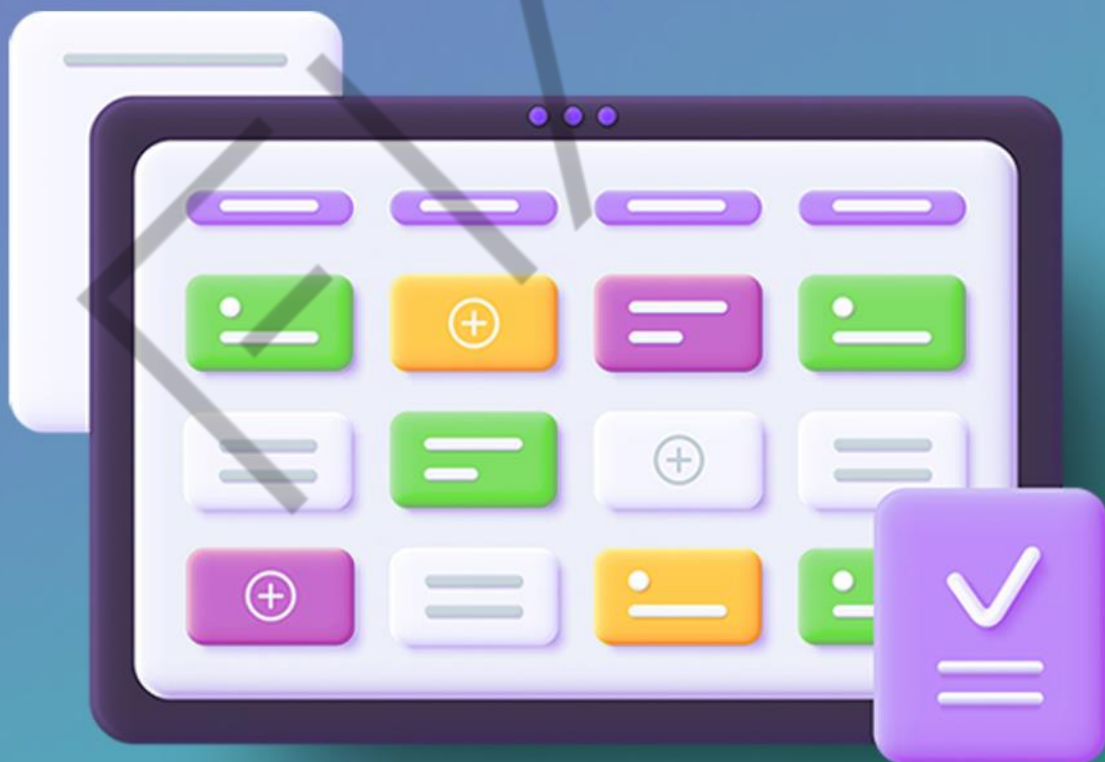


DEVELOPMENT ENVIRONMENT

MATERIALIZANDO **O PROJETO**



9

LISTA DE FIGURAS

Figura 1 – Quadro de tarefas Kanban12



LISTA DE QUADROS

Quadro 1 – Backlog do Produto	7
Quadro 2 – Backlog inicial do primeiro Sprint.....	8
Quadro 3 – Backlog do primeiro Sprint, primeira atualização	9
Quadro 4 – Lista de <i>softwares</i> a serem instalados.....	10
Quadro 5 – <i>Backlog</i> atualizado do primeiro Sprint	10
Quadro 6 – Quadro de tarefas do primeiro Sprint	12

EXEMPLO

SUMÁRIO

1 MATERIALIZANDO O PROJETO	5
1.1 Detalhando o escopo	5
1.2 Planejamento de nossa primeira história de negócios	7
1.3 Planejando nossa primeira história técnica	8
1.4 Mas essa história ainda não acabou	9
1.5 Organizando as atividades em um Kanban	11
REFERÊNCIAS	15

EMANIP

1 MATERIALIZANDO O PROJETO

1.1 Detalhando o escopo

Vamos retomar um pouco do trabalho em gestão, caso contrário, perderemos o fio da meada no caos da desorganização. Para isso, vamos pegar emprestados alguns conceitos de gestão de projetos ágeis que utilizam um *framework* conhecido como Scrum.

A lista do que precisa ser feito nesse projeto de desenvolvimento é chamada de *Backlog* do Produto. Trata-se de uma “lista viva”: itens poderão ser retirados e adicionados conforme vão sendo revelados durante o detalhamento de escopo do projeto. O responsável por definir o que fará parte da lista e o que é prioridade, concebendo a visão do produto, é o profissional conhecido como Dono do Produto. Esse papel será desempenhado por seu tutor.

Outra maneira de remover itens do *Backlog* do Produto é entregá-los prontos. E isso cabe a você!

A maneira mais usual de se retratar os itens dessa lista é por meio de narrativas, chamadas de histórias do usuário. Você já viu isso antes, lembra-se? Quando você pondera sobre o assunto, faz muito sentido: compreendemos o mundo a partir de relatos; é por meio deles que entendemos os personagens, seus desejos e suas motivações. Por que não tratar as necessidades desse projeto da mesma maneira?

As narrativas possuem este formato: “Como _____ (papel), eu gostaria de / quero poder _____ (funcionalidade ou necessidade) para _____ (benefício)”. Sendo assim, já poderíamos listar:

Backlog do Produto:

- Como usuário, quero poder cadastrar e atualizar meus dados pessoais para manter um histórico.
- Como usuário, quero poder informar meus recebimentos para realizar um acompanhamento adequado.

Materializando o projeto

- Como usuário, quero poder registrar meus gastos para realizar um acompanhamento adequado.
- Como usuário, quero poder registrar os meus investimentos para planejar a vida financeira.

Como chegamos a esse levantamento inicial? Bem, temos várias técnicas para isso: podemos entrevistar o cliente em reuniões, enviar questionários por escrito, entre várias possibilidades, chegando ao que chamamos de requisitos de sistema. Em Gestão de Projetos isso é chamado de escopo, trata-se do que deve ou não ser feito em nosso projeto.

Como é cedo demais para construir tais histórias integralmente, vamos adicionar ao Backlog do Produto uma história de cunho mais técnico.

Backlog do Produto:

- Como usuário, quero poder cadastrar e atualizar meus dados pessoais para manter um histórico.
- Como usuário, quero poder informar meus recebimentos para realizar um acompanhamento adequado.
- Como usuário, quero poder registrar meus gastos para realizar um acompanhamento adequado.
- Como usuário, quero poder registrar meus investimentos para planejar a vida financeira.
- Como desenvolvedor, preciso montar uma infraestrutura adequada para construir outras histórias.

O Dono do Produto priorizou as seguintes histórias:

Backlog do Produto:

História	Prioridade
Como equipe de projetos, precisamos formalizar o projeto desenvolvendo seus primeiros artefatos.	1
Como desenvolvedor, preciso montar uma infraestrutura adequada para construir outras histórias.	2
Como usuário, quero poder cadastrar e atualizar meus dados pessoais para manter um histórico.	3
Como usuário, quero poder informar meus recebimentos para realizar um acompanhamento adequado.	4
Como usuário, quero poder registrar meus gastos para realizar um acompanhamento adequado.	5
Como usuário, quero poder registrar meus investimentos para planejar a vida financeira.	6

Quadro 1 – Backlog do Produto
Fonte: FIAP (2016)

Adicionamos a história “Como equipe de projetos, precisamos formalizar o projeto desenvolvendo seus primeiros artefatos”, que inclui atividades cuja fase você já começou.

Parece que já temos um trilha para seguir.

1.2 Planejamento de nossa primeira história de negócios

As atividades serão dispostas em ciclos de duas a quatro semanas. Tais ciclos também são chamados de Sprints, pois, dada a sua curta duração, lembram corridas de 100 metros rasos. Ao final de cada Sprint, a equipe (formada por você) deverá entregar algo para o cliente, representado pelo Dono do Produto (nesse caso, seu tutor).

Nosso primeiro ciclo será de quatro semanas e uma das atividades é desenvolver o termo de abertura do projeto, essencial para formalizarmos o projeto e apresentá-lo como principal responsável por ele.

Além disso, é necessário definir qual equipamento será utilizado para esse fim: é uma estação de trabalho? Um notebook/laptop? Seja qual for, é essencial que você tenha permissões de administrador, assim, poderá instalar os *softwares* necessários.

BACKLOG DO SPRINT	
Meta do Sprint: concluir a história de usuário “Como equipe de projetos, precisamos formalizar o projeto desenvolvendo seus primeiros artefatos”.	
Atividade	Esforço (em horas)
Desenvolver um termo de abertura do projeto.	4 horas
Definir e levantar a infraestrutura física.	2 horas

Quadro 2 – Backlog inicial do primeiro Sprint
Fonte: FIAP (2016)

Você deve ter observado que já preenchemos com o esforço das atividades em horas; embora tenhamos adiantado tais estimativas, quando falamos em Scrum, esse tipo de decisão é um consenso da equipe de projeto, e você faz parte dela! Ou melhor, você é a equipe! Sendo assim, caso tenha reservas acerca dos valores predefinidos, faça sua contraproposta!

1.3 Planejando nossa primeira história técnica

A prioridade número dois é a história “Como desenvolvedor, preciso montar uma infraestrutura adequada para construir outras histórias” e, para cumpri-la, vamos decompor a história em atividades que serão necessárias para que o objetivo seja atingido.

Nosso objetivo secundário será construir a infraestrutura necessária para que o desenvolvedor possa elaborar todo o restante. Mas qual seria essa infraestrutura?

A propósito, você pode criar uma infraestrutura isolada para trabalhar em seu projeto, de forma a separá-la do seu ambiente pessoal ou de entretenimento. Para tal, seria necessário criar uma máquina virtual e trabalhar a partir dela. Caso não esteja familiarizado com o conceito de virtualização, em linhas gerais, seria instalar um *software* que simula os comportamentos de um *hardware* (similar ao *hardware* real que você possui), possibilitando instalar outro sistema operacional sobre esse *hardware* fictício, criando assim um ambiente isolado.

Embora seja altamente recomendada, essa atividade é considerada opcional. Para fazê-lo, sugerimos o uso do Oracle Virtualbox (<http://www.virtualbox.org/>) ou do Docker (<http://www.docker.com/>), ambos gratuitos e disponíveis para as três principais plataformas (Windows, MacOS X e Linux).

Materializando o projeto

Recomenda-se a atualização do seu sistema operacional, pois quanto mais atualizado ele estiver, maior a estabilidade, segurança e compatibilidade com os *softwares* que serão instalados.

Um *software* processador de textos também é essencial; no entanto, você provavelmente já possui um desses instalado, geralmente o Microsoft Word. Se não for o seu caso, instale-o. Caso você não possua licença para isso, saiba que você pode baixar vários produtos da Microsoft (como o Windows e o próprio pacote Office) pela plataforma FIAP ON, por meio de nosso convênio acadêmico com a Microsoft; é importante que você mantenha seu ambiente regularizado e atualizado. Caso você não utilize Windows, sugerimos a instalação do LibreOffice (<http://www.libreoffice.org/>) e/ou o processador de texto livre de sua preferência. Vamos manter a atividade no *backlog*, se você já o possuir, marcaremos a atividade imediatamente como concluída, combinado?

Por enquanto, temos:

BACKLOG DO SPRINT	
Meta do Sprint: concluir a história de usuário “Como equipe de projetos, precisamos formalizar o projeto desenvolvendo seus primeiros artefatos” e “Como desenvolvedor, preciso montar uma infraestrutura adequada para construir outras histórias”.	
Atividade	Esforço (em horas)
Desenvolver um termo de abertura do projeto.	4 horas
Definir e levantar a infraestrutura física.	2 horas
[OPCIONAL] Instalação de um ambiente de virtualização.	2 horas
Instalação ou atualização do sistema operacional.	4 horas
Instalação de um software processador de textos.	3 horas

Quadro 3 – Backlog do primeiro Sprint, primeira atualização

Fonte: FIAP (2016)

1.4 Mas essa história ainda não acabou

Bem, o que detalhamos até o momento pode ser considerada uma infraestrutura básica; precisamos relacionar os *softwares* específicos e criar as atividades que os instalam.

Veja, abaixo, um quadro com os *softwares* que utilizaremos:

Materializando o projeto

Software	Descrição	Website
Eclipse	IDE para desenvolvimento Java	http://www.eclipse.org/
Apache Tomcat 9.0.56	Servidor Web do tipo WebContainer	http://tomcat.apache.org/
Oracle SQL Developer Data Modeler 21.4	Ferramenta de modelagem relacional para o banco de dados Oracle	http://www.oracle.com/technetwork/developer-tools/datamodeler/overview/index.html
Oracle SQL Developer 21.4	Cliente para o servidor de banco de dados Oracle	http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html
Python	Linguagem de Programação	https://www.python.org/
VS Code	Editor para desenvolvimento em várias linguagens	https://code.visualstudio.com/
Postman	Plataforma para trabalhar com API	https://www.postman.com/
NodeJS 16.13.1	Ambiente de execução JS Server Side	https://nodejs.org/en/

Quadro 4 – Lista de *softwares* a serem instalados
Fonte: FIAP (2021)

Este é, portanto, o *backlog* do Sprint atualizado:

BACKLOG DO SPRINT	
Meta do Sprint: concluir a história de usuário “Como equipe de projetos, precisamos formalizar o projeto desenvolvendo seus primeiros artefatos” e “Como desenvolvedor, preciso montar uma infraestrutura adequada para construir outras histórias”.	
Atividade	Esforço (em horas)
Desenvolver um termo de abertura do projeto.	4 horas
Definir e levantar a infraestrutura física.	2 horas
[OPCIONAL] Instalação de um ambiente de virtualização.	2 horas
Instalação ou atualização do sistema operacional.	4 horas
Instalação de um <i>software</i> processador de textos.	3 horas
Instalação do Eclipse.	2 horas
Instalação do Apache Tomcat e sua integração com o Eclipse.	3 horas
Instalação do Oracle SQL Developer Data Modeler.	2 horas
Instalação do Oracle SQL Developer.	2 horas
Instalação do Python.	1 hora

Quadro 5 – *Backlog* atualizado do primeiro Sprint
Fonte: FIAP (2019)

Talvez você já possua o navegador web Google Chrome instalado. Se este for o seu caso, marcaremos a atividade como concluída, certo?

Conforme combinado, os tempos propostos podem ser revistos, afinal, instalar um VisuAlg no Linux com WINE pode levar mais de uma hora, dada a complexidade.

1.5 Organizando as atividades em um Kanban

Agora que temos uma lista de tarefas para o primeiro Sprint, é necessário realizar um controle de suas execuções: essa determinada tarefa foi concluída? O que está sendo feito neste exato momento? Existem obstáculos para cumprir alguma delas?

Há algumas possibilidades para organizar nossas tarefas. A situação ideal é elaborar um cronograma com todas as atividades, dependências entre elas, responsáveis e respectiva duração, em uma linha temporal. Softwares como Microsoft Project são muito utilizados para isso, e você aprenderá a elaborar um mais adiante. Entretanto, propomos usar uma ferramenta mais simples e ágil: o Kanban.

No fim dos anos 1940, a Toyota começou a organizar seus processos utilizando o mesmo sistema utilizado por supermercados no controle de estoque de suas prateleiras. O sistema consistia na passagem de cartões de sinalização (ou *kanban*, em japonês) entre as equipes: quando um determinado insumo na linha de produção faltava, um cartão era passado para a equipe de suprimento com detalhes do que precisava ser repostado.

Esse sistema de cartões passou a ser utilizado em áreas, entre elas no desenvolvimento de *software*. O quadro de tarefas (ou *taskboard*) é dividido em pelo menos três colunas: “a fazer”, “fazendo” e “feito”, e os cartões, cada um representando uma atividade, migra de uma coluna para outra de acordo com seus progressos.

A ferramenta é fácil de usar e muito prática, como você deve ter percebido. Além disso, é totalmente alinhada com o conceito ágil de equipes autogerenciadas: qualquer membro da equipe pode mover os cartões das atividades pelas quais é responsável, sinalizando aos demais o seu progresso. Rapidamente, obtém-se uma visão do que está sendo feito, do que já foi feito e do que resta fazer.

Materializando o projeto



Figura 1 – Quadro de tarefas Kanban
Fonte: Shutterstock (2018)

Vamos organizar nossas tarefas?

O quadro se inicia assim:

A fazer	Fazendo	Feito
Definir e levantar a infraestrutura física.		
[OPCIONAL] Instalação de um ambiente de virtualização.		
Instalação ou atualização do sistema operacional.		
Instalação de um <i>software</i> processador de textos.		
[OPCIONAL] Instalação do VisuAlg.		
Instalação do Pencil.		
Instalação do Astah Community.		
Instalação do Google Chrome.		
Instalação do Eclipse Neon.		
Instalação do Apache Tomcat e sua integração com o Eclipse.		
Instalação do Oracle SQL Developer Data Modeler.		
Instalação do Oracle SQL Developer.		

Quadro 6 – Quadro de tarefas do primeiro Sprint
Fonte: FIAP (2019)

Materializando o projeto

Repare que as tarefas de caráter opcional podem ser removidas do quadro. Não pretende realizá-las? Remova do quadro. Já tem o Google Chrome instalado? Pois, agora, mova as tarefas para a coluna “Feito”.

Esperamos que o quadro seja útil não apenas para organizar as tarefas de seu projeto de graduação, mas também outros projetos pessoais, acadêmicos e profissionais, ao longo de sua vida.

Dicas

Há uma grande resistência em utilizar Scrum em projetos grandes. Isso é possível, mas não significa que seja fácil de ser implementado.

O conceito para trabalhar com grandes times é chamado de *Scrum of Scrums*, pois no Scrum temos limite de componentes (9 pessoas). Portanto, quando falamos em escalar o Scrum, estamos falando em criar tantos times de 9 pessoas quantos forem necessários para atender à demanda do projeto.

Nesse cenário de vários times trabalhando ao mesmo tempo, cada time terá seu *Scrum Master* e seu *Product Owner*, mas talvez haja vários problemas decorrentes disso, entre eles:

- Conflitos de liderança.
- Falhas de comunicação.
- Planejamento descentralizado.
- Duplicação de esforço e retrabalho.
- Problemas com a integração dos produtos.
- Dependências entre as atividades.
- Difícil controle de mudanças.

Para solucionar esses problemas, existem várias sugestões, inclusive um guia específico só para tratar desse assunto, chamado de *Nexus Guide*.

Vamos listar alguns tópicos que ajudam na solução de alguns itens:

- Dividir o escopo do projeto em incrementos mais independentes possíveis entre si, diminuindo as dependências entre os times.

Materializando o projeto

- Criar papéis para a centralização de decisões no nível do *Scrum Master* e do *Product Owner*.
- Criar um *Product Backlog* único.
- Criar marcos (pontos de controle) para unificação de *releases* e publicação de uma nova versão.
- Fazer reuniões diárias conjuntas para que representantes de outros times possam participar, com o objetivo de compartilhar problemas, dependências e conhecimento.
- Criar uma reunião de Scrum de Scrums para o acompanhamento das atividades, verificação de dependências e centralização das mudanças.

Não é o objetivo aqui esgotarmos esse assunto, mas sim oferecer uma visão geral e pontos críticos que devem ser observados em eventual necessidade de utilizar vários times.

IMPORTANTE:

Para mais detalhes a fim de escalar o Scrum, consulte o *Nexus Guide* no site: www.scrum.org.

Algumas ferramentas gratuitas:

- **ScrumHalf** – brasileira. Facilita o uso do Scrum. Tem um quadro Kanban virtual e facilita a colaboração e o acompanhamento da equipe. É de fácil manutenção, prioriza o *Product Backlog* e gera o *Burn Down Chart*, entre outros: <http://myscrumhalf.com/?lang=pt>.
- **Trello** não é uma ferramenta de Scrum, mas automatiza o quadro *Kanban*: <https://trello.com/>.

REFERÊNCIAS

ATLASSIAN. **Kanban**: How the kanban methodology applies to software development. [s.d.]. Disponível em: <<https://www.atlassian.com/agile/kanban>>. Acesso em: 14 dez. 2023.

SUTHERLAND, Jeff. **Scrum**: A arte de fazer o dobro de trabalho na metade do tempo. Rio de Janeiro: Casa da Palavra/Leya, 2014.

EXEMPLO