
PROJETO PRÁTICO MULTIDISCIPLINAR: UM RELATO DE EXPERIÊNCIA MOTIVADORA DO ESTUDO NO ENSINO MÉDIO

Resumo

O aprendizado de forma colaborativa relaciona a teoria e a prática na solução de problemas visando formar profissionais bem preparados para o mercado de trabalho. A possibilidade de associar o conhecimento teórico e prático durante os estudos proporciona aos alunos uma vivência prática de sua futura profissão. Este artigo tem por objetivo, relatar a experiência dos alunos participantes de um projeto prático multidisciplinar, desenvolvido nas disciplinas de Programação para *Web II* do curso técnico em Informática para a Internet, do Instituto Federal de Goiás campus Luziânia, envolvendo e relacionando conteúdos das disciplinas de Engenharia de Software e Banco de Dados. A metodologia aplicada foi uma abordagem qualitativa caracterizada, sobretudo, pela sua natureza exploratória, descritiva, holística e interpretativa, com desenvolvimento de projetos de softwares e realização de entrevistas. Como resultados são apresentados às experiências, dificuldades, lições aprendidas e percepções dos alunos e professor.

Palavras-chave: Projeto de Software, metodologia de ensino, multidisciplinaridade.

INTRODUÇÃO

Na busca de um novo modelo de ensino onde os alunos não desempenhem somente o papel de simples receptores de informações, surge à possibilidade de que os alunos passem a serem os próprios construtores de seu conhecimento (LÊ, 2002).

Tal modelo vem atender às exigências atuais do mercado de trabalho, que demandam profissionais com iniciativa e que saibam trabalhar em equipe para resolver problemas complexos (LAFHEY *et al.*, 2003).

Neste sentido o emprego da multidisciplinaridade pode ser uma via para estabelecer esse perfil. A multidisciplinaridade é considerada importante para acabar com um ensino extremamente especializado, concentrado em uma única disciplina e ocorre quando “a solução de um problema torna necessário obter informação de duas ou mais ciências ou setores do conhecimento” (MORIN, 2002).

Segundo Lima (2009) os cursos de tecnologia da informação, deveriam utilizar um Processo de Design de *Software* (PDS) adequado ao contexto escolar, para contribuir no sucesso do aprendizado. Lima (2009) afirma ainda que estes devem consistir em: projetos com escopo pequeno ou médio a serem desenvolvidos dentro do período de um semestre ou ano letivos; com a presença de clientes reais; com uso de processos e tecnologias considerados estado da arte; com aprendizado colaborativo; com agilidade, como consequência direta do tempo limitado, do escopo restrito e do aprendizado colaborativo; com avaliação contínua e iterativa do progresso do projeto para que o aprendizado dos alunos possa ser avaliado ao longo da disciplina.

Assim, foi definido um projeto prático multidisciplinar desenvolvido na disciplina de Programação para *Web II* do curso técnico em Informática para a Internet do Instituto Federal de Goiás campus Luziânia, envolvendo e relacionando conteúdos das disciplinas de Engenharia de Software e Banco de Dados.

O objetivo do presente trabalho foi o desenvolvimento de projetos de softwares, de forma a levar o aluno a estar mais próximo possível a um ambiente real de desenvolvimento de software. A metodologia utilizada buscou promover a interação do aluno com os conteúdos estudados.

O estudo individual dos conteúdos relacionados ao desenvolvimento de software ao longo do currículo dos cursos de informática, como, disciplinas de programação, banco de dados, engenharia de software, dificulta aos alunos elaborar uma visão sistêmica do processo de desenvolvimento. Assim, são necessárias estratégias de ensino que possibilitem aos alunos da área de Computação uma maior apropriação dos conteúdos e práticas relacionados ao desenvolvimento de sistemas.

Ao associar o conhecimento teórico e prático, podemos proporcionar aos alunos envolvidos uma vivência da prática de sua futura profissão. Não somente no sentido técnico, mas também social no que diz respeito ao trabalho em equipe, liderança e comprometimento (PINTO *et al.*, 2010).

A escolha pela aplicação deste projeto foi feita devido às referências encontradas em trabalhos, tais como:

- Jaqueira *et al.* (2012), em que se aplicou um projeto multidisciplinar em alunos de graduação, envolvendo as disciplinas de engenharia de software e banco de dados, utilizando a metodologia ágil Scrum e um conjunto de ferramentas de desenvolvimento de software.
- Yamamoto *et al.* (2005), que apresenta vários projetos integrando disciplinas tais como: estrutura de dados II, probabilidade e estatística, matemática discreta, teoria da computação e banco de dados, aplicados em alunos do curso de Ciência da Computação do primeiro, segundo, terceiro e quarto período.
- Alves e Benitti (2006), que apresenta a integração entre disciplinas de engenharia de software em curso de Ciência da Computação, realizado através de aplicação de um processo de software definido, que engloba três disciplinas de engenharia de software, a partir do quinto período.

- Cunha *et al.* (2008), que aborda a metodologia de ensino adotada em um estudo de caso realizado no primeiro semestre de 2008, no curso de pós-graduação em engenharia eletrônica e computação, descrevendo a integração entre disciplinas de projeto de sistemas de banco de dados, tecnologia da informação e teste de software em projetos reais.
- Segura *et al.* (2009), que descreve um estudo de caso aplicado na disciplina de banco de dados I para alunos de bacharelado em ciência da computação, que buscou mostrar como a estratégia de PBL (Problem Based Learning) é suportada por AVA (Ambientes Virtuais Colaborativos) em problemas do mundo real.

Este artigo está estruturado da seguinte forma. Na Seção 2 é apresentada a revisão bibliográfica com algumas das tecnologias adotadas no projeto. A seção 3 apresenta os procedimentos metodológicos de investigação, utilizados na pesquisa. Na Seção 4 são apresentados os resultados obtidos, que consistem nos relatos, lições aprendidas e experiências. Por fim, a Seção 5 traz as conclusões e sugestões para trabalhos futuros.

REVISÃO BIBLIOGRÁFICA

Nesta seção são apresentadas algumas das tecnologias adotadas para o desenvolvimento dos projetos.

JAVA

A linguagem de programação Java tem como principais características ser interpretada, orientada a objetos, com portabilidade e segura, esta foi desenvolvida na década de noventa por uma equipe de programadores chefiada por James Gosling na empresa Sun Microsystems (JAVA, 2015).

A Sun disponibiliza a maioria das distribuições Java gratuitamente e obtém receita com programas mais especializados como o Java Enterprise System. Em 13 de novembro de 2006, a Sun liberou partes do Java como software livre, sob a licença GNU General Public License. A liberação completa do código fonte sob a GPL ocorreu em maio de 2007. Em 2009 a Oracle Corporation adquire a empresa responsável pela linguagem Java, a Sun Microsystems, por US\$ 7,4 bilhões. Com o objetivo de levar o Java e outros produtos da Sun a seu portfólio (JAVA, 2015).

A linguagem de programação Java é à base de praticamente todos os tipos de aplicativos em rede e é o padrão global para desenvolvimento e fornecimento de aplicativos incorporados, jogos, conteúdo on-line e software corporativo. Com mais de nove milhões de desenvolvedores em todo o mundo, o Java permite desenvolver aplicativos e serviços incríveis com eficiência (JAVA, 2015).

A Plataforma Java, Standard Edition (Java SE), permite desenvolver aplicativos seguros, portáteis e de alto desempenho para a maior variedade possível de plataformas de computação. Disponibilizando aplicativos em ambientes heterogêneos, as empresas podem agilizar a produtividade do usuário final, a comunicação e a colaboração, além de reduzir

drasticamente o custo de propriedade de aplicativos tanto de empresas quanto de clientes (JAVA, 2015).

O Java Platform, Enterprise Edition (Java EE), é o padrão do setor para computação Java empresarial. Com novos recursos que melhoram o suporte a HTML5, aumentam a produtividade do desenvolvedor e aprimoram ainda mais a forma de atender às demandas corporativas, o Java EE 7 permite que os desenvolvedores escrevam menos códigos, tenham suporte melhor para os mais recentes aplicativos da *Web* e estruturas e tenham acesso à mais capacidade de expansão e funcionalidades mais avançadas (JAVA, 2015).

REPOSITÓRIO DE CÓDIGO E GERENCIADOR DE VERSÕES

Foi utilizado o repositório de código do Google Code que fornece um ambiente de desenvolvimento colaborativo livre para projetos de código aberto (CODE.GOOGLE, 2015). Combinado com o Subversion que é um sistema de controle de versão de código aberto. Fundado em 2000 por CollabNet, Inc., o projeto Subversion tem um sucesso incrível na última década. Subversion tem desfrutado e continua a desfrutar de ampla adoção, tanto na arena de fonte aberta e no mundo corporativo (SUBVERSION, 2015).

Subversion é desenvolvido como um projeto da Apache Software Foundation, e como tal, é parte de uma rica comunidade de desenvolvedores e usuários. Com estes os acadêmicos puderam controlar as alterações no sistema, através do compartilhamento dos projetos em repositórios que guardam a versão anterior à alteração (SUBVERSION, 2015).

POSTGRESQL

Para desenvolvimento da base de dados foi utilizado o gerenciador de banco de dados postgresSQL que é um poderoso sistema gerenciador de banco de dados objeto-relacional de código aberto. Tem mais de 15 anos de desenvolvimento ativo e uma arquitetura que comprovadamente ganhou forte reputação de confiabilidade, integridade de dados e conformidade a padrões (POSTGRESQL, 2015).

Roda em todos os grandes sistemas operacionais, incluindo GNU/Linux, Unix (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), e MS Windows. É totalmente compatível com ACID, tem suporte completo a chaves estrangeiras, junções (JOINS), visões, gatilhos e procedimentos armazenados (em múltiplas linguagens). Inclui a maior parte dos tipos de dados do ISO SQL:1999, incluindo INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, e TIMESTAMP. Suporta também o armazenamento de objetos binários, incluindo figuras, sons ou vídeos. Possui interfaces nativas de programação para C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, entre outros, e uma excepcional documentação (POSTGRESQL, 2015).

Como um banco de dados de nível corporativo, o PostgreSQL possui funcionalidades sofisticadas como o controle de concorrência multiversionado, recuperação em um ponto no tempo, tablespaces, replicação assíncrona, transações agrupadas (savepoints), cópias de segurança a quente (online/hot backup), um sofisticado planejador de consultas (otimizador) e registrador de transações sequencial (WAL) para tolerância a falhas. Suporta conjuntos de

caracteres internacionais, codificação de caracteres multibyte, Unicode e sua ordenação por localização, sensibilidade a caixa (maiúsculas e minúsculas) e formatação. É altamente escalável, tanto na quantidade enorme de dados que pode gerenciar, quanto no número de usuários concorrentes que pode acomodar (POSTGRESQL, 2015).

PROCEDIMENTOS METODOLÓGICOS

Nesta seção são apresentados os aspectos metodológicos utilizados na condução do estudo.

Participaram da pesquisa 22 (vinte e dois) alunos da disciplina de programação para *web II*, do terceiro ano do curso técnico em informática para internet. Sendo assim, esta pesquisa tem como diferencial em relação a outros estudos semelhantes, envolver alunos do ensino técnico de nível médio da área de informática.

Neste projeto os alunos ficaram livres para escolha das tecnologias, mas foram orientados a utilizar softwares livres e clientes reais. Assim, algumas das tecnologias adotadas no desenvolvimento do projeto foram: Java para o desenvolvimento de sistemas *web*; Google Code como repositório de código; Subversion para gerenciar versões e PostgreSQL para gerenciamento dos dados. A realização do projeto ocorreu no quarto bimestre do terceiro ano. Ao final foi realizada a apresentação formal dos softwares desenvolvidos.

Para acompanhamento dos projetos, foi definido um modelo de relatório, a ser entregue semanalmente. O Professor disponibilizou horário de atendimento semanal e era responsável por: orientar sobre o uso correto das metodologias de desenvolvimento e das tecnologias adotadas e acompanhar o andamento dos projetos, orientando no desenvolvimento das tarefas, de modo a contribuir para o bom andamento dos projetos.

Com o objetivo de avaliar a experiência do projeto multidisciplinar para posteriormente relatar seus pontos positivos e desafios, foram analisados os procedimentos dos alunos em suas práticas no decorrer do projeto, tais como: a participação, o rendimento dos alunos e as dificuldades enfrentadas por ambos.

Foram avaliados pelo professor o andamento e progresso dos projetos e a frequência dos alunos no horário de atendimento. Através dos relatórios semanais o professor da disciplina avaliou se os requisitos do software solicitados tinham sido atendidos.

No final, foi solicitado aos alunos que respondessem a um questionário de avaliação da realização dos seus projetos. O questionário consistia em quatro partes principais: a primeira com os dados do projeto, a segunda com os requisitos do software, a terceira sobre o uso das tecnologias livres, e por ultimo na quarta parte foi apresentado algumas perguntas sobre a opinião deles em relação ao projeto desenvolvido, tais como: O que você achou do trabalho prático? Quais as suas principais dificuldades? O trabalho propiciou um melhor aprendizado? E ainda um espaço para comentários espontâneos, a fim de verificar a percepção destes em relação ao aprendizado adquirido com o projeto.

RESULTADOS OBTIDOS

Esta seção apresenta os, relatos dos alunos, as lições aprendidas, as experiências do ponto de vista dos alunos e professor envolvidos no projeto e alguns dos sites desenvolvidos.

Com relação ao tópico referente aos requisitos do software desenvolvido, todas as respostas afirmaram que o processo foi ruim, complicado e com diversos problemas. É natural que seria encontrado um grande numero de dificuldades inerentes ao processo de engenharia de requisitos, tais como: o fato dos requisitos mudarem de acordo com o aprendizado do cliente a cada validação ou interação com o software. (NUSEIBEH AND EASTERBROOK, 2000). Portanto, a afirmação dos alunos sobre as dificuldades na elicitação dos requisitos e o fato dos mesmos necessitarem de modificação no decorrer do projeto, mostrou-se natural, principalmente para uma equipe de iniciantes.

Algumas das experiências adquiridas com o projeto foram: a não confirmação da utilização de clientes reais conforme orientado, a falta de garantia de que as reuniões com os clientes do software foram efetivamente realizadas. Através de entrevistas com os alunos, percebeu-se que apesar da maioria aparentemente ter utilizado clientes reais, os mesmos não se reuniram com frequência. As razões apresentadas por estes foram desde a falta de tempo do cliente e da equipe, que também foi citado como desafio em Pinto *et al.* (2010), até a compreensão correta de que os requisitos mudam com o tempo.

Percebe-se que talvez em projetos reais o melhor fosse que coubesse ao próprio professor entrar em contato com os clientes reais que tenham disponibilidade de participar de reuniões mensais e que em contrapartida receberão o software gratuitamente. Ou o próprio professor poderia desempenhar o papel de cliente, porém pode acontecer do professor ficar sobrecarregado com as tarefas de cliente e avaliador, além do fato de que os alunos não teriam a experiência de um cliente real.

Quando aos relatos dos alunos em relação à pergunta: O que você achou do trabalho prático? Podemos citar:

“Serviu para um bom aprendizado da matéria, nunca mais vou esquecer a matéria”.

“Tenho muita dificuldade com provas, com o trabalho tivemos uma chance melhor de aprender”.

“Um desafio, com um nível de dificuldade grande, mas uma oportunidade para aprendizagem prática”.

“Gostei muito, faz com que o aluno se esforce e aprenda mais”.

“Foi bastante interessante, pois, foi um aprendizado dinâmico e bem bacana.”.

“O trabalho foi difícil, por ser uma primeira experiência. Mas contribuiu com o aprendizado.”.

“Difícil, porem gratificante”.

“Apesar de ter tido um esforço muito grande, foi proveitoso e rendeu aprendizado.”.

“O trabalho prático propiciou grandes aprendizados”.

“Interessante, pois foi uma maneira de resgatar conhecimentos anteriores e até mesmo o que não havia compreendido anteriormente. Foi bom colocar em prática isso e ver funcionar”.

“Em síntese o trabalho foi produtivo, mesmo sendo tão completo e um tanto complicado, foi importante ter esse trabalho como ferramenta de curso na qual nos possibilitou colocar em prática tudo o que foi estudado nos três anos”.

Quanto à pergunta: Quais as suas principais dificuldades? Relataram que:

“Relembrar as matérias, iniciar o trabalho e saber por onde começar”.

“São muitos detalhes, e o projeto requer conhecimento de outras matérias o que dificultou um pouco”.

“Tive muitas dificuldades no começo, mas depois de vários erros de programação, ficou fácil”.

“A implementação do código é um tanto complexo.”.

“Fazer todo o código linha a linha sem que ocorresse erro.”.

“A utilização do netbeans que travava muito, os diagramas e conteúdos anteriores e o tempo, por conciliar com outras atividades.”.

“As principais dificuldades foram: iniciar, pensar em qual tema e durante a realização do mesmo errei muitas partes.”.

“Principalmente a parte de HTML e CSS”.

“A implementação do carrinho de compras e o login, devido ao fato de não compreender muito bem o conceito de sessão”.

“A principal dificuldade foi relembrar o que foi estudado nos últimos anos”.

A respeito da pergunta: O trabalho propiciou um melhor aprendizado, responderam que:

“Sim, a compreensão da disciplina ficou bem mais divertida”.

“Sim, mas a base de muitos problemas, ansiedade, stress e nervosismo”.

“Sim, com o trabalho conseguimos absorver melhor o conteúdo proposto, coisa que com provas não conseguimos fazer da mesma forma.”.

“Sim, aprendi a ter muita paciência com Java, tive que rever muita coisa que não lembrava”.

“Muito, fui capaz de entender cem vezes melhor que na teoria.”.

“Sim, aprendi muitas coisas novas”.

“Sim, pois houve uma procura por novos códigos, o login, por exemplo.”.

“Sim, pois revemos conteúdos que aprendemos a tempos atrás e querendo ou não somos obrigados a estudar um conteúdo para lembrar, e assim melhoramos o nosso aprendizado”.

“Sim, pois levou agente ao extremo, e fez agente buscar novos conhecimentos”.

“Sim, pois houve troca de conhecimento com os colegas.”.

“Sim, os conteúdos realmente se integram. Fazer um projeto que envolve conteúdo de diversas disciplinas nos possibilita ter uma visão mais aprofundada do processo como um todo.”.

“Sim, pois o projeto nos permitiu ter uma experiência de desenvolvimento de software muito próxima ao cotidiano do mercado de trabalho.”.

“Sim, no final do projeto vimos à dificuldade de se fazer um site legal e agradar o cliente”.

“Sim, ajudou muito, pois foram recuperados os aprendizados de outros anos e outras matérias”.

“Sim, é bom ver seu código funcionar”.

“Sim, foi legal relembrar tudo o que foi estudado e ter colocado em prática, no processo de fazer é um pouco chato, porém depois de feito é satisfatório”.

Estas foram às respostas mais relevantes. Sendo que alguns se reservaram ao direito de não responder todas as perguntas. Houve também um relato, de que a maior dificuldade encontrada, foi a pouca experiência com programação para *Web* e o tempo para realização das atividades, houve ainda uma sugestão de utilização de algum framework de desenvolvimento, de preferencia com ampla utilização e fácil consulta.

Foi relatada também pouca orientação, porém poucos alunos compareceram aos horários de atendimento para tirar dúvidas ou enviaram e-mails. Percebe-se que isto pode ser um indicativo da falta de interesse do aluno em se dedicar ao projeto em horário extraclasse. Isto leva a pensar se não deveria existir uma disciplina onde os alunos teriam uma carga horária, dedicada ao projeto.

Um ponto positivo que vale a pena ser ressaltado é que todos os alunos desenvolveram uma versão executável do projeto e atenderam de 60% a 80% dos requisitos solicitados. Podemos dizer que, o saldo da experiência multidisciplinar foi positivo, pois foram geradas versões funcionais dos softwares.

São apresentados alguns dos sites desenvolvidos com o projeto.

Cacau Construções. O trabalho foi desenvolvido com intuito de fornecer um site de exibição materiais de construção, o site, portanto não tem o objetivo de venda, mas sim de uma exibição dos materiais que se encontrariam em uma loja física. Outra funcionalidade contida no site é a possibilidade de cadastro de clientes da loja e também dos funcionários.

Para o desenvolvimento do site Cacau Construções foram designados alguns requisitos: o desenvolvimento de um Projeto *Web* dotado de no mínimo 3 classes com funções diversas, que

sejam capazes de cadastrar clientes, cadastrar produtos e cadastrar funcionários, o desenvolvimento de JSP's para a interação com o usuário.

Para o desenvolvimento do código referente ao Projeto *WEB* utilizou-se o NetBeans (Linguagem Java). Para construção dos diagramas de Caso de Uso, de Classes e Sequência com Astah Community. Para o armazenamento de dados e administração do SGBD PostgreSQL utilizou-se PgAdmin. A Figura 1 ilustra o site de construção desenvolvido.

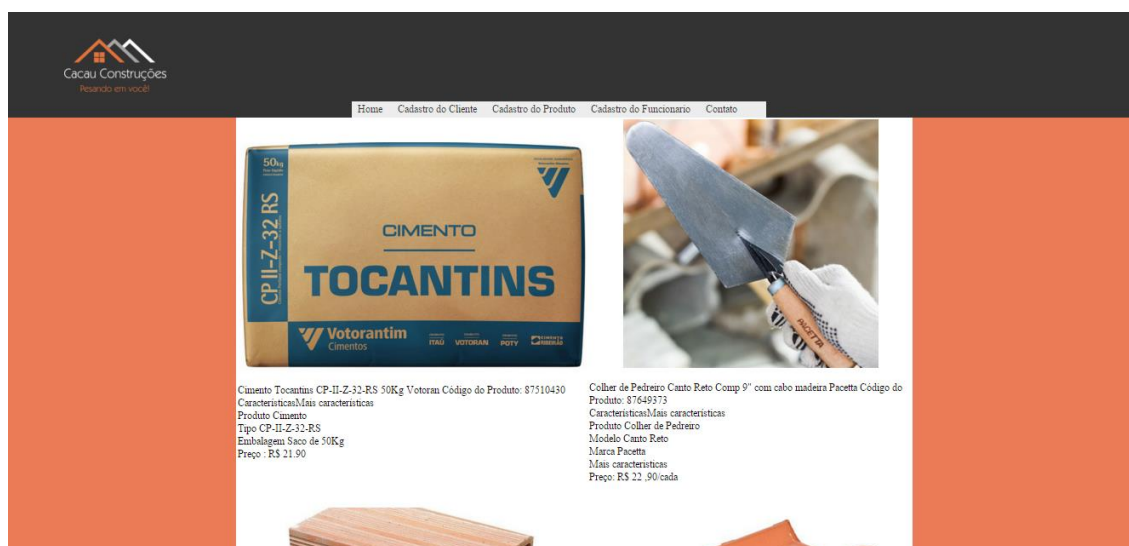


Figura 1: Exemplo de Sistema Web: Site de construção, desenvolvido no projeto

Site IFGAMES. O site funciona como uma loja virtual para a compra de jogos. É um ambiente que permite o cadastro de clientes e a edição dos dados, a visualização dos produtos, além de dispor de um carrinho de compras e funcionalidades exclusivas para os administradores, como ver relatório de vendas, cadastrar e editar produtos, entre outras funcionalidades. A Figura 2 apresenta a pagina inicial do site IFGAMES.

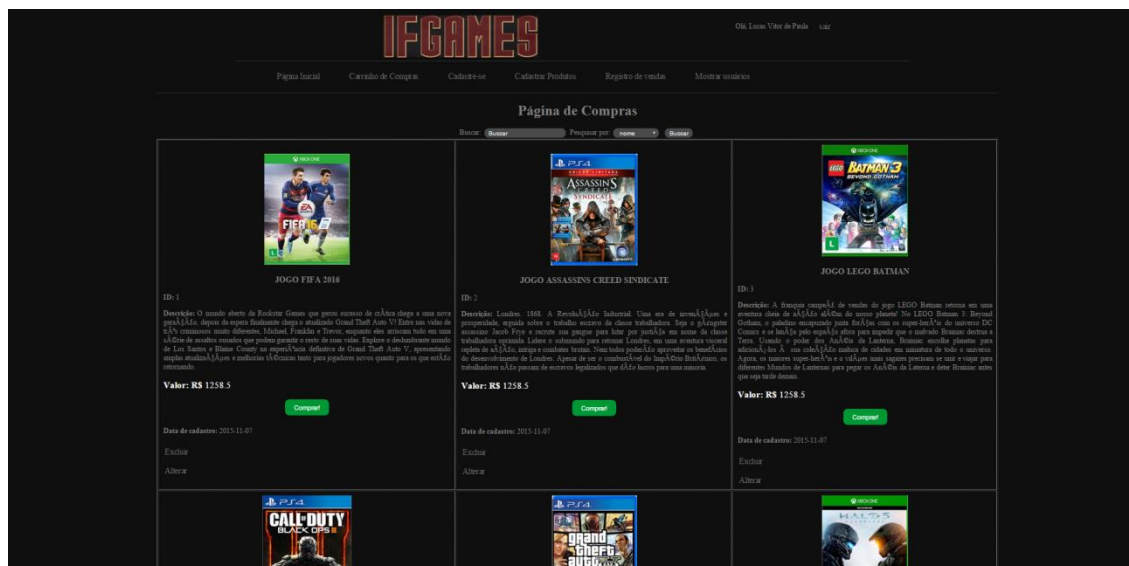


Figura 2: Exemplo de Sistema Web: site IFGAMES, desenvolvido no projeto

Consultório odontológico. O site tem as funções de gravar, excluir, editar e listar. Para o desenvolvimento visual do site, foi usado o Bootstrap onde são disponibilizados exemplos. Os códigos extraídos desse site são em html, e é preciso a conexão com a internet para que sejam atribuídos recursos de um biblioteca, e as ações sejam executadas. A Figura 3 ilustra o site de consultório odontológico desenvolvido.



Figura 3: Exemplo de Sistema Web: de consultório odontológico, desenvolvido no projeto

WQD (QUESTÕES). O presente projeto tratase de de um sistema feito em Java. Possui um servidor (Glassfish), e um banco de dados (PostgreSQL). Esse sistema foi feito pensando nas pessoas que gostam de estudar respondendo questões. Nele, há usuários, os quais podem criar suas próprias contas e inserir questões no banco do sistema. Além disso, eles podem alterar seus dados e, se for preciso, excluir suas contas. Isso acontece também com as questões, que podem ser corrigidas e excluídas. Por fim, na página principal, aparecem as questões que podem ser respondidas quantas vezes for necessário. A Figura 4 apresenta a pagina de responder questões do site WQD (QUESTÕES).

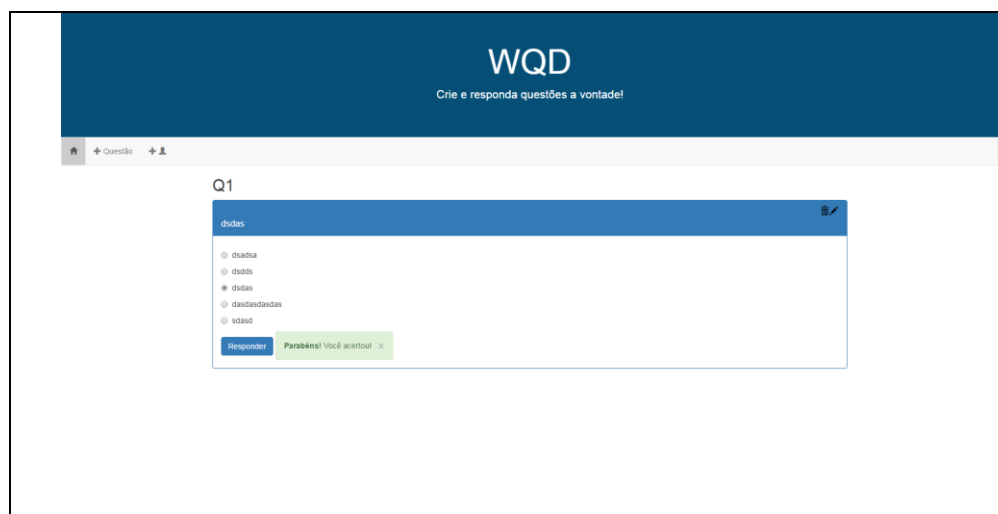


Figura 4: Exemplo de Sistema Web: de responder questões, desenvolvido no projeto

Portal da Corrupção. Esse projeto foi nomeado por Portal da Corrupção, foi pensado para o cadastro de eleitores e mostrar para as pessoas políticos que tenham problemas de corrupção. Nele será possível se cadastrar como eleitor, e buscar nas abas; Ficha Suja, Políticos e Escândalos. A Figura 5 ilustra o site Portal da Corrupção desenvolvido.



Figura 5: Exemplo de Sistema Web: Portal da Corrupção, desenvolvido no projeto

Seus Sabores. Trata-se de um site interativo onde usuários e administradores são responsáveis por tal. Podendo então contribuir com receitas para incrementar o site, alterar e excluir.

Como se trata de um sistema de receitas culinárias, foi preciso organizar a forma do conteúdo de uma forma a facilitar o uso por quem acessa, estabelecendo os atributos como ingredientes, modo de preparo, tempo de preparo, rendimento, grau de dificuldade, um código e identificação e o tipo.

Depois se se elaborou os diagramas que orientam na criação do código, como os diagramas de caso de uso para definir os aspectos obrigatórios pré-definidos no programa de instrução do trabalho, como inserir, alterar, excluir e listar.

Há uma página para cada função, listar, cadastrar, alterar e a página de erro. Todas devidamente trabalhadas com a interface do site, que é simples, porém atende a necessidade, com o uso de HTML e CSS. Com disposição de imagens e formulários, além do login. A Figura 6 apresenta a pagina de login do site Seus Sabores.



Figura 6: Exemplo de Sistema Web: site Seus Sabores, desenvolvido no projeto

Patinhas. É um sistema que tem como objetivo registrar, ou seja, inserir e manipular, dados referentes à adoção, apadrinhamento e reportagem de animais encontrados que necessitam ser encaminhados há um local para acomodação deste.

Nele é possível realizar o cadastramento de alguma adoção, algum animal encontrado ou apadrinhamento, busca e listagem destes através da página inicial. É possível também, através da página de listagem, modificar ou excluir esses registros. A Figura 7 apresenta a pagina inicial do site Patinhas.

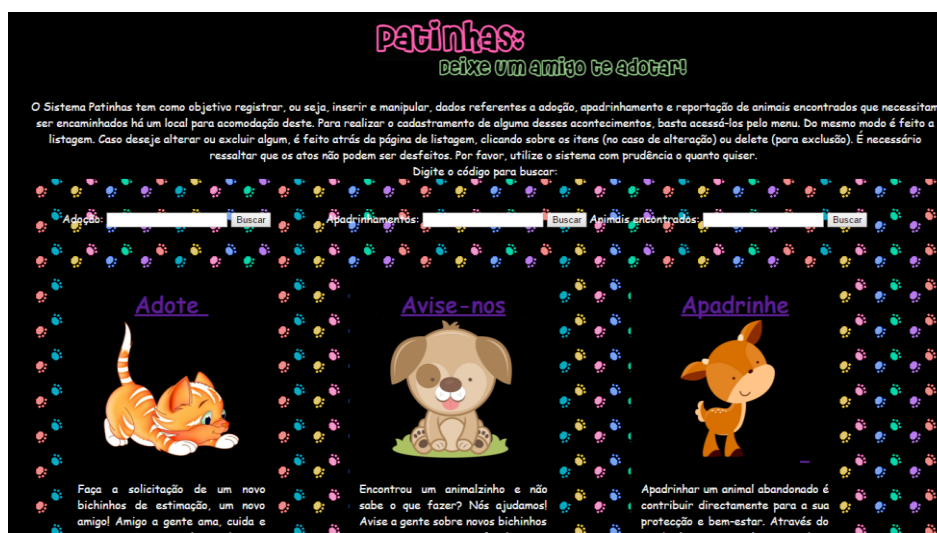


Figura 7: Exemplo de Sistema Web: site Patinhas, desenvolvido no projeto

Percebe-se que o interesse dos alunos foi mais acentuado para aqueles que desenvolveram projetos inovadores e tinham inclusive vontade de colocar o mesmo em produção para os clientes. Como uma forma de motivar e estimular os alunos cabe ao professor elaborar propostas de projetos mais inovadoras e com o uso de novas tecnologias.

Outra observação relevante é que o conteúdo da disciplina foi prejudicado em virtude das orientações e apresentações dos projetos. Se por um lado projetos práticos auxiliam na fixação do conteúdo das disciplinas por outro lado tais projetos consomem muito tempo para acompanhamento e execução. No caso, esse tempo precisa ser retirado do espaço dedicado ao conteúdo da disciplina. Esta observação reforça a necessidade de uma disciplina dedicada apenas ao projeto.

CONSIDERAÇÕES FINAIS

Este artigo apresentou um relato de experiência da aplicação de um projeto prático multidisciplinar desenvolvido na disciplina de Programação para *Web II* envolvendo conteúdos das disciplinas de Engenharia de Software e Banco de Dados de um curso técnico em informática para internet.

A execução do projeto multidisciplinar exigiu muito dos alunos, pois precisaram aprender novos conceitos, rever conceitos utilizados anteriormente e aplicar ao mesmo tempo estes no desenvolvimento do trabalho.

Na prática diversos conceitos e práticas puderam ser trabalhados, levando alguns participantes a terem o primeiro contato real ou aprimorar os conhecimentos que tinham, tais como:

- Utilização de práticas de metodologias ágeis;
- Gerência de configuração: nomenclatura de documentos, controle de versão, backup;
- Gerência de projeto: planejamento, estimativas, distribuição e acompanhamento de tarefas, e utilização de ferramenta para gestão;
- Gerência de qualidade de software: definição de um processo com padrões e procedimentos, e definição de métricas;
- Plano de teste de software: planejamento e execução de testes, definição de casos de testes, ferramentas de gestão de mudanças;
- Gerência de Comunicação: definição e padronização dos diferentes meios de comunicação e suas formas de utilização.

Concluiu-se a partir desta experiência a necessidade de uma disciplina exclusiva para a prática do projeto, onde os alunos poderiam se dedicar e ter um tempo maior para aulas destinadas ao uso das ferramentas e à realização do trabalho.

Outra constatação relevante foi que a prática autodidata pode ser desenvolvida nos alunos através deste tipo de projeto, ampliando o caráter investigativo nos alunos, assim como, o interesse em procurar ajuda.

Diversas outras conclusões podem ser extraídas desse projeto, tais como: o cliente nem sempre estará presente para acompanhar a evolução e dar seu *feedback*, é preciso elaborar um cronograma de execução, o processo deve garantir padrões e a qualidade do produto, entre outros.

Porém, a principal conclusão que se pode chegar é que, a aplicação prática, em paralelo ao ensino da teoria, solidifica fortemente o aprendizado e visa garantir um maior aproveitamento dos conceitos teóricos.

A contribuição teórica do presente estudo assume uma maior relevância, principalmente, ao diferenciar-se de outros estudos pesquisados, ao trazer em foco um projeto prático multidisciplinar aplicado em alunos de um curso técnico de nível médio com faixa etária entre 14 a 18 anos.

Como sugestão para trabalhos futuros, propõe-se aplicar o projeto em uma disciplina exclusiva de modo a verificar se as dificuldades aqui apresentadas serão superadas, e comparar os resultados encontrados nesse projeto a outros desenvolvidos com alunos de nível superior.

REFERÊNCIAS

ALVES, A. G.; BENITTI, F. B. V. **Processo de Desenvolvimento Integrando Disciplinas de Engenharia de Software**. Anais do XXVI Congresso da SBC. Campo Grande, MS, 2006.

CODE.GOOGLE. **Code.Google**. <https://code.google.com/> Acesso em Outubro de 2015.

CUNHA, A. M.; BRAGA E SILVA, G.; MONTE-MOR, J. A.; DOMICIANO, M. A.P.; VIEIRA, R. G. **Estudo de Caso Abrangendo o Ensino Interdisciplinar de engenharia de Software**. Fórum de Educação em Engenharia de Software, 2008.

JAQUEIRA, A.; COELHO, R.; LUCENA, M. **Conduzindo Projetos Ágeis em um Contexto Multidisciplinar: Um Relato de Experiência**. Fórum de Ensino de Engenharia de Software (FEES). Congresso Brasileiro de Software (CBSOFT), 2012.

JAVA. **Java**. <http://www.oracle.com/br/java/overview/index.html> Acesso em Outubro de 2015.

LAFFEY, J. M., MUSSER, D., REMIDEZ, H., AND GOTTDENKER, J. **Networked systems for schools that learn**. Communications of the ACM, 46(9):192–200, 2003.

LÊ, T. **Collaborate to learn and learn to collaborate**. In Proceedings of the Seventh world conference on computers in education, pages 67–70, Copenhagen, Denmark. ACM, 2002.

LIMA, G. A. N., AGUIAR, Y. P. C., LULA, B. LULA JR. **Impacto do Apoio Metodológico Ferramental a Aspectos de Usabilidade no Ensino Prático da Engenharia de Software**. In XVII WEI – XXIX Congresso da Sociedade Brasileira de Computação, Bento Gonçalves, 2009.

MORIN, E. **A cabeça bem feita: repensar a reforma, reformar o pensamento**. 6. ed. Rio de Janeiro : Bertrand Brasil, 2002.



NUSEIBEH, B. EASTERBROOK, S. **Requirements Engineering: A Roadmap**. Proceedings of International Conference on Software Engineering (ICSE-2000), Limerick, Ireland, ACM Press, 2000.

PINTO, C. L. Q., ROCHA, C. R. C., VILARIM, G. **Desafios da Prática da Interdisciplinaridade em Cursos de Ciência da Computação: a Experiência do UNIFESO**. In XVIII WEI – XXX Congresso da Sociedade Brasileira de Computação, Belo Horizonte, 2010.

POSTGRESQL. **Postgresql**. <http://www.postgresql.org.br/old/sobre> Acesso em Outubro de 2015.

SEGURA, R. A.; ARAÚJO JÚNIOR, C. F.; SILVEIRA, I. F. **Aprendizagem Baseada em Problemas Apoiada por Ambientes Virtuais: um Estudo de Caso em Banco de Dados**. Anais do XXIX Congresso da SBC, Bento Gonçalves, RS, 2009.

SUBVERSION. **Subversion**. <https://subversion.apache.org/> Acesso em Outubro de 2015.

YAMAMOTO, F. S.; SILVA, A. F.; ZANUTTO, J.; ZAMPIROLI, F. A. **Interdisciplinaridade no Ensino de Ciência da Computação**. Anais do XXV Congresso da SBC, Unisinos, São Leopoldo, RS, 2005.