

# INICIAÇÃO DO PROJETO

## PRÉ TAREFA:

### TERMO DE ABERTURA DO PROJETO

Após a primeira reunião com o interessado o Product Owner (PO) deve elaborar o Template do Termo de Abertura do Projeto (TAP), que define os compromissos das partes para executarem o projeto.

### (NOVO) Customizar o SpinOff

Dependendo do tipo, tamanho ou complexidade do projeto pode ser necessário customizar o SpinOff para o projeto.

Neste caso o Scrum Master deve criar o documento Caso de Desenvolvimento e orientar o Time de Desenvolvimento sobre as customizações realizadas. O objetivo é orientar os membros do time e novos membros das mudanças realizadas em relação ao processo padrão SpinOff.

## DEFINIR O ESCOPO

### Levantar necessidades do cliente

Assim que um cliente manifesta interesse em automatizar seus processos de negócio e depois de assinar o TAP, o Time de Desenvolvimento planeja e executa as primeiras reuniões para identificar as necessidades e problemas para propor soluções de software. O objetivo é ter tantas reuniões quantas forem necessárias para se definir o escopo do sistema.

Prepare a entrevista antes, ver o Guia - Técnica de Entrevista, para realizar o levantamento de requisitos utilizando a Técnica de Entrevista. Utilize a Ata de Reunião para registrar as questões e posteriormente complemente-a com as respostas. Lembrando que é uma boa prática gravar as entrevistas para depois, logo após a reunião, consolidar o entendimento utilizando a Guia - Workshop de Requisitos. E toda Ata de Reunião deve ser aprovada pelos interessados ou fornecedores de requisitos, assinando ou dando o "de acordo" por e-mail.

Uma vez identificado os problemas e necessidades registre-as no documento Visão.

### Definir o sistema (escopo)

Os requisitos podem ser definidos como “serviços que o sistema deve fornecer para preencher uma ou mais necessidades”. Ver Guia - Requisitos de Sistema de Software.

Os problemas e necessidades devem ser bem entendidos para propor uma boa solução e devem ser documentados no documento Visão, o qual deve fornecer uma visão geral do sistema a ser desenvolvido.

Com base nas necessidades definidas no Visão (na forma de Histórias de Usuário) deve-se elaborar o Modelo de Caso de Uso. Normalmente as Histórias são mapeadas ao Modelo de Caso de Uso um-para-um, os quais definem as principais funcionalidades que compõem o escopo do sistema.

Uma boa prática é fazer uma descrição resumida para cada caso de uso do Modelo de Caso de Uso. Sugere-se a ferramenta Astah (UML) para modelagem e descrição dos casos de uso.

Todos os termos e jargões da área de negócio devem ser identificados e definidos no Glossário.

### **Aprovar o escopo do sistema**

O Desenvolvedor deve apresentar a definição do sistema ao Product Owner (PO) e/ou cliente para validá-los. Lembrando que esse é o escopo inicial e que poderá haver modificações ao longo do desenvolvimento e as modificações serão tratadas como aditivos ao projeto, através do processo de Gerenciamento de Configuração e Mudanças. Ver a tarefa Controlar Mudanças no Sistema.

## **PLANEJAMENTO**

### **Planejar o projeto**

Após definido o sistema o Time de Desenvolvimento se reúne para planejar o projeto. O Scrum Master é o responsável em elaborar e manter o plano do projeto (Planilha de Planejamento e Controle do Projeto) atualizado.

Neste momento é elaborado a primeira composição do Backlog do Produto (Product Backlog) com base no Modelo de Caso de Uso e no Visão. Nele o time deve listar todos os Casos de Uso, que poderão ser desmembrados em tarefas menores.

### **Criar plano de gestão de riscos**

É muito importante durante o planejamento do projeto identificar os riscos ao projeto.

O Time de Desenvolvimento e o Scrum Master devem se reunir para identificá-los e planejar a eliminação, mitigação ou contingência de cada um.

O Visão e as respostas aos questionamentos do Guia - Identificação de Riscos ajudam a identificar os riscos do projeto.

Todos os riscos identificados devem ser relacionados na planilha de gestão de riscos (Planilha de Planejamento e Controle do Projeto) para monitoramento e controle.

Em todas as Reunião de Revisão do Sprint (Sprint Retrospective) no final de cada Sprint os riscos devem ser reavaliados.

### **Priorizar os itens do Backlog do Produto**

O Backlog do Produto (Product Backlog) é uma lista priorizada de trabalhos a serem concluídos pelo time. É continuamente, atualizada pelo Product Owner (PO) que se baseia no feedback vindo do usuário final (à medida em que são expostos a novos incrementos de produtos) e, também, em resposta às evoluções técnicas e cenários de negócios.

Para a priorização das tarefas do Backlog do Produto pode-se dar de duas formas a critério do Time de Desenvolvimento e conduzido pelo PO:

1. Informalmente: Definido pelo cliente ou pelo PO, ver Guia - Método de Priorização VRDC;
2. Formalmente: Aplicando-se um questionário ao cliente (Questionário Kano). Neste caso o PO deve usar o método de priorização conforme Guia - Método Kano para Priorização de Requisitos.

### **Estimar os itens do Backlog do Produto**

Neste momento são definidos os responsáveis pelas tarefas. Lembrando que o time é autogerido, portanto as tarefas não devem ser impostas, mas sim escolhidas pelos desenvolvedores, que deverão ajudar nas suas estimativas de esforço.

Para estimar o esforço em horas das tarefas do Backlog do Produto (Product Backlog), o Time de Desenvolvimento pode se utilizar de dois métodos:

1. Técnica de Estimativa Planning Poker, conforme Guia - Planning Poker ou

## 2. Técnica de Estimativa Análise de Pontos de Função, conforme Guia - APF.

### **Criar o plano de entregas**

Neste momento o time de desenvolvimento deverá criar um plano de entregas (uma aba da Planilha de Planejamento e Controle do Projeto) em alto nível contendo o que será entregue ao final de cada Sprint.

O plano de entregas define quantas entregas serão realizadas aos interessados com base no escopo inicial (Entregas correspondem a uma configuração do sistema, que inclui artefatos e Incremento do Produto (Product Increment)) a serem disponibilizadas para uso dos interessados.

### **Aprovar o Plano do Projeto**

Uma vez que o Plano do Projeto (Planilha de Planejamento e Controle do Projeto) estão prontos, o Scrum Master marca uma reunião com o cliente e/ou Product Owner (PO) para apresentar o plano e obter sua aprovação.

Somente após a aprovação, marco do primeiro Sprint, é que se pode iniciar o desenvolvimento propriamente dito.

## **Lista para Identificação de Riscos**

Fontes de riscos são as possíveis causas da ocorrência de riscos nos projetos. Este guia apresenta questionamentos que podem auxiliar o time a identificar riscos comuns em projetos de software. Elas são aqui representadas e agrupadas de acordo com as **categorias de riscos**. Uma vez identificados os riscos devem ser monitorados através da planilha de planejamento e controle do projeto.

### **Gestão:**

Percebe-se falta de comprometimento da organização em relação ao projeto?

O projeto possui baixa prioridade para a organização?

O orçamento do projeto é insuficiente para a sua conclusão?

Existem restrições de orçamento no projeto?

Percebe-se que as estimativas de custo do projeto são imprecisas?

Existe dificuldade de definir um valor monetário (precificação) para os produtos do projeto?

O prazo atribuído ou estimado para o projeto é insuficiente para a conclusão de suas atividades?

O projeto possui restrições de prazo?

A quantidade de profissionais alocados à equipe é insuficiente para executar as atividades planejadas para o projeto?

A equipe do projeto não possui habilidades e experiência adequadas às características do projeto?

A equipe do projeto nunca trabalhou junta anteriormente?

Existe alta rotatividade da equipe alocada no projeto?

Percebe-se a existência de conflitos interpessoais entre os membros da equipe do projeto?

Percebe-se a falta de planos formais para direcionamento e controle das atividades do projeto?

A equipe do projeto é incapaz de realizar as atividades estabelecidas no processo?

A quantidade de estações de trabalho é insuficiente para a equipe do projeto?

A quantidade de licenças de software essenciais ao desenvolvimento do projeto é insuficiente?

As ferramentas de software disponibilizadas para a realização das atividades do projeto são inadequadas?

#### **Técnicos:**

Os requisitos do projeto foram especificados pelo Cliente?

Os requisitos do projeto são instáveis?

As especificações dos requisitos do projeto são incompletas?

Os fornecedores de requisitos não estão disponíveis para tal atividade?

O projeto não utiliza protótipos para representar e validar os requisitos?

O projeto possui restrições tecnológicas que possam afetar negativamente o projeto?

Os requisitos do projeto não foram validados pela equipe do projeto?

A arquitetura tecnológica do projeto não foi validada pela equipe do projeto?

O sistema quando em produção deve movimentar um grande volume de dados e acessos?

Percebe-se dificuldade em garantir os requisitos de portabilidade, manutenibilidade e confiabilidade definidos para o sistema?

A arquitetura de hardware é incompatível com os requisitos tecnológicos do sistema?

O sistema caracteriza-se por possuir uma grande quantidade de interfaces com sistemas externos?

O projeto envolve uma grande diversidade de tecnologias?

Os requisitos do projeto são difíceis de implementar tecnicamente?

O sistema possui características de tempo real e tolerância a falhas?

#### **Externos:**

O Cliente não aprova ou demora a aprovar acordos, mudanças, etc.?

Percebe-se falta de entendimento do cliente sob aspectos técnicos do projeto?

Percebe-se falta de comprometimento do cliente nas aprovações dos produtos do projeto?

Percebem-se problemas de relacionamento entre o cliente e a equipe do projeto?

O projeto depende de ações de outros fornecedores do Cliente, como para a migração de dados ou troca de informações entre fornecedores?

Existem restrições quanto à subcontratação de produtos e serviços do projeto?

Percebem-se problemas de relacionamento entre a equipe do projeto e a equipe do subcontratado?

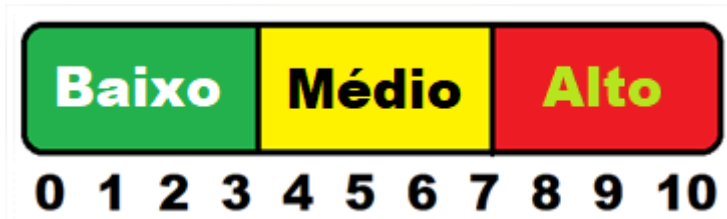
O subcontratado é inexperiente na tecnologia e características do projeto?

## Método VRDC de priorização

VRDC é um acrônimo para as palavras Valor, Risco, Dependência e Complexidade técnica.

O método VRDC de priorização visa destacar com a pontuação mais alta os itens mais importantes do Backlog e que estes itens deveriam ser entregues primeiro.

A técnica consiste em pontuar cada elemento VRDC conforme as regras abaixo:



**Valor:** Pontue de 0 a 10

**Risco:** Pontue de 0 a 10

**Dependência:** Pontue somente 10 caso exista dependência

**Complexidade Técnica:** Pontue de 0 a 10

**V = Valor:** Quanto de valor a história de usuário representa para o usuário final?

**R = Risco:** Qual é o risco desta história não ser entregue?

**D = Dependência:** A história que será pontuada, irá gerar insumos para “outras” histórias no Backlog?

Ex: Temos duas histórias no Backlog, A e B. A história A não consegue ser entregue pois necessita que a história B seja entregue antes. Diante disso a história B receberá a pontuação “10” no método VRDC pois ela precisa ser completada o mais rápido possível para que a história A também seja entregue.

**C = Complexidade Técnica:** Qual a complexidade técnica (tecnologia) da história?

Com base na soma da pontuação final, as histórias terão um peso, resultado desta somatória. Com base neste peso, você conseguirá ter uma visão sobre quais histórias são as mais “pesadas” do seu Backlog.

