

Quando a Teoria é Aplicada: Um Relato de Experiência da Aplicação de Metodologias de Desenvolvimento de Sistemas Computacionais dentro do PETComp-UFMA

Italo Dovera¹, Matheus Bessa¹, Brenno Nascimento¹, Felipe Silva¹,
Rafaela Gomes², Geraldo Braz Junior³, Adriana Nogueira² e Luis Rivero^{1,3}

¹Programa de Educação Tutorial do curso de Ciência da Computação (PETComp)
Universidade Federal do Maranhão (UFMA) São Luís - MA - Brasil

²Programa de Pós-Graduação em Saúde e Tecnologia (PPGST),
Universidade Federal do Maranhão (UFMA) – São Luís, MA – Brasil

³Programa de Pós-Graduação em Ciência da Computação (PPGCC),
Universidade Federal do Maranhão (UFMA) – São Luís, MA – Brasil

{italo.luigi, matheus.levy, brenno.izaias, felipe.henrique,
rca.gomes}@discente.ufma.br, {geraldobraz, adriana.nogueira,
luis.rivero}@ufma.br

Abstract. *In computing, there is a fast and constant advance of new software development tools, which leads professionals in the area to also constantly improve their knowledge of these new tools. However, this rapid evolution makes it difficult for development teams to choose an efficient methodology. In this context, this article aims to report the experience of developing systems, in addition to presenting tools and methodologies that help groups beginning in this process. The team of the Computer Tutorial Education Program (PETComp) at the Federal University of Maranhão defined a training and development process, with functions divided between development and design teams. In all, a team of students with low experience in development carried out activities such as: prototyping, validation meetings and learning of tools. As a result, websites for events, websites for undergraduate courses and mobile applications were created.*

Resumo. *Na computação, existe um avanço rápido e constante de novas ferramentas de desenvolvimento de software, o que leva os profissionais da área ao aprimoramento também constante de seus conhecimentos sobre essas novas ferramentas. Porém, essa rápida evolução dificulta a escolha de uma metodologia eficiente aos times de desenvolvimento. Nesse contexto, este artigo visa relatar a experiência de desenvolver sistemas, além de apresentar ferramentas e metodologias que auxiliem grupos iniciantes em tal processo. A equipe do Programa de Educação Tutorial em Computação (PETComp) da Universidade Federal do Maranhão definiu um processo de treinamento e desenvolvimento, com funções divididas entre equipes de desenvolvimento e design. Ao todo, uma equipe de discentes com pouca experiência em desenvolvimento realizou atividades como: prototipação, reuniões de validação e aprendizagem de ferramentas. Como resultado, obteve-se a criação de sites para eventos, sites para o curso de graduação e aplicativos móveis.*

1. Introdução

No desenvolvimento de software, o trabalho em grupo é essencial tanto no mercado de trabalho quanto no âmbito acadêmico. As empresas da área, como as *startups*, aplicam métodos ágeis para se adaptar melhor as estratégias de negócios assim gerenciando melhor seus times e tarefas [Giardino et al. 2014]. Para que esse trabalho seja desenvolvido de forma fluida e que seja entregue uma aplicação de qualidade, as equipes de desenvolvimento seguem práticas de gerenciamento de projetos e equipes tais como *SCRUM*, *BOPE* e *Extreme Programming* dividindo a equipe e as tarefas [Nunes 2017]. No entanto, existem fatores que podem dificultar a adoção de processos de software adequados, principalmente para grupos constituídos por pessoas inexperientes no processo de desenvolvimento. Fatores como dificuldades técnicas ou resistência da equipe podem levar o projeto ao fracasso [Dias Jr et al. 2014]. Outro fator que pode apresentar desafios é o trabalho remoto, visto que dentro da realidade de cada integrante, não é possível confirmar se todos têm as tecnologias necessárias para desenvolver um sistema. Nesse contexto remoto, aspectos como a falta de comunicação efetiva, a falta de confiança e a dificuldade de coordenar um time são os principais agravantes que podem prejudicar o projeto [Shameem et al. 2017].

Considerando o contexto acima, este artigo visa relatar a experiência da equipe de desenvolvimento do Programa de Educação Tutorial (PET) do curso de Ciência da Computação (PETComp) da Universidade Federal do Maranhão (UFMA). O PET tem se tornado um destaque devido a sua abrangência e potencial formador. Segundo o Ministério da Educação (MEC), o PET é um grupo formado por discentes com tutoria de um docente que tem por base o tripe da universidade: ensino, pesquisa e extensão. Esse programa viabiliza o desenvolvimento de atividades extracurriculares, ajudando na formação acadêmica dos envolvidos, e melhorando a qualidade acadêmica dos cursos de graduação a que ele está direcionado. Nesse contexto, serão descritas as atividades do PETComp/UFMA durante a realização de projetos de desenvolvimento, bem como abordar as relações entre o grupo e apresentar as dificuldades e desafios encontrados no decorrer das atividades. Também serão expostas ferramentas e metodologias de desenvolvimento facilitadoras utilizadas, que se mostraram essenciais para o desenvolvimento eficaz das funções requisitadas no processo de desenvolvimento. Isso para que outros grupos de profissionais na área de computação iniciantes na criação de sistemas possam seguir um processo de software que ajude o andamento do projeto e tente diminuir as dificuldades encontradas por eles.

O restante deste trabalho está organizado como segue. Na Seção 2 serão abordados os trabalhos relacionados. Na Seção 3 será explicada a metodologia utilizada neste trabalho. Por sua vez, na Seção 4 serão apresentados os resultados. A discussão dos resultados será feita na Seção 5. Por fim, a conclusão e trabalhos futuros são apresentados na Seção 6.

2. Trabalhos Relacionados

Aragão e Andrade (2017) buscaram fazer uma melhoria na modelagem do processo de desenvolvimento de uma equipe já bem definida, pois notou-se um distanciamento entre a versão do processo documentada da versão usada na prática. O projeto realizado em parceria entre uma universidade federal brasileira e uma fabricante de celulares, passou

por um procedimento constituído por entrevistas para coletar *feedback* da equipe sobre o trabalho realizado, além de observação das atividades e processos executados pela mesma. Como resultado, percebeu-se que havia divergências entre os membros a respeito dos conceitos sobre as atividades, tal fato decorria da variância de conhecimento por parte dos integrantes. Além disso, foi criado um modelo que permitiu uma visualização geral mais fiel a respeito das funções, além de serem apresentadas as lições aprendidas no decorrer do processo.

No contexto de desenvolvimento de software, Dias et al. (2014) realizaram um estudo de caso que abordou a compreensão de duas equipes a respeito do processo de desenvolvimento utilizado numa empresa pública de tecnologia a qual estavam inseridos. Desta forma, através de uma pesquisa qualitativa, pôde-se medir o nível de entendimento dos grupos quanto aos procedimentos definidos pela empresa. Os resultados evidenciaram problemas principalmente relacionados a aspectos da aprendizagem e utilização do processo, visto que mesmo com uma documentação sobre os processos disponíveis, percebeu-se que os recursos necessários para aprender sobre o projeto são insuficientes, gerando a falta ou o entendimento parcial. Tal aspecto mostra-se como obstáculo ao desdobramento eficaz de qualquer projeto a ser realizado pela equipe.

Mudando um pouco o contexto, Ferreira et al. (2020) buscaram relatar a experiência dentro do Instituto Federal Goiano, campus Iporá, apresentando objetivos de conforme as necessidades e demandas socioeconômicas locais, visando formar profissionais capacitados e inseridos num ambiente de integração entre ensino, pesquisa e extensão no decorrer da formação acadêmica. Nesse cenário, são apresentados softwares desenvolvidos no curso de tecnologia em análise e desenvolvimento de sistemas, assim como são expostos os conhecimentos necessários e relacionados às disciplinas cumpridas no curso. O artigo constatou que os discentes obtiveram sucesso em aplicar os conhecimentos obtidos no decorrer de suas formações, produzindo softwares satisfatórios.

Aproximando-se mais do contexto em que este artigo está inserido, Pereira et al. (2013) consideram os desafios que envolvem o desenvolvimento de software *open source* em uma universidade pública brasileira. Baseados em análises de dados que sugerem que um processo de testes bem definido propicia o desenvolvimento de softwares de boa qualidade em laboratórios acadêmicos, o artigo apresenta um estudo de caso. No estudo, são analisados 12 *softwares* desenvolvidos em um período de 3 anos por estudantes graduandos e graduados, com a supervisão de professores. Através da implantação de um processo denominado BOPE, que mistura a utilização do *Scrum*, *Extreme Programming* e PMBoK, concluiu-se que times formados em sua maior parte por graduandos conseguem desenvolver e manter softwares inovadores de longa duração.

Constata-se que existe uma série de trabalhos que abrangem sistemas desenvolvidos para áreas distintas, porém, os autores deste trabalho não identificaram outro trabalho que vise apresentar uma visão geral do impacto das atividades de desenvolvimento no contexto do programa PET na área de computação, considerando uma metodologia de treinamento durante o processo de desenvolvimento. A análise, desse contexto, permitiria analisar a percepção de pessoas inexperientes em meio a um período de isolamento social, sendo esses indivíduos discentes de Ciência da Computação desenvolvendo sistemas *web* e *mobile*. Desta forma, este artigo visa abordar dificuldades que também podem ser encontradas por outras pessoas ao iniciarem neste campo, com ênfase nos meios de

contornar e solucionar tais adversidades.

3. Metodologia

Realizados de 2020 ao início de 2022, os projetos de desenvolvimento foram executados com base em reuniões internas e externas aos integrantes do PET, objetivando desenvolver as funcionalidades conforme o esperado, estabelecer instruções aos responsáveis pelo avanço das atividades e avaliar o progresso dos aperfeiçoamentos demandados. Contando com um tutor orientador e alunos de graduação, as funções eram divididas entre o grupo de design, encarregado de desenvolver a prototipação através do preparo das imagens e interfaces, e o grupo de desenvolvimento, responsável diretamente pela codificação e implantação da aplicação. A seguir é detalhado o processo seguido para que os discentes aplicassem os conceitos vistos durante as disciplinas do curso no contexto de projetos de desenvolvimento realizados no âmbito do programa PET.

3.1. Perfil dos Discentes

Os discentes que ingressam o grupo PETComp normalmente possuem pouca ou nenhuma experiência com um ambiente de desenvolvimento profissional, ou mesmo programação. A partir dos dados coletados nas inscrições de seletivo no grupo, têm-se as Tabelas 1 e 2. Como é possível ver nas tabelas, os novos membros do grupo são majoritariamente recém-ingressos na graduação.

Alguns desses ingressantes, quando demonstram interesse na área de desenvolvimento no processo seletivo do PETComp, são desafiados a desenvolver um sistema. Tal sistema e requisitos mínimos são definidos pela comissão organizadora e apresentados para os desafiados. Para que todos possuam o conhecimento necessário para o prosseguimento da atividade, uma série de minicursos são ministrados pelos avaliadores abordando a linguagem utilizada como também os *frameworks*. Essa atividade tem como propósito avaliar as habilidades técnicas e capacidade de resolução dos candidatos, como também capacitar os possíveis novos membros e desenvolvedores do grupo.

Tabela 1. Idade de Ingresso dos Discentes

Idade	Frequência	Porcentagem
17	3	9,68%
18	6	19,35%
19	7	22,58%
20	4	12,90%
21	2	6,45%
22	7	22,58%
25	1	3,23%
30	1	3,23%

3.2. Definição e Treinamento

O grupo PETComp dispõe de várias equipes internas que fazem determinadas atividades e com pessoas que revezam tais atividades entre si. Para a atividade de desenvolvimento não é diferente, entram pessoas com vários níveis de experiência e juntos desenvolvem

Tabela 2. Período de Ingresso dos Discentes

Período	Frequência	Porcentagem
2	19	61,29%
3	9	29,03%
4	2	6,45%
5	1	3,23%

sistemas computacionais visando a garantia da qualidade. Considerando que foram realizados diferentes projetos em plataformas *web* e *mobile* no período de 2020 até o início de 2022, a seguir serão apresentadas algumas formas de organização das etapas que compuseram o decorrer das atividades.

Antes da equipe de desenvolvimento iniciar a sua função era necessário que houvesse, no mínimo, um esboço visual da estrutura do projeto, sendo o ideal obter-se, como resultado, uma representação fiel quanto à aparência geral do sistema a ser desenvolvido. Essa tarefa primordial iniciou-se em reuniões de exposição das ideias dos clientes, sendo desenvolvidas em conjunto com a equipe de *design*, almejando uma aparência condizente com a proposta, mas, ao mesmo tempo, viável de ser implementada pelos desenvolvedores. Tal etapa mostra-se essencial, pois visa esclarecer a disposição das funcionalidades, além de representar a identidade visual do projeto, aspecto fundamental para oferecer uma experiência agradável aos usuários.

Introduzindo a próxima etapa, o primeiro passo para desenvolver um software de qualidade é o domínio da sua ferramenta. O grupo dispõe de minicursos e treinamentos esporádicos de diversos *frameworks* que facilitam o trabalho de jovens desenvolvedores, como o *Wordpress*, *Expo*, *React Native*, etc. Durante o processo de desenvolvimento de um software no grupo PETComp da UFMA, primeiramente era necessário que todos os integrantes tivessem o mínimo de conhecimento sobre o *framework* que seria utilizado, como também sua linguagem. O acervo de minicursos e treinamentos esporádicos sobre determinadas tecnologias garantem que todos tenham oportunidades iguais de ter o conhecimento necessário para o bom andamento do projeto. A Figura 1 mostra um digrama ilustrando as etapas da metodologia de desenvolvimento de um sistema.

3.3. Levantamento de Requisitos

A primeira etapa do processo de desenvolvimento é o levantamento de requisitos, pois se não houver entendimento das especificidades do sistema, mesmo com uma boa codificação do sistema, as necessidades do cliente não serão atendidas [Chichinelli 2017]. Portanto, para evitar que o sistema tenha que ser refeito várias vezes durante seu desenvolvimento, a primeira etapa é definida com levantamento dos requisitos.

O levantamento é feito em uma ou mais reuniões virtuais entre o tutor e o cliente, nas quais são discutidas as necessidades e objetivos do cliente. Nesta etapa fica a cargo somente do tutor gerar os documentos necessários, tanto para o desenvolvimento quanto para a prototipação do sistema, e repassá-los as equipes de *design* e desenvolvimento. A próxima etapa será a geração dos protótipos de telas do sistema.

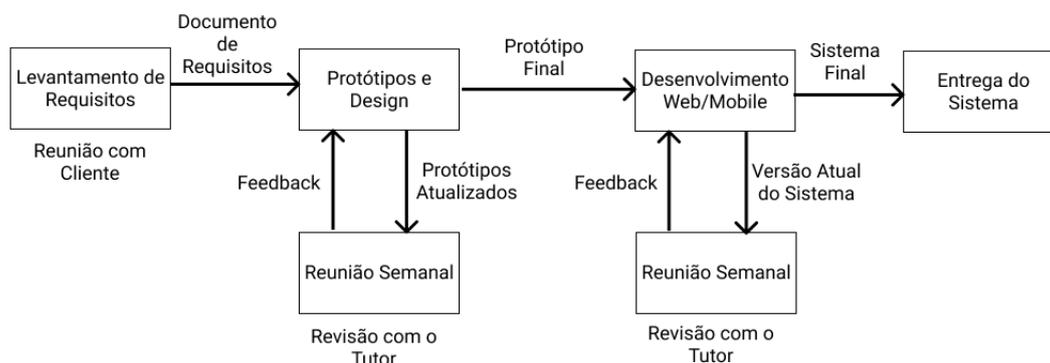


Figura 1. Fluxo de Documentos

3.4. Protótipos e Design

Após o levantamento de requisitos, a etapa seguinte é o planejamento de suas características visuais e funcionais. O *design* tem como uma de suas principais funções tornar o produto convidativo, de maneira a atrair e satisfazer o usuário.

Diante dos documentos gerados pelo tutor após a reunião com o cliente de maneira virtual, a equipe tem a função inicial de coletar tudo aquilo que o contratante deseja que tenha em seu produto, trabalhando em cima dessas informações, avaliando se são sugestões coerentes com a ideia do projeto e também apresentando novas ideias. Com as informações em mãos, foi utilizada uma ferramenta de prototipagem chamada *Figma* para montar o “esqueleto” do aplicativo, tornando possível a visualização e alteração do produto.

Com o primeiro protótipo finalizado, o próximo passo foi analisar a paleta de cores que seria utilizada na aplicação. As cores foram escolhidas conforme a proposta definida. Por exemplo, no caso do aplicativo desenvolvido Mamaprev, por ser um aplicativo de orientação e que envolvia o aspecto maternal, foram escolhidos tons mais claros com um contraste mais suave, trazendo a sensação de conforto ao usuário.

Tendo a primeira versão do protótipo e a paleta definidas, é chegada a hora de decidir o modelo de arte a ser utilizado. Logotipos, imagens, menus, ícones e etc. devem ser produzidos seguindo um mesmo conceito para manter a linearidade do aplicativo. As artes foram produzidas através das ferramentas *Inkscape*, este sendo gratuito, e *Adobe Illustrator* por conta de sua facilidade de trabalhar com vetorização, além de facilitar possíveis alterações que possam ser solicitadas nos arquivos.

Finalizadas as artes, construiu-se um novo protótipo já com cores, ícones e imagens inseridas, apresentando ao cliente exatamente como seria o resultado do produto. Por fim, foi utilizada a ferramenta *Google Drive* para compartilhar protótipos e artes com a equipe responsável pelo desenvolvimento do aplicativo, indicando o modelo de aplicação a ser produzido.

3.5. Desenvolvimento *Mobile*

Após o processo de entendimento básico mencionado anteriormente, no âmbito de desenvolvimento *mobile*, foi utilizado o *React Native*, biblioteca da linguagem *JavaScript* utilizada para desenvolver sistemas *iOS* e *Android*. Além disso, fez-se necessário a utilização do *Expo*, ferramenta que facilita a visualização do seu código em uma tela. Tal recurso pode ser usufruído tanto baixando o aplicativo em um dispositivo móvel, quanto emulando a utilização do *Android/iOS* em seu computador, porém, vale ressaltar que este segundo método possui requisitos de hardware consideráveis, podendo requisitar bastante da memória RAM de sua máquina, portanto é necessário avaliar a situação e optar pelo artifício mais adequado.

No período da atividade de desenvolvimento, ocorreu a implementação de dois aplicativos específicos, o Mamaprev e o Aplicativo de Doenças Clínicas, que podem ser visualizados na Figura 2. A etapa de desenvolvimento do Mamaprev foi de 1 ano no total. O aplicativo teve um desenvolvimento inicial com a entrega planejada e um segundo para correções e aprimoramentos pedidos pelos clientes. No total 3 desenvolvedores participaram do projeto, com 2 trabalhando concomitantemente nas duas fases. Para o Aplicativo de Doenças Clínicas, o período de desenvolvimento foi de 2 meses, com um total de 4 desenvolvedores trabalhando nele simultaneamente.

No âmbito do planejamento e divisão de tarefas, foi utilizado um quadro *Kanban* junto ao aplicativo *Trello*. No grupo essas ferramentas se tornam importantes para a diminuição do retrabalho e aumento da comunicação efetiva, tornando claro quem é responsável por quais tarefas. Também, auxilia na documentação do código junto ao *Github*, permitindo checar quais alterações ocorreram no projeto, quando e quem é o autor delas.



Figura 2. Telas do Aplicativo do Mamaprev e Doenças Clínicas.

Voltando ao cenário do processo de desenvolvimento dos petianos, a comunicação durante o progresso foi facilitada através do programa *Discord*. Disponível tanto para *web* quanto *mobile*, a aplicação é composta por servidores, possibilitando chamadas de voz e/ou vídeo em diferentes salas, além de possibilitar conversas por texto, compartilhamento

de tela, arquivos e *links*. Tal ferramenta proporcionou uma comunicação diária entre os integrantes da equipe de desenvolvimento, aspecto fundamental para um desenvolvimento eficiente que se revelou dificultado por conta período de isolamento social.

O processo de avaliação, aprovação e solicitação de novas demandas se dava por reuniões com os clientes, sendo estas intermediadas pelo tutor orientando do grupo PET de computação. Tais encontros eram realizados pela plataforma *Google Meet*, que facilita a comunicação através de chamadas realizadas por voz e/ou vídeo, além de possibilitar a apresentação da tela do computador, proporcionando uma experiência eficaz de correção e acréscimo de funcionalidades. O resultado de tais reuniões eram documentados e apresentados de maneira instrucional à equipe de desenvolvimento, favorecendo a evolução do projeto.

3.6. Desenvolvimento Web

O processo de desenvolvimento foi baseado em protótipos feitos pela própria equipe de desenvolvimento, eles eram gerados a partir de um editor gráfico, o *Figma*. O *framework* de desenvolvimento foi definido concomitantemente à criação dos protótipos para que assim haja tempo para treinar novos integrantes para a equipe de desenvolvimento. Após a aprovação e validação do protótipo pelo tutor do grupo, a construção do site iniciou.

Foi designado um líder para o desenvolvimento de cada sistema, e esse líder ficaria encarregado de passar ao tutor, e ao restante do grupo PET, atualizações do processo. Assumiu a posição de liderança quem possuía alguma experiência com desenvolvimento, mesmo que não tivesse experiência com determinado *framework*. Coube ao líder da atividade dar suporte e verificar o andamento dos integrantes.

A equipe fez uso do *GIT*, sistema de controle de versões, para não correr o risco de perder toda a produção por conta de algum erro da máquina ou do próprio desenvolvedor, dado que existiam integrantes inexperientes na equipe. Baseados no protótipo toda a equipe se reuniu para fazer a página inicial e depois de pronta, as outras páginas eram distribuídas. Cada integrante desenvolveu uma página e dependendo de sua complexidade, ao acabá-la, o mesmo foi alocado a outra página para desenvolver.

O tutor do grupo verificou semanalmente o desenvolvimento e, quando pronto, avaliou o site com a equipe responsável pelo mesmo. Caso houvesse alguma modificação, ela era feita e apresentada na semana seguinte durante a reunião. Caso estivesse pronto, o tutor validou o site e o mesmo foi disponibilizado para o público. A Figura 3 mostra um site desenvolvido pelo PETComp para a recepção de calouros do período de 2020.2.

3.7. Entrega do Sistema

Após o sistema ter sido desenvolvido e revisado, pela equipe de desenvolvimento e pelo tutor, ele vai para a fase final de validação pelo cliente, na qual o sistema é disponibilizado para teste. No caso dos sistemas *web* o site é disposto em um servidor de hospedagem *web* e para os sistemas *mobile* são disponibilizados através de duas formas, dependendo da necessidade do cliente. Em sistemas *Android* é utilizado um *Android Application Pack (APK)*, o qual permite a instalação do sistema. Para sistemas *iOS* a instalação é feito por intermédio do *Expo*, uma ferramenta utilizada no desenvolvimento *mobile* com *React Native* que permite o fácil acesso às *API's* nativas do dispositivo sem precisar instalar qualquer dependência ou alterar código nativo. O cliente instala o aplicativo *Expo* em seu



Figura 3. Página inicial de um site desenvolvido para a organização de um evento.

dispositivo celular e a equipe de desenvolvimento disponibiliza o sistema em um servidor *Expo*. Este fornece um *QR Code* para instalação do sistema. A equipe de desenvolvimento então repassa para o cliente este *QR Code* para ele escanear e assim é feita a instalação do sistema no dispositivo celular do cliente. Isto é feito pois não é possível gerar um instalador para sistema *iOS*.

O cliente após ter acesso ao sistema agora pode fazer todos testes desejados. Além disso, caso alguma falha tenha sido ignorada em alguma etapa do processo de desenvolvimento e for identificada pelo cliente, este pode solicitar correções. E assim é feito o refinamento do sistema.

4. Resultados

Os projetos a partir de vários ciclos são aprimorados e desenvolvidos. Em seguida são detalhados esses ciclos.

4.1. Composição de Documentos de Requisitos

A partir de uma demanda como a organização de um evento ou de uma requisição de desenvolvimento por terceiros, é levantada a proposta do sistema pelo tutor do grupo PETComp e é discutido em reuniões semanais os detalhes do que deve ser feito e os prazos para a conclusão. O processo de levantamento de requisitos foi uma das grandes dificuldades encontradas pelo grupo no processo, por isso é importante que a pessoa com mais experiência em desenvolvimento fique responsável por essa atividade. É necessário que haja essa modelagem para que os envolvidos entendam o que deve ser feito, mas também possam se adaptar às alterações que podem surgir [Pressman and Maxim 2016].

Pôde-se perceber que há uma negligência por desenvolvedores iniciantes na definição formal de requisitos, fato que pode ser um problema no desenvolvimento de sistemas grandes, mas que, em contrapartida, dá certa flexibilidade para o surgimento de alterações e correções.

4.2. Preparo para o Desenvolvimento

A disponibilidade dos minicursos e tutoriais aos membros do grupo PETComp mostrou-se como fator essencial para o prosseguimento regular do projeto, isso pois um conhecimento básico do funcionamento das ferramentas e da linguagem de programação a serem utilizadas são imprescindíveis para a configuração do ambiente de desenvolvimento e iniciação do uso das tecnologias. Tais noções devem preceder o início de qualquer projeto, já que se mostra como aspecto requerido para a criação de sistemas, tanto os simples quanto os complexos.

Os conhecimentos necessários para desenvolvimento de sistemas, dentro do PETComp, são constantemente documentados e atualizados. Tal prática decorre da alta rotatividade dos membros do grupo, facilitando o entendimento e absorção de tais conteúdos pelos novos ingressantes, além de evitar dificuldades de introdução às novas tecnologias. No processo seletivo de novos integrantes, são realizadas apresentações didáticas para as equipes. Tais atividades servem como exercícios de revisão, capacitando os petianos e aprimorando a base de seus conhecimentos. As práticas apresentadas mostraram-se essenciais para o desenvolvimento eficiente e condizente com a proposta dos sistemas desenvolvidos.

4.3. Processo de Desenvolvimento

Um dos princípios do processo de desenvolvimento é montar equipes que sejam eficientes e que haja respeito e coordenação entre seus membros, além de ter uma forma de manter comunicação [Pressman and Maxim 2016]. Por isso, é dado muito foco no início do projeto em reuniões de desenvolvimento através de ferramentas de comunicação como as plataformas *Discord* e *Google Meet*.

Outra ferramenta que mostrou ser de grande importância foi a de controle de versionamento, o *GIT*, e a de repositórios remotos, o *Github*. Essas ferramentas permitem o acesso de qualquer membro do grupo ao código e também fazer alterações no mesmo. Também, um acompanhamento das tarefas realizadas pelo método *Kanban* foi essencial para a organização dos projetos em andamento. Em complemento a essas ferramentas foi utilizado o *Google Drive* para permitir a obtenção dos designs e imagens pela equipe de desenvolvimento.

5. Discussões

Neste estudo analisamos a metodologia e organização utilizada dentro do PET, ressaltando necessidades e condições especiais para que a atividade seja feita de forma plena com alunos de graduação. Entre elas, mostra-se de grande importância o investimento no treinamento contínuo de novos integrantes, para que disponham dos conhecimentos e tecnologias necessárias no desenvolvimento de *software* e *design*. Também, a inclusão de profissionais e discentes mais experientes que consigam auxiliar e direcionar os integrantes com menos prática na área de desenvolvimento se mostra significativo para a conclusão integral do projeto.

Acerca dos aspectos positivos e negativos da atual organização do processo de desenvolvimento, foi realizada uma coleta em entrevista das opiniões dos discentes. Observou-se que uma das etapas mais importantes, nas suas visões, é o treinamento.

Abaixo, o comentário de um destes discentes quando perguntado sobre dificuldades com que se deparou durante os softwares.

“A falta de experiência e conhecimento dos desenvolvedores é um problema comum no desenvolvimento de sistemas e sites no grupo. Porém, são contornados de duas formas principais: através de um ciclo de treinamento, através de minicursos e tutorias, sejam eles pagos ou gratuitos; e também através do auxílio de um membro do grupo que tenha mais conhecimento sobre desenvolvimento.” - Discente 01.

A capacitação, então, se torna ferramenta primordial para a continuação da atividade. Entretanto, tal etapa não é realizada de forma regular, sendo apresentada apenas em momentos de nova oportunidade de planejamento de sistema. Como descrito pelo relato a seguir:

“A aplicação dos treinamentos internos geralmente não segue um ciclo muito frequente. Isso gera 2 problemas principais: a equipe como um todo não passa por um processo de treinamento sobre determinados tópicos; e os problemas são corrigidos de maneira não otimizada, ou seja, correções rápidas pra resolver bugs que poderiam ser corrigidos de maneira mais eficiente e prática.” - Discente 02.

Portanto, apesar do treinamento ajudar em momentos de necessidade, quando os desenvolvedores não possuem o conhecimento técnico necessário para tal, a falta de uma constante qualificação dos discentes pode acabar prejudicando o projeto. Para contornar o problema, seria necessário um contínuo investimento para cursos. Estes investimentos, todavia, requerem tempo, podendo se tornar uma dificuldade para a equipe, tendo em vista que o grupo possui outras atividades em paralelo, que podem consumir o tempo disponível para treinamento. Além disso, alguns cursos podem ter custo elevado, o que pode dificultar sua aquisição. Como solução para este problema, os discentes que participam dos treinamentos, costumam desenvolver novos treinamentos e/ou atualizar os treinamentos antigos de tecnologias que sofreram atualizações.

6. Conclusões e Trabalhos Futuros

Discentes inexperientes possuem dificuldade em seguir um processo bem definido de desenvolvimento. Desta forma, este artigo apresentou dificuldades encontradas no decorrer do desenvolvimento de sistemas *web* e *mobile* no âmbito do programa PET, no grupo PETComp da UFMA, assim como as ferramentas que foram utilizadas para contornar tais empecilhos.

Oferecendo o ponto de vista de equipes iniciantes apresentando sistemas concretos e finalizados, o trabalho visa apresentar meios de começar e progredir no desenvolvimento de projetos. Dividido em seções de reuniões de planejamento inicial, prototipagem e desenvolvimento. Os pontos apresentados podem ser relevantes aos leitores que se identificarem com tal estrutura, ou até mesmo nortear uma forma de organização, podendo-se utilizar das informações apresentadas neste artigo para aprimorar o desempenho ao realizar atividades.

Em resumo, os resultados e análise realizada apresentam uma maneira satisfatória de organizar o desenvolvimento de software dentro da graduação, no contexto de um grupo de discentes inexperientes, ajudando a capacitar o aluno ao mesmo tempo que disponibiliza uma oportunidade de experiência em desenvolvimento de softwares que serão

utilizados por usuários em situações reais.

Como trabalho futuro, pretende-se analisar a visão dos discentes que participaram da atividade de desenvolvimento, após sua participação na atividade e estando no mercado de trabalho. Com isso, espera-se verificar de que forma a atividade poderia ser adaptada para atender as demandas da indústria de software e/ou preparar melhor os discentes em contextos reais de desenvolvimento de sistemas computacionais.

Referências

- Aragão, B., Santos, I., Nogueira, T., Mesquita, L., and Andrade, R. (2017). Modelagem interativa de um processo de desenvolvimento com base na percepção da equipe: Um relato de experiência. In *Anais do XIII Simpósio Brasileiro de Sistemas de Informação*, pages 428–435. SBC.
- Chichinelli, M. (2017). A importância das técnicas de levantamento de requisitos no processo de desenvolvimento de software. *Revista Empreenda UniToledo Gestão, Tecnologia e Gastronomia*, 1(1).
- Dias Jr, J., Belo, A., and Dias, F. (2014). Percepções sobre um processo de software sob o ponto de vista das equipes de desenvolvimento: Um estudo de caso. *XIII Simpósio Brasileiro de Qualidade de Software (SBQS), Blumenau, Brasil*, pages 136–150.
- Ferreira, N. C., de Oliveira, D. P., Grande, E. T. G., and de Oliveira, D. C. (2020). Relato de experiência dos softwares desenvolvidos no curso de tecnologia em análise e desenvolvimento de sistemas no instituto federal goiano campus iporá. *Research, Society and Development*, 9(10):e9329109381–e9329109381.
- Giardino, C., Unterkalmsteiner, M., Paternoster, N., Gorschek, T., and Abrahamsson, P. (2014). What do we know about software development in startups? *IEEE software*, 31(5):28–32.
- Nunes, W. S. (2017). Implantação de práticas de integração contínua: um relato de experiência em um laboratório de pesquisa e desenvolvimento de software.
- Pereira, I. M., de Senna Carneiro, T. G., and Pereira, R. R. (2013). Developing innovative software in brazilian public universities: Tailoring agile processes to the reality of research and development laboratories. In *Proceedings of the 4th Annual Conference on Software Engineering and Applications (SEA 2013)*, page 16.
- Pressman, R. and Maxim, B. (2016). *Engenharia de Software-8ª Edição*. McGraw Hill Brasil.
- Shameem, M., Kumar, C., and Chandra, B. (2017). Challenges of management in the operation of virtual software development teams: A systematic literature review. In *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 1–8. IEEE.