

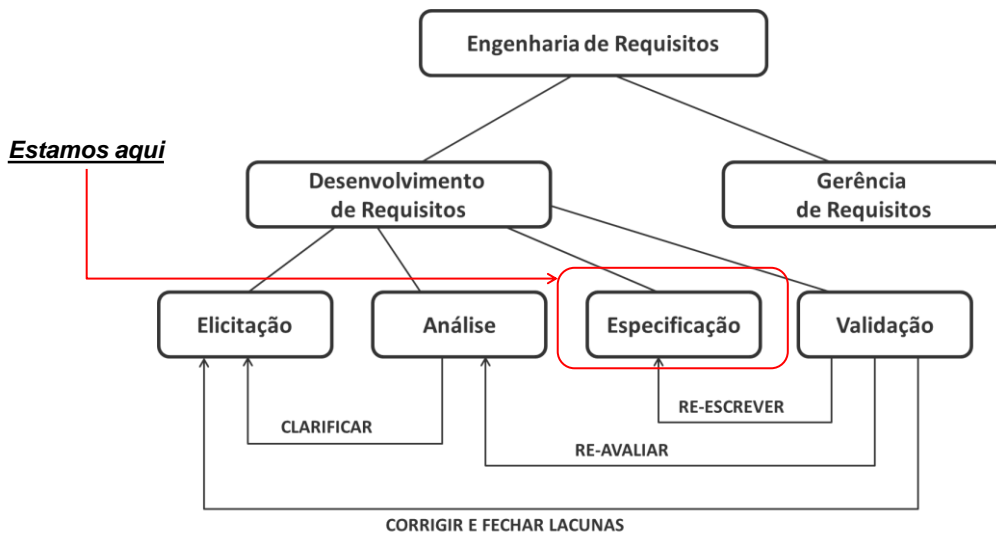
Requisitos de Software

ESPECIFICAÇÃO

1

1

Engenharia de requisitos



2

Requisitos de Software

2



Objetivo geral da especificação

- Comunicação clara e eficaz é o princípio fundamental do desenvolvimento de requisitos.
- A pessoa de requisitos escolherá a maneira mais eficaz de comunicar cada tipo de requisitos para cada tipo de envolvido no projeto.
- Registrar os requisitos de uma forma organizada para que as partes interessadas (stakeholders) possam revisar e chegar a um acordo.
- Requisitos funcionais e não funcionais do produto muitas vezes são armazenados em uma especificação de requisitos de software (SRS, sigla em inglês).

3

Requisitos de Software

3

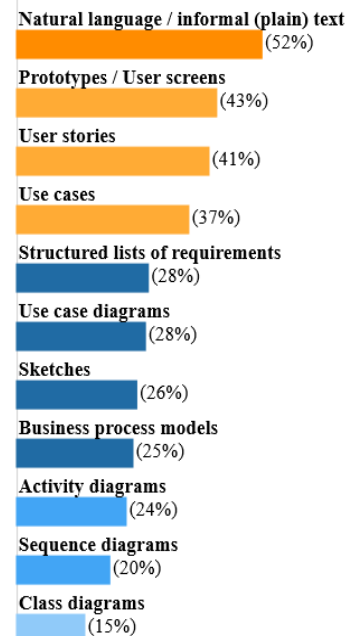
Técnicas de documentação

- Linguagem natural e informal (sem estrutura) mais usado no mundo
- Esboços de interfaces são excelentes para registrar comportamento do ponto de vista do usuário
- Histórias de usuários e Casos de uso são as técnicas baseadas em cenários

Mais usadas

Naming the Pain in Requirements Engineering (NaPiRE) constitutes a globally distributed family of surveys on Requirements Engineering (RE) practices and problems

<http://napire.org/#/explore>



4

Requisitos de Software

4

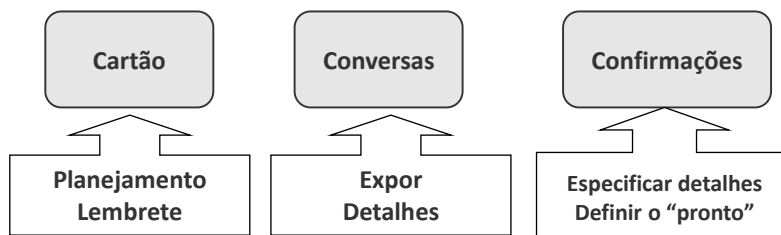
HISTÓRIAS DE USUÁRIO

5

5

Histórias de usuário

- Uma história é uma descrição de funcionalidade ou aspecto que traz valor a um stakeholder.

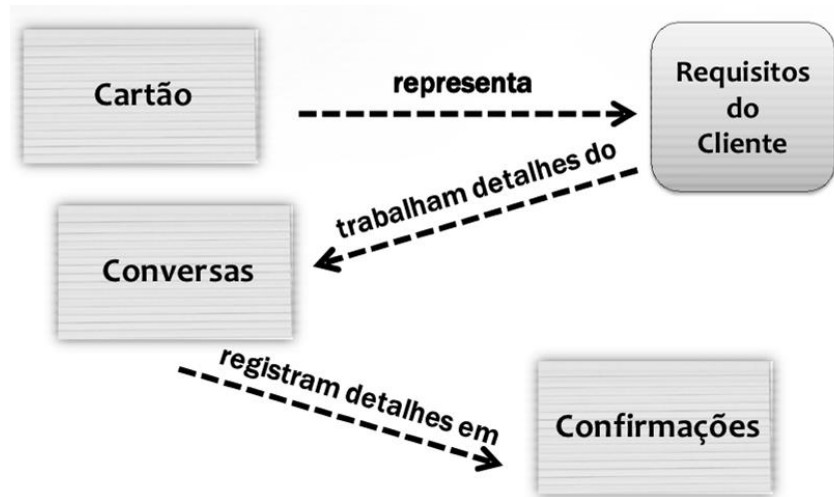


3

6

Histórias de usuário

- Estrutura “3 Cs”



<https://ronjeffries.com/xprog/articles/expcardconversationconfirmation/>

7

Requisitos de Software

7

Histórias de usuário

Cartão

Como um candidato,
eu posso buscar vagas de emprego

- Apresenta um claro valor de negócio e objetivo do usuário
- No entanto, difícil começar a implementar e testar baseado somente nessa descrição

8

Requisitos de Software

8

Histórias de usuário

Conversas

**Como um candidato,
eu posso buscar vagas de emprego**

**+ Opções de busca: por cidade, por estado,
por título do cargo**

- Detalhes que são expostos durante a elicitación
- Ajudando a definir os limites (escopo)
- Sem exagero de detalhes

9

Requisitos de Software

9

Histórias de usuário

**Como um candidato,
eu posso buscar vagas de emprego**



VERSO

+ Opções de busca:

Confirmações

Confirmações:

+ Busca sem campo algum preenchido não é realizada (falha)
+ Apenas cidades com vagas são apresentadas (sucesso)

- Como o cliente avaliará o sistema?
- Como saber que já está pronto?
- Uso de exemplos como critérios de aceitação

10

Requisitos de Software

10



Histórias de usuário

- Servem para requisitos não-funcionais? Sim.

Como cliente, desejo executar seu produto em todas as versões do Windows a partir do Windows 7.

Como usuário, quero que o site esteja disponível 99,999% do tempo que tento acessá-lo, para não ficar frustrado e encontrar outro site para usar.

Como CTO, quero que o sistema use nosso banco de dados de pedidos existente em vez de criar um novo, para que não tenhamos mais um banco de dados para manter.

Como usuário, desejo que as instruções de direção sejam as melhores 90% do tempo e razoáveis 99% do tempo.

Como alguém que fala uma língua latina, talvez eu queira executar seu software algum dia.

11

Requisitos de Software

11



Modelo INVEST

Modelo de avaliação **qualitativa**

- **I**ndependente
- **N**egociável
- **V**aluable (com valor)
- **E**stimável
- **S**mall (pequena)
- **T**estável

Aplicável a histórias que representam requisitos funcionais

12

Requisitos de Software

12



Modelo INVEST - Independência

Exemplos

- #1 Um cliente pode pagar um pedido com Cartão Visa
- #2 Um cliente pode pagar um pedido com Cartão Hipercard

Problemas

- Mesmo objetivo principal de negócio
- Dependência de funcionalidade
- Dependência de estimativa

Alternativas

- Combinar histórias (“... pagar pedido com cartão de crédito”)
- Dividir de outra forma:
 - “... pagar um pedido com um tipo de cartão de crédito”
 - “... pagar um pedido com outros dois tipos de cartão de crédito”

13

Requisitos de Software

13



Modelo INVEST - Negociável

Exemplo

- #1 Um cliente pode adicionar itens ao carrinho de compras usando um link direto na descrição do produto
- #2 Um cliente pode visualizar o carrinho de compras caso haja algum item adicionado

Problemas

- Objetivos de baixo nível
- Detalhes de interface gráfica e regras de negócio

Alternativas

- Focar no objetivo principal
- Deixar os detalhes para as conversas e confirmações
 - “Um cliente pode organizar seu pedido em um carrinho de compras”
 - “Nota: Opção de adicionar no carrinho diretamente da página do produto”
 - “Nota: Não exibir carrinho vazio”

14

Requisitos de Software

14



Modelo INVEST - Valor

Exemplo

- #1 As conexões com o BD são feitas através de um pool de conexões
- #2 Log deve ser guardado em banco de dados

Problemas

- Tem valor apenas para a equipe de desenvolvimento
- Mesmo que trouxessem valor para stakeholder, não são histórias!

Alternativas

- Focar em histórias que tragam valor para o usuário ou para stakeholders
- Focar em atributos de qualidade e não em decisões arquiteturais
 - “Sistema deve suportar 1.000 transações por segundo”
- Faça com que os clientes/stakeholder escrevam histórias, usando sua linguagem e seu entendimento, o que lhe traz “sensação” de valor.

15

Requisitos de Software

15



Modelo INVEST - Estimável

Exemplo

- #1 Usuário pode realizar compras online
- #2 Coordenadores podem solicitar empenho

Problemas

- Conhecimento de domínio
- Conhecimento técnico
- História grande demais

Alternativas

- Participação conjunta de clientes e equipe na elicitação
- Tecnologias desconhecidas devem ser alvos de estudos de viabilidade
- Histórias grandes devem ser quebradas!

16

Requisitos de Software

16

Modelo INVEST – Testável

Exemplos

- #1 Como usuário, eu quero que o sistema seja rápido.
- #2 Como professor, eu quero gerenciar minhas aulas.

Problemas

- Normalmente uma história não é testável porque não passou nos critérios anteriores
- Histórias grande demais ou com muitas dependências

Alternativas

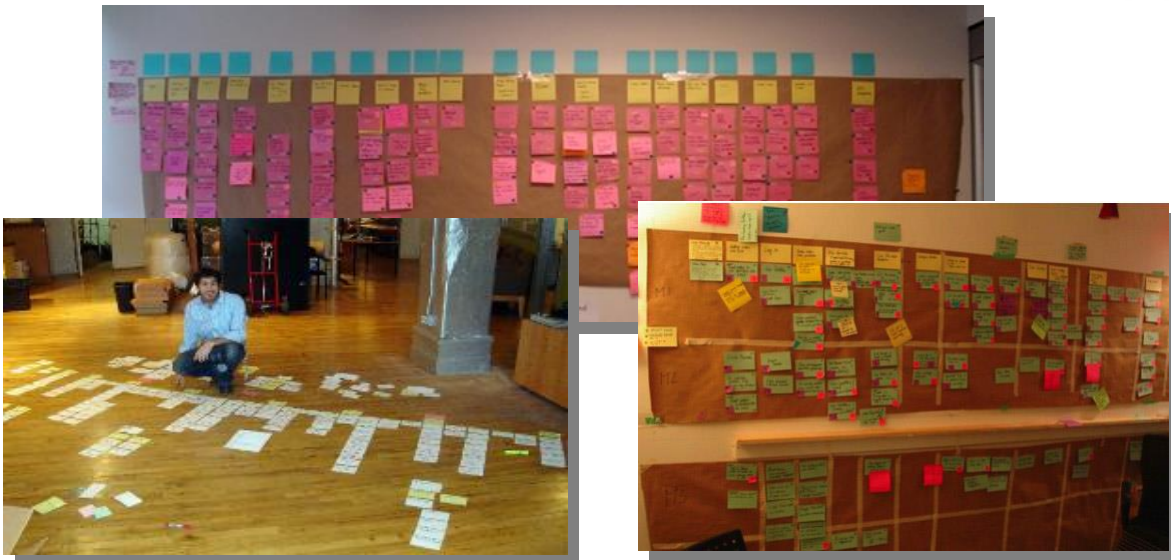
- Ser mais específicos nas restrições não-funcionais
- Quebrar as histórias em partes menores e independentes

17

Requisitos de Software

17

Organizando as histórias



18

Requisitos de Software

18

Histórias de usuário

Contextualizando as histórias

- Relacionar uma história com um requisito de alto nível permite compreender o objetivo maior
- Relacionar uma história com um cenário/narrativa permite compreender o contexto de uso

Como lidar com um projeto com muitas histórias?

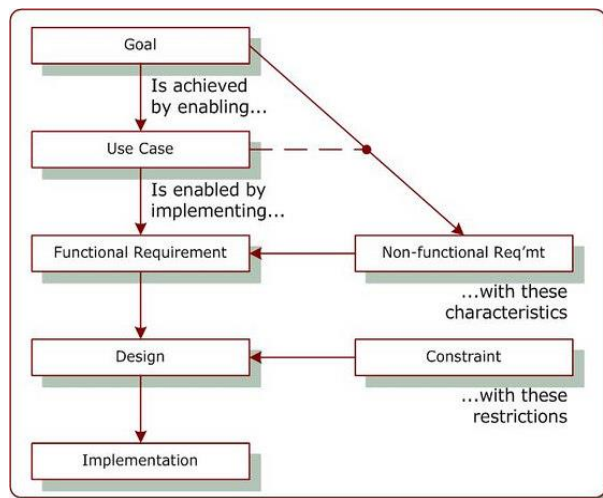
- Objetivos (requisitos de negócio/stakeholder)
- Temas
- Épicos

19

Requisitos de Software

19

Histórias x Objetivos



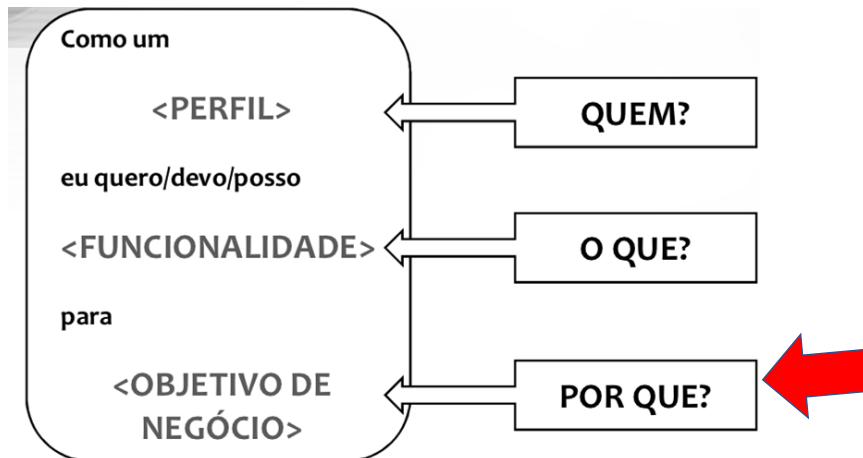
20

Requisitos de Software

20

Histórias x Objetivos

- Um formato para o Cartão



21

Requisitos de Software

21

Histórias x Objetivos

As a/an	I want to...	so that...
moderator	create a new game by entering a name and an optional description	I can start inviting estimators
moderator	invite estimators by giving them a url where they can access the game	we can start the game
estimator	join a game by entering my name on the page I received the url for	I can participate
moderator	start a round by entering an item in a single multi-line text field	we can estimate it
estimator	see the item we're estimating	I know what I'm giving an estimate for
estimator	see all items we will try to estimate this session	I have a feel for the sizes of the various items
moderator	see all items we try to estimate this session	I can answer questions about the current story such as "does this include ..."

22

Requisitos de Software

22



Histórias x Temas

Temas

- Agrupamento livre de estórias para fins de comunicação e planejamento
- Exemplos: “Relatórios”, “Controle de Acesso”, “Painel de Administração”, “Financeiro”, “Setor Pessoal”

Se você fez a elicitação corretamente...

- ... os temas surgiram naturalmente, e foram usados na organização das estórias durante a elicitação

23

Requisitos de Software

23



Histórias x Épicos

Épicos

- História composta
 - Compreende múltiplas estórias menores
 - Opção: dividir em micro-cenários/funções (CRUD ou pequenos objetivos)
 - Opção: dividir pelo dados (entidade principal, entidades agregadas)
- História complexa
 - Inerentemente grande e difícil de ser dividida
 - Exemplos: tecnologias desconhecidas, novos algoritmos
 - Opção: dividir em investigação (spike) e desenvolvimento

Se você fez a elicitação corretamente...

- ... épicos foram identificados, como requisitos de stakeholder ou de solução em alto nível

24

Requisitos de Software

24

Fatiando Épicos

Compostos

- Como candidato, posso manter currículos
- Como candidato, posso editar um currículo
- Como candidato, posso remover um currículo
- Como candidato, posso ter múltiplos currículos
- Como candidato, posso ativar/desativar currículos



Complexos

- Como empregador, posso pagar por uma divulgação de vaga usando cartão de crédito
- Investigar processamento de cartão de crédito através da Internet
- Como empregador, posso realizar pagamentos de serviços com um cartão de crédito



25

Requisitos de Software

25

Temas, Épicos, Histórias, Objetivos, Conversas



Cartões físicos são ótimos para um planejamento colaborativo.

26

Requisitos de Software

26

Temas, Épicos, Histórias, Objetivos, Conversas

Backlog do Projeto

ID	Tema	Como um	Eu posso...	A fim de...	Conversas	Confirmações

Mas para gerenciar, uma planilha é ajuda mais.

Link template: <https://drive.google.com/file/d/18XrXS667wax5OdwMmCtohmUwsoeEi71/view?usp=sharing>

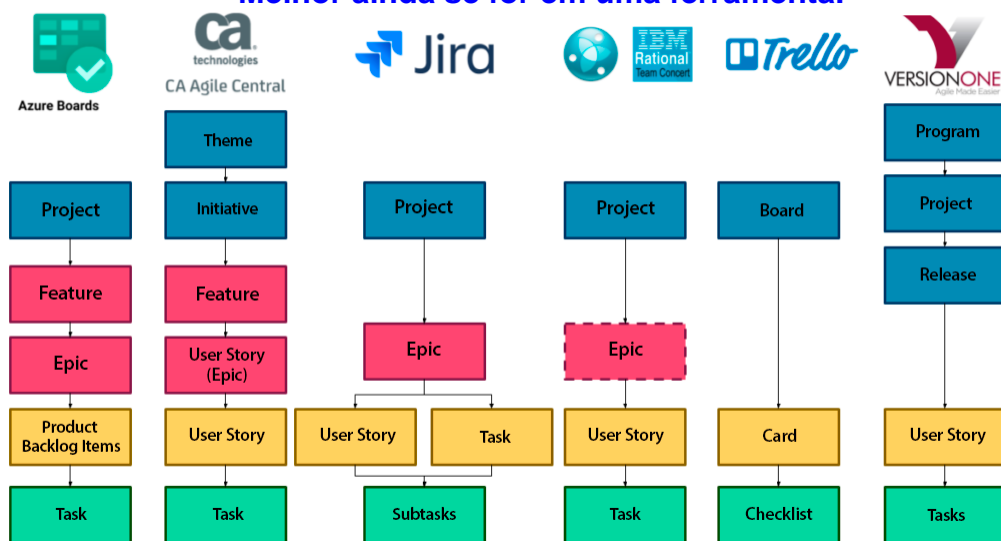
27

Requisitos de Software

27

Temas, Épicos, Histórias, Objetivos, Conversas

Melhor ainda se for em uma ferramenta.



28

Requisitos de Software

28



Conclusões

- Especifiquei, e agora?
- #1 Discuta com clientes, desenvolvedores, e testadores se o nível atual de documentação está adequado
 - Aos cliente: “Está faltando algo importante?”
 - Aos desenvolvedores: “Consegue implementar isso?”
 - Aos testadores: “Consegue elaborar testes para isso?”
- #2 Verifique a consistência das especificações com artefatos anteriores

29

Requisitos de Software