



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

YALLY DE LIMA GALDINO

**UMA ABORDAGEM PARA SUPORTE À ANÁLISE DE RISCOS
NO MANTIS BUG TRACKER**

CAMPINA GRANDE - PB

2022

YALLY DE LIMA GALDINO

**UMA ABORDAGEM PARA SUPORTE À ANÁLISE DE RISCOS
NO MANTIS BUG TRACKER**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharela em
Ciência da Computação.**

Orientador: Professor Dr. Hyggo Oliveira de Almeida.

CAMPINA GRANDE - PB

2022

YALLY DE LIMA GALDINO

**UMA ABORDAGEM PARA SUPORTE À ANÁLISE DE RISCOS
NO MANTIS BUG TRACKER**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharela em
Ciência da Computação.**

BANCA EXAMINADORA:

Professor Dr. Hyggo Oliveira de Almeida.

Orientador – UASC/CEEI/UFCG

Professor Dr.(a.) Melina Mongiovi Cunha Lima Sabino

Examinador – UASC/CEEI/UFCG

Professor Tiago Lima Massoni

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 06 de Abril de 2022.

CAMPINA GRANDE - PB

ABSTRACT

Mantis Bug Tracker is a web tool whose focus is to manage software defects (bugs) during its development and evolution process. With it, one has a database of the defects reported by users, development and quality teams. The motivation for using a tool such as Mantis is that the high quantity of bugs in software generates wear and tear for the client, devalues the product, brings costs with rework and risks of financial and business losses. In 2009, for example, a Google programmer added a backslash to URLs and the site was flagged as malware for about an hour, generating a total loss of almost US\$ 3 million. In this work, the goal is to present an approach to support risk analysis in Mantis Bug Tracker, raising the critical points of a system through the analysis of bugs reported by customers. Based on the study of these bugs, to understand their cause and thus create actions to eliminate them or to have a faster response in case they appear. This will bring benefits to the developing company in terms of time, cost, learning curve and greater customer satisfaction.

Uma abordagem para suporte à análise de riscos no Mantis Bug Tracker

Yally de Lima Galdino

Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
yally.galdino@ccc.ufcg.edu.br

Hyggo Oliveira de Almeida

Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
hyggo@computacao.ufcg.edu.br

RESUMO

Mantis Bug Tracker é uma ferramenta web cujo foco é gerenciar defeitos (*bugs*) no software durante o seu processo de desenvolvimento e evolução. Com ele, tem-se um banco de dados dos defeitos relatados pelos usuários, equipes de desenvolvimento e de qualidade. A motivação para o uso de uma ferramenta como a Mantis é que a alta quantidade de *bugs* em um software gera desgaste para o cliente, desvaloriza o produto, traz custos com retrabalho e riscos de prejuízos financeiros e de negócio. Em 2009, por exemplo, um programador da Google adicionou uma barra invertida nas URLs e o site foi sinalizado como malware por cerca de uma hora, gerando um prejuízo total de quase US\$ 3 milhões. Neste trabalho, tem-se como objetivo apresentar uma abordagem para suporte à análise de riscos no Mantis Bug Tracker, levantando os pontos críticos de um sistema através da análise dos *bugs* relatados pelos clientes. Com base no estudo desses *bugs* entender a sua causa e assim criar ações para eliminá-los ou ter uma resposta mais rápida caso eles apareçam. Isso trará benefícios para a empresa que o desenvolve em relação a tempo, custo, curva de aprendizado e maior satisfação do cliente.

Palavras-chave:

Bug, Risco, Gestão de Risco, Mantis Bug Tracker.

1. INTRODUÇÃO

Falhas em software podem atrapalhar o seu funcionamento e serem grandes aliadas para crimes cibernéticos. Em casos mais graves, deverá ser realizada a correção imediata, pois os *bugs* podem comprometer o sistema de segurança do aparelho no qual o software está sendo executado e, conseqüentemente, comprometer dados pessoais do dono. Um *bug* é um risco negativo para o sistema. A alta quantidade de *bugs* de um software traz instabilidade para o sistema, custos para a empresa e desvalorização do produto.

Diante dos problemas que um *bug* pode causar em um sistema, existem diversas ferramentas para detecção e rastreamento dos mesmos. Uma das ferramentas mais populares no mercado é o Mantis Bug Tracker, que provê rastreamento de *bugs* com a

flexibilidade de personalizar etiquetas de problemas e processos de fluxo de trabalho.

Contudo, estas ferramentas são utilizadas para registrar os *bugs* de projetos mas não possuem um mecanismo de análise automática dos casos. Os registros acabam não sendo utilizados em projetos futuros ou são corrigidos apenas os casos sem que seja estudada a causa do problema, fazendo com que a empresa repita erros críticos já cometidos no passado.

Considere uma empresa da área de software da Paraíba, aqui chamada de EmpresaX, por restrições de confidencialidade, na qual a autora deste artigo trabalha, que possui cerca de 150 funcionários e tem em média 120 casos por dia relatados através do Mantis Bug Tracker. Quando o usuário relata um novo *bug*, a equipe de distribuição da empresa adiciona a ele o setor que irá tratá-lo, equipe à qual o setor pertence e um marcador indicando em qual parte do sistema ocorreu o erro. O Mantis possibilita criar alguns campos personalizados para colocar nos casos relatados como equipe, setor e marcador. Os marcadores foram personalizados pela empresa, permitindo indicar qual o assunto do erro. Porém, o Mantis não traz uma análise desses marcadores.

Neste trabalho, foca-se em como utilizar conhecimento sobre *bugs* relatados pelos usuários de projetos para criar uma gestão de risco a nível de desenvolvimento. Com o objetivo de identificar quais foram os marcadores mais utilizados de um setor em um certo período de tempo, a base de dados de *bugs* será importada do Mantis para que se faça uma análise quantitativa dos marcadores e, com isso, identificar os pontos que apresentam mais erros. Exemplos de alguns campos que poderão ser exportados para a planilha são: número do caso; data em que foi aberto; data em que foi devolvido para o cliente; setor, equipe, marcadores; entre outros.

Ao sabermos os pontos do sistema que mais apresentam erros, podemos analisar quais os *bugs* que mais se repetem nesses pontos. Com isso, podemos criar soluções para evitá-los ou, se não for possível evitá-los, métodos para que sejam solucionados rapidamente. Tendo os pontos mais críticos do sistema mapeados, a equipe de qualidade da EmpresaX pode testar com mais atenção os pontos que mais apresentam erros a fim de tentar evitar que ocorram nas atualizações seguintes. Como resultado da análise

dos *bugs* e formas de resolvê-los, propõe-se uma abordagem de análise de risco para dar suporte aos gestores na análise dos pontos críticos do sistema. Por fim, apresenta-se a validação do método com uma pesquisa qualitativa.

2. FUNDAMENTAÇÃO TEÓRICA

Risco é um evento ou uma condição incerta que, se ocorrer, tem um efeito positivo ou negativo em um ou mais objetivos do projeto, como escopo, prazo, custo e qualidade [1]. Dentro da gestão de um projeto existe uma área chamada Gestão de Riscos que tem como objetivo reduzir a probabilidade e impacto de eventos negativos e aumentar a probabilidade e o impacto dos eventos positivos. No caso deste projeto, vamos focar em reduzir a probabilidade e impacto de eventos negativos.

Segundo o guia PMBOK (*Project Management Body of Knowledge*), o gerenciamento dos riscos do projeto inclui os processos de condução do planejamento, da identificação, da análise, do planejamento das respostas, da implementação das respostas e do monitoramento dos riscos em um projeto. Ele visa identificar e gerenciar os riscos que não são considerados pelos outros processos de gerenciamento de projetos. Ambientes de alta variabilidade, por definição, incorrem em mais incerteza e risco. Para resolver isso, os projetos gerenciados por abordagens adaptativas fazem uso de revisões frequentes de produtos de trabalho incremental e de equipes de projetos multifuncionais para acelerar o compartilhamento do conhecimento e garantir que os riscos sejam compreendidos e gerenciados. Os riscos são considerados na seleção do conteúdo de cada iteração, e os riscos também serão identificados, analisados e gerenciados durante cada iteração [1].

Um *bug* de software é um erro ou falha no software de computador que faz com que ele produza um resultado incorreto ou inesperado, ou se comporte de maneira não intencional [2]. Os erros podem resultar em inúmeros efeitos, com níveis variando da inconveniência ao usuário do programa até erros mais graves que podem fazer com que o programa deixe de funcionar ou congele e conduza à negação de serviço [3].

Segundo Roger S. Pressman, o objetivo do controle da qualidade de software e da gestão da qualidade em geral é, em sentido mais amplo, eliminar problemas de qualidade no software. Tais problemas são conhecidos por diversos nomes como *bugs*, falhas, erros ou defeitos. O mesmo também diz que aqueles que realmente querem software confiável descobrirão que devem, primeiro, encontrar um meio de evitar a maioria dos *bugs* e, como resultado, o processo de programação se tornará mais barato, programadores eficientes, não devem perder tempo depurando e, para início de conversa, não devem introduzir erros[4].

O crítico de arquitetura romano, Vitruvius, lançou a noção de que prédios bem projetados eram aqueles que apresentavam solidez, comodidade e deleite. O mesmo poderia ser dito em relação a software de boa qualidade. Solidez: um programa não deve apresentar nenhum *bug* que impeça seu funcionamento. Comodidade: um programa deve ser adequado aos propósitos para os quais foi planejado. Deleite: a experiência de usar o programa deve ser prazerosa. Temos aqui os princípios de uma teoria de projeto de software[4].

O rastreamento de *bugs* (*Bug Tracking*) é uma das etapas mais importantes em qualquer ciclo de vida de desenvolvimento de software. Sistemas de rastreamento de *bugs* ou de problemas são usados para relatar *bugs* e outros problemas, bem como para controlar se estes foram corrigidos ou não[5]. O Mantis Bug Tracking é uma ferramenta web *open source*, desenvolvida em PHP, customizável e totalmente balanceada entre simplicidade e desempenho para controle de *bugs*. Ela foi desenvolvida com o intuito de gerenciar defeitos e falhas de outros sistemas. Porém, dada sua facilidade de implantação, customização e custo baixo, vem sendo adaptada para diversos outros contextos, entre eles para o gerenciamento de requisitos ou tarefas (*tasks*) por times de desenvolvimento. Sua principal vantagem é fornecer um ambiente web de fácil acesso [6].

3. SOLUÇÃO

Para atingir os objetivos citados anteriormente, foi criado um mecanismo para a análise de riscos com base no PMBOK para realizar a gestão de riscos. O mecanismo inclui os seguintes passos: planejar o gerenciamento dos riscos; identificar os riscos; realizar a análise qualitativa e quantitativa; planejar as respostas aos riscos e controlar os riscos.

Nas seções a seguir, explicamos como aplicamos as técnicas de gestão de risco do PMBOK em nosso contexto. Ela é composta por cinco etapas distintas. Na Seção 3.1 descrevemos o planejamento dos riscos. Na Seção 3.2 mostramos como é feita a identificação do risco. Na Seção 3.3 apresentamos a análise qualitativa e quantitativa. Na Seção 3.4 mostramos como foi realizado o planejamento das respostas aos riscos. Por fim, na Seção 3.5 apresentamos uma proposta de como controlar os riscos.

3.1 Planejar o Gerenciamento dos Riscos

Planejar o gerenciamento dos riscos consiste em planejar como o gerenciamento será executado, monitorado e controlado. Para o nosso caso, ele foi dividido nas seguintes fases:

1. Escolher um setor da Empresa para analisar os casos, como exemplo considere a EmpresaX;
2. Identificar quais marcadores são mais utilizados naquele setor. Um exemplo de marcador pode ser "Login", ou seja, em um sistema que precisa que o usuário esteja autenticado, se nessa tela houver um erro, poderíamos colocar esse marcador;
3. Verificar no Mantis quais os *bugs* mais frequentes relatados por marcadores, a fim de achar um padrão entre eles;
4. Analisar os *bugs* que foram listados no passo anterior e definir sua causa, grau de risco, estratégia e ação;
5. Executar as ações citadas no passo anterior;
6. Avaliar as ações.

3.2 Identificando os Riscos

Na etapa de identificação dos riscos é realizado um mapeamento dos riscos do sistema. Para a identificação dos mesmos,

utilizamos o Mantis Bug Tracker e uma planilha no Google Sheets para listar os riscos identificados. A seguir, apresenta-se o passo a passo realizado.

3.2.1 Definir quais serão os casos analisados, escolhendo o setor e um período de tempo

Para este trabalho, no caso da EmpresaX, foi escolhido um setor aqui chamado de Setor A, por restrição de confidencialidade, e os casos foram relatados entre 01/11/2021 e 18/02/2022.

3.2.2 Filtrar os casos selecionados no Mantis

Para isso, através da função “Ver Tarefas” no Mantis, na opção “Usar Filtro de Data”, insere-se a data escolhida, com os seguintes filtros:

- Relator: qualquer
- Estado: qualquer
- Categoria: Bug e atendimento
- Atribuído a: qualquer
- Setor: setor escolhido (no nosso caso, Setor A)

Ao clicar em “Aplicar filtro” o sistema traz todos os casos correspondentes aos dados escolhidos, como podemos observar na Figura 1.

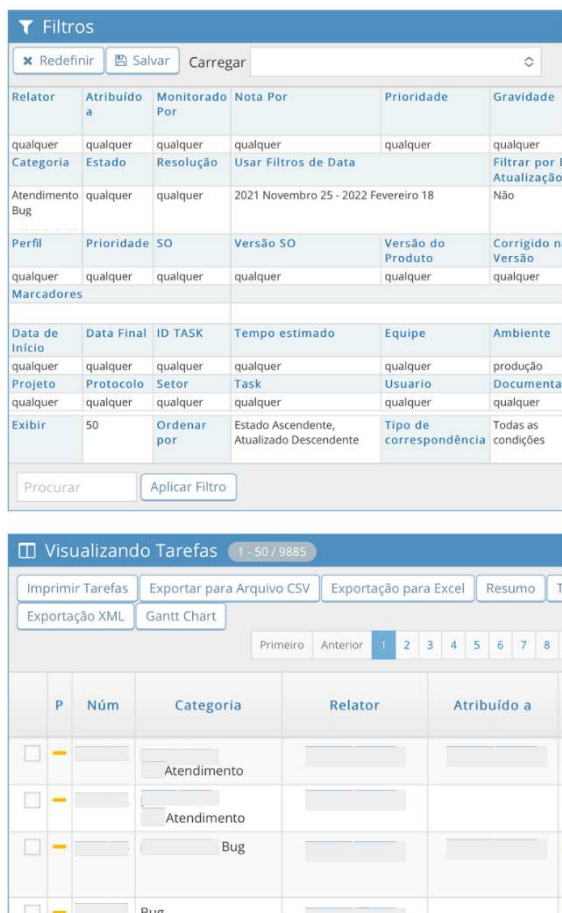


Figura 1: Imagens do Mantis após clicar em “Aplicar Filtro”.

3.2.3 Importar os casos para uma planilha.

Para importar os casos para uma planilha, foi realizada a seguinte sequência de passos:

1. Após clicar em “Aplicar Filtro”, aparecerão todos os casos que possuem as informações selecionadas. Deve-se então clicar no botão “Exportar para Arquivo CSV”;
2. Criar uma planilha no Google Sheets <https://www.google.com/sheets/about/>;
3. Na planilha ir em “Arquivo” → Importar → Upload → Selecionar o arquivo CSV exportado do Mantis → Substituir página atual.

3.2.4 Verificar quais os marcadores mais utilizados do setor

Para isso, pode-se utilizar os dados exportados da planilha, analisando quais marcadores são mais encontrados. Na Figura 2, ilustra-se um gráfico gerado com a contagem de marcadores.

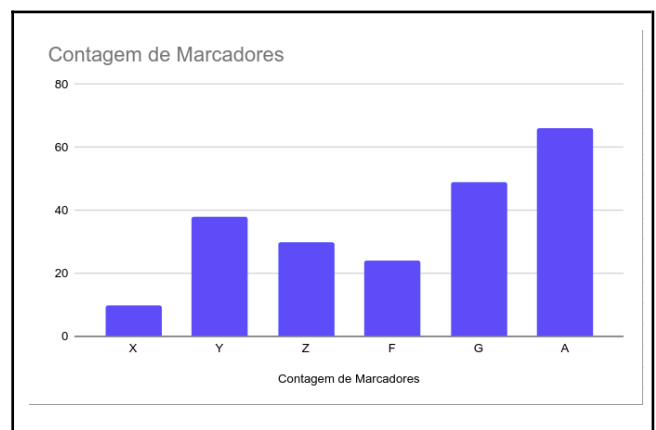


Figura 2: contagem de marcadores.

Com base nos marcadores que apresentaram mais erros, foi realizada uma análise sobre quais são esses erros.

3.2.5 Verificar quais os erros dos marcadores

Para identificar os erros de um marcador específico, pesquisamos no filtro de “Ver Tarefas” do Mantis o marcador escolhido no campo “Marcadores”.

O Mantis traz a lista de bugs com os dados filtrados. Ao clicar na palavra “Resumo”, ele tenta alinhar os casos deixando em ordem alfabética os bugs de acordo com a primeira palavra do resumo. Assim facilita-se a identificação de erros iguais.

3.2.6 Registrar os bugs frequentes de cada marcador

Foi utilizada uma planilha de apoio para registrar os *bugs* no formato ilustrado no Quadro 1.

NOME DO MARCADOR	
BUG	ID
Bug n1	001
Bug n2	002
Bug n3	003
Bug n4	004

Quadro 1: Bugs frequentes de cada marcador e seu ID.

Com isso tem-se uma lista dos *bugs* mais frequentes por marcador.

3.3 Análise Quantitativa e Qualitativa

Com essa análise pode-se classificar o nível de risco do *bug* e discutir uma possível causa para o problema. Para isso, foram realizadas entrevistas semi-estruturadas por meio da plataforma Google Meet com o suporte da EmpresaX para adicionar a Causa e o Grau de Risco de cada *bug*.

Os Graus de riscos são classificados em:

- **Alto:** causa um alto impacto no sistema e possui uma alta probabilidade de acontecer. São os riscos que precisam de mais atenção;
- **Moderado:** possui uma probabilidade mediana de acontecer e causa um impacto moderado no sistema;
- **Baixo:** causa impacto insignificante no sistema e a probabilidade de acontecer é baixa. São casos que não precisam de muita atenção.

Como resultado dessa fase, adicionamos mais duas colunas à tabela: Causa e Grau de Risco (Quadro 2). Com essas informações podemos melhor tratar os *bugs* na próxima fase e priorizá-los.

NOME DO MARCADOR			
BUG	ID	CAUSA	GRAU DE RISCO
Bug n1	001	Causa Bug 1	Alto
Bug n2	002	Causa Bug 2	Alto
Bug n3	003	Causa Bug 3	Moderado

Bug n4	004	Causa Bug 4	Moderado
Bug n5	005	Causa Bug 5	Baixo

Quadro 2: Bugs frequentes de cada marcador com seu ID, Causa e Grau de Risco.

3.4 Planejamento de Respostas

Através de um *brainstorming* com os integrantes do setor e convidados que tenham domínio do sistema, adicionam-se mais duas colunas à planilha: Estratégia e Ação.

Para isso, primeiro colocamos a estratégia de cada risco, de acordo com os tipos de estratégia do PMBOK:

- **Escalar:** risco fora do escopo do projeto ou acima da autoridade do gestor do projeto.
- **Prevenir:** eliminar a ameaça ou proteger o projeto do seu impacto.
- **Transferir:** passar a responsabilidade de uma ameaça a terceiros.
- **Mitigar:** reduzir a probabilidade de ocorrência e/ou impacto de uma ameaça.
- **Aceitar:** reconhecer a existência de uma ameaça, mas nenhuma ação proativa é tomada.

Depois de escolher a estratégia, discutimos qual(is) a(s) melhor(es) ações a serem tomadas para aquele risco. Com isso, a planilha terá as informações apresentadas no Quadro 3.

NOME DO MARCADOR					
BUG	ID	CAUSA	GRAU DE RISCO	ESTRATÉGI A	AÇÃO
Bug n1	001	Causa Bug n1	Alto	Estratégia para o Bug n1	Ação para o Bug n1
Bug n2	002	Causa Bug n2	Alto	Estratégia para o Bug n2	Ação para o Bug n2
Bug n3	003	Causa Bug n3	Moderado	Estratégia para o Bug n3	Ação para o Bug n3
Bug n4	004	Causa Bug n4	Moderado	Estratégia para o Bug n4	Ação para o Bug n4
Bug n5	005	Causa Bug n5	Baixo	Estratégia para o Bug n5	Ação para o Bug n5

Quadro 3: Planilha completa dos Bugs do Marcador.

3.5 Controlar os Riscos

Para controlar os *bugs* do sistema, cada líder do setor deve realizar o levantamento dos casos mais comuns da semana, preencher a planilha com a sua causa, Grau de Risco, estratégia e ação. Junto com o líder da sustentação, devem colocar em prática as ações citadas e verificar se o número de casos de cada marcador está diminuindo. Caso contrário, será necessário colocá-los novamente na planilha para preencher com novas estratégias e novas ações.

Para verificar se o número de casos por marcador está diminuindo, basta colocar no filtro do Mantis o marcador e o período no qual você deseja comparar. Por exemplo, para visualizar a diferença entre o mês de Janeiro e o mês de Março, deve-se colocar em “Usar filtro de data”: “Data início” = 01/01/2022 e “Data de Término” = 31/01/2022. Assim, aparecerá o número de casos, Figura 3 do mês de Janeiro, depois basta mudar o mês e aparecerá a quantidade de casos do mês de Março, Figura 4, e você pode observar se houve diminuição de casos.

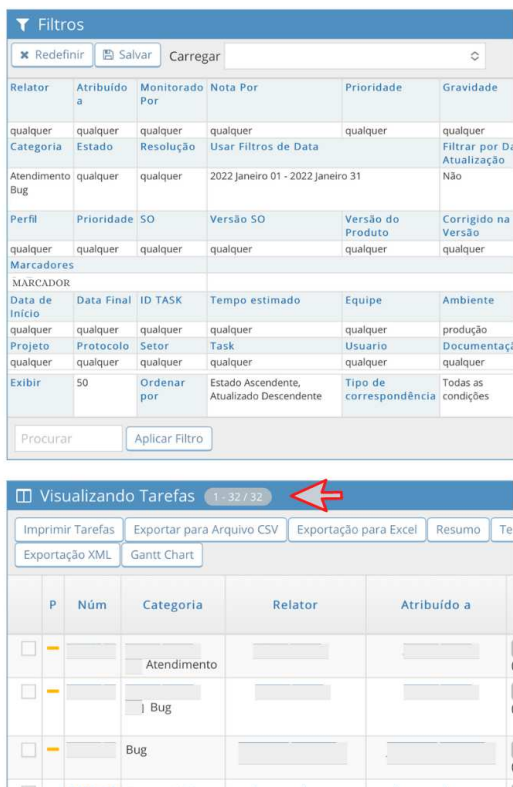


Figura 3: Casos no Mantis de Janeiro.



Figura 4: Casos no Mantis de Março.

4. AVALIAÇÃO

Com o intuito de validar se essa abordagem é útil, apresentamos na EmpresaX para os líderes de setor, sustentação e coordenador de equipe por meio da plataforma Google Meet. Decidimos fazer uma pesquisa qualitativa de avaliação tanto para sabermos se traz benefícios para a empresa quanto para avaliar a importância dos dados encontrados.

Para a avaliação foi aplicado um questionário através do Google Forms, plataforma de gerenciamento de pesquisa em que o usuário pode aplicar pesquisas e coletar informações, com questões com respostas fechadas, de múltipla escolha, e uma pergunta aberta.

Na Seção 4.1 definimos o que queremos responder com a avaliação realizada. Os resultados e discussões são apresentados na Seção 4.2.

4.1 Questionário

As 4 questões iniciais tiveram como objetivo obter informações sobre os participantes da pesquisa:

1. Qual o seu cargo?
2. Sua equipe ou setor tem em média quantos casos por dia?
3. Você como gestor, já fez algum tipo de gestão de risco para sua equipe ou setor?
4. Sobre os Marcadores, você já fez alguma análise deles?

Já as questões seguintes foram elaboradas a fim de saber a opinião dos participantes sobre a abordagem e a importância dos dados encontrados:

5. Você acha que essa abordagem será útil para a empresa?
6. Quais áreas da empresa você acha que se beneficiaram com a abordagem (Suporte, Qualidade, Desenvolvimento, Análise, administrativo, RH, comercial, comunicação, jurídico)?
7. Você acha que depois de aplicada haverá diminuição de casos relatados?
8. De 0 a 10, o quão importante você acha que são os dados obtidos (sendo 0 pouco importante e 10 muito importante)?

4.2 Resultados e Discussões

O formulário contou com 12 participantes, que estão em cargos de liderança na empresa, sendo que 58,3% dos participantes eram líderes de setores, 16,7% líderes da sustentação e 25% coordenador de equipe.

Ao perguntarmos aos participantes quantos casos em média são abertos por dia para sua equipe ou setor, 50% dos participantes informaram que são abertos para sua equipe ou setor em média 10 casos por dia, 25% marcaram a opção de 60 casos ou mais, 16% escolheram a opção de 20 casos e 8,3% dos participantes marcaram a opção de em média 30 casos.

Dos 12 participantes, 7(58,3%) informaram que nunca fizeram gestão de riscos para sua equipe ou setor e 5 participantes (41,7%) informaram que já fizeram. Aos que marcaram “sim”, pedimos que os mesmos nos informassem o que foi feito nessa gestão de riscos e as respostas foram:

- Participante n1: “Fizemos um levantamento de casos para identificar quais erros eram mais recorrentes e identificar potenciais falhas futuras.”
- Participante n2: “Orientação quanto a dúvidas dos casos; Testar bem antes de passar o caso para desenvolvimento; Relacionar os casos corretamente e abrir causa para erros recorrentes. Com isso, as demais etapas do processo são mais precisas e ocorre a diminuição de *bugs* abertos em produção, sendo esse o principal risco da equipe.”
- Participante n3: “Toda equipe de Suporte focada 100% em uma sala para tirar suas dúvidas, assim comunicado com suporte de apoio, caso não entenda a situação, falar direto com analista.”
- Participante n4: “Tento deixar sempre uma pessoa com um conhecimento principal e um secundário para que na ausência de alguém possa cobrir mais facilmente. Ex. pessoa 1 tem conhecimento azul principal e amarelo secundário, pessoa 2 tem conhecimento principal amarelo e secundário azul. Assim se um se ausentar por

qualquer motivo tenho como cobrir mais facilmente aquela área.”

- Participante n5: “Execução ordenada entre QA, Sustentação e Devs no momento de subida de pacote à produção, onde têm que ser divididos os devs que pegam *bug* de prod, e sustentação alinhada a zerar os casos para não impactar a subida.”

Em relação a se eles já fizeram alguma análise sobre os marcadores, 58,3% informaram que não fizeram e 41,7% informaram que já fizeram uma análise dos marcadores.

Como é possível observar na Figura 5, ao perguntar aos participantes quais as áreas da empresa que eles acham que se beneficiarão após a aplicação dessa abordagem, 12 participantes (100%) votaram no Suporte, 10(83,3%) na equipe de Qualidade, 11(91,7%) na equipe de Desenvolvimento, 7(58,3%) na de Análise, 5(41,7%) no RH, 4(33,3%) no Comercial e nenhum dos participantes votou nas áreas Administrativo, Comunicação e Jurídico.

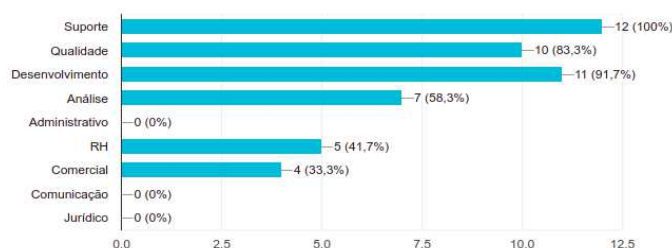


Figura 5 - Quais áreas da empresa você acha que se beneficiarão após a aplicação dessa abordagem? Opções: Suporte, Qualidade, Desenvolvimento, Análise, Administrativo, RH, Comercial, Comunicação e Jurídico.

Ao perguntar aos mesmos se eles achavam que depois da aplicação haveria redução dos casos, 100% respondeu que sim. Pedimos também para eles classificarem em uma escala de 0 a 10, sendo 10 muito importante e 0 insignificante, a importância dos dados obtidos. Todos os participantes escolheram de 7 ou mais, sendo que 1 (8,3%) atribuiu 7; 2 (16,7%) atribuíram 8; 5 (41,7%) atribuíram 9; e 4 (33,3%) atribuíram 10 (Figura 6).

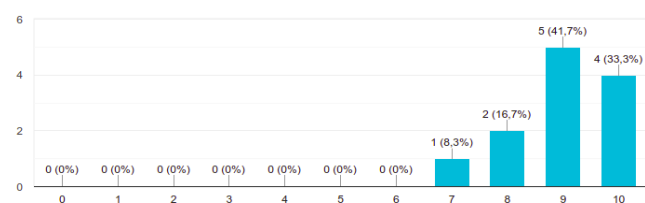


Figura 6 - De 0 a 10, o quão importante você acha que são os dados obtidos (sendo 0 pouco importante e 10 muito importante)?

5. CONCLUSÕES E TRABALHO FUTUROS

Neste artigo descrevemos o desenvolvimento de uma técnica para análise de riscos no Mantis Bug Tracker, a partir dos casos relatados pelos usuários dos sistemas da EmpresaX. Acreditamos que esta técnica possa ser utilizada para ter um auxílio na análise de risco em nível de desenvolvimento, ao sabermos quais os pontos do sistemas geram mais *bugs* e quais são os *bugs* mais frequentes, podendo tratá-los melhor a fim de solucionar os erros, assim como, ao fazer uma nova funcionalidade, sabermos os pontos mais críticos do sistema.

Com a pesquisa que aplicamos nos funcionários, podemos validar que a abordagem de fato será útil e seis áreas da empresa se beneficiaram com ela. Podemos concluir também que os dados encontrados serão de grande importância para a empresa.

Como trabalhos futuros, pretendemos automatizar a visualização dos marcadores, para que seja possível ver em um gráfico os marcadores mais utilizados. Para isso, é preciso alterar o código fonte do Mantis e introduzir uma função no campo “Resumo” para que ele gere um gráfico com essas informações.

6. REFERÊNCIAS

- [1] Guia PMBOK® 6a. ed. – EUA: Project Management Institute, 2017. BORGES, Carlos; ROLLIM, Fabiano.
- [2] WIKIPEDIA. Software bug. Disponível em: https://en.wikipedia.org/wiki/Software_bug. Acesso em: 28 mar. 2022.
- [3] BRASIL ESCOLA. Bug. Disponível em: <https://brasilescola.uol.com.br/informatica/bug.htm>. Acesso em: 28 mar. 2022.
- [4] PRESSMAN, Roger S.; MAXIM, Bruce R.. Engenharia de software: uma abordagem profissional. 8 ed. Porto Alegre: AMGH, 2016.
- [5] SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 529 p. Tradução Ivan Bosnic e Kalinga G. de O. Gonçalves.
- [6] MANTIS BUG TRACKER. MantisBT makes collaboration with team members & clients easy, fast, and professional. Disponível em: <https://www.mantisbt.org/>. Acesso em: 28 mar. 2022.