

Softwares são soluções para problemas...

## I – O que é uma metodologia de desenvolvimento de software?

Softwares são complexos, problemas são complexos, comunicação é difícil e geralmente possui problemas

- Como o cliente se comunica com alguém que produz software? **Normalmente:** O cliente não sabe comunicar o que quer, o desenvolvedor não sabe entender o que o cliente precisa, o cliente não sabe que o problema dele pode ser resolvido de forma mais eficiente e não quer mudar
- Como as pessoas que desenvolvem o sistema se comunicam entre si?
  - Como garantir que todos estão resolvendo o mesmo problema? – Cada um pode interpretar o problema de forma diferente
  - Como garantir que todos saibam as decisões tomadas? Como documentar essas decisões?
  - Como garantir que um trabalho não seja feito duas vezes por pessoas diferentes?

Uma forma de tentar resolver estes problemas é utilizando uma metodologia de desenvolvimento, ou seja, um conjunto de passos bem definidos que levem ao desenvolvimento de software de qualidade

**Nem toda metodologia usa a UML** – as metodologias ágeis prezam por entregar código continuamente, recebendo sempre o feedback do cliente

Iconix é uma metodologia que usa **parte da UML** (uma linguagem), alguns outros **artefatos** e **diagramas**.

- O levantamento de cenários é um artefato, não é um diagrama
- O diagrama de robustez não pertence a UML, mas é de um de seus criadores

## II - Como solucionar um problema criando um software? – usando a metodologia Iconix

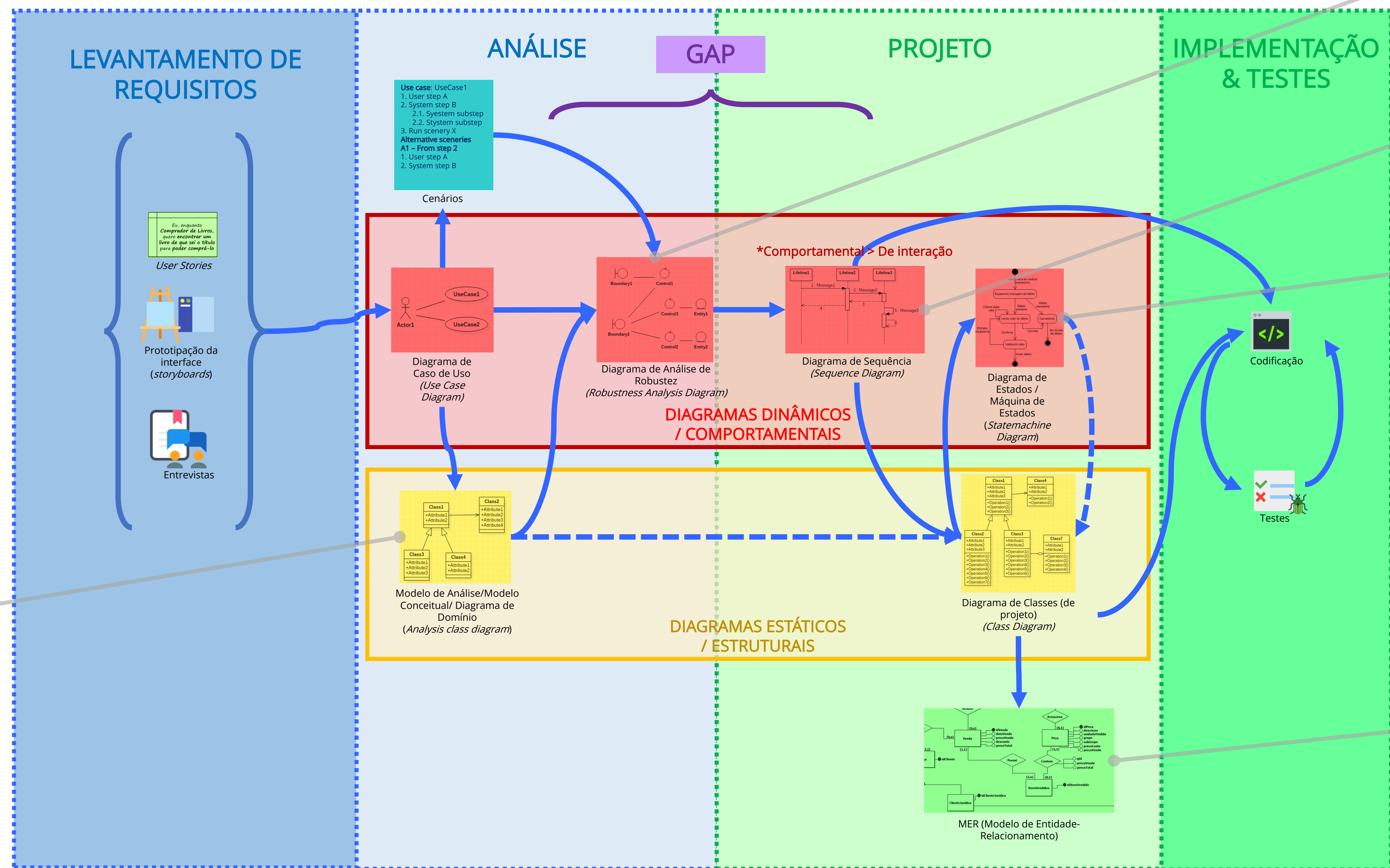
1. Colete dados sobre o problema a ser resolvido
2. Descubra e documente que problema é esse
3. Delimite o escopo da solução
4. Determine o que sua solução irá fazer, quais componentes irá manipular e os termos que vai utilizar para se referir aos componentes do problema e como eles estão relacionados
5. Levante como o usuário vai se comunicar com o sistema passo a passo. Identifique componentes que interagem com os usuários, que processam coisas e que armazenam dados
6. Use as informações levantadas para desenvolver a estrutura de uma solução que possa ser traduzida para código facilmente. Levante como estes componentes interagem entre si ao longo do tempo
7. Levante as partes do sistema que mudam de acordo com regras bem definidas e que podem estar em um conjunto finito de estados – incorpore isso na visão de estrutura e de comportamento
8. Use o projeto para construir o software tanto em sua estrutura (classes, métodos, assinaturas, herança, associações) quanto em seu comportamento (como estes itens estruturais se comportam ao longo do tempo, como interagem entre si e usam os serviços uns dos outros)
9. Procure e corrija possíveis erros que não foram encontrados previamente

Milestones do ICONIX

<https://en.wikipedia.org/wiki/ICONIX>



# PROCESSO ICONIX



A presença do diagrama de robustez é a principal diferença do processo Iconix, ele é utilizado para diminuir o gap entre o diagrama de classes e o de sequência e em última instância entre a fase de análise e de projeto

A principal função do diagrama de sequência é mostrar como os itens estáticos da solução irão interagir entre si ao longo do tempo

O diagrama de estados tem como base as enumerações do diagrama de classes, quando é feito novos métodos e até estados começam a aparecer, estes devem ser incorporados ao diagrama de classes – é um processo iterativo

O Diagrama de classes de projeto auxilia na documentação do que foi levantado sobre o problema, é como um dicionário visual mostrando os dados manipulados e como se relacionam.

Não deve mostrar **como** o problema será resolvido mas sim **qual** o problema.

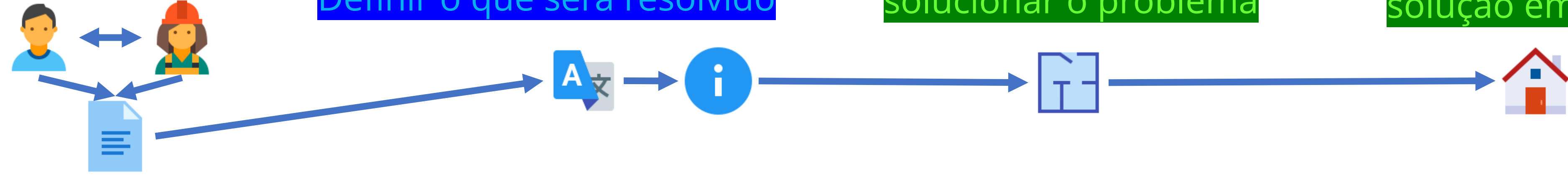
É de grande auxílio na limitação do escopo – separar o que é o ambiente analisado e o que será resolvido com software

**Mapeamento objeto-relacional:** um diagrama de casos de uso pode ser diretamente transformado em um MER, basta saber quais são as entidades e quais dados serão armazenados

Em Java o framework *Hibernate* usa a estrutura de código e anotações para criar um banco de dados relacional automaticamente

O mesmo acontece em C# com o *Entity Framework*

O que será resolvido? Entender o problema Definir o que será resolvido Como software pode solucionar o problema Colocar o plano de solução em prática



LADO DO PROBLEMA

[PLANO DE SOLUÇÃO]

LADO DA SOLUÇÃO