UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

CAMPUS CORNÉLIO PROCÓPIO

DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL

DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO

CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

RICARDO CARREIRO ALVIM

**DESENVOLVIMENTO DE UMA FERRAMENTA DE LEVANTAMENTO DE REQUISITOS COM BASE NO MODELO R4C**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO

2016

RICARDO CARREIRO ALVIM

**DESENVOLVIMENTO DE UMA FERRAMENTA DE LEVANTAMENTO DE REQUISITOS COM BASE NO MODELO R4C**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina Trabalho de Conclusão de Curso, do curso de Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para a obtenção do título de Tecnólogo.

Orientador: Prof. Dr. Alexandre L’Erário

CORNÉLIO PROCÓPIO

2017

**RESUMO**

ALVIM, Ricardo Carreiro. **Desenvolvimento de uma ferramenta de levantamento de requisitos com base no modelo r4c**. 2017. 29 f. Trabalho de Conclusão de Curso (Graduação) – Tecnologia em Análise e Desenvolvimento de Sistemas. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2017.

Este projeto trata da relação entre o crowdsourcing e suas interações em uma grande rede e a engenharia de requisitos, onde os processos de elicitações podem ser facilitados quando explorados em uma grande rede, contanto que as adaptações e ajustes sejam realizados. Estas ações entre o levantamento de requisitos, crowdsourcing e o framework Scrum foi relacionado por RIBEIRO (2016) que propôs um modelo chamado de r4c. Este projeto todavia este modelo proposto não possui uma ferramenta para realizar todas ações realizadas, sendo este o propósito deste trabalho, tendo como base o processo r4c, desenvolver uma ferramenta para o processo de crowdsourcing, assim auxiliando no processo de levantamento de requisitos.

**Palavras-chave:** Crowdsourcing, Engenharia de Requisitos, Elicitação, Engenharia de Software, r4c, Crowdsourcing e Scrum.

**ABSTRACT**

ALVIM, RICARDO. **DIGITAL MARKEYTING PROCESS MANAGEMENT TOOL DEVELOPMENT**. 2016. 29 f. Proposta de Trabalho de Conclusão de Curso (Graduação) – Tecnologia em Análise e Desenvolvimento de Sistemas. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2016.

This proposal approach concepts and explain a global view of Digital Marketing and it’ evolution in technology context. Furthermore, is approached the role of social media in presente and how they can help in the process of capture of new clientes. The growth of social medias makes the appearance of digital networks to accord a market share of enterprises and organizations that seek new approaches of interatiions. However, like many others sectors it’s necessary a adequated management of requirements and of process, something that proposes seeks to point: a solution through of a tool of process management oriented to Digital Marketing.

**Keywords**: Digital Marketing Management Tool, Digital Marketing Tool.

**LISTA DE IMAGENS**

[Figura 1. Diagrama de Caso de Uso - Visão Geral](http://academico.docx#_Toc453568669) 19

[Figura 2. Diagrama de Classes - Regras de Negócio](#_1ci93xb)

[Figura 3: Diagrama de Classes - Representação de Banco de Dados](#_2bn6wsx)

[Figura 4: Adição de novo cliente](#_3as4poj)

[Figura 5: Adição de uma nova campanha](#_1pxezwc)

[Figura 6: Adição de um novo post - Facebook](#_49x2ik5)

[Figura 7: Visão Geral da Campanha](#_2p2csry)

**LISTA DE SIGLAS**

|  |  |
| --- | --- |
| API | Aplication Program Interface |
| WYSIWG | What You See Is What You Get |
| UML | Unified Modeling Language |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Sumário**

[**1.1.** **INTRODUÇÃO** 8](#_Toc481958357)

[**1.2.** **PROBLEMA** 9](#_Toc481958358)

[**1.3.** **JUSTIFICATIVA** 9](#_Toc481958359)

[**1.4.** **OBJETIVOS** 9](#_Toc481958360)

[**1.4.1.** **OBJETIVO GERAL** 9](#_Toc481958361)

[**1.5.2.** **OBJETIVOS ESPECÍFICOS** 10](#_Toc481958362)

[**1.5.3.** **ESTRUTURA DO TRABALHO** 10](#_Toc481958363)

[**2.** **FUNDAMENTAÇÃO TEÓRICA** 10](#_Toc481958364)

[**2.1.** **CROWDSOURCING** 10](#_Toc481958365)

[**2.2.** **ENGENHARIA DE REQUISITOS** 12](#_Toc481958366)

[**2.3.** **REQUISITOS EM CROWDSOURCING** 15](#_Toc481958367)

[**2.4.** **TRABALHOS RELACIONADOS** 15](#_Toc481958368)

[**3.** **MÉTODOS E PROCEDIMENTOS** 16](#_Toc481958369)

[**3.1.** **SCRUM** 16](#_Toc481958370)

[**3.2.** **SCRUM SOLO** 17](#_Toc481958371)

[**3.3.** **MODEL VIEW CONTROLLER** 18](#_Toc481958372)

[**3.4.** **JAVA SERVER FACES** 18](#_Toc481958373)

[**3.5.** **ASTAH** 18](#_Toc481958374)

[**4.** **DESENVOLVIMENTO** 20](#_Toc481958375)

[**5.** **CONSIDERAÇÕES FINAIS** 20](#_Toc481958376)

## **INTRODUÇÃO**

A engenharia de requisitos é um tópico de grande relevância para a Engenharia de Software, sendo que um dos problemas frequentes no desenvolvimento do software é a elicitação de requisitos.

A elicitação é uma etapa de desenvolvimento para levantar as informações necessárias para o desenvolvimento de um software. Ao longo dos anos a Engenharia de Software evoluiu em formas de processo de software, passando de modelos cascata, evoluindo para modelos incrementais e atualmente, modelos agilistas. Todos eles desenvolvidos com o propósito de melhorar o processamento de desenvolvimento de software e também os problemas relacionados a volatilidade de requisitos, como isso impacta em custos de produção, organização da equipe e o racionamento de recursos.

De início acreditou-se que as mesmas premissas dos modelos tradicionais de engenharia civil onde o projeto era desenvolvido do início ao fim sendo totalmente planejado antes da construção do mesmo. Com a evolução dos processos de software, este tipo de solução se mostrou insuficiente e necessário ser divididas em etapas, com mecanismos de validação, dando assim origem em processos interativos e incrementais que juntamente com experiências de mercado se transformaram em processos agilistas como o Scrum, onde a gestão do projeto é priorizada.

Contudo, paralelamente a esta evolução, ocorreram também mudanças significativas para a indústria de software como a criação do sistema operacional Linux, do movimento de software livre e a mídia de web em que o usuário e entusiasta de um software, produto ou serviço deixa de ser um mero protagonista no processo de evolução ou construção de um produto, para passar a ser um protagonista do mesmo, onde ele pode colaborar diretamente ou indiretamente com evoluções no mesmo, seja por feedbacks ou por colaborar com alterações de códigos fontes existentes, melhorando-os e os submetendo em um processo de aprovação em comunidades onde estas alterações e sugestões são julgadas e aplicadas as devidas soluções. Este ato de colaboração em uma grande rede, onde as informações podem vir de diversos tipos de usuário e pode ser utilizada para a melhora de produtos e serviços é chamada de crowdsourcing.

Mesmo softwares de código fechado estão mudando suas concepções para terem mecanismos de feedback e sugestões de mudanças dos usuários, onde as mesmas são ponderadas e podem serem acrescentadas, melhoradas ou resolvidas. Um exemplo atual do uso de crowdsourcing é o Windows Insider Preview, que é uma versão rolling release do sistema operacional Windows onde existe um mecanismo de obtenção de feedback de usuários sobre as alterações do sistema, reportagem de problemas e sugestão de novas funcionalidades. Estas alterações são analisadas por uma equipe da Microsoft que pode ou não adicionar uma sugestão, correção ou alteração enviada por um usuário Windows nas próximas builds ou de testes do sistema operacional.

Por tanto, o crowdsourcing já é uma realidade associada a engenharia de requisitos, e como poderá ser visto no capítulo dedicado ao tema de crowdsourcing, existem diversas soluções de mercado ou da academia para o gerenciamento de requisitos, sendo uma delas o modelo r4c que será analisado adiante.

## **PROBLEMA**

O problema apontado por esta proposta é o acompanhamento de requisitos de software por meio de modelo de crowdsourcing por meio do desenvolvimento de uma ferramenta.

## **JUSTIFICATIVA**

O crescimento do número de usuários de redes sociais, o aumento do interesse de empresas em Marketing Digital e a necessidade de uma visão global de todos os processos relacionados a uma campanha, ação ou estratégia de marketing implicam em ferramentas para o gerenciamento dos dados relacionados e sua transformação em dados relevantes em termos estratégicos, auxiliam na tomada de decisões em agências e empresas que possuem interesse em Marketing Digital.

## **OBJETIVOS**

## **OBJETIVO GERAL**

O objetivo geral deste projeto é desenvolver uma ferramenta para requisitos em crowdsoucing baseado no modelo r4c.

Deixe claro que é a implementação de um trabalho anterior. É a continuação de um TCC

## **OBJETIVOS ESPECÍFICOS**

Esta proposta tem como objetivos específicos do desenvolvimento da ferramenta de gestão de processos:

1. Compreender o processo de crowdsoucing;
2. Compreender o modelo r4c;
3. Desenvolver uma ferramenta com base no r4c;
4. Observar as interações com o Scrum; e
5. A adoção de um processo ágil (Scrum Solo) para o desenvolvimento desta proposta.

## **ESTRUTURA DO TRABALHO**

Este trabalho está estruturado em capítulos onde o capítulo 2 descreve a fundamentação teórica sobre crowdsourcing, engenharia de requisitos e requisitos em crowdsourcing, onde são apresentados os conceitos elementares sobre os mesmos e descrevendo como os mesmos se associam.

O capítulo 3 descreve os métodos e procedimentos utilizados pelo projeto como o Scrum e o Teste Experimental. O capítulo 4 descreve o desenvolvimento da ferramenta, seus processos e tecnologias utilizadas. Já o capítulo 5 descreve o teste experimental realizado e no capítulo 6, os resultados do teste são apresentados. No capítulo 7, é concluída as informações do projeto e apresentando os resultados obtidos.

## **FUNDAMENTAÇÃO TEÓRICA**

Este capítulo apresenta uma visão geral dos temas de crowdsourcing, engenharia de requisitos e requisitos, demonstrando a associação entre os temas e a relevância dos mesmos, apresentando conceitos acadêmicos e soluções de mercado associada aos mesmos.

## **CROWDSOURCING**

Crowdsourcing é um termo cunhado por Jeff Howe, em 2006, para representar o processo de obtenção de idéias, serviços e conteúdo em um canal de grande público tendo em mente a internet onde estas características e ações se desenvolviam rapidamente graças as novas possibilidades na internet.

O termo foi criado em um contexto de que serviços da web como a Wikipédia, as distribuições Linux e demais projetos open source. Além disso, na época, também estava em expansão serviços na web como as redes sociais, devido ao advento da Web 2.0, que segundo O’RELLY (2005) permitiu a evolução para uma web de conteúdos dinâmicos e alta interatividade. Junto com elas, segundo Leite et al. (2013) temos o surgimento de blogs, plataformas de compartilhamento de vídeos, *wikis* e outras aplicações voltadas a interatividade com usuários. Esta interatividade descrita por (Leite et al., 2013) caracterizam alguns pontos dos pilares do crowdsourcing que veremos adiante.

Segundo Brabham, Daren (2008) o crowdsourcing é um ato de uma empresa ou organização transferir as funções realizadas por uma rede indefinida, tornando assim este trabalho de forma colaborativa e distribuída.

Estas pessoas da chamada rede indefinida, segundo (POENG, BABAR. 2014) se tornaram prestadoras de tarefas para estas organizações, por isto o processo de crowdsourcing também é visto como um processo de terceirização. No IEE Explorer da IEEE, uma das mais relevantes bases sobre o escopo deste projeto, que é a engenharia de software, possui mais de 900 artigos relacionados ao tema crowdsourcing sendo que estes números crescem com o passar dos anos, comprovando a importância do tema.

Além da redução de custos, o crowdsourcing é um processo de obtenção de feedback de usuários, contudo, estes usuários, segundo HOSSAIN (2012) devem estar cientes de tudo que ocorre no projeto e na equipe, sendo chamado este processo de Consciência em Grupo.

O crowdsourcing, contudo, pode ser classificado por suas atividades e seus pilares. Segundo HOSSEIN et al. (2005) podem ser divididas em quatro pilares, que serão detalhadas a seguir.

O primeiro pilar é relacionado as pessoas envolvidas, que constituem as atividades do processo (HOSSEIN et al. 2014). O segundo pilasr é a tarefa propriamente dita que irá atrair o público que irá participar da atividade de crowdsourcing.

O quarto pilar é o sistema, meio ou canal onde ocorrerá os processos de obtenção de informações que podem ou não ser um meio digital (HOSSEIN et al, 2014)

Além dos pilares, também há 5 características principais como a diversidade, a know-less, largeness e a suitability.

A diversidade refere-se a distinção entre as pessoas que participam do processo, classificados em gênero, localidade e faixa etária.

A segunda característica chamada de unknow-less que trata do anonimato, contudo, este pode ser subdividido em dois tipos, quando o crowd não sabe quem está realizando o processo ou quando um dado participante não conhece os demais.

Já a característica largeness refere-se ao tamanho da abrangência de colaboração que em contextos muito abrangentes não se deve fazer com que o crowd assuma muitas responsabilidades para evitar confusão na multidão. A quarta característica é a suitability que refere-se aos ajustes necessários quanto as habilidades dos usuários para realizar uma tarefa (HOSSEIN et al, 2014).

Das características mais detalhadas e importantes para o escopo deste projeto se trata das características do segundo pilar do crowdsourcer. Esta seria relacionada aos incentivos que seria o ânimo para desempenhar o processo onde o crowdsourcer pode fornecer incentivos como financeiro, social ou de entretenimento. Outra característica do segundo pilar refere-se a audiência aberta ao público em geral ou ao menos que tenham relevância ao escopo do projeto. (HOSSEIN et al., 2015).

A partir de estudos associando as áreas da engenharia de software e o crowdsourcing, analisou-se que, o uso dos recursos de obtenção de feedback do crowdsourcing pode facilitar o processo de levantamento de requisitos de modo a atender as necessidades dos usuários projetando um software sob medida. Estes estudos estão relacionados a empirical validation, requirements identification, stakeholders discovery, requirements engineering, requirement-driven social adaptation e feedback-based requirement (HOSSEIN et al., 2015)

## **ENGENHARIA DE REQUISITOS**

Segundo Pressman (s/a) a Engenharia de Requisitos pode ser descrita como uma ação associada a Engenharia de Software que é iniciada na atividade de comunicação e continua na modelagem.

Em outras palavras, a Engenharia de Requisitos permite entender as necessidades do cliente não apenas sob aspecto funcional, isto é, das funcionalidades de um software, mas também as consequências de determinadas escolhas no projeto de software.

Segundo CHRISTELE e Kong (s/a), já no levantamento de requisitos os problemas ocorrem, por problemas no escopo, tendo a má definição dos limites do projeto ou então com detalhes desnecessários ou imprecisos, gerando confusão e problemas de entendimento. Também ocorrem problemas quando os clientes não possuem o total domínio dos limites e capacidades de seus sistemas computacionais. Outro problema é quando o cliente ou usuário omite certos requisitos por considerarem certas informações obvias demais, acreditando que o analista já tenha todo o conhecimento do domínio e regras de negócios internalizadas pela empresa ou organização.

Esta questão do domínio é um problema, pois se por um lado um analista com conhecimentos do domínio do projeto ajuda a melhorar na precisão dos requisitos, sabendo assim ponderar e adequar todas as perguntas ao escopo do projeto, isso torna os custos de projeto mais altos devido a uma mão de obra mais qualificada.

Um outro problema da engenharia de requisitos é a volatilidade dos requisitos, que podem alterar a qualquer momento no projeto. Sendo justamente por isto, metodologias de processo de desenvolvimento de software e frameworks de processos com tendência agilista tentam criar mecanismos onde a alteração de requisitos seja menos custosa, previsível e em certos casos, como no framework Scrum, desejável.

No entanto, todo processos de desenvolvimento de software passa por uma etapa onde são especificados os requisitos. Segundo PRESSMAN (s/a) esta especificação pode ser um documento textual, um conjunto gráfico, cenários de uso, protótipos, modelagem matemática ou combinação destes.

Um dos modelos citados por PRESSMAN é o SRS, software requirements specifications, que em tradução livre seria Especificação de Requisitos de Software que é um modelo de documento escrito detalhando todos os aspectos do software desenvolvido por Karl Wiegers da Process Impact Inc.

Este modelo tenta mapear os aspectos do software como proposta, escopo, referências, perspectiva, característica, classes de usuário, ambiente operacional, restrições, documentos para usuários, hipóteses, características, interfaces e requisitos não funcionais; assim passando por todos os aspectos possíveis de um software.

Entender como estes requisitos fazem parte do projeto do software é importante para compreender um modelo como o r4c para a obtenção de requisitos por meio de uma grande rede e transformar então os requisitos em informação relevante para o projeto de software.

## **REQUISITOS EM CROWDSOURCING**

Dado o caráter do crowdsourcing de sua ampla disponibilidade em uma grande rede, chamada por Howe de uma maneira de terceirizar a solução de problemas fazendo uso da inteligência coletiva (HOWE, 2006) sendo constituídos pelos pilares do crowdsourcing.

Segundo HOSSEIN et al (2015) o crowdsourcing pode colaborar na etapa de elicitação de requisitos no desenvolvimento de um projeto, em especial em sistemas utilizados por um grande número de usuários que possam colaborar com o processo. A partir dos estudos descritos, pode-se afirmar que o crowdsourcing auxilia na eficiência da elicitação de requisitos, onde a sabedoria coletiva pode auxiliar no processo de elicitação e atender de maneira precisa e concisa as necessidades do cliente do projeto de software.

## **TRABALHOS RELACIONADOS**

Os trabalhos relacionados podem ser categorizados em trabalhos acadêmicos e softwares que fazem uso dos conceitos do crowdsourcing para obtenção de requisitos.

Dos artigos acadêmicos temos o trabalho de RIBEIRO (2016) que específica o modelo r4c para uso de requisitos em crowdsourcing, sendo que este artigo se baseia no de HOSSEIN et al (2015) que apresenta maneiras para o ajuste do crowdsourcing para aperfeiçoar a elicitação de requisitos. Neste trabalho houve uma atividade de levantamento de requisitos por meio da ferramenta Google Forms com a questão “Para você o que um celular deve possuir?” solicitando aos alunos do curso de Engenharia Elétrica gerasse 10 requisitos. Nos resultados foi verificado que, embora o processo r4c possua aderência a proposta de gerenciar requisitos seja verdadeira, o processo manual inviabiliza o procedimento, sendo inclusive um dos trabalhos futuros sugeridos, a criação de uma ferramenta automática ou semi automática para a melhoria do mesmo.

HOSSEIN et al (2015) defende o uso do crowdsourcing na etapa de elicitação apresentando em seu artigo formas de realizar o ajuste para o aprimoramento de requisitos e para isso, realizou um experimento onde envolveu dois grupos de 14 participantes, envolvendo usuário e desenvolvedores. Além disso foram envolvidas outras 34 pessoas da comunidade de engenharia de requisitos. Na execução, os aspectos humanos foram divididos e mapeados de modo a se estabelecer como seria realizado o processo onde os formulários aplicados foram os mesmos, tendo apenas as variações de nível e conhecimento, demonstrando assim a preocupação com os diferentes níveis de conhecimento e visão dos envolvidos, estando assim alinhado o processo de modo a se extrair de forma adequada as informações dos usuários.

Os tempos de execução foram calculados e dois avaliadores avaliaram os resultados, com um terceiro observando para evitar problemas. No final do processo, observou-se a ligação entre crowdsourcing e atributos de qualidade induzido com 34 relações classificadas em 10 categorias, sendo que cada uma delas expressava uma característica do crowdsourcing.

Considerando que não haviam conhecimentos suficientes para aumentar a qualidade dos requisitos induzidos, foi confirmada a relação entre o Crowd e requisitos.

Forno e Zanatta (2016) realizaram uma proposta de processo de testes funcionais onde este processo foi dividido em três etapas, sendo que a primeira contemplou o teste de Somerville (2011), para investigar as tarefas que poderiam ser executadas fazendo uso do crowdsourcing. Em seguida foram aplicadas as correções para permitir o uso do crowdsourcing no processo e por fim, a execução propriamente dita em uma empresa de desenvolvimento de software, tendo como alvo um sistema ERP (Enterprise Resource Planning) onde foram elaborados 50 casos de teste que, na execução do projeto conforme o modelo proposto dos autores e os mesmos seriam executados simultaneamente e na ocasião da publicação, a configuração do ambiente estava em andamento.

Contudo, mesmo, não tendo sido finalizado e com maiores resultados, os autores concluem que a proposta era viável e inovadora por permitir que usuários externos pudessem colaborar com a evolução do produto. Este tipo de abordagem de testes dirigidos por feedback é utilizado por softwares como o Windows em seu programa chamado de Windows Insider Preview.

Já VLAANDERN et al (2010) realizaram uma aplicação real com o propósito de mostrar como os métodos ágeis podem trabalhar em conjunto com a gestão de produto. As experiências relatadas no artigo de VLAANDEER et al em uma empresa fazem o uso do framework Scrum, destacando os ajustes aplicados e como a partir do caso de uso podem ser ajustados, aplicados e mensurados em um cenário real.

A primeira etapa onde se levanta as ideias da empresa ou cliente é chamado de vision. Outra etapa é a theme onde ocorre a execução formal tornando assim a visão com mais detalhes.

Posteriormente ocorre a definição de requisitos, dividida em três etapas onde a primeira é realizada pela equipe de produtos de software, chamada de SPM, a equipe de SPM traduz os conceitos em listas realizando os ajustes necessários entre os requisitos. Após isso a equipe de desenvolvimento elabora os requisitos, onde podem ser criados documentos para clarificar o processo.

Conclui-se que é necessário, para que a equipe SPM ágil desenvolva com sucesso as tarefas do projeto, determinar o tamanho das tarefas, estruturas de backlog e realizar o refinamento dos requisitos.

Além destes trabalhos, existem outros no IEE Explore, sendo que a grande maioria deles, não existe aplicação prática.

## **MÉTODOS E PROCEDIMENTOS**

Este capítulo será dedicado para apresentação geral do processo de software, tecnologias e arquitetura escolhidas para o desenvolvimento desta proposta de projeto.

## **SCRUM**

Scrum é uma metodologia de desenvolvimento ágil, interativo e incremental usado para o processo de desenvolvimento de software. Foi concebido, segundo SILVA (2014) nos anos 1990 por Jeff Sutherland, John Scummiotales e Jeff McKenna quando os modelos de processo tradicionais atingiram seus limites. O Scrum foi então a solução encontrada pelo mercado e pela academia para contornar os problemas relacionados com o desenvolvimento de software, principalmente o de gestão de projetos. (SCHAWABER; SUTHERLAND, 2015)

No Scrum o início de cada projeto é realizado um *Product Backlog*, uma fase do processo de desenvolvimento em que são definidas as principais funcionalidades do projeto. Essas funcionalidades, de início não serão exatamente detalhadas, elas serão melhor elaboradas posteriormente com diversas reuniões com os envolvidos no projeto e o *Product Owner*, sendo que este último participa ativamente do processo de gestão e desenvolvimento de software, ele é o responsável por defender os interesses do cliente do produto.

Mudanças de requisitos são consideradas positivas e são até mesmo encorajadas, contanto que o *Product* *Owner* se comprometa a não mudar os requisitos durante a chamada *Sprint*, que são pequenas etapas do projeto, juntamente com o *Product Owner,* essas *sprints* são divididas de modo que durem 3 a 4 semanas, sendo que são adaptadas de acordo com as necessidades do *Product Owner* e tem a presença de todos os responsáveis e interessados pelo projeto. A primeira *Sprint*, chamada de *Sprint Planning Meeting* é realizada uma reunião inicial no qual as primeiras dúvidas são levantadas.

Além da Sprint Planning Meeting, todos os dias existe uma reunião diária, chamada de *Daily* *Scrum*, para que os membros da equipe de produção, chamada de *Scrum* *Team*, repassem aos envolvidos os acontecimentos do dia anterior, para verificar os problemas existentes e priorizar o necessário para o dia. No fim de cada Sprint é realizada uma compilação de tudo que foi alcançado no projeto naquela Sprint, chamada de *Sprint* *Review* *Meeting*. Após esta fase, é realizado o Sprint *Retrospective*, que possui o objetivo de identificar o que funcionou bem e o que não funcionou bem durante a Sprint. Na sequência, inicia-se o planejamento para a próxima Sprint até que todos os itens do *Product Backlog* estejam finalizados e então o projeto seria considerado concluído.

O Scrum é baseado em várias etapas de gestão, chamadas de *sprints*, sendo que o *Sprint* *Planning* *Meeting* e o *Daily* *Scrum*, por exemplo, são formas de tentar mapear o escopo do projeto; no caso do Planning Meeting; descobrir os problemas existentes e promover uma constante comunicação entre os membros da equipe, que são chamados de *Scrum Team*. (PAULA, s/d)

## **SCRUM SOLO**

== explique um parágrafo com citações sobre o scrum e entre direto em scrum solo.

A citação do scrum solo está em pagoto,2016, ([10.1109/CISTI.2016.7521555](https://doi.org/10.1109/CISTI.2016.7521555)).

O Scrum Solo é uma metodologia baseada no Scrum, com a ênfase em equipes pequenas ou de apenas um desenvolvedor. Ou seja, o número de pessoas que irão consumir as informações dos artefatos do processo será baixo. Algumas características gerais do Scrum são mantidas como o *Product Backlog*, que é uma avaliação geral do escopo do projeto, e o *Sprint Backlog*, que é uma avaliação geral das atividades de uma determinada Sprint.

As mudanças passam a existir quanto ao número de reuniões, no Scrum Solo possui apenas reuniões semanais, diferentemente das reuniões diárias do Scrum. Porém, no final de cada Sprint, que em média tem uma duração de duas semanas, o desenvolvedor deverá entregar um protótipo funcional dos requisitos coletados pela aquela Sprint. (L’ERÁRIO et al, 2016)

Sendo assim, o Scrum Solo se mostra uma alternativa adequada para um projeto com as características descritas da ferramenta proposta, além das possibilidades de os requisitos serem alterados e a capacidade de gestão do processo citado.

## **Arquitetura candidata**

Colocar um texto, e um diagrama UM explicando a arquitetura candidata..

## **MODEL VIEW CONTROLLER**

Model View Controller, MVC, é um padrão de projeto que visa separar os projetos de software em três camadas a Visão, onde os dados são apresentados para o usuário; o Modelo, onde ficarão as classes de modelo de negócios e a modelagem da abstração dos dados e o Controlador, onde é feita a comunicação com o banco de dados entre o Modelo e a Visão.

**REFERÊNCIAS**

BENGEL, A.; SHAWKI, A.; AGGARWAL, D. **Simplifying web analytics for digital marketing**. 2015 IEEE International Conference on Big Data (Big Data), 2015. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7363968>. Acesso em 12.jun.2016

CENDÓN, B. V. **Ferramentas de busca na web**. Ci. Inf.,Brasília, v. 30, n. 1, p. 39–49, jan./abr. 2001. Disponível em: <http://www.scielo.br/pdf/ci/v30n1/a06v30n1>. Acesso em 12.jun.2016

CHAFFEY, D. **Global social media research summary 2016**. Fev 2016. Disponível em: <http://www.smartinsights.com/social-media-marketing/social-media-strategy/new-globalsocial-media-research/>. Acesso em 12.jun.2016

CONSTANTINIDES, E.; FOUNTAIN, S. J. **Web 2.0: Conceptual foundations and marketing issues**. 2007. Disponível em: <http://www.palgravejournals.com/dddmp/journal/v9/n3/full/4350098a.html>. Acesso em 12.jun.2016

CORDEIRO, G. **Aplicações Java para web com JSF e JPA**. 1. ed. [S.l.]: Casa do Código, 2013.

EXAME. **Faturamento do Facebook supera expectativas, apesar de aumento nas despesas**. 2015. Disponível em: <http://exame.abril.com.br/tecnologia/noticias/faturamento-dofacebook-supera-expectativas-apesar-de-aumento-nas-despesas>. Acesso em 12.jun.2016

FACEBOOK. **Company Info**. 2015. Disponível em: <http://newsroom.fb.com/companyinfo/>. Acesso em 12.jun.2016

FACEBOOK. **Impulsionar suas publicações**. Mar 2016. Disponível em: <https://www.facebook.com/business/help/547448218658012>. Acesso em 12.jun.2016

FACEBOOK. **Métricas de publicação de páginas**. 2016. Disponível em: <https://www.facebook.com/help/336143376466063/>. Acesso em 12.jun.2016

GONZALEZ, C. de O. **O modelo de negócio da Google:** entre a eﬁciência técnico-cientíﬁca e o imperativo econômico do retorno do investimento extraﬁscalidade como instrumento de proteção ambiental no Brasil. Anais do V Congresso de Direito de Autor e Interesse Público, 2011.

IAB. **Números de Investimento em Mídia Online 2015-2016 - AD SPEND BRASIL**. 2016. Disponível em: <http://iabbrasil.net/guias-e-pesquisas/mercado/numeros-de-investimento-emmidia-online-2015-2016—ad-spend-brasil>. Acesso em 12.jun.2016

KENDZERSKI, P. R. **Web Marketing e Comunicação Digital.** 2. ed. [S.l.: s.n.], 2009.

KOTLER, P. **Administração de Marketing**: análise, planejamento, implementação e controle. [S.l.]: Editora Atlas, 1998.

LEITE,N.;CAIXINHA, H.;RAMOS,F. **Proposta de uma aplicação web para monitoração do impacto de notícias nas redes sociais Facebook e Twitter**. Revista Comunicando, v. 2, 2013.

MAHMOOD, M. A. **Advanced Topics in End User Computing**. Idea Group Publishing, 2004. Disponível em: <https://books.google.com.br/books?hl=ptBR&lr=&id=OnlxwDq4YTEC&oi=fnd&pg=PA16&dq=digital+marketing&ots=IST6OynaQ&sig=vGB08fHOnLOCFyS6itnstl23MY4>. Acesso em 12.jun.2016

ORACLE. **JavaServer Faces Technology Overview**. s/a. Disponível em: <http://www.oracle.com/technetwork/java/javaee/overview-140548.html>. Acesso em 12.jun.2016

O’REILLY, T.**What Is Web2.0**. Set 2005.

PARISE, S.; GUINAN, P. J. **Marketing using web 2.0.** Proceedings of the 41st Hawaii International Conference on System Sciences, 2008. Disponível em: <https://www.computer.org/csdl/proceedings/hicss/2008/3075/00/30750281.pdf>. Acesso em 12.jun.2016

PAULA, R. de S. A. **Scrum na melhoria do gerenciamento de projetos de software:** Um estudo sobre a implantação de Scrum para otimizar o processo de gerenciamento de projetos de software. Engenharia de Software Magazine, v. 23, s/d.

SANTIAGO, M. P. **Coleção Gestão Empresarial**. [S.l.]: Associação Franciscana de Ensino Senhor Bom Jesus, 2002.

SAPIRO, A. et al. **Gestão de Marketing**. [S.l.]: Editora FGV, 2007.

SCHWABER, K.; SUTHERLAND, J. **Guia do Scrum - Guia deﬁnitivo para o Scrum: As regras do jogo**. [S.l.]: Scrum.org, 2015. SILVA, L. A. da. iFrame - framework para o desenvolvimento de aplicações web. 2014.

STATS, I. W. **INTERNET USAGE STATISTICS:** The Internet Big Picture World Internet Users and 2015 Population Stats. Nov 2015.

TOMAS, R. N.; MESCHGRAHW, R. P.; ALCANTARA, R. L. C. **As redes sociais e o comportamento de compra do consumidor:** O reinado do ”boca-a-boca” está de volta?. Revista Brasileira de Marketing, 2012.