

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE GRADUAÇÃO DE TECNOLOGIA EM SISTEMAS PARA
INTERNET**

TIAGO LUIZ GOMES DE OLIVEIRA

**PORTAL DE AUXÍLIO PARA NOVATOS EM PROJETOS DE
SOFTWARE LIVRE**

TRABALHO DE CONCLUSÃO DE CURSO

CAMPO MOURÃO - PR

2016

TIAGO LUIZ GOMES DE OLIVEIRA

**PORTAL DE AUXÍLIO PARA NOVATOS EM PROJETOS DE
SOFTWARE LIVRE**

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de Tecnólogo em Tecnologia em Sistemas para Internet.

Orientador: Prof. Igor Fábio Steinmacher

CAMPO MOURÃO - PR

2016

Dados Internacionais de Catalogação na Publicação

PORTAL DE AUXÍLIO PARA NOVATOS EM PROJETOS DE SOFTWARE LIVRE/ Tiago Luiz
Gomes de Oliveira. – 2016.
48 f. : il. ; 30 cm

Orientador: Prof. Igor Fábio Steinmacher.
Trabalho de Conclusão de Curso (Graduação) – Universidade Tecnológica Federal do Paraná. Curso de
Graduação de Tecnologia em Sistemas para Internet. Campo Mourão - PR, 2016.
Bibliografia: f. .



ATA DA DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO

Às **vinte horas e trinta minutos** do dia **quatorze de julho de dois mil e quinze** foi realizada na sala E103 a sessão pública da defesa do Trabalho de Conclusão do Curso Superior de Tecnologia em Sistemas para Internet do acadêmico **Tiago Luiz Gomes de Oliveira** com o título **PORTAL DE AUXÍLIO PARA NOVATOS EM PROJETOS DE SOFTWARE LIVRE**. Estavam presentes, além do acadêmico, os membros da banca examinadora composta pelo professor **Igor Fabio Steinmacher** (Orientador-Presidente), pelo professor **Ivanilton Polato** e pelo professor **Igor Scaliante Wiese**. Inicialmente, o aluno fez a apresentação do seu trabalho, sendo, em seguida, arguido pela banca examinadora. Após as arguições, sem a presença do acadêmico, a banca examinadora o considerou **APROVADO** na disciplina de Trabalho de Conclusão de Curso e atribuiu, em consenso, a nota 7,0 (Sete). Este resultado foi comunicado ao acadêmico e aos presentes na sessão pública. A banca examinadora também comunicou ao acadêmico que este resultado fica condicionado à entrega da versão final dentro dos padrões e da documentação exigida pela UTFPR ao professor Responsável do TCC no prazo de **onze dias**. Em seguida foi encerrada a sessão e, para constar, foi lavrada a presente Ata que segue assinada pelos membros da banca examinadora, após lida e considerada conforme.

Observações:

Campo Mourão, 14 de julho de 2015

A folha de aprovação assinada encontra-se na coordenação do curso

RESUMO

OLIVIERA, Tiago Luiz Gomes de. PORTAL DE AUXÍLIO PARA NOVATOS EM PROJETOS DE SOFTWARE LIVRE. 48 f. Trabalho de Conclusão de Curso – Curso de Graduação de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Campo Mourão - PR, 2016.

Quase todos os projetos de software livre dependem de voluntários para continuar crescendo e melhorando. Por isso, a entrada e retenção de novatos é de suma importância. No entanto, os novatos tem dificuldades para fazer sua primeira contribuição em um projeto devido, entre outros fatores, à falta e dificuldade de acesso à informação sobre os projetos como: documentação, ajuda para configurar espaço de trabalho, como enviar contribuição, entre outras. Considerando a dificuldade que os novatos possuem para obterem informações acerca dos projetos e dada a importância dos novatos nesses projetos, este trabalho objetivou criar um portal que reúne informações de diversos projetos com intuito de disponibilizar informações aos novatos de forma organizada para facilitar os primeiros passos dos novatos em projetos de software livre.

Palavras-chave: Software livre, Novatos, Contribuinte Voluntário, Barreiras

ABSTRACT

OLIVIERA, Tiago Luiz Gomes de. . 48 f. Trabalho de Conclusão de Curso – Curso de Graduação de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Campo Mourão - PR, 2016.

...

Keywords: Open Source Software, Newcomers, Volunteer Contributor, Barriers

LISTA DE FIGURAS

FIGURA 1	– Modelo de Barreiras desenvolvido por Steinmacher et al. (2014)	16
FIGURA 2	– Cadastro de conta	20
FIGURA 3	– Informações básicas	21
FIGURA 4	– Busca de informações no OpenHub	21
FIGURA 5	– Adicionar Requisitos do projeto	22
FIGURA 6	– Adicionar informações sobre o fluxo de contribuição	23
FIGURA 7	– Adicionar informações sobre tarefas fáceis	24
FIGURA 8	– Encontrar auxílio	25
FIGURA 9	– Cadastro de recursos do projeto para configurar o Espaço de Trabalho	26
FIGURA 10	– Cadastro de Informações para Submissão mudanças	27
FIGURA 11	– Canais de comunicação	28
FIGURA 12	– Tela Inicial	29
FIGURA 13	– Sobre o Projeto	30
FIGURA 14	– Graficos: Linguagens e Linhas de código do Projeto	30
FIGURA 15	– Características do Projeto	31
FIGURA 16	– Fluxo de contribuição do projeto	32
FIGURA 17	– Tarefa fácil para iniciar	33
FIGURA 18	– Como encontrar um orientador	34
FIGURA 19	– Pesquisar antes de perguntar	35
FIGURA 20	– Canal de Bate-papo IRC	36
FIGURA 21	– Informações sobre a Lista de Discussão	37
FIGURA 22	– Ajuda para configurar workspace	38
FIGURA 23	– Entender código	39

SUMÁRIO

1	INTRODUÇÃO	1
2	SOFTWARE LIVRE	4
2.1	CONCEITO	4
2.2	LICENÇAS DE SOFTWARE LIVRE	4
2.3	BENEFÍCIOS DO SOFTWARE LIVRE	6
2.4	DIFICULDADES E LIMITAÇÕES	7
2.5	COMUNIDADES E CONTRIBUIÇÃO	7
2.5.1	ESTRUTURA/COMPONENTES DE UMA COMUNIDADE	8
2.5.2	PROCESSO DE CONTRIBUIÇÃO	10
2.5.2.1	CONTRIBUIÇÃO VOLUNTÁRIA	10
3	TRABALHOS RELACIONADOS	13
4	DESENVOLVIMENTO DO PROJETO	15
4.1	O MODELO DE BARREIRAS DE STEIMACHER	15
4.2	MÓDULOS DO PORTAL	19
4.2.1	MÓDULO AUTENTICADO/MANUTENÇÃO DE PROJETO	19
4.2.1.1	CADASTRO DE USUÁRIO	19
4.2.1.2	CADASTRO DE INFORMAÇÕES BÁSICAS	20
4.2.1.3	CADASTRO DE REQUISITOS DO PROJETO	21
4.2.1.4	CONSIDERAÇÕES SOBRE O FLUXO DE CONTRIBUIÇÃO	22
4.2.1.5	ENCONTRAR TAREFA FÁCIL	23
4.2.1.6	ENCONTRAR AUXÍLIO	24
4.2.1.7	CADASTRO DE RECURSOS PARA CONFIGURAR O ESPAÇO DE TRABA-	
	LHO	25
4.2.1.8	CADASTRO DE INFORMAÇÕES PARA SUBMISSÃO DE MUDANÇAS	26
4.2.1.9	CADASTRO DOS CANAIS DE COMUNICAÇÃO	27
4.2.2	MÓDULO DE VISUALIZAÇÃO DO PROJETO	28
4.3	RECURSOS UTILIZADOS	39
5	CONCLUSÕES	45
	REFERÊNCIAS	47

1 INTRODUÇÃO

Software livre encontra-se em processo de popularização mais evidente, estando em quase todos os lugares, desde smartphones até grandes sistemas. Como exemplo, temos o Mozilla Firefox que é um projeto de software livre muito difundido. Mas, o sucesso de projetos como este, é sustentado principalmente pela comunidade que o desenvolve. Essa comunidade possui integrantes de dois tipos fundamentais: o contribuinte remunerado e o voluntário. O contribuinte remunerado é aquele que de alguma forma é pago por uma entidade para participar e produzir no projeto. Já o voluntário é aquele que dedica seu tempo e conhecimento em benefício da comunidade e do projeto sem receber remuneração.

Alguns projetos possuem em sua comunidade contribuintes pagos para colaborar, para esses a motivação inicial é o "salário". No entanto, conforme Madey et al. (2002), a comunidade é uma característica importante de projetos de software livre, sendo constituída significativamente por voluntários: desenvolvedores que participam livremente dos projetos que consideram atraentes. Esses projetos se baseiam, uns de maneira mais forte que outros, na atração de novos voluntários para continuarem crescendo e melhorando. (MADEY et al., 2002)

A atração e retenção de voluntários novatos não é algo tão simples. Geralmente, os novatos não conseguem contribuir e evoluir rapidamente, devido a falta e dificuldade de acesso às informações sobre os projetos. Conforme Cubranic et al. (2005), muitas vezes os novatos necessitam aprender aspectos técnicos e sociais do projeto sem ajuda e buscar informações sobre os projetos, e "... não é fácil acessar essas informações devido ao grande volume, à falta de ferramentas para navegar nos repositórios, e à dificuldade de fazer as conexões entre os itens relacionados logicamente em fontes diferentes". Steinmacher et al. (2012), cita que, em alguns casos, ao solicitarem informações em lista de discussões, os novatos não são atendidos de maneira adequada pelos demais desenvolvedores criando um potencial motivo de desistência. (STEINMACHER et al., 2012)

A falta de um ambiente adequado também é um complicador para o novato. De acordo

com Park e Jensen (2009), em alguns estudos foi constatado que a forma de organização dos ambientes de desenvolvimento e os recursos fornecidos pelos projetos de software livre, são suficientes para apoiar o trabalho dos contribuintes já estabelecidos. No entanto, essa suficiência não se aplica necessariamente aos novatos. Pois, estes possuem necessidades diferentes dos desenvolvedores já ativos no projeto. A organização considerada pelo desenvolvedor pode ser o contrário para o novato. Situações como as citadas anteriormente, podem vir a desencorajar os iniciantes e prejudicar o sucesso do software.

A característica voluntária e não remuneratória dos projetos de software livre, gera um laço de compromisso mais vulnerável entre o contribuinte e o projeto. Com isso, o contribuinte não tem uma obrigação para com o projeto, a qualquer momento o mesmo pode desvincular-se. Visto que muitos projetos de software livre dependem dos novatos e as dificuldades destes em se orientar em um meio desconhecido. Então, como é possível sanar os problemas dos novatos por informações sobre os projetos de software livre e fornecer-lhes uma maneira mais amigável e confortável de contribuírem com Projetos de Software Livre?

Visto que o problema central é a dificuldade que os novatos possuem para ingressarem no mundo do software livre e dada a importância dos novatos nesses projetos, é necessário fornecer uma maneira mais eficiente de provê-los de informações de forma que os mesmos tenham uma maior facilidade de contribuir para os projetos. Nesse sentido, o objetivo deste trabalho é fornecer um portal web, que propicie ao novato um ambiente que forneça um certo nível de informação adequada as suas necessidades e reduza as dificuldades enfrentadas para realizar sua primeira contribuição em um projeto de software livre.

Para e tal objetivo, foram traçadas as seguintes metas:

- Fornecer informações de como o novato pode encontrar um integrante mais experiente dentro do projeto para auxiliá-lo;
- Fornecer informações ao novato de como ele pode encontrar uma tarefa fácil para iniciar;
- Fornecer informações sobre o fluxo de contribuição do projeto;
- Fornecer informações de como o usuário pode enviar sua contribuição ao projeto;
- Fornecer informação de como o usuário pode configurar o espaço de trabalho;
- Apresentar os recursos que são utilizados pelo projeto, tais como: linguagens, bibliotecas, drives, plataforma, etc;

- Apresentar os mecanismos de comunicação do projeto, tais como: lista de discussão, fóruns, blogs e chat;
- Assegurar que somente usuários cadastrados e autenticados no portal, podem inserir, alterar e remover seus respectivos projetos.
- Prover controle de acesso ao portal para manutenção de projetos;
- Prover interface para cadastro de novos usuários e projetos;
- Disponibilizar opção de busca de informações sobre o projeto na API do OpenHub;
- Integrar o portal com a API do OpenHub para obtenção dos dados de sobre os projetos.

As definições dos requisitos para esse ambiente baseiam-se parcialmente nas barreiras encontradas por Steinmacher et al. (2014). Assim, o portal deve fomentar o aprendizado sobre o projeto e motivar a participação dos novatos. O novato deve poder buscar projetos e tarefas de acordo com suas habilidades e interesses. Deve-se oferecer um espaço organizado e padronizado onde o novato possa encontrar dicas, componentes, ferramentas e materiais que possam auxiliá-lo a romper as barreiras iniciais, e tornar mais eficiente o processo de contribuição em um projeto de software livre.

O restante do trabalho está estruturado da seguinte forma: No Capítulo 2, falamos um pouco sobre Software Livre, seu conceito, benefícios, dificuldades e limitações, comunidades e contribuição, estrutura e componentes de uma comunidade, o processo de contribuição destacando os tipos de contribuição voluntária e comercial. No Capítulo 3, apresentamos os trabalhos que se relacionam ao nosso. No Capítulo 4, demonstramos os objetos de interface que foram produzidos e os recursos e tecnologias utilizados na implantação do portal. No Capítulo 5, é apresentado conclusão e indicativas para trabalhos futuros.

2 SOFTWARE LIVRE

2.1 CONCEITO

O conceito de software livre está pautado em torno da liberdade de expressão, em poder fazer o que se tem vontade sem escusa de consciência, repressão. Portanto, para entender o conceito deve-se pensar em "liberdade de expressão" e não ausência de custo/desoneração. Portanto o que importa é o usuário ter liberdade para com o software e não aspectos relativos à preço. De acordo com GNU (2014), "Software livre se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o software".

O conceito livre não deve ser tomado como impedimento ao comércio, ou seja, 'Software Livre' não quer dizer 'não comercial'. Portanto, mesmo o software sendo livre, o usuário tem a liberdade de distribuí-lo e cobrar por esta distribuição. Anteriormente já havia mencionado que os preceitos do software livre se pautam na liberdade de ação de expressão, ou seja, não quer dizer que um software livre não possa ser vendido ou que um usuário não possa cobrar por ele. E, além disso, a possibilidade de vender um software livre é independente da forma como se deu a sua obtenção do mesmo pelo usuário, ou seja, não importa se pagou ou conseguiu gratuitamente (GNU, 2014).

No entanto, apesar de possuírem uma gama de liberdades, as possibilidades de usufruto do software livre estão também vinculados aos tipos de licenças que são aplicados a eles. Algumas licenças são mais permissivas, já outras são um pouco mais restritivas.

2.2 LICENÇAS DE SOFTWARE LIVRE

Ao decidir tornar um software livre, quem possui os direitos sobre esse, deve escolher os termos que serão aplicados à esse projeto, os direitos que estará transferindo para outras pessoas e as condições de uso para desse software.

Existem vários tipos de licença de software livre, e ainda é possível que o proprietário crie sua própria licença. No entanto, não seria o ideal criar uma nova licença, visto que já existem várias disponíveis. O mais prudente é verificar, dentre as existentes, a que mais se encaixa nos objetivos desejados e utilizá-la. Além disso uma nova licença dificulta para os futuros usuários do software visto que terão de ler e analisar os termos de da nova licença.

A seguir apresentamos breve descrição de algumas licenças aplicadas ao software livre são:

- GNU General Public License (GPL)

Essa licença permite a execução do programa para qualquer finalidade, permite que o programa seja estudado e adaptado, que seja redistribuído como cópias, e, também que seja aperfeiçoado. Contudo, permite que se mantenha os direitos do autor e impede que essa informação seja usada de uma maneira que limite as liberdades relativas à essa licença, impedindo, por exemplo, que alguém venha a se apoderar do código do software ou restringir o uso do mesmo por outras pessoas.

- Berkeley Software Distribution (BSD) License

Essa licença possibilita ao detentor dos direitos autorais do software autorizar outras pessoas a utilizá-lo, modificá-lo e distribuí-lo. Para isso, as únicas exigências são que o nome do autor original não seja utilizado em trabalhos derivados sem permissão. Essa restrição, tem o objetivo de proteger e salvaguardar o autor original de qualquer relação com possíveis alterações e modificações realizadas, por terceiros, no software, que possam vir a ocasionar qualquer tipo de problema. No caso de redistribuição do código fonte ou binário, modificado ou não, é necessário que seja mencionado o copyright original e os termos da licença.

- MIT License

Essa é uma licença, que estabelece que qualquer pessoa que obtém uma cópia do software e seus arquivos de documentação associados pode lidar com eles sem restrição, inclusive sem limitação dos direitos a usar, copiar, modificar, mesclar, publicar, distribuir, sublicenciar e/ou vender cópias do software. As condições impostas para tanto são apenas manter o aviso de copyright e uma cópia da licença em todas as cópias ou porções substanciais do software.

- Apache License V2.0

Essa licença se assemelha com a BSD, mas algumas características são diferentes, como, por exemplo, a questão relativa às patentes de software, que define que toda pessoa que contribuir para o software em questão, autoriza também uma licença em termo mundial e perpétua para uso de suas patentes que sejam necessárias para uso ou distribuição do código que ele produziu juntamente com o software original. Também há possibilidade dessa licença ser adotada por referência e também pode-se incluir um arquivo chamado NOTICE, junto aos arquivos do projeto, cujo conteúdo precisará ser mantido e redistribuído junto com o código.

- GNU Library or Lesser General Public License (LGPL)

Essa licença possibilita que seja desenvolvido programas mistos, onde parte do código do programa está disponível, ou seja, é aberto e outra parte é restrita, ou seja, seu código aberto. Outra característica dessa licença é a possibilidade de associação com programas que não estejam sob os termos da GPL ou LGPL, como os programas de Software proprietário.

2.3 BENEFÍCIOS DO SOFTWARE LIVRE

A luta contra o software proprietário deu ao software livre o poder de proporcionar vários benefícios à sociedade. Dentre as vantagens do software livre, podemos destacar algumas citadas por Costa (2012), como: o baixo custo social, a liberdade de uso, independência de um único fornecedor, desembolso inicial bem perto de zero, robustez e segurança, possibilidade de adequar aplicativos e redistribuir uma versão alterada.

Diferente do software proprietário, que é desenvolvido e distribuído por um único fornecedor, o software livre tem sua localização distribuída, é possível adquirir o software tanto da fonte original quanto de outros desenvolvedores, empresas e usuários. O desembolso inicial próximo de zero, se refere ao fato de mesmo sendo livre o software pode auferir custos tanto para ser adaptado, por uma empresa, por exemplo, que deverá pagar desenvolvedores para este fim, quanto para adquirir o software que também pode ser comercializado.

Além disso, nos dias atuais é possível constatar a inegável contribuição do software livre para difusão do conhecimento e desenvolvimento tecnológico. Pois, ações como a adaptação, incremento, correção, melhoria etc., oriunda de um esforço despendido por diversos

contribuintes e participantes de todas as partes do mundo. Dessa forma, o saber de cada um é disseminado através do software livre, quando o mesmo é redistribuído. Por exemplo, uma melhoria em algum componente do software, redistribuída posteriormente, seguindo as liberdades básicas, beneficiará os futuros usuários do software.

2.4 DIFICULDADES E LIMITAÇÕES

Apesar de suas possibilidades e benefícios, o software livre também apresenta dificuldades e limitações para sua utilização em larga escala. Dificuldades que na maioria são geradas por aqueles que desconfiam ou não acreditam no modelo de desenvolvimento de Software Livre, ou as dificuldades também podem ser geradas por problemas de adaptação dos usuários, falta alternativas de aplicativos/recurso entre muitos outros fatores que influenciam na decisão de adoção de Software Livre.

Observamos que o software livre, a exemplo das versões de Sistema Operacional Linux, ainda não possuem grande popularização devido à dificuldade que os usuários encontram ao utilizá-los. Muitos usuários não se familiarizam e adaptam com os conceitos adotados por eles.

Softwares livres costumam ter interfaces pouco intuitivas, instalações complicadas características que espantam qualquer pessoa. Por isso, muitos programas de código aberto têm copiado funções e o layout de programas populares, como o Windows e o pacote Microsoft Office, para se tornarem familiares aos usuários.

2.5 COMUNIDADES E CONTRIBUIÇÃO

Um projeto de software livre é formado e mantido por uma comunidade, um grupo de pessoas engajadas em torno de um objetivo comum. Evidentemente quando se pretende formar uma comunidade desse tipo, uma das necessidades básicas e mais importantes é atrair e convencer pessoas que se interessem em participar doando seu tempo, esforço e conhecimento. Mas afinal, como é composta/formada uma comunidade de software livre? Nascimento e Santoro (2009) afirmam que "a comunidade é geralmente composta por um moderador e diversos participantes, onde todos contribuem de alguma forma para alcançar o objetivo da comunidade".

Conforme afirma em sua obra, Raymond (2002) apresenta dois modelos de desenvolvi-

mento, a Catedral, que é baseado no desenvolvimento centralizado, controlado, típico do desenvolvimento de software proprietário. E, o modelo Bazar, onde o desenvolvimento do software é disperso geograficamente, aberto para participação de todos, é o adotado pelas comunidades de desenvolvimento livre.

Em seus relatos, Raymond (2002), ainda fala sobre a desconfiança em torno do desenvolvimento no estilo Bazar. Ele contesta a eficácia de se tentar organizar, coordenar e engajar em um único propósito, uma diversidade imensa de personalidades e culturas diferentes. No entanto, ele mesmo observou os benefícios do desenvolvimento em comunidade, abrindo seu projeto, o Fetchmail.

Muito se pensa que as comunidades de software livre são todas iguais, ou seja homogêneas, no entanto, as comunidades se organizam em torno de objetivos específicos, assim, os contribuintes também aderem àquela que mais lhe parece amigável, que atende os seus ideais. As "comunidades, por sua vez, embora convirjam em inúmeros pontos, detêm vários atributos que a elas são muito próprios, configurando-se, assim, um cenário plural, democrático e, por certo, livre". (ALENCAR et al., 2009)

2.5.1 ESTRUTURA/COMPONENTES DE UMA COMUNIDADE

Uma comunidade surge quando mais de uma pessoa se reúne em torno de um objetivo comum. Atualmente há comunidades de sucesso como Debian e Ubuntu. Essas e outras comunidades precisam de uma estrutura para seu funcionamento, uma forma de se organizar para produzir e compartilhar. Em uma análise sobre as comunidades Fedora, Ubuntu, Slackware e Debian, Alencar et al. (2009), apresentou tópicos que representam a dinâmica interna e externa das comunidades analisadas, como: um porta voz, canais de comunicação, hierarquia e relação de poder.

Conforme Alencar et al. (2009), a figura do porta voz, seria:

"... alguém que, ao passar pelos desafios e provas inerentes à cultura meritocrática e fundamentar suas atividades nas relações sociais em meio ao grupo, conseguiu engariar um capital simbólico tão significativo a ponto de ter as qualificações necessárias para responder pela comunidade."

Com relação às comunidades que analisou, Alencar et al. (2009), a figura do porta-voz deve ser "... um símbolo, uma figura cuja conduta se identifique pontualmente com a filosofia das distribuições ao incorporar seus traços elementares, estando, assim, apto a representar seus

membros". Muitas vezes a figura do representante não é necessariamente provida por um cargo que o possibilite ser. Em virtude do envolvimento e participação incisiva na comunidade alguns representantes são reconhecidos pelos outros membros e assim designados a serem a figura que os representa. Como exemplo, tem-se o caso do representante da comunidade Debian, no Brasil, conforme apresentado por Alencar et al. (2009), os demais integrantes desta comunidade apontam como porta-voz o paranaense Felipe Augusto van de Wiel, que se envolveu com o Debian apenas em 2002, mas o sua afinco na comunidade o habilitou a ser apontado como representante.

A disponibilidade dos integrantes das comunidades, onde os membros podem estar em qualquer lugar no mundo, gera a necessidade de mecanismos de comunicação. Algumas comunidades promovem encontros presenciais em eventos, congressos, palestras e fóruns. No entanto, grande parte da comunicação se dá no meio cibernético, como: Listas de discussão, Fóruns, IRC¹, Blogs e E-mail.

O fato das comunidades de software livre se tratarem de uma agregação de membros das mais variadas localidades e características, não significa que estas comunidades sejam um ambiente desorganizado, onde impera a anarquia. Assim, os líderes devem tomar cuidado com a sua postura diante dos colaboradores, pois, uma atitude inadequada pode culminar com a decisão desse não contribuir mais no projeto ou até migrar para outro. (ALENCAR et al., 2009)

Enquanto projetos geridos somente por uma comunidade, a hierarquia é definida em assembleias. Nas distribuições comerciais, as empresas que as mantêm desempenham papel fundamental nas decisões. Alencar et al. (2009), ainda destaca dois graus de hierarquia em comunidades de software livre. O Debian, por exemplo, é uma estrutura que se organiza com um líder, um comitê técnico, um secretário e delegados, apesar disso quem realmente exerce o grau máximo de decisões e poder são os desenvolvedores do projeto, exercendo o poder com Resolução Geral ou eleições internas. Num segundo nível de estrutura, mais inferior a dos conselhos, estão aqueles que são responsáveis em organizar e coordenar cada subprojeto, os líderes, que possuem tarefa de manter o nível de atividade do projeto para tarefas de tradução, documentação, incentivo de novatos, entre outras.

Em uma publicação realizada por Fogel (2005), podemos ver que os meios digitais são parte fundamental da estrutura técnica de um projeto de software livre. A estrutura que ele

¹Internet Relay Chat (IRC) é um protocolo de comunicação utilizado na Internet. Ele é utilizado basicamente como bate-papo (chat) e troca de arquivos, permitindo a conversa em grupo ou privada

cita é composta de um web site, que será a "casa" do projeto; controle de versão, necessário para controlar as mudanças de código; lista de discussão/fórum, para realizar a divulgação e discussão de melhorias, entre outros feitos; controle de erros, para acompanhar o trabalho e programar lançamentos; e, um canal de bate-papo, para realizar conversações rápidas e mais simples.

2.5.2 PROCESSO DE CONTRIBUIÇÃO

Muitas vezes quando se fala em processo de contribuição em comunidades de software livre, a ideia que se têm é de que isso é feito basicamente por voluntários, pessoas que despendem seu tempo, conhecimento para contribuir sem remuneração, apenas se pautando na aquisição de conhecimento. No entanto, o processo de contribuição sofre influência tanto de voluntários como de contribuintes que recebem para participarem das comunidades.

De acordo com, Capra e Wasserman (2008), a produção de software livre está baseada em duas categorias, a saber: "comunidade" e "comercial". Os projetos de comunidade "... são liderados por uma comunidade de desenvolvedores ou partes interessadas e são distribuídos sob uma licença de código aberto aprovada, por exemplo, GPL, BSD, ou Apache. Já os projetos de categoria comercial, "... são liderados por uma empresa, que normalmente desenvolveu a maior parte ou todo o código, e depois vende assinaturas e serviços para o produto desenvolvido". Neste trabalho nosso interesse está pautado nos projetos envolvendo comunidades de software livre.

2.5.2.1 CONTRIBUIÇÃO VOLUNTÁRIA

O contribuinte voluntário não possui nenhuma forma de obrigação para com o projeto e a comunidade, ou seja, ele não possui uma carga horária definida, obrigação em torno de uma tarefa, enfim, ele é livre para trabalhar e contribuir como ele quiser e puder.

Já que não recebe valor monetário em troca de sua contribuição, o que leva uma pessoa a participar, despender seu tempo livre em torno de uma coisa que não lhe traga benefício financeiro algum? Sobre isso, Himanen (2001), afirma que "sem considerar os motivos sociais, é muito difícil compreender os motivos que levam os hackers [programadores] a empregar seu tempo livre no desenvolvimento de programas que são dados a terceiros posteriormente". Ou seja, o voluntário pode participar por algo muito além do retorno financeiro, assim para:

"... compreender a razão desse trabalho voluntário é preciso entender, portanto, que o fator organizacional dessa comunidade de Software Livre não está associado somente ao seu caráter mercantil, mas também à paixão e ao prazer de criar juntos algo que seja reconhecidamente valioso entre seus pares. Nesse "modelo livre", as trocas ocorrem, além do interesse no mercado, baseadas em elogios, sentimentos de recompensa, amizade, denúncia em fóruns públicos, entre outras inúmeras formas". (JUNIOR; SOARES, 2013)

A motivação maior de um voluntário que participa de uma comunidade de software livre é o retorno social de sua contribuição, dos laços afetivos formados e da possibilidade de participar de algo que venha a contribuir para melhorar o mundo. Conforme depoimento de um participante da comunidade GNOME - GNU Network Object Model Environment, a satisfação em participar de uma comunidade de software livre provém do fato de estar junto das pessoas, de ganhar novos amigos, da satisfação de ver o projeto sendo concretizado e progredindo e do reconhecimento provindo dos demais usuários.

Em uma afirmação feita por Linus Torvalds, em seu livro *Só por Prazer - Linux - Bastidores de sua criação*, ele trata o processo de contribuição voluntária e o resultado deste, como uma parte do programador, algo que também acaba se tornando parte dele. O produto do esforço do contribuinte é uma obra de arte, uma coisa que gera um vínculo indissociável e nenhuma forma de remuneração seria capaz de substituir a satisfação e o prazer do resultado obtido. A relação com o resultado do trabalho realizado é tão forte que acaba se tornando parte do próprio ser, uma relação que não é possível de ser rompida.

Santos (2002), afirma que quando se participa de um projeto motivado por sentimentos como paixão e prazer, vai além do simples e mero cumprimento e obediência de carga horária e jornada, o tempo se torna algo trivial e as atividades acabam se confundindo com o lazer.

"É inegável que os hackers mantêm com o trabalho uma relação particular, na qual o aspecto lúdico, a diversão e o prazer são capitais, assim como é incontestável que semelhante relação implique uma especial gestão do tempo, já que trabalhando movidos pela satisfação, experimentam o tempo de forma muito mais elástica do que aqueles para quem a atividade é uma obrigação e mesmo um fardo". (SANTOS, 2002)

A falta de uma obrigação formal para com o projeto e também a ausência de aspectos contratuais que garantam uma efetividade de participação, poderiam comprometer o andamento, a qualidade e o sucesso do projeto. Contudo, projetos de renome e sucesso como "(...) Slackware e Debian são os maiores exemplos de distribuições não-comerciais, nas quais pesam apenas a força das intensas atividades de seus voluntários". (ALENCAR et al., 2009)

A ampliação do conhecimento, que o compartilhamento de ideias proporcionado pela organização em comunidade é uma forma de ganho para todos os envolvidos. Da mesma forma que eu aprendo com os outros, os outros também podem aprender comigo. No entanto, aquele que desenvolve software livre não pode viver apenas do compartilhamento de ideias, ele precisa de algo para suprir suas necessidades básicas. Essas constatações são feitas por Souza (2014).

"Nas supostas palavras de Bernard Shaw, 'Se você tem uma maçã e eu tenho uma maçã, se as trocarmos, cada um de nós continuará com apenas uma maçã. Mas eu tenho uma ideia, e você tem uma ideia. Se as trocarmos um com o outro, ambos teremos duas ideias'. O desenvolvedor de Software Livre além das tuas ideias precisam de algumas maçãs também, afinal são gente como a gente. Penso logo existo, mas se não como logo, logo não mais existo". (SOUZA, 2014)

O contribuinte voluntário ao participar de projetos de software livre, sem que receba nenhum benefício financeiro por essa participação, se baseia em algo que desvincula sua contribuição do fator monetário. Para ele, sua participação é baseada, entre outros aspectos, no aprendizado, ou seja, compartilhar o seu conhecimento e aprender com outras pessoas; no retorno social, em estar envolvido em algo que pode vir a ajudar outras pessoas. Portanto, para o contribuinte, ao participar de forma voluntário de um projeto de software livre o dinheiro não é fator mais importante envolvido.

3 TRABALHOS RELACIONADOS

Em um estudo realizado por Steinmacher et al. (2012a), procurou-se fazer uma identificação preliminar dos motivos que levam os novatos a desistirem dos projetos de software livre. A fonte de dados utilizados para a fundamentação dos trabalhos foi provida de lista de discussões e gerenciador de tarefas, no caso o Hadoop Common. Os resultados obtidos revelaram que menos de 20% dos novatos continuaram nos projetos. Outro ponto observado é que o autor e o tipo das respostas influenciam os novatos a desistirem ou não dos projetos.

Park e Jensen (2009) examinaram os benefícios das ferramentas de visualização para o progresso dos novatos em projetos de software livre, visto que, essas ferramentas podem proporcionar maior facilidade em explorar e descobrir os projetos de software livre e, assim, simplificar o processo de aprendizagem do novato. Os resultados do experimento mostraram que as atuais ferramentas e recursos disponíveis, para os desenvolvedores, não são suficientes para que os mesmos possam extrair informações mais específicas. Isso gera uma dificuldade de obter tais informações, o que pode ter impacto negativo sobre a motivação e eficiência dos novatos. Dessa forma, a pesquisa também revelou que o fornecimento de informações visuais para os novatos pode reduzir os desafios que eles enfrentam quando estão aprendendo sobre um novo projeto. Os usuários das ferramentas de visualização também fizeram críticas positivas sobre a utilização das mesmas.

Verificando a necessidade de auxílio adequado ao novato, um trabalho conduzido por Steinmacher et al. (2012b), observando esta necessidade propôs a implantação de um sistema de recomendação, que vise buscar e identificar um desenvolvedor mais experiente que se enquadrasse nas dificuldades enfrentadas pelo novato e desta maneira fornecer orientação ao mesmo no processo inicial de contribuição a um projeto de software livre. Para esse fim ele utilizou-se das técnicas de análise temporal e social. A meta posterior do trabalho visava à concretização desta proposta de ferramenta que identifique especialistas que estão envolvidos em atividades onde se encontram novatos.

A forma com que os novatos são recepcionados e interagem com a comunidade, também são importantes para a sua permanência, visto que um novato que não é bem recepcionado tende a não permanecer no projeto, o mesmo pode ocorrer caso a interação não seja tão bem efetivada. Maciel (2014) realizou um trabalho com o objetivo de identificar os padrões de entrada e migração de novatos em projetos de software livre. A pesquisa baseou-se em análise de redes sociotécnicas e alterações do relacionamento do entre os desenvolvedores novatos e experientes do projeto Hadoop Common. Os seus resultados mostraram que o canal mais utilizado para a entrada de novatos é a lista de e-mail. Outra constatação foi a dificuldade de retenção de novatos em projetos de software livre, pois, foi observado que os novatos permaneceram por pouco tempo nos projetos independente do meio utilizado; e, a maior parte da interação dos novatos é feita com veteranos, porém, muitos deles não são respondidos e também acabam deixando o projeto.

Em seu trabalho, Zhou e Mockus (2012), tentaram identificar a motivação que leva um indivíduo a passar da condição de novato a um participante efetivo do projeto, ou seja, verificaram o processo de adesão de novatos em projetos de software livre. O estudo foi conduzido utilizando dados obtidos nos projetos da Mozilla e Gnome. Os resultados demonstraram que, para um novato se tornar um contribuinte efetivo, além do seu interesse, também está intimamente vinculada com o ambiente do projeto em si. Os novatos que começam com tarefas mais básicas e simples, como realização de comentários na comunidade e também aqueles que conseguem que um problema relatado por eles sejam corrigidos, tem uma alta chance de permanecer no projeto.

Um trabalho mais recente de Steinmacher et al. (2014), objetivou identificar e organizar as barreiras de integração para os recém-chegados a projetos de software livre. Para esse fim, analisou fontes de dados obtidas através da revisão sistemática da literatura, para observar barreiras já identificadas por outros pesquisadores; informações de alunos que, para pesquisa, se propuseram a contribuir para projetos de software livre; respostas obtidas através de questão enviada para projetos de software livre; e, entrevistas semiestruturadas com promotores de diferentes projetos. Como resultado da pesquisa, obteve um modelo de barreiras, as quais descrevem as dificuldades encontradas pelos novatos em projetos de software livre. O modelo obtido fornece base para que os novos projetos de software livre permitam uma atenção e facilidade maior aos novatos em projetos de software livre. O modelo é apresentado na Figura 1.

4 DESENVOLVIMENTO DO PROJETO

Neste trabalho implementamos um portal web que permite o cadastro de projetos de software livre com informações para os novatos realizarem sua primeira contribuição. Tais informações foram organizadas e apresentadas de forma padronizada, visando auxiliar o novato a encontrar o projeto adequado às suas características e realizar a sua interação com o projeto. Para o desenvolvimento do portal, os requisitos foram levantados considerando como parâmetros o modelo de barreiras elaborado por Steinmacher et al. (2014).

4.1 O MODELO DE BARREIRAS DE STEIMACHER

Em seu trabalho, por meio de entrevistas com participantes de projetos de software livre e constante revisão da literatura, foram identificadas dificuldades que os novatos encontram para participarem de projetos de software livre. Desse trabalho de Steinmacher et al. (2014), foi projetado o Modelo de Barreiras, apresentado na Figura 1, o qual sistematiza as dificuldades que os novatos encontram ao tentarem contribuir com projetos de software livre.

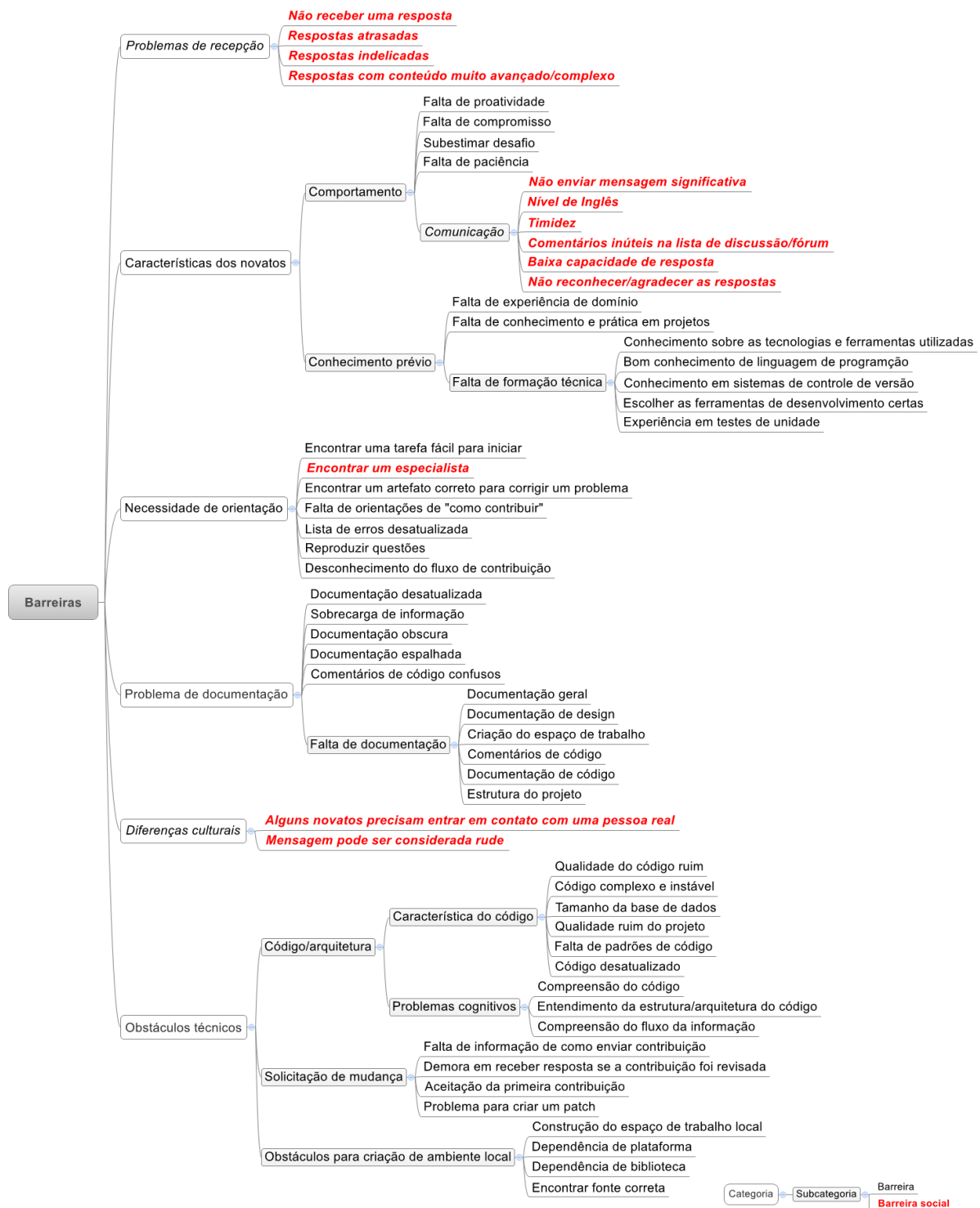


Figura 1: Modelo de Barreiras desenvolvido por Steinmacher et al. (2014)

Os resultados obtidos por Steinmacher et. al (2014), por meio da análise dos dados, gerou cinco categorias de barreiras, a saber: Problemas de recepção, Características dos nova-

tos, Necessidades de orientação, Problema de documentação, Diferenças culturais e Obstáculos técnicos.

A categoria Problemas de Recepção do Modelo de Barreiras, destaca pontos como a falta de respostas, respostas atrasadas, indelicadas e respostas com conteúdo muito avançados. Nessa categoria o agente principal a ser conscientizado seria o que se situa dentro do projeto e que interage com o novato, para que o atenda da melhor forma possível, levando em conta suas dificuldades e limitações. No entanto, é difícil tratar e garantir que situações como essas não ocorram. Pois, não podemos simplesmente controlar ou mudar o estado emocional do integrante do projeto para que sempre aja de forma cordial e compreensiva com os novatos. A nossa proposta é tentar dar condições de que seja disponibilizado o máximo de informação para o novato, dando assim a condição do mesmo encontrar as informações por conta própria e, dessa forma, reduzir solicitações de informações já disponibilizadas no Portal.

Da categoria de Características dos Novatos, tentaremos tratar das barreiras da subcategoria Comportamento, que são o não envio de mensagem significativa e comentários inúteis na lista de discussão/fórum. Essas barreiras, tentamos amenizar dando espaço para inserção de informações de como o novato pode se inscrever na lista de discussão, de como se comportar dentro da lista e como formatar adequadamente uma mensagem.

Evidentemente que o Modelo de Barreiras, Figura 1, destaca outras características relacionadas ao novato. Muitas delas se relacionam com o aspecto motivacional do novato. Por exemplo, experiência com testes de unidade, conhecimento em linguagem de programação, conhecimento das tecnologias utilizadas pelo projeto, controle de versão, falta de proatividade, compromisso, paciência. Essas barreiras mostram um pouco do perfil de parte dos novatos que iniciam na contribuição em projetos de software livre.

A superação de muitas das barreiras, mencionadas no parágrafo acima, vai da iniciativa do novato em aplicar esforço no entendimento e aprendizado dos aspectos do projeto bem como das tecnologias utilizadas por ele. Para tentar aprender e atender esses requisitos, o novato primeiramente deve conhecê-los. Dessa forma, no Portal, tentaremos possibilitar que essas informações sejam fornecidas ao novato. No portal trataremos essas informações como: Requisitos Básicos/Técnicos e Requisitos Adicionais/Comportamentais. O primeiro trata dos recursos disponibilizados pelo projeto, tais como: linguagens, ferramentas de desenvolvimento, bibliotecas, entre outras. O segundo abrange questões de comportamento dentro do projeto, como a necessidade de proatividade, compromisso, respeito para com os demais integrantes,

entre outras.

Outra categoria de barreira importante apresentada no Modelo de Barreiras, é quanto à Necessidade de Orientação. O modelo cita nesta categoria, 7(sete) barreiras, das quais 4(quatro) tentaremos abrandar no Portal. Essas barreiras são: necessidade de encontrar uma tarefa fácil para iniciar, encontrar um especialista, orientação de como contribuir e desconhecimento do fluxo de contribuição. Essas barreiras estão destacadas na aba "Como Iniciar" dentro de cada projeto no Portal. Essa aba possui outras três, que são sobre o fluxo de contribuição, como escolher tarefa fácil e como encontrar um orientador.

A subcategoria de Solicitação de Mudança, da categoria de Obstáculos Técnicos, denota os problemas encontrados na fase de envio da contribuição. O novato consegue realizar intervenções no código, no entanto, não consegue encontrar uma informação adequada de como ele pode enviar sua contribuição, quando consegue, a resposta sobre a revisão de sua modificação é demorada. Essa situação tentamos amenizar representando, no Portal, na aba "Enviar contribuição", informações sobre como o novato pode enviar sua contribuição.

Na subcategoria de Obstáculos para criação de ambiente local, dentre as barreiras encontradas, uma refere-se ao problema do novato não conseguir configurar o seu espaço de trabalho. O novato tinha condição e vontade de contribuir com o código do projeto, no entanto, sem conseguir configurar o ambiente de trabalho adequadamente, esse desejo fica frustrado. Dessa forma, nosso Portal, aborda essa situação na aba "Configure seu espaço de trabalho", disponibilizando informações de ajuda para o novato configurar seu espaço de trabalho.

O Modelo de Barreiras, sistematizado por Steinmacher et al. (2014), aprofundou o conhecimento dos aspectos referentes às barreiras encontradas pelos novatos para ingressarem e contribuírem para os projetos de software livre. A importância desse trabalho está na possibilidade de disseminar ideias e mecanismos para facilitar a entrada de novatos em projetos de software livre. Os responsáveis e integrantes dos projetos, conhecendo e tendo ciência das dificuldades dos novatos, podem criar uma revisão do estado de todo o ambiente do projeto, verificando os pontos críticos para os novatos e assim adequá-los para ser absorvido e facilitar a interação do novato.

Sobre os dados obtidos do trabalho de Steinmacher et al. (2014), nosso Portal, tentará abordar algumas das barreiras evidenciadas. Para um trabalho mais completo e abrangente seria importante atingir o máximo de barreiras, no entanto, tempo e material humano foram limitados para esse fim. Por isso, inicialmente, o Portal se aterá por absorver uma quantidade de barreiras

condizentes com nossas condições.

Na seção 4.2, apresentaremos as interfaces desenvolvidas com base nos dados do Modelo de Barreiras de Steinmacher et al. (2014), bem como as barreiras que cada uma tenta abrandar. A relação das interfaces com as barreiras é apresentando, mais precisamente, na subseção 4.2.2, que mostra como os usuários irão visualizar as informações dos projetos.

4.2 MÓDULOS DO PORTAL

O projeto foi subdividido fundamentalmente em duas partes: visualização e manutenção. A parte de visualização refere-se à exibição das informações referentes de um projeto específico. A parte de manutenção destina-se ao cadastro, alteração e exclusão de projetos. A parte de visualização são as interfaces que exibem as informações dos projetos aos novatos.

4.2.1 MÓDULO AUTENTICADO/MANUTENÇÃO DE PROJETO

A seguir, nesta seção apresentamos as interfaces utilizadas para as tarefas de manutenção de informações de projeto.

4.2.1.1 CADASTRO DE USUÁRIO

O cadastro de conta do usuário é importante para que o usuário possa realizar atividades de manutenção de projetos, bem como para a realização de comentários sobre as informações de projetos. A criação de conta é realizada através do auto-cadastro, ou seja, o próprio usuário acessa o portal e realiza a inserção dos seus dados e confirma o cadastro.

A Figura 2, representa a interface de realização dessa etapa. É um formulário simples, onde o usuário informa o nome, e-mail e senha. Após a inserção dos dados, submete o formulário que após verificações realiza o cadastro do usuário ou não. Caso ocorra sucesso, o usuário estará apto a realizar cadastro, alteração e exclusão de projetos cadastrados por ele no Portal.

AJUDA NOVATOS Projetos ▾ option ▾ Cadastre seu projeto Cadastre-se Enter ▾

Cadastrar

*Nome

*Email

*Senha

*Confirmação de Senha

* Campos de preenchimento obrigatório

Figura 2: Cadastro de conta

4.2.1.2 CADASTRO DE INFORMAÇÕES BÁSICAS

Na Figura 3, apresentamos o formulário para a realização da inserção de informações básicas do projeto no portal. As informações básicas são: o nome, a página principal ou site do projeto, a data de criação, a página de download, o local onde se encontra o código fonte, o local onde se encontra o gerenciador de tarefas, o endereço da lista de e-mail e a descrição do projeto. No entanto, caso o mesmo projeto já esteja cadastrado no OpenHub, basta o usuário informar o nome do projeto, conforme Figura 4, que as informações básicas e outras informações mais específicas são carregadas automaticamente, diminuindo a replicação de trabalho.

Figura 3: Informações básicas

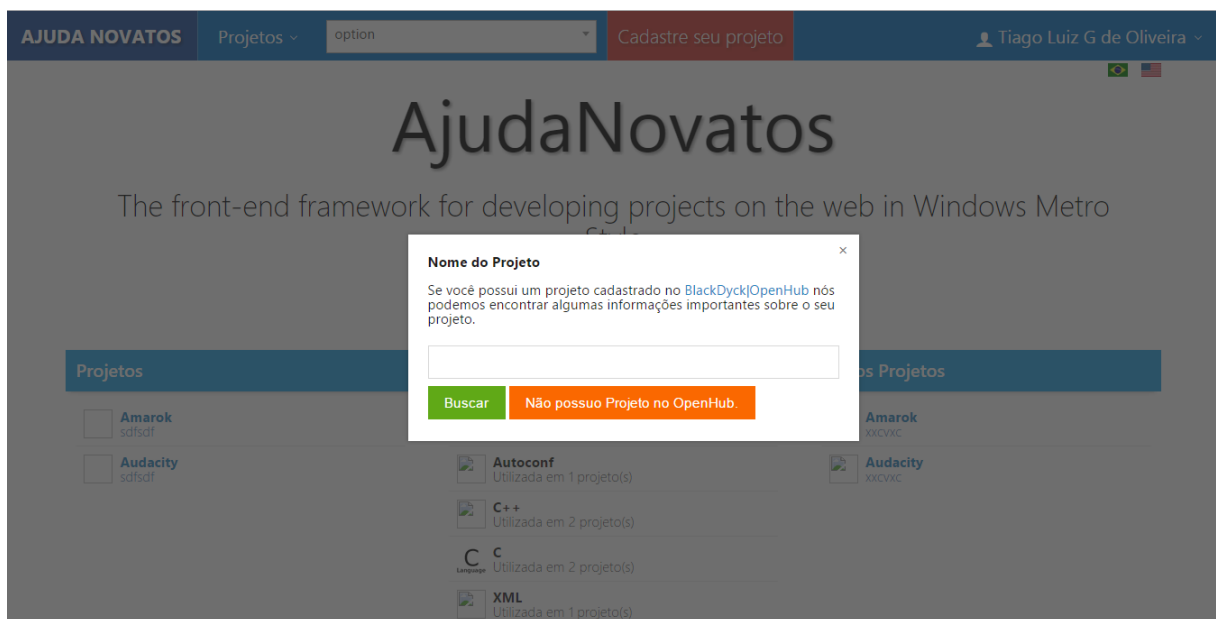


Figura 4: Busca de informações no OpenHub

4.2.1.3 CADASTRO DE REQUISITOS DO PROJETO

Os requisitos do projeto, são os conhecimentos básicos necessários para que o novato possa participar do projeto de forma mais fluente e natural. Os requisitos são divididos em Básicos e Adicionais. Os primeiros, são recursos utilizados no ambiente do projeto, tais como:

linguagens de programação, plataformas, entre outras, ideais para que o novato possa contribuir com o projeto. Os Adicionais, são características de comportamento e relacionamento por parte do novato dentro do projeto. A figura 5, exibe a interface de inserção dessas informações.

Figura 5: Adicionar Requisitos do projeto

4.2.1.4 CONSIDERAÇÕES SOBRE O FLUXO DE CONTRIBUIÇÃO

Nesta interface o usuário pode inserir considerações sobre o fluxo de contribuição do projeto. Na interface de visualização das informações sobre o fluxo de contribuição, é apresentado conforme a Figura 16, um esquema que dirige o novato para diferentes áreas do Portal. As considerações seriam informações complementares sobre o processo de contribuição dentro do projeto.

Adicionar meu projeto

[Salvar Projeto](#)
[Cancelar](#)
[DADOS](#)
[REQUISITOS](#)
[POR ONDE COMEÇAR?](#)
[ESPAÇO DE TRABALHO](#)
[COMUNICAÇÃO](#)
[ENTENDER CÓDIGO](#)
[SUBMISSÃO DE MUDANÇAS](#)
[FLUXO DE CONTRIBUIÇÃO](#)
[ENCONTRAR UMA TAREFA FACIL](#)
[ENCONTRAR UM AUXILIADOR](#)

O fluxo de contribuição é o processo no qual o novato deve passar até realizar sua primeira contribuição.

Condições sobre o fluxo de contribuição.

Figura 6: Adicionar informações sobre o fluxo de contribuição

4.2.1.5 ENCONTRAR TAREFA FÁCIL

Na Figura 7, apresentamos a interface de cadastro de informações sobre tarefas fáceis. Nessa interface o usuário pode cadastrar link para Feeds¹ de tarefas fáceis do projeto. Necessariamente o usuário deve filtrar esses feeds, por exemplo, no gestor de erros e verificar quais se adequam ao grau de conhecimento de um novato.

Juntamente a isso ele pode utilizar a área de texto para inserir informações adicionais, de outros locais que disponibilizam tarefas fáceis para que os novatos possam iniciar no projeto. Por exemplo, é possível informar arquivos que precisam de tradução, caso o novato, possua bom conhecimento, poderá começar dentro do projeto traduzindo arquivos.

¹Web Feed (vindo do verbo em inglês "alimentar") é um formato de dados usado em formas de comunicação com conteúdo atualizado frequentemente, como sites (sítios) de notícias ou blogs

FLUXO DE CONTRIBUICAO **ENCONTRAR UMA TAREFA FACIL** ENCONTRAR UM AUXILIADOR

Nesta área você pode informar aos usuários onde e como eles podem encontrar tarefas fáceis para iniciarem.
Uma sugestão é você filtrar/pesquisar na lista de arquivo do projeto, as tarefas fáceis para um iniciante e disponibilizar aqui, caso possua, os feeds dos arquivos.

O projeto possui Feeds para tarefas fáceis? Sim Não

Inserir feed

Título para do feed

URL para do feed

Inserir

Considerações sobre outras formas de encontrar uma tarefa fácil para iniciar.

File ▾ Edit ▾ Insert ▾ View ▾ Format ▾ Table ▾ Tools ▾

← → Formats **B** *I* [List icons] [Link icon] A ▾ A ▾

Figura 7: Adicionar informações sobre tarefas fáceis

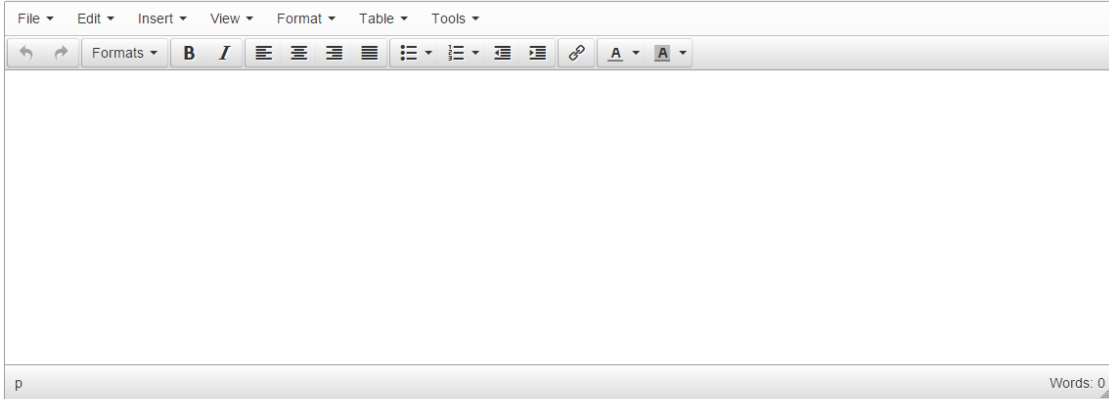
4.2.1.6 ENCONTRAR AUXÍLIO

Na Figura 8, o responsável pelo projeto pode informar onde é possível se encontrar os usuários mais experientes dentro do projeto. Para que assim, os novatos possam ter um ponto de referência em caso de dúvidas.

Adicionar meu projeto

[Salvar Projeto](#)[Cancelar](#)[DADOS](#)[REQUISITOS](#)[POR ONDE COMEÇAR?](#)[ESPAÇO DE TRABALHO](#)[COMUNICAÇÃO](#)[ENTENDER CÓDIGO](#)[SUBMISSÃO DE MUDANÇAS](#)[FLUXO DE CONTRIBUIÇÃO](#)[ENCONTRAR UMA TAREFA FACIL](#)[ENCONTRAR UM AUXILIADOR](#)

O projeto disponibiliza mentores aos novatos? Se sim, onde os mesmos podem ser encontrados?



The image shows a rich text editor interface. At the top, there is a menu bar with options: File, Edit, Insert, View, Format, Table, and Tools. Below the menu bar is a toolbar with various icons for text formatting, including bold (B), italic (I), bulleted list, numbered list, link, and text color. The main area of the editor is a large, empty white space. At the bottom of the editor, there is a status bar showing a paragraph symbol (p) on the left and 'Words: 0' on the right.

Figura 8: Encontrar auxílio

4.2.1.7 CADASTRO DE RECURSOS PARA CONFIGURAR O ESPAÇO DE TRABALHO

Um ambiente de trabalho bem organizado é importante para a produzir e contribuir com o código do projeto. Muitas vezes essa é uma tarefa complicada para o novato, pois, o mesmo desconhece a melhor forma e como configurar adequadamente o seu ambiente. Na Figura 9, é apresentada a funcionalidade de inserção dessas informações. O usuário pode adicionar informações e dicas de auxílio ao novato na configuração do seu espaço de trabalho, bem como, citando quais tipos de hardwares e plataformas indicadas, se a configuração do espaço de trabalho é mais indicada para o sistema operacional linux, windows ou mac, entre outras formas de configurações.

Figura 9: Cadastro de recursos do projeto para configurar o Espaço de Trabalho

4.2.1.8 CADASTRO DE INFORMAÇÕES PARA SUBMISSÃO DE MUDANÇAS

Na literatura e no modelo de barreiras de Steinmacher et al. (2014), uma das dificuldades relatadas pelos novatos é no que se diz respeito à submissão das mudanças feitas por eles. O novato realiza suas mudanças no projeto, porém, não sabe como submetê-las. Em virtude dessa situação, buscamos disponibilizar área para inserção dessas informações. Na figura 10, é possível se visualizar como o responsável pelo cadastro do projeto pode realizar a inserção e disponibilização dessas informações aos novatos, mostrando a ele quais procedimentos necessita tomar para enviar sua primeira contribuição.

AJUDA NOVATOS Projetos option Cadastre seu projeto Tiago Luiz G de Oliveira

Adicionar meu projeto

Salvar Projeto Cancelar

DADOS REQUISITOS POR ONDE COMEÇAR? ESPAÇO DE TRABALHO COMUNICAÇÃO ENTENDER CÓDIGO SUBMISSÃO DE MUDANÇAS

Aqui você deve fornecer ao novato informações de como ele pode realizar a sua primeira contribuição

File Edit Insert View Format Table Tools

Formats B I [List Icons] [Link Icon] A A

Figura 10: Cadastro de Informações para Submissão mudanças

4.2.1.9 CADASTRO DOS CANAIS DE COMUNICAÇÃO

A comunicação com os demais integrantes da comunidade pode ser importante para o novato. A falta de canais de comunicação com o restante da comunidade e dificuldade de acessar esses canais, pode dificultar para o novato a busca de informações com outros participantes do projeto, e assim, também pode prejudicar seu estabelecimento no projeto. Por isso, é importante, caso o projeto possua, apresentar esses mecanismos ao novato, para que ele possa participar de discussões, divulgação de melhorias, relato de erros e solicitação de ajuda. Na Figura 11, é apresentado a interface em que o responsável pelo projeto irá informar onde se encontram o endereço dos canais de comunicação usados pelo projeto.

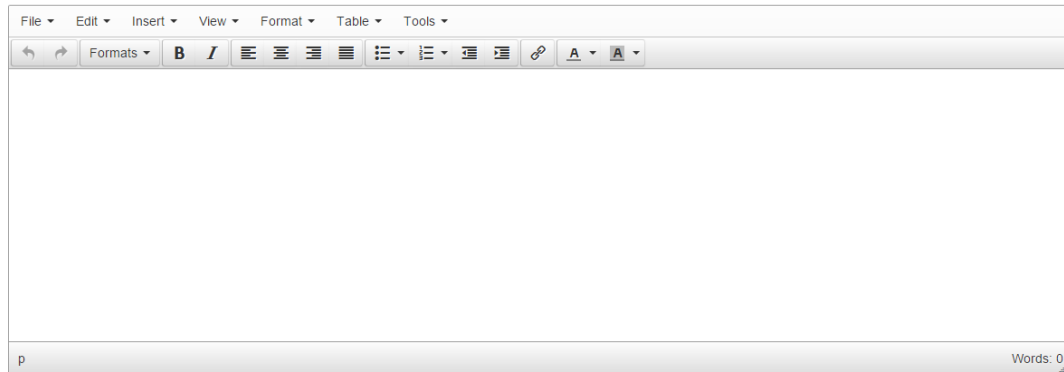
Esta parte destina-se a apresentar ao novato os canais de comunicação que são utilizados pelo projeto.

O projeto possui IRC? Sim Não

Informe o endereço para a página do chat do projeto.

URL para o Chat IRC do projeto

Considerações sobre o IRC.



The image shows a web form for configuring IRC channels. It features a menu bar with 'File', 'Edit', 'Insert', 'View', 'Format', 'Table', and 'Tools'. Below the menu is a toolbar with icons for undo, redo, bold, italic, bulleted list, numbered list, link, and text color. The main area is a large empty text box. At the bottom left, there is a small 'p' icon, and at the bottom right, it says 'Words: 0'.

Figura 11: Canais de comunicação

4.2.2 MÓDULO DE VISUALIZAÇÃO DO PROJETO

Nesta seção são apresentados visão das interfaces de visualização das informações dos projetos.

Na Figura 12, apresentamos a página inicial do sistema. Aqui o novato pode visualizar a descrição do Portal, ou seja, para que ela serve e o que o novato irá encontrar. Na barra superior há um menu para visualização dos projetos; um autocomplete para busca dinâmica pelo nome do projeto; botão de opção de cadastro do projeto; botão de opção de criação de conta; e, menu para acesso ao formulário de login ao portal.



Figura 12: Tela Inicial

Após o novato selecionar ou buscar um projeto, ele visualiza a página do projeto. Esta será organizada da seguinte forma: Descrição do projeto, Requisitos, Como iniciar no projeto, Canais de comunicação, Como entender o código e Ajuda para enviar contribuição. Inicialmente é apresentado uma descrição do projeto, sua finalidade, funções e aplicações, entre outros detalhes. Tais possibilidades de interação baseiam-se no modelo de barreiras apresentado anteriormente.

Na Figura 13, apresentamos a interface de exibição dos dados do projeto cadastrado. Há descrição do projeto, site do projeto, links que incluem a página de wiki do projeto, gerenciador de erros, entre outros. Além disso existe gráficos que exibem informações dos projetos que foram extraídas do OpenHub. Na Figura 14, mostram os gráficos referentes às linguagens e seus respectivos percentuais de participação no projeto, e sobre o código do projeto que exibe o total de linhas de código, total de linhas em branco e total de comentários.

Os dados para criação dos gráficos, como já dito anteriormente, são obtidos da API do OpenHub. Quando o usuário fornece o nome o projeto, na etapa de cadastro do projeto, o Portal realiza consulta na API do OpenHub. Caso o projeto esteja presente no OpenHub, a API retorna os dados em formato XML. O Portal processa o XML, do qual retinamos as informações exibidas na aba "Sobre" do projeto. Vale ressaltar que os gráficos referem-se e são possíveis de serem gerados, somente se o projeto a ser cadastrado, estiver no OpenHub.

AJUDA NOVATOS
Projetos ▾
Cadastre seu Projeto
joao da silva ▾

LibreOffice

Curtir
Compartilhar

SOBRE
REQUISITOS
COMO INICIAR
FALE COM A COMUNIDADE
CONFIGURE SEU ESPAÇO DE TRABALHO
ENTENDER O CÓDIGO
ENVIAR CONTRIBUIÇÃO

LibreOffice é uma suite de escritório integrada com base em licenças copyleft e compatível com a maioria dos formatos e padrões de documentos. LibreOffice é apoiado pela The Document Foundation, o que representa uma grande comunidade independente de empresas, desenvolvedores e outros voluntários movidos pelo objetivo comum de trazer ao mercado o melhor software para produtividade pessoal. LibreOffice é open source e gratuito para download, usar e distribuir.

Total de Linhas de Código do Projeto **7170283**

Total de Commits **416561**

Número de participantes no projeto **1372**

Site do projeto: <http://www.libreoffice.org>

Links:

- [Wikipedia](#)
- [Wiki](#)
- [Nabble Web Acces to Mailing Lists](#)
- [Ask Libreoffice](#)
- [Bugzilla](#)
- [Easy Hacks](#)
- [Developer starter page](#)

Figura 13: Sobre o Projeto

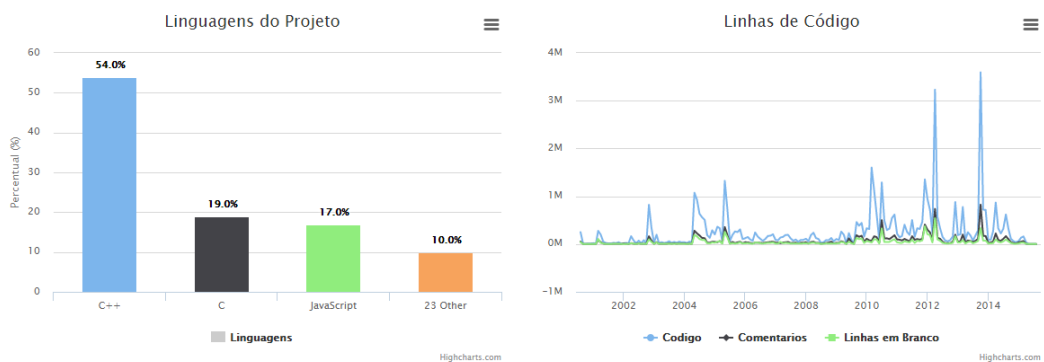


Figura 14: Graficos: Linguagens e Linhas de código do Projeto

Na Figura 15, são apresentadas os requisitos do projeto, ou seja, os habilidades e requisitos necessários para um bom andamento no projeto. Nesta tela o novato, tem noção do que ele precisa entender e aprender, como por exemplo, sobre as linguagens, características sociais e recursos disponíveis. É uma forma de ele medir se os seus conhecimentos e habilidades são ideais e suficientes para o projeto. Caso ele julgue suas habilidades não tão adequadas ao que foi indicado, ele já saberá o que pode melhorar. Por exemplo, se o projeto é baseado em Java e o novato não possui muita familiaridade com a linguagem, e mesmo assim queira participar, ele já terá em mente que aprender e se familiarizar com Java será necessário.

A exibição dos Requisitos Básicos dessa interface relaciona-se com as barreiras da

subcategoria Falta de formação técnica do Modelo de Barreiras. Esta subcategoria trata das características dos novatos, sobre a falta de conhecimento em recursos utilizados pelos projetos, como: controles de versão e testes de unidade. Os Requisitos Adicionais, referem-se ao comportamento desejado, que o participante tenha dentro do projeto, eles relacionam-se com a subcategoria de Comportamento do Modelo de Barreiras.

The screenshot shows the 'Ajuda Novatos' website interface for the LibreOffice project. At the top, there is a navigation bar with 'AJUDA NOVATOS', a dropdown menu for 'Projetos', a search bar, a 'Cadastre seu Projeto' button, and a user profile for 'joao da silva'. Below the navigation bar, the LibreOffice logo is displayed. A secondary navigation bar contains tabs for 'SOBRE', 'REQUISITOS', 'COMO INICIAR', 'FALE COM A COMUNIDADE', 'CONFIGURE SEU ESPAÇO DE TRABALHO', 'ENTENDER O CÓDIGO', and 'ENVIAR CONTRIBUIÇÃO'. The 'REQUISITOS' tab is selected, showing two columns of requirements:

- Requisitos Básicos**
 - Algoritmos são habilidades necessárias
 - A linguagem de programação predominante é C / C ++. No entanto, Java e Python já são utilizados.
 - O código é armazenado em um repositório GIT.
 - O projeto utilizou Gerrit como a ferramenta de revisão de código. Você deve enviar seus patches para lá.
 - Linha de comando é usada para construir o projeto.
 - Linux é desejável, no entanto, é possível utilizar outro SO
- Requisitos Adicionais**
 - Proatividade, para tentar resolver problemas antes de pedir ajuda à comunidade;
 - Compromisso com as tarefas que você escolheu;
 - Respeite a comunidade, eles estão desenvolvendo um produto comercial, confiável, utilizado por milhões de pessoas, eles precisam manter alta qualidade;
 - Entenda as diferenças, algumas pessoas são diretas, críticas ou parecem ser rudes. Se você considerar uma mensagem inadequada ou rude, tente entender que há pessoas de diferentes culturas envolvidas. Solicite mais informações, se você não entender uma resposta recebida. Mas, sempre aja de forma respeitosa.

Figura 15: Características do Projeto

Um dos pontos destacados no Modelo de Barreiras de Steinmacher et. al. (2014), é a necessidade de orientação sobre o fluxo de contribuição dos projetos. Ou seja, o novato não tem noção dos passos, do que é envolvido até que ele possa chegar a contribuir com o projeto. Na Figura 16, é apresentado a interface onde o novato pode visualizar considerações sobre o fluxo de contribuição do projeto, e, um esquema que mostra os pontos de conhecimento necessário para que ele possa contribuir com o projeto. Ou seja, para se chegar até o ponto de contribuir para o projeto o novato tem algumas etapas, que é desejável que ele siga para realizar sua contribuição. Notoriamente antes de chegar até a contribuição ele pode alternar entre as etapas, não necessariamente precisando seguir a ordem indicada na Figura 16.

O fluxo exibido na Figura 16, é o indicado para um novato. Para contribuir com o projeto, ele inicia verificando suas habilidades, se são suficientes para contribuir com o projeto; Configura o seu espaço de trabalho, para poder contribuir com o código; Procura uma tarefa dentro do projeto que seja fácil para que possa iniciar; e, faz a submissão da(s) sua(s) mu-

dança(s). Em todas essas etapas ele pode recorrer ao suporte da comunidade, buscando ajuda nos fóruns, lista de discussões ou através de bate-papo.

A interface representada na Figura 16, tenta abrandar justamente os aspectos referentes à barreira Desconhecimento do fluxo de contribuição da Categoria de Necessidade de orientação. Ao selecionar uma das opções no fluxo, o usuário é direcionado para uma área vinculada dentro do Portal. Por exemplo, ao selecionar a opção "Configurar Workspace", o Portal exibe as informações da aba "Configure seu espaço de trabalho".



Figura 16: Fluxo de contribuição do projeto

Uma das dificuldades relatadas no modelo de barreiras é a falta de orientações de como encontrar uma tarefa fácil para iniciar. Não saber e não ter informações de como começar e iniciar com uma tarefa muito complexa pode trazer muitas dificuldades para o novato e até mesmo acabar levando-o a desistir do projeto. Assim, começar com uma tarefa que seja mais fácil possibilita a ele interagir com o projeto, realizar sua primeira contribuição, se familiarizar e aprender de forma mais efetiva o código e outros aspectos do projeto. Na Figura 17, é apresentado como o novato visualizará as tarefas fáceis que o projeto possui.

As tarefas fáceis são filtradas pela equipe do projeto, que posteriormente pode inserir, caso possua, feed para exibir lista de tarefas fáceis. Ou então, informar manualmente por meio de texto, links e imagens exibindo ao novato opções para acessar as tarefas fáceis. Os demais usuários cadastrados no Portal, podem contribuir com comentários sobre as tarefas fáceis apre-

sentadas na interface da Figura 17, indicando quais tarefas estão em desacordo com o nível dos novatos ou quais seriam mais adequada.

Com a exibição de opções de tarefas fáceis ao novato no Portal, tentamos amenizar os aspectos negativos da barreira Encontrar uma tarefa fácil para iniciar da categoria Necessidade de Orientação. Para que assim, o novato possa ir progressivamente ampliando sua participação no projeto e tendo condições de se envolver em tarefas mais complexas.

LibreOffice

Curir Compartilhar

SOBRE REQUISITOS **COMO INICIAR** FALE COM A COMUNIDADE CONFIGURE SEU ESPAÇO DE TRABALHO ENTENDER O CÓDIGO ENVIAR CONTRIBUIÇÃO

Aqui você encontra informações de como iniciar no projeto.

FLUXO DE CONTRIBUIÇÃO **ESCOLHER UMA TAREFA FÁCIL** ENCONTRAR UM ORIENTADOR

Como encontrar uma tarefa fácil para iniciar

Use os comentários para informar a comunidade que está a trabalhar nessa tarefa e também para obter apoio / pedir ajuda.

Tarefas fáceis para todas as habilidades em todos os módulos	Tarefas fáceis para C++ em todos os módulos
<p>[Bug 38841] Fuzz XML file filters Field Value Product LibreOffice Component LibreOffice AssignedToName Not Assigned ReportedByName Björn Michaelsen Status NEW Resolution --- Priority... Apr 22, 2014 1:59:05 PM</p> <p>[Bug 38844] Reduce XOR rendering Field Value Product LibreOffice Component Draw AssignedToName Not Assigned ReportedByName Björn Michaelsen Status NEW Resolution --- Priority mediu... Sep 8, 2014 8:20:00 AM</p> <p>[Bug 54169] LibO doesn't obey OS setting "only show ..." Field Value Product LibreOffice Component UI AssignedToName Not Assigned ReportedByName Samuel Mehrbrodt Status NEW Resolution --- Priority high Se... Jul 10, 2015 6:24:44 PM</p> <p>[Bug 55066] FILEOPEN: flaws in importing 123 files Field Value Product LibreOffice Component Calc AssignedToName Not Assigned ReportedByName Horst Status NEW Resolution --- Priority high Severity no... May 21, 2015 7:26:00 AM</p>	<p>[Bug 54169] LibO doesn't obey OS setting "only show ..." Field Value Product LibreOffice Component UI AssignedToName Not Assigned ReportedByName Samuel Mehrbrodt Status NEW Resolution --- Priority high Se... Jul 10, 2015 6:24:44 PM</p> <p>[Bug 55066] FILEOPEN: flaws in importing 123 files Field Value Product LibreOffice Component Calc AssignedToName Not Assigned ReportedByName Horst Status NEW Resolution --- Priority high Severity no... May 21, 2015 7:26:00 AM</p> <p>[Bug 70625] Add --cat parameter to make git diffs pr... Field Value Product LibreOffice Component LibreOffice AssignedToName Deena Francis (lids co in) ReportedByName Michael Meeks Status NEW Resolution ... Aug 15, 2014 1:52:38 PM</p> <p>[Bug 64575] Photo Album: Better Image Layout Field Value Product LibreOffice Component Impress AssignedToName Enes Ateş ReportedByName Samuel Mehrbrodt Status NEW Resolution --- Priority mediu... Mar 22, 2015 5:13:08 AM</p>

Figura 17: Tarefa fácil para iniciar

Em diversas situações os novatos necessitam de ajuda de alguém mais experiente dentro do projeto. No trabalho de Steinmacher et. al. (2014), que resultou no Modelo de Barreiras, uma das barreiras identificadas foi a de "Encontrar um especialista" da categoria de "Necessidade de orientação". Caso o novato possua dificuldade na realização de alguma tarefa dentro do projeto, ele pode recorrer aos integrantes mais experientes dentro do projeto ou até um integrante relacionado com a tarefa em que ele está trabalhando.

Na Figura 18, apresenta-se interface que exibe dicas de como o novato pode encontrar ajuda de outros integrantes mais experientes em determinada tarefa. A figura representa a situação de um erro o Bug 71043 do projeto LibreOffice. Em destaque, com bordas, temos a lista de usuários relacionados ao erro. Assim, o novato pode pedir informação sobre algum detalhe do projeto, de implementação, ou seja, pode solicitar mais detalhes para ajudá-lo em contribuir para a resolução do erro.

Com essa interface, mostrada na Figura 18, objetiva-se tentar ajudar o novato a encontrar pessoas mais experientes no projeto, reduzindo/amenizando os aspectos negativos da barreira "Encontrar um especialista".

SOBRE REQUISITOS **COMO INICIAR** FALE COM A COMUNIDADE CONFIGURE SEU ESPAÇO DE TRABALHO ENTENDER O CÓDIGO ENVIAR CONTRIBUIÇÃO

Aqui você encontra informações de como iniciar no projeto.

FLUXO DE CONTRIBUIÇÃO ESCOLHER UMA TAREFA FÁCIL **ENCONTRAR UM ORIENTADOR**

Como encontrar ajuda de outros participantes

É possível tirar dúvidas e obter o apoio de desenvolvedores usando lista de discussão e IRC

Quando você escolhe uma tarefa que você pode pedir ajuda e orientação no Bugzilla, comentando a tarefa e pedindo apoio adequado.

Você pode verificar quem são os especialistas por olhando para a lista CC da tarefa (como você pode observar na Figura abaixo).

Você também pode verificar a página Encontre o especialista, que lista os especialistas por módulo LibreOffice.

Bug 71043 - Use STACK lint tool to clean code ...

Status: NEW

Product: LibreOffice

Component: Libreoffice

Version: 4.2.0.0.alpha0+ Master

Platform: Other All

Importance: medium normal

Assigned To: José Guilherme Vanz

QA Contact:

Reported: 2013-10-30 11:30 UTC by Michael Meeks

Modified: 2014-07-09 11:17 UTC ([History](#))

CC List: 5 users

- dennisrocze
- libreoffice
- lo_bugs
- sbergman
- serval2412

Figura 18: Como encontrar um orientador

A falta de atenção adequada às questões enviadas por novatos também é um dos problemas identificados no modelo de barreiras. Ou seja, os novatos não recebem resposta, ou então as respostas acontecem atrasadas, são indelicadas ou possuem conteúdo muito avançado para o grau de conhecimento do novato. Nesse aspecto, os canais de comunicação são mecanismos importantes para os novatos. Por exemplo, o canal de comunicação IRC, permite ao novato se conectar e realizar interações com os outros envolvidos na comunidade, fortalecendo seus laços sociais dentro do ambiente, aprendendo mais sobre a comunidade e projeto, tirando dúvidas, realizando comentários entre outras interações. Outra opção, que pode evitar o desconforto e a má recepção para com os novatos, é a pesquisa nos canais de comunicação do projeto.

Na Figura 19, tem-se como destaque a opção do usuário realizar uma busca antes de perguntar. Esta opção tenta abranger as barreiras da categoria de "Problemas de recepção", mais especificadamente as de não recebimento de resposta, respostas atrasadas e respostas indelicadas. O novato informará o conteúdo de dúvida que ele deseja obter ajuda e o Portal exibe outra aba do navegador, mostrando os resultados de busca na lista de discussão do projeto, com auxílio do motor de busca do Google.

Ao darmos a opção para o novato tentar realizar busca de suas dúvidas na lista de discussão do projeto, automaticamente poderemos reduzir a necessidade dele realizar incursões junto aos demais integrantes do projeto a fim de tentar resolver uma dúvida que já foi discutida em algum momento. Pois, pode ocorrer que ao solicitar uma informação que já foi discutida para um usuário mais experiente, ele pode agir de forma não tão cordial, ser grosso ou simplesmente dizer: - procure nos arquivos do projeto! Assim, conseqüentemente o novato irá realizar menos solicitações que podem ser desnecessárias e evitar situações desagradáveis com outros participantes do projeto.



Figura 19: Pesquisar antes de perguntar

O bate-papo é uma forma de relacionamento importante. Por meio dele os usuários podem realizar interação em tempo real e divulgar suas ações, discutir soluções e melhorias, solicitar ajuda, entre outras situações. No portal o novato também pode ter acesso, caso o projeto possua, ao bate-papo IRC. Na Figura 20, é apresentado a interface de como é a visualização do IRC dos projetos.

O serviço de IRC exibido no Portal, são os mesmos utilizados no projeto. O objetivo de disponibilizar uma área de acesso ao chat do projeto é de evitar que o novato precise sair da página do Portal para chegar até esse mecanismo. Ironicamente o chat pode ser uma fonte de respostas indelicadas e de frustrações para o novato. No entanto, essa é uma situação que se pode prever e evitar, mas em algum momento o novato irá necessitar recorrer a um mecanismo desse tipo.

No chat, o novato pode interagir e se ambientar em tempo real com vários integrantes do projeto, com suas personalidades e particularidades. Isso pode significar uma amenização da barreira Timidez da subcategoria de Comunicação.

Canal de comunicação do projeto

Aqui você pode esclarecer suas dúvidas de forma síncrona, recebendo respostas de prompt do LibreOffice Developers.

Comece apresentando-se como um recém-chegado e dizendo que você está tendo problemas

Por exemplo:

Olá, eu sou um novato tentando colocar a minha primeira contribuição para Libre Office. Estou enfrentando problemas [durante os meus primeiros passos / encontrando uma tarefa / configurar o meu espaço de trabalho].

Alguém pode me ajudar [esclarecendo algumas dúvidas]

Conecte-se ao canal especificado abaixo, espere até que você entrar no canal, e interagir com os desenvolvedores nice, atenciosa e experientes

The screenshot shows a web interface titled "Connect to freenode IRC". It contains the following elements:

- Nickname:** A text input field containing "newcomer_mycoach4".
- Channels:** A text input field containing "#libreoffice-dev".
- Auth to services:** A checkbox that is currently unchecked.
- reCAPTCHA:** A section with a visual captcha image showing the number "1585" and a "Reload" button.
- Audio captcha:** A checkbox that is currently unchecked.
- Connect:** A button at the bottom right of the form.

Figura 20: Canal de Bate-papo IRC

A Lista de discussão, é um mecanismo que permite a troca de mensagens via e-mail entre um grupo de pessoas. Por meio dela, também é possível que se estabeleça a discussão e debate de determinado assunto com os participantes que estão cadastrados nesta lista.

Diferentemente do IRC, que é em tempo real, a Lista é o contrário. Ela é mais conveniente na impossibilidade de tempo para parar e fazer conversação em tempo real com outros integrantes. Ou seja, por exemplo, o novato precisa de ajuda para solucionar um problema. Ele pode enviar essa dúvida para todos os associados na lista do projeto, que responderão a mensagem em um momento mais apropriado, podendo também pesquisar reunir, pesquisar mais informações e elaborar uma resposta mais completa a essa dúvida. Da mesma forma o novato, caso esteja associado na lista de e-mail, também receberá todas as mensagens enviadas, por outros participantes, para o grupo.

Na lista de discussão, o comportamento dos seus participantes deve ser mais formal, ou seja, as mensagens devem ser o mais claro e objetivo possível, evitando fugir do assunto tratado. A Figura 21, mostra a interface de visualização de informações sobre a Lista de Discussão do projeto. Essa interface exhibe informações ao novato de como ele pode se inscrever na lista do

projeto e de como ele deve se comportar na lista.

O Modelo de Barreiras de Steinmacher et. al. (2014), apresenta barreiras identificadas sobre os problemas de comunicação dos novatos. Na subcategoria de Comunicação, tem-se barreiras como o não envio de mensagem significativa e comentários inúteis na lista de discussão/fórum. Assim, no Portal, ao darmos sugestão de comportamento e de mensagem para envio na lista de discussão, tentamos evitar a ocorrência desse tipo de situação.

PESQUISE ANTES DE PERGUNTAR BATE-PAPO(IRC) **LISTA DE DISCUSSÃO**

Inscrevendo-se na lista

Há uma desenvolvedores lista para os nossos desenvolvedores de discussão, e para codificadores interessado em fazer contribuições de código. Você pode se inscrever e obter mais informações a partir daqui:

<http://lists.freedesktop.org/mailman/listinfo/libreoffice>

Você também pode verificar se o seu país tem uma lista de discussão regionais aqui: https://wiki.documentfoundation.org/Local_Mailing_Lists

**** Mensagem Importante: Por favor, lembre-se: tudo o que você postar para as listas de discussão públicas, incluindo o seu endereço de e-mail e outras informações pessoais contidas em sua mensagem, será arquivada publicamente e não pode ser excluído. Então, por favor, faça pós sabiamente

Como se comportar na lista

OK! Se você não encontrou qualquer solução para o seu problema ou se preferir sen um e-mail para a lista de discussão, seguir em frente.

Se ele é o primeiro e-mail que você está enviando, por favor:

- Ser educado, a comunidade é composta de voluntários
- Apresentar-se
- Identificar-se como um recém-chegado
- Apresentar as suas motivações e seus objetivos
- Mostram que você tentou resolver o problema e encontrar a solução
- Perguntar diretamente a sua pergunta
- Enviar uma mensagem significativa e correta (pedir a alguém para revisar o Inglês, se você não está confiante)

Sugestão Mensagem

Olá,

Meu nome é [seu nome] e sou um novato tentando colocar a minha primeira contribuição de código para o LibreOffice. Estou enfrentando problemas [durante os meus primeiros passos / encontrando uma tarefa / configurar o meu espaço de trabalho]. Alguém pode me ajudar [esclarecendo algumas dúvidas / me mentoring]?

Já tenho [mencionar as coisas que você tem feito já para tentar resolver o seu problema] [Se você estiver recebendo um erro, incluir na mensagem]

[Mencione o sistema operacional que você está trabalhando e as ferramentas que você está usando]

Desde já, obrigado

Figura 21: Informações sobre a Lista de Discussão

No trabalho de Steinmacher et. al. (2014), uma das barreiras relatadas pelos novatos, é quanto a construção do espaço de trabalho local. Ou seja, os novatos possuíam dificuldades em criar e configurar adequadamente o seu ambiente de desenvolvimento. Assim, o objetivo da interface apresentada na Figura 22, é tentar sanar esse problema, dando possibilidade ao novato de poder acessar informações de auxílio para esse fim.

Como exemplo, a figura exhibe opção de acesso para informações de configuração de ambiente de trabalho em tipos de sistemas operacionais distintos. Além disso, o novato tem opção de realizar uma busca de ajuda na lista de discussão do projeto. Os integrantes do projeto, podem definir as informações mais precisamente sobre o processo de configuração e criação do espaço de trabalho e disponibilizar ao novato. Isso pode evitar que o novato tenha que ficar vasculhando a internet em busca de materiais de auxílio.



The screenshot shows the LibreOffice website's navigation menu with the following items: SOBRE, REQUISITOS, COMO INICIAR, FALAR COM A COMUNIDADE, CONFIGURE SEU ESPAÇO DE TRABALHO (highlighted), ENTENDER O CÓDIGO, and ENVIAR CONTRIBUIÇÃO. There are also buttons for 'Curtir' and 'Compartilhar'.

Mecanismos de ajuda para configurar seu espaço de trabalho

Recursos disponibilizados pelo projeto

Você pode usar os seguintes recursos fornecidos pelo projecto de criação de seu espaço de trabalho.

Configurar espaço de trabalho (Linux):

- Construir dependências em Linux
- Como construir diretrizes gerais para a construção no Linux.
- Tutorial em vídeo (Youtube)
- Correr LibreOffice em um IDE (KDevelop) - Linux

Configurar espaço de trabalho (Windows):

- Construir dependências no Windows
- Executando Como construir em Windows - Dicas e Truques
- Correr LibreOffice de VisualStudio
- Setup Workspace Local (Mac):

Como construir Espaço de Trabalho no Mac

- Como construir local de trabalho para Mac

Perguntar se o projeto possui endereço do histórico de arquivos da lista de email
 Buscar uma solução no Google.

Buscar no Google

Figura 22: Ajuda para configurar workspace

Outra dificuldade dos novatos apresentada no Modelo de Barreiras, é o localizar documentações do projeto. A documentação é uma fonte importante de informações tanto para novatos quanto para desenvolvedores permanentes do projeto, no entanto, o acesso e localização da mesma pelo novato é bem mais difícil. Na Figura 23, é apresentado a interface de visualização das informações sobre onde encontrar documentação do projeto, convenções de código, entre outras documentações referentes ao projeto.



Figura 23: Entender código

No portal foi disponibilizado espaço para usuários novatos realizarem comentários e avaliar cada um tópicos. No entanto, para evitar comentários feito a esmo, será necessário a realização de cadastro de login.

4.3 RECURSOS UTILIZADOS

Para o desenvolvimento dessa aplicação, as seguintes tecnologias foram escolhidas:

- Java Server Pages(JSP)

Como é um sistema web dinâmico, necessitamos de um serviço de páginas que nos possibilite esse recurso. Esta tecnologia também devido a maior familiaridade com a linguagem Java, do tempo reduzido para o aprendizado de uma nova tecnologia e pelo fato do JSP ser escrito nessa linguagem.

JSP é uma linguagem de script com especificação aberta que tem como objetivo primário a geração de conteúdo dinâmico para páginas da Internet. Ela possibilita a criação de dinamismo nas páginas. Com HyperText Markup Language (HTML) puro, não conseguiríamos fazer o dinamismo em nossas páginas, por exemplo, consultar por critérios específicos, uma lista de usuários, no banco de dados e exibir na interface do usuário.

Isso seria não seria possível com o HTML puro, no entanto, com a utilização do JSP, conseguimos realizar essa tarefa. Adicionamos código JSP no HTML, o servidor interpreta e gera o código HTML necessário. Como é gratuita e por ter sua especificação aberta, existem vários servidores que suportam JSP, entre eles podemos destacar: Tomcat, GlassFish, JBoos, entre outros. Em nosso Portal estaremos utilizando o GlassFish. Um servidor desses é imprescindível, pois, como já mencionado, ele irá interpretar o código JSP e apresentar ao usuário somente o HTML.

Mais sobre a especificação JSP pode ser encontrada em: <http://www.oracle.com/technetwork/java/javaee/jsp/index.html>.

- **Mysql Database**

O Portal reunirá informações e dados que necessitam estar disponíveis permanentemente para diversos usuários, da mesma forma necessitamos garantir a consistência e integridade desses dados. Para que isso seja possível, necessitamos utilizar um Sistema de Gerenciamento de Banco de Dados (SGBD) que garanta a consistência dos dados e realize este procedimento. Para o desenvolvimento do Portal utilizamos o Mysql.

O Mysql é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. É atualmente um dos bancos de dados mais populares. Entre suas características podemos destacar o excelente desempenho e estabilidade, e compatibilidade com diversas linguagens de programação.

O MySQL é um produto regido pela licença General Public License (GPL). E pode ser encontrado em <https://www.mysql.com/>.

- **Hibernate ORM**

O hibernate é um framework de mapeamento relacional para java. O hibernate é distribuído com a licença LGPL v2.1, podendo ser utilizado gratuitamente tanto em projetos comerciais como projetos de código aberto.

O hibernate abstrai o processo de persistência no banco de dados. Todo processo de comunicação entre a aplicação e o banco de dados é intermediado por ele, essa característica contribui para que o desenvolvedor evite ter que ficar se preocupando em escrever instruções SQL para realizar operações de recuperação ou persistência de dados do software.

Todo o mapeamento do objeto relacional, ou seja, as tabelas do existentes no nosso banco de dados são representadas através de classes Java em nossa aplicação, no entanto nem todas as nossas classes vão representar tabelas, pois, há classes que necessitam ser persistidas, já outras funcionam apenas como Beans. O hibernate possui diversos métodos que são utilizados para operações de recuperação e persistência dos dados, isso evita que o programador fique se preocupando com instruções SQL como selects, join e entre outras. Além disso, o framework pode ser capaz até de resolver as particularidades com que cada Sistema de Gerenciamento de Banco de Dados - SGDB, funciona.

O hibernate será utilizado para gerenciar o processo de persistência dos dados no banco de dados evitando a criação manual de queries sql. Mais sobre o hibernate pode ser encontrado em: <http://hibernate.org/orm/>.

- Metro UI CSS

Framework css, que será utilizado para proporcionar ao portal uma interface mais amigável.

Um framework css, serve para reduzir o esforço de entendimento e aprendizado do CSS, trazendo uma série de vantagens como, por exemplo: a padronização do estilo da página; arquivos modularizados; flexibilidade do estilo, que pode ser alternado e combinado de diversas maneiras dentro da página; redução de tempo, pois, os designer não precisará desenvolver o estilo da página do início, podendo assim, se preocupar apenas com outros aspectos particulares da página; e, reduz possíveis esforços de manutenção da página.

O Metro UI CSS, é distribuído sob a licença MIT. Esta licença permite que seja obtida cópia gratuita do framework e seus arquivos associados. E ainda, entre outros usos, modificar, mesclar, publicar e negociá-los sem restrições, dando os devidos créditos ao autor do framework, incluindo o aviso Copyright (c) 2012-2015 Sergey Pimenov nas cópias inteiras do software ou partes dela.

O Metro UI CSS, está disponível para download na sua página <http://metroui.org.ua/> e atualmente se encontra na versão 3.

- Vraptr 4

Framework que será utilizado para a padronização do projeto web, seguindo as especificações do Padrão de Projeto Modelo Visão e Controle (MVC).

Um framework MVC, divide a representação da informação em três partes, Modelo(model), Visão(view) e Controle(controller). O modelo se trata dos dados da aplicação, das regras de negócio, lógica e funções. A visão consiste nas saídas que representam os dados, como por exemplo, uma tabela com nome de pessoas cadastradas no banco de dados da aplicação. O controle é o responsável pela intermediação entre o modelo e a visão.

Atualmente o Vraprot, encontra-se na versão 4, mantido pela Caelum e disponibilizado sob os termos da Licença Apache. Download e mais informações sobre o framework podem ser encontradas em: <http://www.vraprot.org/>.

- Jquery

Biblioteca Javascript, que será utilizada para evitar a incompatibilidade de implementação Javascript em diferentes navegadores. Jquery faz a compactação de tarefas que necessitam de várias linhas de código para serem representadas e disponibiliza funções que produzem os mesmos efeitos. Assim, o programador utiliza menos código para suas funções javascript.

Com Jquery é possível fazer manipulação do HTML e do CSS, fazer efeitos e animações na página, realizar chamadas assíncronas AJAX, entre outras funcionalidades genéricas.

Em nosso portal utilizamos Jquery em muitas situações, por exemplo, para realizar a remoção de um projeto. O usuário visualiza os seus projetos, e seleciona a opção de excluir, o identificados do projeto é enviado ao método responsável no controle, após a remoção o projeto é removido da página. Isso, sem que a página toda seja enviada e recarregada. Assim economizamos a quantidade de informações enviadas pela rede ao servidor.

Jquery é distribuído sob os termos da Licença MIT, e disponibilizado gratuitamente no seguinte endereço: <http://jquery.com/>.

- GlassFish Server 4

GlassFish é um servidor de aplicação de código aberto liderado pela Sun Microsystems para a plataforma J2EE². Glassfish também possui uma versão proprietária que é chamada Sun GlassFish Enterprise Server. A versão gratuita é software livre, que é distribuída sob

²Java Platform, Enterprise Edition, ou em português Plataforma Java, Edição Empresarial) é uma plataforma de programação para servidores na linguagem de programação Java

os termos de duas licenças a Common Development and Distribution License (CDDL) e GNU General Public License (GPL).

Como mencionado mais acima, utilizaremos para geração de páginas dinâmicas o JSP. E para que consigamos utilizar as funcionalidades disponibilizadas pelo JSP, necessitamos de um servidor que dê suporte à essa tecnologia. O Glassfish é o responsável por interpretar o JSP e disponibilizar o HTML resultante à interface do usuário.

Glassfish pode ser encontrado e baixado no seguinte endereço: <https://glassfish.java.net/>.

- API do OpenHub O OpenHub (antes Ohloh.net) é uma comunidade que online que oferece resultados de análises de código de projetos de software livre.

A API - Application Programming Interface, em português Interface de Programação de Aplicações, disponibilizada pelo OpenHub é denominada API Ohloh e baseado em REST - Representational State Transfer, em português Transferência de Estado Representacional. Através dela é possível obter diversas informações dos projetos, como por exemplo: total de linhas de código do projeto, total de comentários, total de linhas em branco, descrição do projeto, total de commits, entre outras.

Utilizamos a API para obtenção de dados e informações mais detalhadas sobre os projetos e assim evitar a replicação de trabalho para o cadastro de informações do projeto.

Para utilizar a API, primeiramente criamos uma conta e uma chave de API no OpenHub. É possível criarmos 5(cinco) chaves de API, e cada uma possibilita até 1000(mil) consultas diárias.

Por ser um serviço baseado em REST, o Portal realiza a consulta de informações de um projeto por meio do nome. Caso o projeto exista no OpenHub, as informações são retornadas no formato XML, que após processamento damos ênfase somente nas informações que são adequadas às nossas necessidades.

A página do OpenHub na internet pode ser acessada por meio do endereço <https://www.openhub.net>. A API, possui mais informações de uso e está disponível no GitHub através do seguinte endereço https://github.com/blackducksw/ohloh_api.

A permissão de uso da API é desprovida de qualquer empecilho, sendo possível até vender cópias do Software sendo necessário que se faça a inclusão de aviso de copyright

"Copyright (c) 2013 Black Duck Software, Inc. and its contributors" em todas as cópias ou partes substanciais.

- Tinymce

TinyMCE é uma plataforma web independente, um editor de texto baseado em Javascript, HTML e controle de edição WYSIWYG. Sua utilização e distribuição é como Código Aberto sob as condições da Licença LGPL.

Muitas vezes necessitamos inserir informações formatadas como título em negrito, listas com marcadores, tabelas, texto e imagens, entre outros. Isso é muito complicado utilizando-se a área de texto que é o componente HTML padrão. Para conseguir um efeito, por exemplo, de título em negrito o usuário deveria ter conhecimento de HTML para esse fim. Com a utilização de um editor como o TinyMCE, o usuário tem um ferramenta com opções familiares às encontradas em editores de texto como o Microsoft Word, tornando mais fácil, rápido e produtiva a inserção de informações formatadas.

No Portal o editor é utilizado para fornecer uma forma familiar para o usuário poder formatar adequadamente as informações no momento do cadastro das informações do projeto.

O TinyMCE pode ser baixado em sua página <http://www.tinymce.com/>.

- Highcharts

O Highcharts é uma plataforma desenvolvida em JavaScript, para geração de diversos tipos de gráficos dinâmicos em páginas web. Sua utilização é gratuita para fins não comerciais, para fins comerciais é necessário obter uma licença de uso. Em qualquer uma das licenças, gratuita ou não, o usuário tem permissão para baixar o código fonte e fazer suas próprias edições. Isto permite modificações pessoais e uma grande flexibilidade.

No Portal ele é utilizado para geração dos gráficos referentes ao projeto, como o gráfico de linguagens e o gráfico sobre o código.

O Highcharts pode ser baixado no seguinte endereço: <http://www.highcharts.com/>.

5 CONCLUSÕES

Neste trabalho, realizamos a implementação de um Portal Web para auxiliar novatos em projetos de software livre, possibilitando que informações sobre diversos projetos sejam disponibilizados para os novatos. Para este trabalho, utilizamos os requisitos obtidos na literatura que tratava sobre as barreiras enfrentadas pelos novatos em projetos de software livre para realizarem sua primeira contribuição e também para continuarem participando dos projetos. O modelo de barreiras descrito por Steinmacher et al. (2014), forneceu embasamento para a realização deste trabalho, visto que se trata de uma pesquisa aprofundada sobre novatos em software livre.

Do processo de desenvolvimento desse Portal, como todo trabalho desse tipo possuem, houve muitas dificuldades e aprendizados. Desenvolver um projeto como trabalho de conclusão de curso e sozinho, é uma tarefa complicada, sobretudo quando se está iniciando no desenvolvimento de software e se tem de conciliar o projeto com o trabalho, vida social e outras disciplinas do curso.

Devido à característica do projeto, o número de tecnologias utilizadas foi, até certo ponto, grande. E aprender utilizar todas essas tecnologias foi um desafio que exigiu certo grau de dedicação. No entanto, apesar do processo exigente, no final o conhecimento adquirido nas tecnologias servem para aperfeiçoar e aumentar o nível de conhecimento.

Sobretudo, ter a consciência de que esse pode ser o início de algo que venha a ser útil para alguém, é algo muito satisfatório. É fato, que vale destacar que o projeto ainda precisa de muitos ajustes, o que é normal para qualquer tipo de software. Nenhum projeto nasce grande, abrangente e completo o suficiente para não ser atualizado e melhorado. Assim, esperamos que o projeto seja aprimorado e melhorado no intuito de servir como uma ferramenta de apoio para muitos novatos e também não novatos, que queiram contribuir com o mundo do software livre.

Como prosseguimento e melhoria deste portal, seria interessante, em trabalhos futuros,

a aplicação de um sistema próprio de obtenção de dados dos projetos, baixando o código diretamente dos repositórios de código fonte e aplicando métricas de software. Com essa complementação seria possível obter dados mais específicos e adequados às necessidades dos novatos e eliminaria a dependência dos dados disponíveis no OpenHub. Também como trabalho futuro, é necessário conduzir testes de usabilidade e experiência de usuário no portal.

REFERÊNCIAS

- ALENCAR, A. F. d.; MACHADO, M. B.; EVANGELISTA, R.; SILVEIRA, S. A. d.; AGUIAR, V. M. Software livre cultura hacker e ecossistema da colaboração. **Produção de Terceiros Sobre Paulo Freire (PTPF); Livros**, 2009.
- CAPRA, E.; WASSERMAN, A. I. A framework for evaluating managerial styles in open source projects. In: **Open Source Development, Communities and Quality**. [S.l.]: Springer, 2008. p. 1–14.
- COSTA, R. Candido da. Conhecendo o software livre. In: **Anais do Congresso Nacional Universidade, EAD e Software Livre**. [S.l.: s.n.], 2012. v. 1, n. 1.
- CUBRANIC, D.; MURPHY, G. C.; SINGER, J.; BOOTH, K. S. Hipikat: A project memory for software development. **Software Engineering, IEEE Transactions on**, IEEE, v. 31, n. 6, p. 446–465, 2005.
- FOGEL, K. **Producing open source software: How to run a successful free software project**. [S.l.]: "O'Reilly Media, Inc.", 2005.
- GNU. **O Sistema Operacional Linux**. 2014. Disponível em: {<http://www.gnu.org/philosophy/free-sw.html>}.
- HIMANEN, P. A ética dos hackers e o espírito da era da informação: a importância dos exploradores da era digital. **Rio de Janeiro: Campus**, 2001.
- JUNIOR, G. C. S.; SOARES, D. d. Q. A força da ação voluntária em comunidades de software livre: uma reflexão antropológica. **VI Jornada Internacional de Políticas Públicas**, 2013.
- MACIEL, A. C. Padrões de socialização de novatos em projetos de software livre. Campo Mourao, 2014.
- MADEY, G.; FREEH, V.; TYNAN, R. The open source software development phenomenon: An analysis based on social network theory. **AMCIS 2002 Proceedings**, p. 247, 2002.
- NASCIMENTO, L.; SANTORO, F. Análise de interações nas comunidades virtuais de software livre. **Simpósio Brasileiro de Sistemas de Informação, Brasília: Brasil**, 2009.
- PARK, Y.; JENSEN, C. Beyond pretty pictures: Examining the benefits of code visualization for open source newcomers. In: IEEE. **Visualizing Software for Understanding and Analysis, 2009. VISSOFT 2009. 5th IEEE International Workshop on**. [S.l.], 2009. p. 3–10.
- RAYMOND, E. S. A catedral e o bazar. 1998. URL: <http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>, 2002.

SANTOS, F. C. d. Peripécias de agosto: alguns episódios da cena hacker. **Fronteiras, São Leopoldo-RS**, v. 4, n. 2, p. 79–101, 2002.

SOUZA, M. S. **Software Livre e o mito do voluntariado**. 2014. Disponível em: {<http://jornalgggn.com.br/blog/luisnassif/software-livre-e-o-mito-do-voluntariado>}.

STEINMACHER, I.; SILVA, M. A. G.; GEROSA, M. A. Barriers faced by newcomers to open source projects: a systematic review. In: **Open Source Software: Mobile Open Source Technologies**. [S.l.]: Springer, 2014. p. 153–163.

STEINMACHER, I.; WIESE, I. S.; CHAVES, A. P.; GEROSA, M. A. Newcomers withdrawal in open source software projects: Analysis of hadoop common project. In: IEEE. **Collaborative Systems (SBSC), 2012 Brazilian Symposium on**. [S.l.], 2012. p. 65–74.

STEINMACHER, I.; WIESE, I. S.; GEROSA, M. A. Recommending mentors to software project newcomers. In: IEEE. **Recommendation Systems for Software Engineering (RSSE), 2012 Third International Workshop on**. [S.l.], 2012. p. 63–67.

ZHOU, M.; MOCKUS, A. What make long term contributors: Willingness and opportunity in oss community. In: IEEE PRESS. **Proceedings of the 34th International Conference on Software Engineering**. [S.l.], 2012. p. 518–528.