documentos e processos. Diante desse cenário, a plataforma SIGESTAGIOS foi projetada a fim de mitigar possíveis erros e redundância no preenchimento dos dados e facilitar a geração dos relatórios referentes ao estágio. Diante dos dados

disponibilizados no sistema, com apenas um clique é possível gerar todos os relatórios referentes ao estágio.

Este projeto teve como objetivo fundamental entregar uma ferramenta tecnológica que possibilitasse um controle centralizado e facilitado dos documentos, tanto por parte dos estudantes como por parte dos docentes e supervisores do componente de estágio curricular da UESPI-Parnaíba. Não há mais necessidade de impressões desnecessárias destes documentos, visto que eles são mantidos na base de dados do sistema, ou por meio de preenchimento de formulários ou por meio de *upload* de arquivos. Além de ser possível saber se o documento foi validado e/ou revisado pelo docente e/ou pelo supervisor.

Para conseguir atingir os objetivos foi fundamental o estudo e uso de técnicas de desenvolvimento de *software* e de tecnologias como o *framework* Django, Django REST Framework, Vue.js, assim como integração com a API do IBGE na aplicação. Dessa maneira, bastante conhecimento foi adquirido.

De acordo com o resultado do questionário de usabilidade SUS, o sistema obteve um *score* bem acima da média. Isso quer dizer que o sistema foi, de maneira geral, bem intuitivo para os usuários voluntários. Além de apresentar uma boa usabilidade, o desenvolvimento dessa ferramenta possibilita um passo inicial quanto a futuros trabalhos que tenham como objetivo o melhoramento da plataforma, ao tempo que disponibiliza os diagramas modelos do sistema. Outra contribuição é o próprio sistema como ferramenta que pode auxiliar o corpo docente e discente da instituição ao qual foi planejado e desenvolvido e possivelmente outras instituições.

Como trabalhos futuros, no que se refere a este sistema, além da adição de funcionalidades, como gerenciar os documentos sem a necessidade de *upload* de arquivos, é importante ressaltar não só o aperfeiçoamento de algumas das funcionalidades, como, por exemplo, as mensagens de notificação aos atores envolvidos no processo, mas também a realização de mais testes para validação da proposta.

6 Referências

ALMEIDA, B. C. **Desenvolvimento de um Sistema Web para Controle de Faltas da Universidade Federal de Ouro Preto**. Universidade Federal de Ouro Preto. Monografia. Minas Gerais, 2018.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **Uml: Guia do Usuário**. 2 ed. Rio de Janeiro: Elsevier, 2005.

BRENDON, Hudson. **Criando API REST com Django Rest Framework 1/10**. Grupy RN, 2016. Disponível em: <<https://medium.com/grupy-rn/criando-api-rest-com-django-rest-framework-1-10-e77208f7e5b9>>. Acesso em: 19 jul. 2019.

DIONISIO, EDSON JOSÉ. **PostgreSQL Tutorial**. DevMedia, 2015. Disponível em: <<https://www.devmedia.com.br/postgresql-tutorial/33025>>. Acesso em: 18 jul. 2019.

DJANGO. **Django makes it easier to build better Web apps more quickly and with less code**, 2005. Página inicial. Disponível em: <djangoproject.com/>. Acesso em: 10 dez. 2018.

FERREIRA, J. C. **A tecnologia como instrumento na gestão pública: um estudo de caso do processo administrativo digital no Tribunal Regional Eleitoral do Maranhão**. Universidade Federal do Maranhão. Monografia. Maranhão, 2016.

FREITAS, MARCIO L. **UML**. DevMedia, 2008. Disponível em: <<https://www.devmedia.com.br/uml/8579>>. Acesso em: 18 jul. 2019.

HEUSER, Carlos Alberto. **Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS**. Bookman Editora, 2009.

LEONCIO, D. **Sistema para controle de estágio supervisionado do departamento acadêmico de informática**. Universidade Federal do Paraná. Monografia. Paraná, 2018.

MCLEOD, Saul. **Likert Scale**. SimplyPsychology, 2008. Disponível em: <<https://www.simplypsychology.org/likert-scale.html>>. Acesso em: 23 jul. 2019.

NASCIMENTO, E. S. **Modelagem e Protótipo de um Sistema para Gerenciamento de Reuniões**. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Formiga. Monografia. Formiga, jun. 2017.

OLIVEIRA, C.; MOURA, S. P. **TIC'S na educação: a utilização das tecnologias da informação e comunicação na aprendizagem do aluno**. Disponível em: <<http://periodicos.pucminas.br/index.php/pedagogiacao/article/viewFile/11019/8864>>. Acesso em: 15 jul. 2019.

POSTGRESQL. **About PostgreSQL**. 1996. Disponível em: <postgresql.org/about/>. Acesso em: 10 dez. 2018.

RALLO, Rafael. **Material Design: aprenda tudo sobre o design do Google!**. rockcontent. Disponível em: <<https://rockcontent.com/blog/material-design/>>. Acesso em: 23 jul. 2019.

RIBEIRO, Leandro. **O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML**. DevMedia, 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>>. Acesso em: 09 jun. 2019.

SAURO, J. **Measuring usability with the system usability scale (SUS)**. MeasuringU, 2011. Disponível em: <<https://measuringu.com/sus/>>. Acesso em: 19 jul. 2019.

SILVA, A. dos S. S.; CONCEIÇÃO, E. B. da; SILVA, L. S. da. **Desenvolvimento de um Sistema de Controle de Atividades Complementares em Ruby On Rails - Um Estudo das Gems Bootstrap, Devise e Prawn**. Instituto Federal Fluminense. Monografia. Rio de Janeiro, 2016.

SILVA, M. R. R. da. **Projeto e Desenvolvimento de um Sistema para Gerenciamento de Trabalhos de Conclusão de Curso**. Universidade Federal de Uberlândia. Monografia. Minas Gerais, 2017.

TEIXEIRA, Fabricio. **O que é o SUS (System Usability Scale) e como usá-lo em seu site**. UX Collective BR, 2015. Disponível em: <<https://brasil.uxdesign.cc/o-que-e-o-sus-system-usability-scale-e-como-usa-lo-em-seu-site-6d63224481c8>>. Acesso em: 21 jul. 2019.

VUE. **The Progressive JavaScript Framework**, 2014. Página inicial. Disponível em: <<https://vuejs.org/>>. Acesso em: 23 jul. 2019.

VUETIFY. **Material Design Component Framework**, 2016. Página inicial. Disponível em: <<https://vuetifyjs.com/en/>>. Acesso em: 23 jul. 2019.