

GESTOR DE TCC: DESENVOLVIMENTO DE SISTEMA PARA GESTÃO DE TRABALHOS ACADÊMICOS

Lucas Castejon Alves¹

Murilo Ferreira Urquiza¹

Carlos Eduardo de França Roland².

RESUMO

Observa-se nas universidades atualmente certa dificuldade de orientadores, professores, coordenadores de cursos e alunos, para gerir e desenvolver trabalhos acadêmicos, Trabalhos de Conclusão de Curso (TCC), Iniciação Científica (IC) ou Monografias. Os cronogramas costumam atrasar e os coordenadores e orientadores têm dificuldades para controlar o progresso dos projetos. A partir dessa observação surgiu a ideia de desenvolver um sistema que permita a gerência de projetos de TCC, possibilitando que as partes interessadas fiquem cientes das etapas de execução dos projetos e acompanhem o cumprimento de prazos, tenham controle das etapas e mantenham a comunicação centrada em uma mesma ferramenta. O sistema tem como objetivo organizar todos os processos dos projetos, permitindo a inclusão de orientadores e suas áreas de pesquisa, relação de temas de projetos, alunos envolvidos, dentre outros requisitos. O orientador juntamente com o aluno, definirão o cronograma de desenvolvimento das etapas alocando os marcos. O aluno, por sua vez, fará a inserção dos documentos das etapas, permitindo que o orientador avalie cada entrega. Com barras de progresso do desenvolvimento, as partes, assim como a coordenação do curso, serão notificadas de como está o andamento dos trabalhos. O sistema ainda permitirá o planejamento da realização da banca de examinadores e a emissão dos documentos utilizados nas sessões públicas de apresentação da monografia. A organização que o sistema permitirá, e todas as funcionalidades disponíveis, facilitarão a comunicação entre as partes envolvidas nos projetos, melhorando a produtividade dos trabalhos, e a geração de documentação de todo o processo. Para o desenvolvimento desse projeto foi necessária a realização de pesquisa bibliográfica acerca dos conceitos que contextualizam o tema, a questão problema, os objetivos geral e específicos, bem como dos elementos de projeto e desenvolvimento do sistema, quais sejam arquitetura Cliente/Servidor e Web, *frameworks* de desenvolvimento, Sistemas Gerenciadores de Banco de Dados e ferramentas utilizadas para a Gestão dos Projetos. No trabalho são abordadas as etapas realizadas, como o planejamento incluindo a análise de produtos concorrentes, entrevistas com especialistas, coleta e documentação de requisitos, prototipação, diagramas de atividades, de casos de uso, de máquina de estado e de sequência,

¹ Discente do curso de Bacharelado em Sistemas de Informação do Uni-FACEF

² Docente do curso de Bacharelado em Sistemas de Informação do Uni-FACEF

modelagem de Banco de Dados (BD) e o cálculo das métricas de custo e prazo do projeto. Também é apresentado o desenvolvimento do projeto, como a implementação do BD e dos módulos do sistema, sendo que cada módulo tem o seu Model, View e Template (MTV), seguindo o padrão do *framework* Django. São descritas modificações de escopo do projeto que surgiram, os resultados da implementação do protótipo do sistema apresentado, e, por fim, são apresentadas as Considerações Finais com eventuais problemas identificados, constatações que foram obtidas a partir da experiência do desenvolvimento, com possíveis melhorias futuras.

Palavras-chave: Gestão de Projetos. Gestor de TCC. Sistemas de Informação. Software. Trabalhos Acadêmicos.

ABSTRACT

Nowadays on the universities it is noticed a certain difficulty of mentors, course coordinators, teachers and students on managing and developing academic papers, Term Papers, Scientific Initiation or Monographs. Schedules delay very often and the coordinators and mentors find difficult to control the progress of the projects. Following this observations the idea of develop a system that allows the management of a Term Paper emerged. A system which enables the interested parts to be more aware of the project execution stages and monitor deadlines with compliance, have control of the steps ahead and keep the communication focused on the same tool. The system aims to organize all processes of the projects, allowing the inclusion of mentors and their research areas, list of projects themes, students involved, among other requirements. The supervisor together with the student, define the steps of the development schedule allocating landmarks. The students, in turn, will make the inclusion of the steps of the documents, allowing the mentor to evaluate each delivery. With the development progress bars, all the parties, as well as the course coordinator, shall be notified of how is the progress of the project. The system will also enable the planning the board of examiners and issue the documents used in the public sessions for the presentation of the monograph. The organization that the system will allow, and all available features, facilitate the communication between the parties involved in the projects, improving the productivity of the paper, and generate a documentation of the entire process. For the development of this project was necessary to carry out the bibliographic research about the concepts that contextualize the theme, the problem in case, the general and specific objectives, as well as the project elements and development of the system, on both architectures: Client/Server and Web, development frameworks, Database Management System and tools used for managing the projects. This paper deals with the steps taken, such as the planning including the analysis of competing products, interviews with specialists, collection and documentations of requirements, prototyping and activity, use case, state machine, sequence diagrams, database modeling and the calculation of cost metrics and project deadline. The project development is presented, that consists in the database implementation and the system modules, each module has its own Model View and Template (MVT), following the Django framework pattern. The changes on the scope project are described as they emerged as well as the results of the prototype

implementation from the presented system, and at the end the final considerations are presented alongside the identified problems, observations that were obtained through the user experience on the development, with possible future improvements

Keywords: Project Management. Term Paper Management. Information Systems. Software. Academic Projects.

1 INTRODUÇÃO

Pela experiência vivenciada durante os anos letivos no Centro Universitário Municipal de Franca - Uni-FACEF, foi observada a dificuldade no relacionamento entre orientadores, professores, coordenadores e alunos, para gerir e desenvolver trabalhos acadêmicos, como por exemplo, Artigos, Trabalhos de Conclusão de Curso (TCC), Relatórios de Iniciação Científica (IC), e Monografias. Essa problemática também pode estar sendo vivenciada em outras instituições de ensino, pois é o mesmo cenário acadêmico. Os atrasos no desenvolvimento dos projetos de TCC e a dificuldade dos coordenadores e orientadores em controlar o progresso dos projetos são perceptíveis, prejudicando sua qualidade, refletindo nas avaliações ou até recebendo reprovações, pois o tempo de desenvolvimento costuma ficar aquém do necessário.

Em entrevistas realizadas pelos autores, foram levantados alguns problemas que ocasionam as falhas citadas, como por exemplo, a falha na comunicação entre os orientadores e alunos e qualidade deficitária das informações sobre o projeto, impossibilitando o desenvolvimento dentro dos prazos estabelecidos pela matriz curricular. Neste cenário, surgiu a ideia do desenvolvimento de um sistema para gestão de trabalhos acadêmicos, com o objetivo de melhorar a organização das etapas e melhorar a eficiência do gerenciamento dos trabalhos.

Para iniciar o projeto, foi definida a questão problema: como a tecnologia de informação pode auxiliar a gestão de trabalhos acadêmicos? A hipótese de solução proposta é o desenvolvimento de uma aplicação computacional para auxiliar a gestão de trabalhos acadêmicos como os TCC. A motivação vem da dificuldade de organizar toda a documentação e cumprir os cronogramas desses trabalhos, além da demanda existente no ramo educacional com a carência de oferta de sistemas que façam a gerencia de trabalhos acadêmicos como o proposto.

O objetivo do projeto é o desenvolvimento de um protótipo funcional de artefato de *software*, a ser operado em ambiente *web*, para a gestão de projetos acadêmicos, inicialmente focado em projetos de TCC do Centro Universitário Municipal de Franca - Uni-FACEF.

No capítulo 2, é apresentado o Referencial Teórico sobre algumas tecnologias que foram utilizadas para o desenvolvimento do sistema, que teve como linguagem de programação o Python e o *Framework* Django na versão Web 1.7.1; também a metodologia MVC de desenvolvimento de software com o uso da MTV adotada pelo ambiente Python/Django; o Sistema Gerenciador de Banco de Dados (SGBD) PostgreSQL; PgModeler para modelagem do Banco de Dados. Para a criação

dos diagramas de projeto foi utilizado o Enterprise Architect; para a criação da Estrutura Analítica de Projeto (EAP) o WBS Tool; o sistema para prototipação Draw.io; o Heroku para hospedagem do protótipo; e o Bitbucket para controle de versionamento dos códigos.

A seguir, no capítulo 3, aborda-se o planejamento do projeto, como a realização da análise de produtos concorrentes, entrevistas, coleta de dados, documentação de requisitos, prototipação, diagramas de atividades, casos de uso, máquina de estado e de sequência, modelagem de Banco de Dados (BD) e o cálculo das métricas de custo e prazo do projeto.

Na sequência, no capítulo 4, é apresentado o desenvolvimento do projeto, como a implementação do BD e dos módulos do sistema, sendo que cada módulo tem o seu Model, View e Template (MTV), seguindo o padrão do *framework* Django. Nesta etapa, surgiram modificações de escopo do projeto, que foram descritas neste capítulo.

O capítulo 5 aborda os resultados da implementação do protótipo do sistema apresentado. E por fim, são apresentadas as Considerações Finais com eventuais problemas identificados, constatações que foram obtidas a partir da experiência de desenvolvimento, e possíveis melhorias futuras.

2 REFERENCIAL TEÓRICO

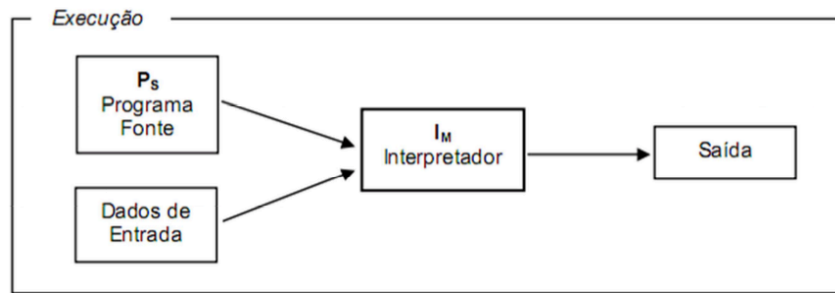
Nesta seção, são apresentados os conceitos e as definições dos elementos que contextualizam tecnologicamente o tema, a questão problema e a hipótese de solução que orientam esse projeto. O objetivo é permitir que o leitor tenha uma visão introdutória sobre os conceitos de cada um dos elementos fundamentais do projeto.

2.1.1 Python

De acordo com Borges (2010, p. 13), Python é uma linguagem de programação (LP) de altíssimo nível, *object oriented* (OO) ou em português orientada a objeto, de tipagem dinâmica e forte, interpretada e interativa.

O interpretador do Python é interativo, ou seja, é possível executá-lo sem fornecer um script (programa) para ele. Ao invés disso, o interpretador disponibiliza uma interface interativa na qual é possível inserir os comandos desejados um por um e ver o efeito da execução de cada um deles (UFF, 2011), como mostra a Figura 1.

Figura 1 - Linguagem Interpretada



Fonte: Caldas (Apud FLORES et al., 2010, p. 38).

2.2 FRAMEWORK DJANGO DE DESENVOLVIMENTO PYTHON

Framework é um conjunto de classes que incorpora um projeto abstrato para soluções de famílias de problemas associados. Em outras palavras *framework* é um projeto e uma implementação parcial de uma aplicação para um dado domínio do problema.

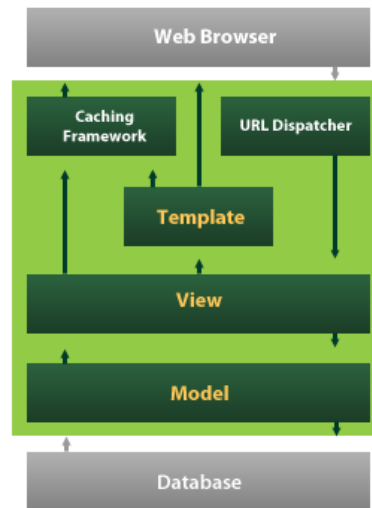
O *framework* para o desenvolvimento é a “base de onde se pode desenvolver algo maior ou mais específico. É uma coleção de códigos-fonte, classes, funções, técnicas e metodologias que facilitam o desenvolvimento de novos softwares” afirma Minetto (2013, p. 17).

2.2.1 MTV adotada pelo ambiente Python/Django

O Django redefine o padrão da tradicional arquitetura de *software* MVC para a definição MTV na qual M corresponde a Model, a camada de acesso aos dados. Esta camada contém qualquer coisa e tudo sobre os dados: como acessá-los, como validá-los, quais comportamentos terão, e seus relacionamentos. T corresponde a Template, a camada de apresentação. Esta camada contém decisões relacionadas à apresentação: como algo deve ser mostrado em uma página *web* ou qualquer outro tipo de documento. V corresponde a View, a camada da lógica de negócios. Esta camada contém a lógica que acessa os dados (Model) e os submete à apresentação (Template) apropriada. A camada View pode ser pensada como uma ponte entre Model e Templates (SILVA, Regis, 2015, online).

Na Figura 2 são representadas as camadas conforme explicadas pelo autor.

Figura 2 - Desenho de camadas MTV



Fonte: Whitton (2015, online).

2.3 SISTEMAS GERENCIADORES DE BANCOS DE DADOS (SGBD)

Na obra de Amadeu (2015, p. 8), é apresentada a definição: “um sistema gerenciador de banco de dados (SGBD) é um sistema de software de uso geral que facilita o processo de definição, construção, manipulação, e compartilhamento de banco de dados entre diversos usuários e aplicações”.

Existem várias opções no mercado entre SGBD gratuitos e com política de licenciamento de uso cobrados. O produto adotado, neste projeto, é o PostgreSQL.

2.4 FERRAMENTA DE DOCUMENTAÇÃO PROJETO

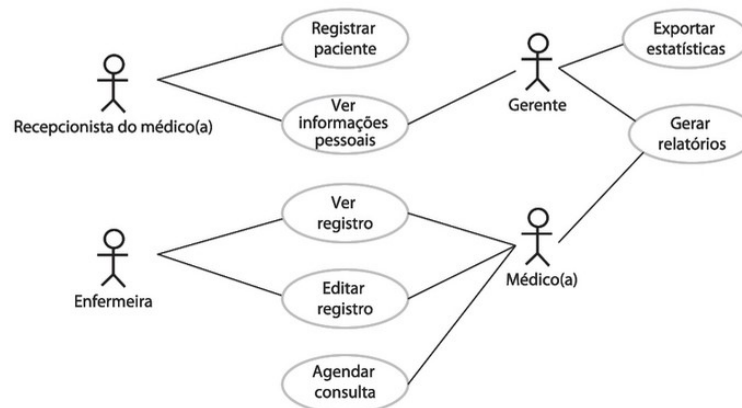
Nesta seção, é apresentada a ferramenta utilizada no desenvolvimento da documentação do projeto, assim como o *software* utilizado como ferramenta.

2.4.1 Unified Modeling Language (UML)

A UML ou Linguagem de Modelagem Unificada, segundo Medeiros (2004), não mostra de forma alguma como um software deve ser desenvolvido. Na verdade, a UML indica diversas formas para se representar as etapas do desenvolvimento de um software. UML é uma linguagem para comunicação humana de aspectos técnicos do desenvolvimento de sistemas de informação. Essa linguagem tem como objetivo padronizar a comunicação das partes envolvidas no desenvolvimento, de forma que mesmo em países diferentes, e até línguas diferentes, possa ser entendido corretamente o que é para ser desenvolvido.

Sommerville (2007) caracteriza a UML como um padrão real para modelagem orientada a objetos, e, com isso, os diagramas de Casos de Uso têm sido cada vez mais usados para a elicitación de requisitos. Medeiros (2004) também reforça a importância de um diagrama de Casos de Uso na construção de software orientado a objetos utilizando a UML. Segundo o autor, este diagrama pode até ser o único instrumento que acompanha o software de seu início até o fim. Toda ação que houver no sistema, será representada como um Caso de Uso na UML. E quem realizará essa ação será um Ator, que pode ser uma pessoa, um sistema, um equipamento, uma entidade externa, ou seja, algo ou alguém que realizará uma ação. Na Figura 3, mostre-se a representação de um exemplo da obra de Sommerville (2011).

Figura 3 – Exemplo de Diagrama de Casos de Uso



Fonte: Sommerville (2011, p. 75).

3 ANÁLISE E PROJETO DO SISTEMA

Nesta seção, são apresentadas as etapas realizadas na análise e desenvolvimento do projeto para a implementação do sistema, abrangendo as características necessárias para o funcionamento do Gestor de TCC em uma instituição. Esse planejamento foi baseado nas necessidades do Centro Universitário Municipal de Franca - Uni-FACEF, para a organização de Trabalhos de Conclusão de Curso, mas já se tem a ideia de posteriormente adaptá-lo para ser uma ferramenta de gestão de trabalhos acadêmicos em geral como trabalhos de Iniciação Científica, ou artigos e monografias diversas de que as instituições necessitem.

A análise, o projeto e o planejamento foram realizados abrangendo o sistema completo, mas para viabilizar ao cronograma acadêmico, foi desenvolvido apenas um protótipo funcional de artefato de *software* com as tecnologias descritas.

3.1 LEVANTAMENTO DE REQUISITOS

Nesta seção, são apresentadas as etapas realizadas para o levantamento de requisitos do sistema Gestor de TCC. Foi realizada entrevista com o professor coordenador do curso de Sistemas de Informação (SI) do Uni-FACEF, e futuro usuário do sistema.

Também foram realizadas pesquisas informais com alunos e professores da instituição, e profissionais da área de desenvolvimento de software.

A partir deste levantamento, foi possível levantar dados e as necessidades que o sistema deve atender para auxiliar o desenvolvimento do TCC, e criar os documentos com os requisitos levantados.

3.1.1 Entrevista com Coordenador de Curso

A entrevista foi realizada com o intuito de se levantar os dados e requisitos para a elaboração do projeto do sistema, bem como o posicionamento sobre a real necessidade desse tipo de produto na instituição em foco.

O resultado da entrevista mostrou alta receptividade por parte do entrevistado, que enfatizou a possibilidade de minimização de problemas com os projetos, uma vez que sistematiza e formaliza os processos. Da entrevista, também puderam ser identificados os requisitos que o sistema deve atender, com os quais foi desenvolvido o Documento de Requisitos. A seguir, foi criado o Diagrama de Casos de Uso para se alcançar a visão sistêmica e holística do projeto, e desenvolvida a prototipação das telas.

Além de auxiliar o desenvolvimento do TCC, o entrevistado mostrou a necessidade do sistema para auxiliar em tarefas administrativas da coordenação, como por exemplo, auxiliar no controle de professores que estarão nas bancas examinadoras das apresentações dos trabalhos, bem como da documentação necessária à realização das seções públicas de apresentação dos TCC. Esses itens também foram incluídos no Documento de Requisitos.

3.1.2 Documentação de Requisitos

Trata-se dos documentos mais importantes do projeto, que servirão de base para os desenvolvedores, gerentes de projetos, engenheiros de software até mesmo para a equipe de testes. De acordo com o proposto pela Engenharia de Software, utiliza-se uma tabela para listar e descrever todos os requisitos necessários para que o sistema funcione em perfeito estado. ID Requisito, Nome Requisito, Descrição do Requisito, são as colunas que iniciam a documentação dos requisitos.

O documento foi elaborado desde o início do desenvolvimento do projeto, em agosto de 2014, e conforme se encontravam novos requisitos do sistema alguns deles eram acrescentados ao documento, tendo sido atualizado até setembro de 2015.

3.2 PROJETO

Os diagramas UML foram desenvolvidos em novembro de 2014, no início do projeto. Durante a implementação de algumas funcionalidades foram identificados novos requisitos, mas não foram contemplados nos diagramas UML por fugirem do escopo do projeto.

3.2.1 Prototipação de Interface

Em uma etapa no desenvolvimento de sistemas, existe a prototipação de interfaces que, por sua vez, define como um usuário operará o sistema e como o sistema apresentará as informações a ele segundo UFES (2015).

Ainda de acordo com o autor, o princípio básico para a prototipação de interfaces usuário é: “Conheça o usuário e as tarefas”. O processo de prototipação não é apenas tecnológico, e sim o estudo do usuário. Quem é o usuário? Como ele aprende a interagir com o novo sistema? O que ele espera do sistema? Essas são as questões dentre muitas outras para definir uma interface para o usuário UFES (2015).

Nesta seção, é apresentada a prototipação das telas de alguns casos que o sistema Gestor de TCC agrega em suas funcionalidades. O primeiro é o caso em que haverá a escolha do tema do projeto pelo aluno. E o segundo é o caso da tela que listará os projetos a que um professor está vinculado.

3.2.3.1 Caso Escolha do Tema do Projeto

De acordo com Oliveira Netto (2008), o processo de desenvolvimento do TCC inicia-se com a escolha do tema que os alunos desejam abordar; em seguida, é formulada a Questão de Pesquisa que orientará o desenvolvimento do projeto; as Justificativas; o Objetivo Geral e Objetivos Específicos; a definição da Metodologia a ser adotada para a realização do projeto; bem como o Cronograma com a estimativa de prazos para a realização das etapas do projeto. Dessa forma, o sistema Gestor proposto deve auxiliar desde a escolha do tema, apresentando as linhas de pesquisa e sugestões das áreas do conhecimento que cada professor orientador tem interesse em suas respectivas disciplinas, cadastradas previamente no sistema. O aluno poderá escolher, dentre as opções, o tema e o professor orientador iniciando-se então o pré-projeto de TCC.

Tem-se, na Figura 5, um recorte do protótipo de tela para a escolha de Tema e convite ao Orientador no sistema Gestor de TCC, onde são exibidos os assuntos que os professores sugerem. Nela o aluno pode definir o tema que ele deseja abordar em seu projeto, e descrever sua ideia resumidamente. O sistema notificará o professor enviando o convite para orientar o trabalho do aluno.

Figura 5 - Protótipo de tela de escolha de tema e orientador

The screenshot shows a web interface with a vertical navigation bar on the left containing two user profiles: Prof. Me. Carlos Eduardo Roland and Prof. Me. Carlos Alberto Lucas. The main content area is divided into two sections. The top section is a dropdown menu with a list of project topics: Sistemas de Informação (Gestão, Análise e Projeto), Gestão de Projetos, IHC, Automação de Processos, Políticas Públicas para Inclusão Digital, Desenvolvimento Web (Python+Django), Aplicativos Android, and Tecnologia na Educação. The 'Gestão de Projetos' option is selected. Below the dropdown is a form with a text input field containing 'Ex: Gestão de Projetos', a 'Descrição...' label, and two buttons: 'Cancelar' and 'Convidar'. The bottom section shows a list of project topics: Engenharia de Software and Aplicativos para pessoas portadoras de necessidades especiais, with a right-pointing arrow button.

Fonte: Elaborado pelos autores.

3.2.3.2 Caso Etapas do Cronograma do Projeto

Iniciando o pré-projeto, os alunos deverão descrever a Questão Problema, as Justificativas, os objetivos Geral e Específicos, e os procedimentos metodológicos a serem utilizadas no desenvolvimento do projeto. Portanto, o sistema também auxiliará nessas etapas, solicitando automaticamente, todas elas, e permitindo que sejam alocados os prazos para o cumprimento de cada uma dessas etapas. O aluno escreve os textos, inserindo-os na etapa para que o orientador receba a notificação para analisar e aprovar ou pedir correções, dando continuidade ao desenvolvimento.

As etapas do projeto são cadastradas pelo orientador ou pelos orientandos em um processo colaborativo. Em cada uma, são informadas a descrição do escopo, data de entrega, e tempo de esforço estimado, que na verdade é uma estimativa de quantidade horas que o aluno dedicará para realizar determinada etapa. Essa funcionalidade de estimar o esforço, foi verificada no Runrun.it (2015), um dos gerenciadores de tarefas que foram avaliados pelos autores.

Na Figura 6, é mostrada a lista de etapas do projeto, exibindo informações já citadas. Nessa tela, é possível visualizar a avaliação que foi feita pelo orientador em cada etapa, que são as estrelinhas indicando as notas atribuídas. Com essas notas em cada etapa, será possível obter uma média de avaliação do projeto, facilitando os professores orientadores e coordenadores a ter um relatório da situação dos projetos.

Figura 6 - Tela com as etapas do cronograma do projeto

Projeto Inclusão Digital

-
-
-
-
-
-
-
-
-
-
-
-

#001 - Introdução

7%

Esforço estimado: 02:00

#002 - Resumo

8%

Esforço estimado: 02:00

#003 - Sumário

Fonte: Elaborado pelos autores.

3.2.3.3 Caso Área de Projetos de um Professor

Os professores poderão vincular-se para orientar mais de um projeto ao mesmo tempo. Portanto, o sistema controlará os dados, listando as etapas de cada projeto separadamente. O professor terá uma tela onde será possível visualizar uma lista de projetos que ele está vinculado como orientador. Nesta tela, também será possível visualizar alguns avisos no seu topo, como convites de alunos para orientar projetos, convite para banca de avaliação, estatísticas de quantidade de projetos em andamento e finalizados, e exibição do tempo restante para entrega de etapas dos projetos.

3.2.2 Métricas de custo e prazo do projeto

Antes da realização do projeto, conforme preconizado nas disciplinas de Engenharia de Software, foi realizado o cálculo de métricas de custo e prazo, resultando que seriam necessárias 494 horas para a implementação do software, que equivale a aproximadamente quatro meses e meio, trabalhando-se 6 horas por dia útil. O custo chegaria a R\$ 11.880,00, levando-se em consideração a base de R\$ 20,00 por hora de trabalho do desenvolvedor. Deve-se ressaltar que este cálculo foi realizado no início da análise e projeto, o que significa que nem todos os requisitos e elementos do

software foram levados em consideração nessas métricas. Mas com elas, já se pode ter uma estimativa acerca da complexidade do projeto.

4 DESENVOLVIMENTO DO PROJETO

Nesta seção, são tratados os conceitos de requisitos funcionais e não funcionais e também apresentados alguns dos requisitos do sistema que foram levantados para seu projeto.

4.1 REQUISITOS FUNCIONAIS

De acordo com Sommerville (2003, p. 84), “Os requisitos funcionais para um sistema descrevem a funcionalidade ou os serviços que se espera que o sistema forneça”. Os requisitos de um sistema obviamente não são os mesmos de outro sistema. Cada software tem suas necessidades, de acordo com o lugar em que será usado, e por quem será usado. São apresentados a seguir alguns dos requisitos funcionais de destaque do projeto.

4.1.1 Usuário realizando as etapas

Já com a etapa cadastrada, o aluno ou orientador quando for realizá-la, clicará no botão Iniciar, e, com isso, começará a contar o tempo de realização da etapa. Assim que terminar de realizá-la, o usuário que realizou a etapa clicará no botão Pausar, encerrando o registro do tempo despendido para a atividade. Quando o usuário voltar a trabalhar nessa etapa, o processo deverá ser repetido. Com isso, o tempo de esforço efetuado em cada etapa será contabilizado, e contando que ele já foi estimado, ajudará na mensuração do tempo restante para concluir a etapa e também o projeto como um todo.

4.1.2 Definição de horas disponíveis para o trabalho

Como afirma Wazlawick (2014), o processo de desenvolvimento do TCC, ou de qualquer outro trabalho acadêmico, exige disciplina e dedicação dos alunos para focarem no estudo e redação da monografia. Pensando nisso, o Gestor também deve proporcionar, aos alunos e professores, a possibilidade de estabelecer a quantidade de horas semanais que se dedicarão ao trabalho, seja estudando, pesquisando ou escrevendo. Tendo essa definição das horas disponíveis para dedicação ao trabalho, após inserir as etapas do projeto, os seus prazos e as estimativas de tempo de esforço, será possível estimar a data de entrega prevista de cada uma das etapas. A alocação de estimativa de horas para as etapas de um projeto e o cálculo de estimativa de data de entrega serão explicados em item posterior.

Para o usuário definir o tempo disponível para trabalhar em cada um dos projetos em que estará envolvido, seja como orientador ou como aluno, terá uma

página para definir as horas disponíveis para todos os projetos, conforme apresentado na Figura 7. À medida que vai atualizando as horas de trabalho disponíveis por semana, em cada projeto, vai atualizando um total de horas disponíveis em todos projetos que será exibida em campo próprio nessa mesma página. Com isso, o usuário saberá quanto tempo está disponibilizando para todos os projetos, e facilmente pode alterar a alocação de tempo para um projeto que recebeu mais tempo que o necessário, por exemplo. Nessa mesma tela de alteração de horas alocadas nos projetos, podem ser mostrados os totais de horas já estimadas nas etapas de um projeto e também em todos os projetos.

Figura 7 - Prototipação da tela do Gestor de TCC indicando horas disponíveis de um projeto

Tempo para trabalho nos projetos

- 🏠 Início
- 📁 Projetos
- 👤 Professores
- 👥 Turmas
- 👁 Banca
- 📖 Biblioteca
- ✉ Mensagens
- ⚙ Configurações
- 🕒 Tempo
- ⚙ Perfil
- 📧 Notificações

Horas já disponibilizadas 4:30

Inclusão Digital - 1:00

Inteligência Artificial - 3:30

Salvar

Fonte: Elaborado pelos autores.

4.1.3 Estimativa de entrega da etapa

Ressaltam Leite Filho e Martins (2006) que a principal dificuldade entre orientador-orientando, evidenciada em entrevistas, foi a ausência de contato, com ênfase na falta de tempo das partes, para realização das atividades de orientação, que ainda evidenciam o baixo contato como um entrave para o desenvolvimento do projeto acadêmico.

Dados de Veiga e Viana (2007, p. 2) mostram o “sofrimento pela responsabilidade do orientando, que deve fazer o máximo em um período de tempo limitado”. Também é claro que existem problemas acadêmicos, pessoais ou até mesmo profissionais, mas que tais obstáculos, se superados até o final do projeto, potencializarão o resultado tanto profissional, quanto pessoal.

Para isso, o Gestor de TCC conta com uma ferramenta inteligente, chamada de Estimativa de Entrega. Essa ferramenta permitirá obter prazos de entrega de cada etapa, possibilitando o usuário organizar melhor seu tempo de trabalho, evitando acumular tarefas para última hora.

Ao cadastrar uma etapa, o usuário definirá o tempo de esforço estimado necessário para realizá-la. Ou seja, o usuário inserirá a quantidade de horas de trabalho naquela etapa que ele observará que é necessária para cumprir seu escopo.

Definidas as horas de trabalho disponíveis por projeto, conforme apresentado no item anterior, e também o esforço estimado em cada etapa dos projetos, o sistema conseguirá estimar a semana de entrega de cada etapa, de acordo com os dados fornecidos, e conseqüentemente também conseguirá estimar a entrega de todo o projeto. Com essa funcionalidade, é possível minimizar os efeitos da falta de tempo para desenvolver o projeto.

Na criação do projeto, os usuários definem a data desejada para a entrega do projeto. Essa data, normalmente, é definida pela coordenação do curso para a entrega do trabalho. Caso ainda não tenha sido definido esse prazo pela coordenação, o usuário terá que estabelecer uma previsão.

4.1.4 Notificações de lembretes disparados pelo site

Outro requisito identificado e que está no projeto, é o envio de notificações via e-mail e também na seção de notificações do sistema, elas serão configuradas por cada usuário. Ou seja, cada usuário informará quais notificações que deseja receber e de que modo. Algumas opções de notificações são: comentários, arquivo enviado, etapa aprovada, ou reprovada, prazo de entrega se aproximando, prazo extrapolando, etc. E os modos de notificação são via e-mail ou no ícone de notificações que dá acesso a seção de notificações.

4.1.5 Interação de alunos e orientador

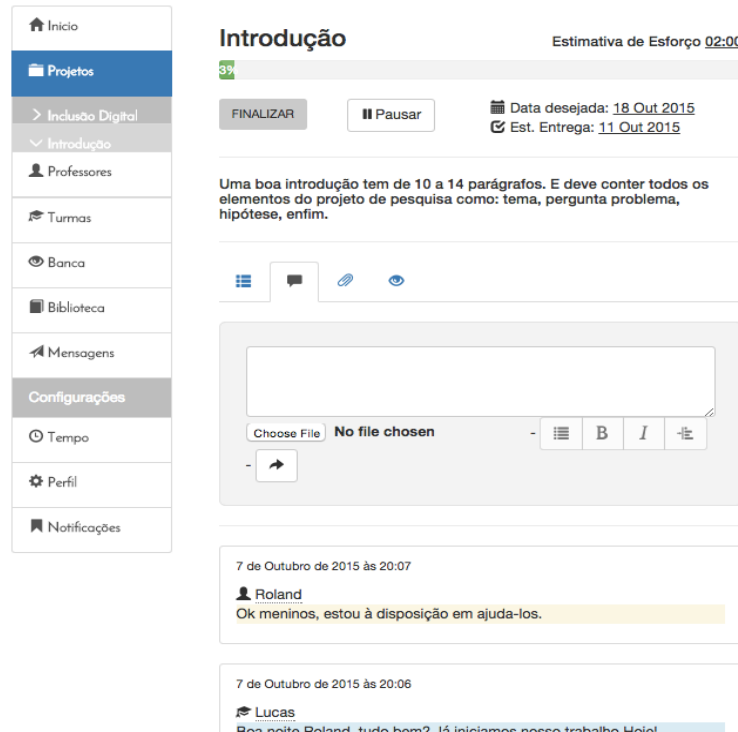
Na Figura 8, é apresentada, a tela com a etapa aberta de um projeto, que está implementada em fase de testes no *software*. Nessa tela, são exibidas interações de alunos e professor orientador, podendo enviar arquivos, corrigindo e relatando informações da etapa. Na etapa fica registrado quando o usuário, seja professor ou aluno, iniciou ou pausou a atividade. O usuário também poderá verificar informações do escopo da etapa, alterar o tempo de estimativa de esforço para a realização da etapa, alterar a data desejada e visualizar a data de entrega calculada pelo sistema.

4.1.6 Banca examinadora da apresentação

O sistema possibilitará ao coordenador do curso, após implementação desse requisito projetado, selecionar os professores que serão os avaliadores de cada projeto na apresentação, de acordo com a disponibilidade de cada um deles, levando em consideração as outras bancas que ele participará. O Gestor também fornecerá

materiais a serem impressos para a banca examinadora como a Ata, Relatórios de Avaliação pelos examinadores e Certificados da Participação para os examinadores, orientadores e orientandos. Esses itens também estão incluídos no Documento de Requisitos. Inclusive, foram identificados casos em que a universidade recebe convidados para a banca. Portanto, esse item também será abordado no sistema.

Figura 8 - Prototipação da tela de uma etapa aberta



Fonte: Elaborado pelos autores.

4.1.7 Traduções digitais e automáticas para a língua de sinais

Ressalta HandTalk (2015), empresa especializada em realizar traduções digitais e automáticas para a Língua de Sinais Brasileira (LIBRAS), que “são mais de 300 milhões de surdos no mundo e a maioria não compreende a língua escrita do seu país e depende exclusivamente da língua de sinais para se comunicar e obter informação”. Com base nesse número, pode-se considerar que existam alunos com esse tipo de deficiência. Esses alunos terão que desenvolver seus trabalhos acadêmicos, ou até mesmo lê-los publicados na Biblioteca Virtual do próprio Gestor de TCC.

Na Figura 9, pode-se ver a imagem de um homem executando a Língua de Sinais, que de acordo com a SPEI (2015), é a língua natural das comunidades surdas. Atualmente, poucas pessoas têm o conhecimento desta língua, a maioria das pessoas acreditam que são apenas jogo de sinais, movimentos aleatórios, apenas para passar a informação ao próximo, mas na realidade existem regras gramaticais e

sintáticas que sustentam a comunicação. Existem cursos, especializações, comunidades, dentre outras sendo, em sua grande maioria, de voluntários.

Figura 9 - Traduções digitais para a língua de sinais no Gestor de TCC



Fonte: Elaborado pelos autores.

Conforme dados do Censo da Educação Superior de 2013 (INEP, 2013), 7.305.977 alunos foram matriculados em ensino superior no Brasil, sendo que 29.034 (0,397%) alunos tinham alguma necessidade especial, e destes, 7.037 (24,2%) alunos eram com deficiência auditiva. O número de deficientes auditivos pode parecer pequeno comparado com o total de matriculados, mas considerando que 7.037 alunos estão com dificuldades de comunicação, e conseqüentemente dificuldades de desenvolver seus trabalhos acadêmicos, em conjunto com os professores, fica a necessidade de se ter funcionalidades que permitam a inclusão desses alunos.

O software Gestor de TCC foi planejado para receber a ferramenta HandTalk de traduções digitais e automáticas. Inicialmente, funciona com o plano básico, apenas quinhentas traduções por mês, isso permite, em média, 16 traduções por dia.

Oferecendo essa funcionalidade de traduzir os textos do Gestor de TCC para Libras, será permitida uma comunicação *online* entre orientadores e seus orientandos com esse tipo de deficiência, pois ainda é pequeno o número de professores que dominam a língua de sinais ficando próximo a zero. Com o auxílio da Tecnologia da Informação, consegue-se ultrapassar essas barreiras como preconceitos e falta de profissionais qualificados para este tipo de inclusão.

4.2 REQUISITOS NÃO FUNCIONAIS

Os requisitos não funcionais não dizem respeito diretamente as funções que o software agregará, mas “podem estar relacionados a propriedades de sistema

emergentes, como confiabilidade, tempo de resposta e espaço em disco” (SUMMERVILLE, 2003, p. 85).

Uma falha em cumprir com algum dos requisitos funcionais pode degradar o sistema, já uma “falha em cumprir um requisito não funcional de sistema pode tornar todo o sistema inútil” (SUMMERVILLE, 2003, p. 85).

A seguir, é elucidado um dos requisitos não funcionais de destaque para o sistema.

4.2.1 Sistema multi-empresa e acessibilidade

Inicialmente, o sistema Gestor de TCC será implantado no Centro Universitário Municipal de Franca - Uni-FACEF, como já relatado, mas posteriormente tem-se a intenção de levá-lo ao mercado e a outras instituições. Por isso, o mesmo deve operar em modo multi empresa, ou seja, permitindo que várias instituições e seus relacionados, acessem o sistema no mesmo local, mas visualizando informações referentes somente a projetos que dizem respeito a cada um deles.

Para cumprir tal requisito, o sistema foi projetado para operar em ambiente *web*. Sendo assim, será um *website* hospedado no Heroku que foi apresentado no capítulo anterior. Assim, de qualquer local e dispositivo que houver acesso à internet, será possível utilizar o Gestor de TCC. O *website* também deverá funcionar com as adaptações necessárias para os dispositivos móveis, como *tablets* e *smartphones*.

4.2.2 Backup do banco de dados

Como medida de segurança, o sistema de hospedagem fará backup automaticamente de todos os dados contidos no Gestor de TCC. Além disso, são gerados *logs* de transações realizadas dentro do sistema.

5 RESULTADOS

Utilizando o método de análise e projeto adotado para o desenvolvimento do Gestor de TCC, levantando os requisitos, criando os diagramas UML, prototipando telas para interfaces, foi possível ter uma visão das regras de negócio do sistema, o que permitiu otimizar o tempo de desenvolvimento. Ainda assim, na fase de implementação, surgiram discussões sobre alguns detalhes do sistema, oportunizando novas ideias que foram implementadas, e que melhoraram o protótipo gerado.

Se não tivesse sido utilizado o método de análise e projeto adotado não seria possível se atingir a visão holística do sistema necessária para a fase de implementação. Além disso, identificar requisitos enquanto se realiza a implementação gera retrabalho ao se ter que refazer os módulos nos quais houve necessidade de alteração durante o processo de desenvolvimento. No caso deste projeto, as novas ideias surgidas durante a implementação acrescentaram ao projeto melhorias sem ter gerado significativo volume de retrabalho.

Foram identificados elementos faltantes no levantamento dos requisitos do Gestor de TCC, que geraram novas reuniões da equipe para encontrar soluções ou mesmo adotar novos caminhos de desenvolvimento. Uma questão que gerou novas decisões foi a relacionada à versão do *Framework* Django. O projeto foi iniciado com versão 1.6 do *framework*, e após alguns meses foi lançada a nova versão 1.7.1, foi então realizada uma reunião e foi decidido pela atualização para a nova versão já considerada estável e testada.

Inicialmente, foi considerado o uso de dois *frameworks* de desenvolvimento simultaneamente: Django e AngularJS. Entretanto, essa opção se mostrou inadequada, pois os produtos se mostraram incompatíveis em alguns pontos, causando dificuldades para a implementação das funcionalidades do projeto. Foi então decidida a adoção de uma única ferramenta e a opção foi manter o Django, cujo domínio de uso, por parte dos desenvolvedores, era maior permitindo melhores resultados tanto sobre aspectos técnicos, quanto sobre prazos de entrega das atividades de projeto.

No período de desenvolvimento da monografia do projeto, não foi encontrada uma ferramenta específica que auxiliasse o desenvolvimento do TCC, por isso, foi adotado o ambiente Google Docs (2015) que é um pacote de aplicativos da Google baseado em AJAX que funciona totalmente *on-line* diretamente no navegador. Também foi utilizado o Runrun.it para organizar tarefas entre orientador e orientandos.

Com as aulas da disciplina de Engenharia de Software no 5º e 6º semestres do curso de Sistemas de Informação do Centro Universitário Municipal de Franca - Uni-FACEF, foi iniciado o projeto com o levantamento de requisitos, juntamente com as entrevistas e *feedbacks* de alunos, professores e coordenadores. Já com o levantamento de requisitos pronto, foi possível prosseguir para análise de dados para o projeto e a modelagem de processos e interface.

Com a ajuda das disciplinas Estrutura de Dados, Cálculos e Lógica de Programação, conseguiu-se elaborar uma das tarefas mais complexas do projeto: planejamento da estimativa de entrega.

Uma atividade que não foi identificada no levantamento de requisitos, mas que com a experiência adquirida com o desenvolvimento do sistema se mostrou necessária, foi a elaboração de meta *tags* para melhorar o posicionamento do *site* no *ranking* do mecanismo de buscas da Google. Inicialmente o *website* ocupava a quarta posição na busca por “gestor de tcc”, e depois da elaboração das *tags* passou a ocupar o primeiro lugar.

6 CONSIDERAÇÕES FINAIS

Diante da dificuldade observada nas instituições de ensino que regulamentam Trabalhos de Conclusão de Curso (TCC), como requisito obrigatório para a conclusão da formação acadêmica, foi projetada e está em fase de implementação a hipótese de solução para a questão problema: como a tecnologia de informação pode auxiliar a gestão de trabalhos acadêmicos? A constatação de que essa hipótese realmente funcionará, poderá ser obtida após a implantação e utilização do sistema na

instituição por um período completo de desenvolvimento dos TCC, sendo então possível se avaliar os resultados obtidos com dados reais de utilização do sistema.

Para o desenvolvimento do software poderiam ser utilizadas outras linguagens de programação, mas a robustez, a estabilidade, a confiabilidade e a grande comunidade colaborativa atuante, além da facilidade sintática e ortográfica que o Python com o *framework* Django proporcionam, e o conhecimento dos autores sobre essa LP, prevaleceram na sua escolha, e a avaliação é que ela foi correta.

A decisão de se adotar somente o *framework* Django como base para o desenvolvimento também foi confirmada como opção acertada, uma vez que o ganho de produtividade em relação a eventuais perdas técnicas, pelo não uso do AngularJS, se mostrou vantajoso.

Há que se considerar que o protótipo desenvolvido neste projeto está em versão Beta, sendo necessária, como a qualquer sistema, a implantação em teste em alguns ambientes acadêmicos para seu amadurecimento e correções. Há, ainda, uma série de melhorias que deverão ser implementadas para que o sistema alcance o *status* de produto, mas conceitual e funcionalmente, os principais requisitos foram identificados, analisados, projetados e implementados. Pretende-se oferecer sua utilização ao Centro Universitário Municipal de Franca - Uni-FACE para, a partir dos resultados de testes realizados, implementar correções e atualizações que potencializem os recursos do Gestor de TCC.

O desenvolvimento do projeto continuará a ser realizado com vistas a tornar o sistema um produto para a implantação em instituições de ensino que exijam a realização de TCC como regulamento. Inicialmente, o foco continuará em TCC, mas como melhorias futuras, pretende-se agregar funcionalidades, para que atenda às rotinas de desenvolvimento de outros trabalhos acadêmicos tais como relatórios de Iniciação Científica e de Extensão Universitária, produção de Artigos Científicos, dissertações de Mestrado, teses de Doutorado, dentre outros.

Algumas novas funcionalidades e melhorias foram sugeridas pela banca examinadora. A primeira delas é a alteração na rotina de convite do aluno para o professor orientar um projeto, solicitando o aval da coordenação do curso antes do convite chegar ao professor. Também confirmar o recebimento de convites e de leituras/visualizações dos comentários e atualizações. Outra sugestão foi limitar o horário em que os professores poderão utilizar o sistema, estando de acordo com o horário de trabalho estabelecido no contrato, evitando que se gerem problemas de cunho trabalhista à instituição. E também solicitar confirmação dos usuários para o recebimento ou não de mensagens, junto ao aceite de um termo que regulamente o procedimento. Foi sugerido para que após aprovados os projetos, sejam enviados à uma comissão científica que avaliará o projeto, para então ser publicado na Biblioteca do Gestor de TCC. Outra questão que será estudada futuramente, a pedido da banca examinadora, é a possibilidade de o sistema integrar com a Plataforma Lattes, criada e mantida pelo CNPq, para que os projetos aprovados, sejam inseridos automaticamente nos respectivos currículos dos participantes, mediante confirmação.

A funcionalidade do sistema de realizar a documentação do processo de desenvolvimento de projetos acadêmicos, oferece um nível de formalização de critérios para avaliação dos trabalhos e justificativa de notas atribuídas. O sistema pode expandir sua abrangência para gestão de relatórios que devem ser produzidos por

estagiários, docentes, coordenadores e instituição de uma forma geral, na medida em que ele permite o planejamento, execução, avaliação e *feedback* de cada projeto gerenciado, agregando desde processos acadêmicos até processos administrativos, pois envolve os estudantes, os docentes, as chefias de departamentos, a instituição e as secretarias.

Todo esforço realizado neste projeto foi válido à medida que possibilitou a aquisição de um corpo de conhecimentos e maturidade para a implementação de Sistemas de Informação, que serão utilizados em futuros projetos que a equipe venha a participar.

REFERÊNCIAS

AMADEU, Claudia Vicci (Org.) Banco de dados. 1 ed. São Paulo: Pearson Education do Brasil, 2015.

BORGES, Luiz Eduardo. Python para Desenvolvedores. Rio de Janeiro: Edição do Autor, 2010. Disponível em: <https://ark4n.files.wordpress.com/2010/01/python_para_desenvolvedores_2ed.pdf>. Acesso em: 19 set. 2015, 10:29:26.

CCM. Os modelos de SGBD. [s. l.]: CCM. Disponível em: <<http://br.ccm.net/contents/66-os-modelos-de-sgbd>>. Acesso em: 07 out. 2015, 10:59:00.

FLORES, Felipe Baratojo *et al.* Linguagem de Programação Interpretada Utilizada na Configuração de Aplicativos Voltados para Sistemas Multi-Plataforma com Implementação em Ambiente Paravirtualizado. São Paulo: Universidade Anhembi Morumbi, 2010. Disponível em: <<http://engenharia.anhembi.br/tcc-10/cco-12.pdf>>. Acesso em: 15 jun. 2015, 12:01:35.

GOOGLE DOCS. Google Docs. [s. l.]: Google, 2015. Disponível em: <<https://www.google.com/docs/about/>>. Acesso em: 08 out. 2015, 09:00:15.

HANDTALK. Handtalk. [s.l.]: HandTalk, 2015. Disponível em: <<http://www.handtalk.me/sobre>>. Acesso em: 16 set. 2015, 21:16:58.

INEP. Sinopses Estatísticas da Educação Superior - Graduação. Brasília: INEP - Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, 2013. Disponível em: <<http://portal.inep.gov.br/superior-censosuperior-sinopse>>. Acesso em 03 set. 2015, 12:09:22.

LEITE FILHO, Geraldo Alemandro; MARTINS, Gilberto de Andrade. Relação Orientador-Orientando e suas Influências na Elaboração de Teses e Dissertações. [s. l.]: FEA, 2006. Disponível em: <<http://www.spell.org.br/documentos/download/10859>>. Acesso em 16 set. 2015, 21:06:11.

MEDEIROS, Ernani. Desenvolvendo software com UML 2.0. São Paulo: Pearson Makron Books, 2004.

MINETTO, Elton Luís. Frameworks para Desenvolvimento em PHP. [s. l.]: Novatec, 2013. Disponível em: <<http://www.novateceditora.com.br/livros/frameworks/capitulo9788575221242.pdf>>. Acesso em: 19 set. 2015, 10:38:02.

OLIVEIRA NETTO, A. A. Metodologia da Pesquisa Científica: Guia Prático para Apresentação de Trabalhos Acadêmicos. Florianópolis: Visual Books, 2008.

RUNRUN.IT, 2015. Runrun.it. [s. l.]: Runrun.it. Disponível em: <<http://www.runrun.it/>>. Acesso em: 14 ago. 2015, 11:10:12.

SILVA, Regis da. Principais dúvidas de quem quer aprender Django. [s. l.]: PythonClub, 2014. Disponível em: <<http://pythonclub.com.br/principais-duvidas-de-quem-quer-aprender-django.html>>. Acesso em: 22 set. 2015, 00:32:16.

SOMMERVILLE, Ian. Engenharia de Software. 6 ed. São Paulo: Addison Wesley, 2003.

_____, Ian. Engenharia de Software. 8 ed. São Paulo: Pearson Addison- Wesley, 2007.

_____, Ian. Engenharia de Software. 9 ed. São Paulo: Pearson Addison- Wesley, 2011.

SPEI. Livro de Língua Brasileira de Sinais - Libras. Brasília: SPEI, 2015. Disponível em: <http://servicos.spei.br/site/arquivos/biblioteca/livros/livro_de_lingua_brasileira_dos_sinais.pdf>. Acesso em: 16 set. 2015, 21:16:12.

UFES. Engenharia de Software. [s. l.]: UFES - Universidade Federal do Espírito Santo, 2005. Disponível em: <<http://www.inf.ufes.br/~falbo/download/aulas/es-g/2005-1/NotasDeAula.pdf>> Acesso em: 07 out. 2015, 11:58:05.

UFF. Tutorial de Introdução ao Python. Niterói: Universidade Federal Fluminense (UFF), 2011. Disponível em: <<http://www.telecom.uff.br/pet/petws/downloads/apostilas/PYTHON.pdf>>. Acesso em:

01 set. 2015, 22:07:47.

VEIGA, Ilma Passos Alencastro; VIANA, Cleide Maria Quevedo Quixada. Orientação acadêmica: uma relação de solidão ou de solidariedade? [s. l.]: UnB, 2007. Disponível em: <<http://30reuniao.anped.org.br/trabalhos/GT04-3345--Int.pdf>>. Acesso em: 08 out. 2015, 00:59:51.

WAZLAWICK, Raul Sidnei. Metodologia de Pesquisa Para Ciência da Computação. 2. ed. Rio de Janeiro: Elsevier, 2014.

WHITTON, Thomas. Django by Example. [s. l.]: Thomas Whitton, 2012. Disponível em: <<http://www.thomaswhitton.com/django-presentation/#/4>>. Acesso em: 04 nov. 2015, 21:35:33.