

Sistemas de Informação

**GESTÃO DE TAREFAS - SCRUM**

*Daniel José Guedes Carneiro*

Índice

[Índice III](#_Toc29048660)

[Índice de figuras VI](#_Toc29048661)

[1. Introdução VIII](#_Toc29048662)

[1.1 Objetivo VIII](#_Toc29048663)

[1.2 Organização do Relatório VIII](#_Toc29048664)

[2 Capítulo 1 – Scrum VIII](#_Toc29048665)

[2.1 Origem do Scrum VIII](#_Toc29048666)

[2.2 Metodologia Scrum IX](#_Toc29048667)

[2.3 Estrutura da metodologia *Scrum* IX](#_Toc29048668)

[2.3.1 *Sprint* IX](#_Toc29048669)

[2.3.2 *Product Backlog* IX](#_Toc29048670)

[2.3.3 *Sprint Planning Meeting* X](#_Toc29048671)

[2.3.4 *Daily Scrum* X](#_Toc29048672)

[2.3.5 *Review Meeting* X](#_Toc29048673)

[2.3.6 *Sprint Retrospective* X](#_Toc29048674)

[2.4 Gestão da Estrutura Scrum XI](#_Toc29048675)

[2.4.1 Scrum Master XI](#_Toc29048676)

[2.4.2 Product Owner XI](#_Toc29048677)

[2.4.3 Equipa de Desenvolvimento XII](#_Toc29048678)

[3 Capítulo 2 – Caso de Estudo XII](#_Toc29048679)

[3.1 Modelo de Gestão de Tarefas Anterior (CADS Software LTD) XII](#_Toc29048680)

[3.2 O novo modelo de gestão de tarefas (CADS Software LTD) XIII](#_Toc29048681)

[3.2.1 Tarefa XIII](#_Toc29048682)

[3.2.2 Diagrama de Contexto XIV](#_Toc29048683)

[3.2.3 Diagrama Nível 1 XV](#_Toc29048684)

[4. Conclusão XV](#_Toc29048685)

[5. Referências e Bibliografia XVII](#_Toc29048686)

Índice de figuras

[Figura 1 - Diagrama de Contexto da gestão de tarefas da equipa de desenvolvimento 7](#_Toc28011621)

[Figura 2 - Diagrama de nível 1 da gestão de tarefas da equipa de desenvolvimento 8](#_Toc28011622)

# Introdução

No âmbito da disciplina de Sistemas de Informação, foi proposto um trabalho à escolha do aluno que visa em descrever a gestão de tarefas da equipa de desenvolvimento de software para a web, da empresa CADS Software LTD sediada em Bournemouth, Reino Unido. Como programador nesta empresa passei tanto pelo modelo de organização anterior como no modelo em que vou desenvolver neste trabalho baseado na metodologia *Scrum*.

## Objetivo

São objetivos deste trabalho a descrição da implementação de um sistema de tarefas com base na metodologia *Scrum* na empresa tal como a elaboração de um diagrama de organização de tarefas. Também uma retrospetiva de como era o sistema anterior e as mudanças que foram efetuadas.

## Organização do Relatório

No primeiro capítulo vou descrever a metodologia *Scrum*, a sua origem, a gestão das equipas e os diferentes postos de trabalho que fazem parte desta metodologia. No segundo capítulo vou acrescentar o modelo antigo de gestão da empresa CADS Software LTD e o novo modelo e analisar as vantagens e desvantagens do novo modelo.

# Capítulo 1 – *Scrum*

## Origem do *Scrum*

Retirado de uma posição tática do râguebi termo Scrum consiste em concentrar todos os atacantes em apenas um lugar no campo para facilitar a oportunidade de um membro da equipa avançar com a bola, sendo que todos trabalham para o mesmo objetivo.

Em 1995 foi criado a metodologia *Scrum* por Jeff Sutherland e Ken Schwaber, estes apresentaram na conferência de Ospsla em Austin (Texas, Estados Unidos da América) uma analogia entre as táticas de râguebi e o desenvolvimento de produtos tendo como visão trabalhar em equipa em direção a um objetivo comum.

## Metodologia *Scrum*

*Scrum* é uma metodologia Agile para gestão e planeamento de projetos de software usada para gerir o trabalho em produtos ou soluções complexas desde o início dos anos 90. É uma estrutura na qual as pessoas podem lidar com problemas com frequentes modificações e complexos, enquanto fornecem produtos de maneira criativa e produtiva. Podemos encarar a metodologia *Scrum* não como um processo, uma técnica ou método definitivo, mas por uma estrutura na qual pode empregar vários processos e técnicas.

## Estrutura da metodologia *Scrum*

## 2.3.1 *Sprint*

Representa um período de tempo no qual um conjunto de atividades deve ser executado. Podemos compreender como uma lista de tarefas distribuídas pela equipa com um prazo de expiração.

Normalmente um *sprint* tem uma duração entre duas semanas a um mês, dependendo da complexidade do projeto.

## 2.3.2 *Product Backlog*

Representa-se por uma lista de funcionalidades a serem implementadas num projeto, ou seja, uma lista de tarefas que não têm um sprint destinado. Esta pode ser constituída por novas funcionalidades a implementar ou então problemas existentes no projeto que necessitam de ser revistos ou alterados. Esta lista é mantida e atualizada pelo *Product Owner* e *Scrum Master* e analisada para futuros *sprints*.

## 2.3.3 *Sprint Planning Meeting*

Esta é uma reunião em que o *Product Owner* planeia e prioriza as tarefas do *Product Backlog* e a equipa seleciona as atividades que será capaz de implementar durante um *Sprint.* Esta reunião é bastante importante para que a *sprint* seja concluída a tempo sendo que a equipa de desenvolvimento faz uma estimativa de quanto tempo demora a efetuar a lista de tarefas.

## 2.3.4 *Daily Scrum*

Como em todas as empresas a comunicação interna é crucial para o sucesso da empresa. Esta é uma breve reunião normalmente matinal com o objetivo de verificar o que foi feito no dia anterior para auxiliar o *Product Owner* e *Scrum Master* a identificar interferências e priorizar o trabalho do dia que se inicia.

## 2.3.5 *Review Meeting*

Para que a equipa esteja ciente das alterações do projeto ou de novas funcionalidades que estão a ser desenvolvidas é criada uma reunião em que visa a apresentação dessas funcionalidades implementadas numa num Sprint. Por vezes desta reunião podem seguir alterações ao projeto uma vez que nesta reunião tudo é visto ao pormenor.

## 2.3.6 *Sprint Retrospective*

Entre todos os sprints deve ser analisado o que correu bem versus o que correu mal, para que tal tenha o conhecimento da equipa normalmente é enviado um email ou feito um documento com uma retrospetiva no fim de um sprint. Neste documento devem ser indicados todos os fatores que influenciam a duração da sprint também como alguma interferência externa (por exemplo férias, mudanças na legislação) ou interna (por exemplo falha na comunicação, compreensão ou alteração do produto na última hora) para que o *Scrum* Master tenha uma noção do que levou a sprint a aumentar ou diminuir a sua duração.

## Gestão da Estrutura Scrum

Dimensão das Equipas

O tamanho das equipas *Scrum* normalmente ronda entre 5 a 7 profissionais, visto que esta metodologia segundo os seus criadores funciona melhor do que em equipas de maior dimensão. O grande intuito desta estrutura não é o foco só no que ocorreu erradamente mas também verificar o que correu bem e no que pode ser melhorado para um próximo *sprint* e sendo numa equipa mais pequena não só é mais fácil identificar e gerir problemas, soluções ou funcionalidades como também de coordenar.

## 2.4.1 Scrum Master

Este é compreendido como o responsável por coordenar e auxiliar as equipas na metodologia *Scrum* e de também garantir que a mesma é compreendida e posta em prática por toda a equipa ou seja ele é responsável por garantir que a equipa segue o processo, como também é um auxilia com a remoção de interferências e obstáculos à equipa sendo que atua como um líder-servo da equipa.

## 2.4.2 Product Owner

O *product owner*, como a própria palavra o diz, tem a posse do produto, isto significa que é um membro da equipa *scrum* e que tem de ter uma compreensão sólida do negócio e mercado e que tem de representar a equipa e os *stakeholders* no projeto.

Este é responsável pelo *product backlog* e por transmitir uma visão das funcionalidades que acrescentam valor para a empresa e para a equipa.

A nível de trabalho de equipa o *product owner* está em contacto direto com a equipa de desenvolvimento e trabalha com a mesma para assegurar que o *sprint* e seus requisitos estão conforme as expectativas, também quando este é concluído e bem como o *product backlog* é constantemente revisto. Podemos também acrescer que a responsabilidade de verificar e manter as especificações tanto como informar a equipa de eventuais modificações pertence tanto ao *product owner* como ao *scrum* *master*.

## 2.4.3 Equipa de Desenvolvimento

Esta equipa engloba todos os programadores, *designers*, *testers*, administradores de bases de dados, arquitetos entre outros necessários para cada empresa. Todos os membros da equipa de desenvolvimento devem possuir em coletivo as habilidades necessárias para desenvolver o que é solicitado num *sprint* e pelo *Product Owner*.

A equipa de desenvolvimento é responsável pela conceção, construção e testes de um produto ou seja tem de efetuar todo o trabalho que lhes é destinado ajudando-se internamente para que a mesma seja concluída a tempo (antes do fim da *sprint*). Podemos também salientar que o grande foco desta equipa é a cooperação para um objetivo, podendo então dizer que cada membro terá de ter uma postura e atitude de um por todos e todos por um.

# Capítulo 2 – Caso de Estudo

## 3.1 Modelo de Gestão de Tarefas Anterior (CADS Software LTD)

No ano de 2017 quando dei início à posição de programador *front-end* na empresa CADS Software LTD, o modelo de organização de tarefas era baseado em *emails* enviados pelos chefes de equipa e pela equipa de desenvolvimento, e de fichas de Microsoft Excel com uma lista de tarefas a ser efetuadas para toda a equipa. Essa lista era partilhada em Microsoft Sharepoint para que todos tivessem acesso a mesma e continha várias folhas com diferentes tarefas:

* **Lista de bugs existentes e novos** - nesta lista incluía todos os problemas que o *software* na versão online continha tanto como novos problemas na versão de testes (*pre-release*), ou seja das novas funcionalidades que já estariam a ser integradas.
* **Lista de novas funcionalidades a implementar** - esta lista englobava todas as funcionalidades que a equipa de vendas e a equipa de desenvolvimento propõem para mais tarde implementar.
* **Lista de prioridades definida pela equipa de vendas** - baseada na lista de novas funcionalidades, selecionam quais as tarefas que teriam mais impacto no software e quais seriam fulcrais para os clientes.

Para além das listas também eram enviados *emails* para todas as equipas a confirmar as que seriam desenvolvidas num período de tempo (normalmente um mês), sendo um processo demoroso e muitas vezes difícil de aferir quais as alterações. Muitas destas alterações eram por vezes efetuadas por email e não alteradas nas listas, tornado o processo de desenvolvimento mais fragmentado na pesquisa da informação correta.

## 3.2 O novo modelo de gestão de tarefas (CADS Software LTD)

Em Janeiro de 2019, após várias pesquisas e análises a diferentes *softwares* a empresa decidiu adotar o pacote de software da *Atlassian* que compreendia os subcomponentes: *Jira*, *BitBucket* e *Confluence*. Esses pacotes de softwares foram instalados e expandidos por todas as equipas de desenvolvimento da empresa num prazo de 3 meses.

A grande vantagem deste pacote de *softwares* é que se interligam entre si, ou seja quando uma tarefa é criada esta cria um id que posteriormente pode ser convertido num ambiente seguro para programar essa tarefa no software (*branch*). Também uma das grandes vantagens é que pode ser criado um sprint, como referido anteriormente, um período de tempo para o qual uma lista de tarefas tem de ser concluída, isto obriga a equipa a se ajudar mutuamente para que tal seja concluído a tempo.

## 3.2.1 Tarefa

* Para a gestão eficiente da equipa é necessário atribuir tarefas consoante as diferentes aptidões dos membros da mesma, para isso são criadas tarefas com imagens, descrições e atributos como:
* **Prioridade** - existem vários tipos de prioridades como Urgente, Normal, Baixa Prioridade;
* **Sprint pertencente** - referência o período de tempo em que uma lista de tarefas tem de ser concluída;
* **Estado da Tarefa** - a tarefa terá vários estados consoante o grau de desenvolvimento da mesma, por exemplo: A Fazer, Em Progresso, Em Revisão e Completada;
* **Atribuição da tarefa** - as tarefas serão distribuídas pela equipa mas somente podem ser adicionadas para um trabalhador. Se necessário a tarefa pode ser atribuída a outro programador para ser completada.

As tarefas podem ser criadas por qualquer membro da equipa e são destinadas ao *Product Backlog* que armazena todas as tarefas por efetuar, mais tarde estas podem ser adicionadas a *sprints* e atribuídas a apenas um membro da equipa.

## 3.2.2 Diagrama de Contexto

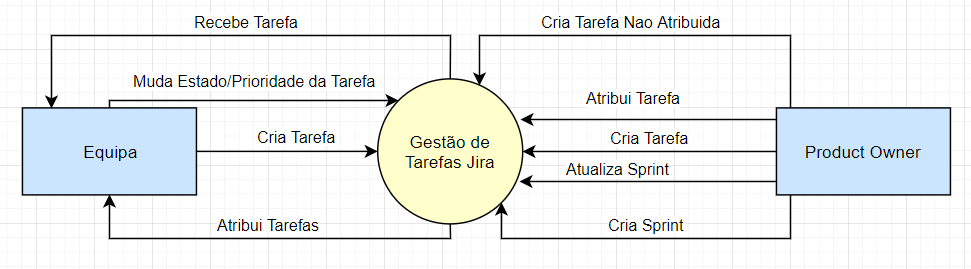


Figura 1 - Diagrama de Contexto da gestão de tarefas da equipa de desenvolvimento

## 3.2.3 Diagrama Nível 1

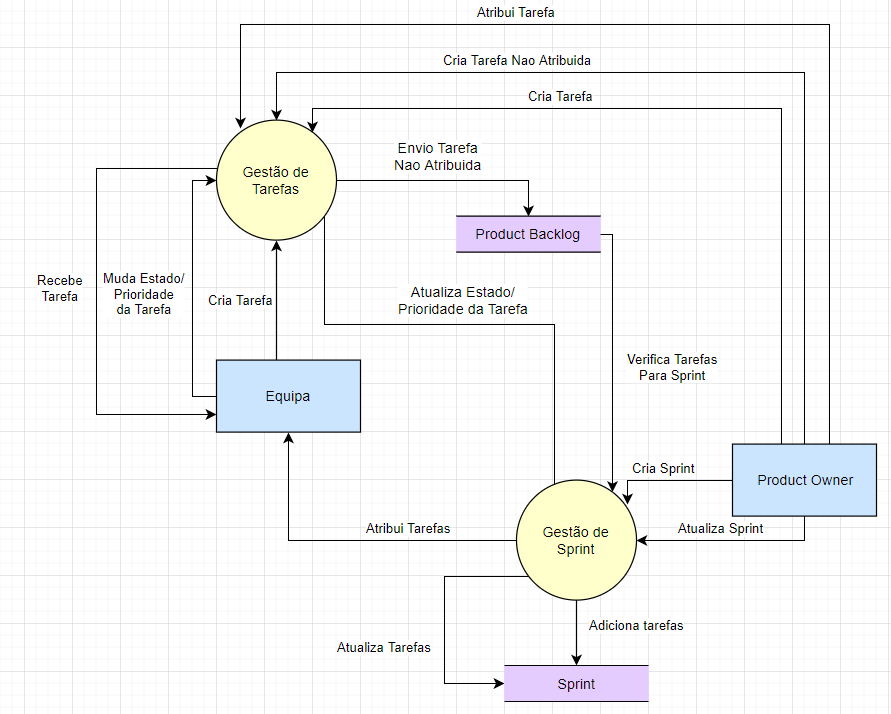


Figura 2 - Diagrama de nível 1 da gestão de tarefas da equipa de desenvolvimento

# 4. Conclusão

Com esta implementação da metodologia *Scrum* foi possível a melhor organização da equipa de desenvolvimento e melhor gestão de documentação para suportar futuras alterações. Também importante de salientar que a comunicação entre os membros da equipa melhorou e tornou-se mais focada sendo que todos os parâmetros de uma tarefa são registados e facilmente acedidos.

No lado de demonstração dos produtos ao cliente também foi possível estimar com mais clareza quando é que as próximas versões irão ser disponibilizadas, o mesmo acontece para novas funcionalidades.

No entanto pelo lado menos favorável este processo requer alterações constantes pela equipa de desenvolvimento sendo que todas as informações devem constar no *software* levando a mais tempo dispensado para estes registos.

# Referências e Bibliografia

*Product Owner*. (23 de 12 de 2019). Obtido de Scrumportugal: http://www.scrumportugal.pt/product-owner/

Schwaber, K., & Sutherland, J. (7 de 2016). *O Guia do Scrum™*. Obtido de Scrumguides: https://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Portuguese-European.pdf

Schwaber, K., & Sutherland, J. (November de 2017). *The Scrum Guide™*. Obtido de Scrumguides: https://www.scrumguides.org/scrum-guide.html

*SCRUM*. (23 de 12 de 2019). Obtido de desenvolvimentoagil.com.br: https://www.desenvolvimentoagil.com.br/scrum/

*Scrum Master*. (23 de 12 de 2019). Obtido de Scrumportugal: http://www.scrumportugal.pt/scrum-master/

*The Scrum*. (23 de 12 de 2019). Obtido de worldrugby.org: https://passport.worldrugby.org/?page=beginners&p=12

*What is the Evolution of Scrum?* (23 de 12 de 2019). Obtido de Visual Paradigm: https://www.visual-paradigm.com/scrum/what-is-the-evolution-of-scrum/