



INSTITUTO FEDERAL
PARANÁ

Engenharia de Software

PROF. AYSLAN T. POSSEBOM

Roteiro

Introdução

Princípios para um bom projeto de interfaces

Dica: início do projeto da interface

Dica: Modelagem de interfaces

- Exemplo: Reserva de um quarto em hotel

Dica: Modelagem de navegação entre telas

Ambiente de desenvolvimento de interfaces

Introdução

Além das atividades normais de desenvolvimento, existem outras atividades importantes para a produção de um software de sucesso:

- **Interface Homem-Máquina (IHM)**: desenvolvimento de interfaces com o usuário
- Produzir interfaces (telas dos programas) amigável com o usuário é um desafio!
- A interface permite a comunicação do usuário com o sistema
- O projeto de interfaces deve levar em conta as necessidades, experiências e capacidades dos usuários de sistemas
- Projetistas de interfaces devem conhecer as limitações físicas e mentais das pessoas e reconhecer que pessoas cometem erros.

Aspectos importantes a serem considerados no projeto de interfaces para o software:

- **Tempo de resposta do sistema**
 - Função da duração e da variabilidade. Duração pode ser longa ou curta. Variabilidade é a média do tempo de resposta.
- **Facilidade de ajuda (help online)**
 - Usuário pode consultar e tirar dúvidas sobre as interfaces sem precisar desistir de usar o software
- **Indicações de mensagens de erro**
 - Mensagens de fácil entendimento. Oferecer conselhos. Aconselhado vir junto com sinais sonoros ou visual
- **Sequência bem definida de comandos**
 - Seguir um “passo a passo” bem definido para que as ações sejam executadas

Projeto de interfaces exige **conhecimentos de fatores humanos e tecnologia de interfaces**

- As interfaces devem atender às habilidades, experiências e expectativas dos usuários
- Devem tornar o sistema simples de ser utilizado
- Devem ter um formato “limpo” onde os usuários consigam encontrar facilmente as funcionalidades que precisam e executá-las

Princípios para um bom projeto de interfaces

Familiaridade de usuário:

- Conter termos e conceitos de “Orientação a Objetos” e não de conceitos de computador
- Ex: Sistema de escritório deve usar conceitos como “cartas”, “documentos”, “folhetos”. Não deve usar “diretórios” ou “identificadores de arquivos”

Consistência de interface:

- A interface deve ser consistente, apresentando um mesmo padrão nos menus e nas funcionalidades
- Ex: Comandos e Menus devem ter o mesmo formato, pontuações de comandos devem ser similares, etc.

Surpresa mínima:

- Ao executar uma ação, o computador deve executar instruções que correspondam às expectativas do usuário em relação a esta ação
- Ex: Se um comando opera em modo conhecido, o usuário deve ser capaz de prever a operação de comandos similares.

Facilidade de recuperação:

- Sistema deve fornecer alguma resistência a erros dos usuários e permitir que eles sejam corrigidos
 - Ex: recurso “refazer”, confirmação de ações destrutivas, deletes, etc.

Guia de usuário:

- Devem fornecer “help”, manuais online, etc.

Diversidade de usuário:

- Sistemas devem apoiar diferentes tipos de usuários
 - Ex: usuários podem ter problemas de visão, então textos maiores podem estar disponíveis quando configurados.

Dica: início do projeto da interface

Desenvolver modelos que representem fielmente a interação que será realizada por um usuário

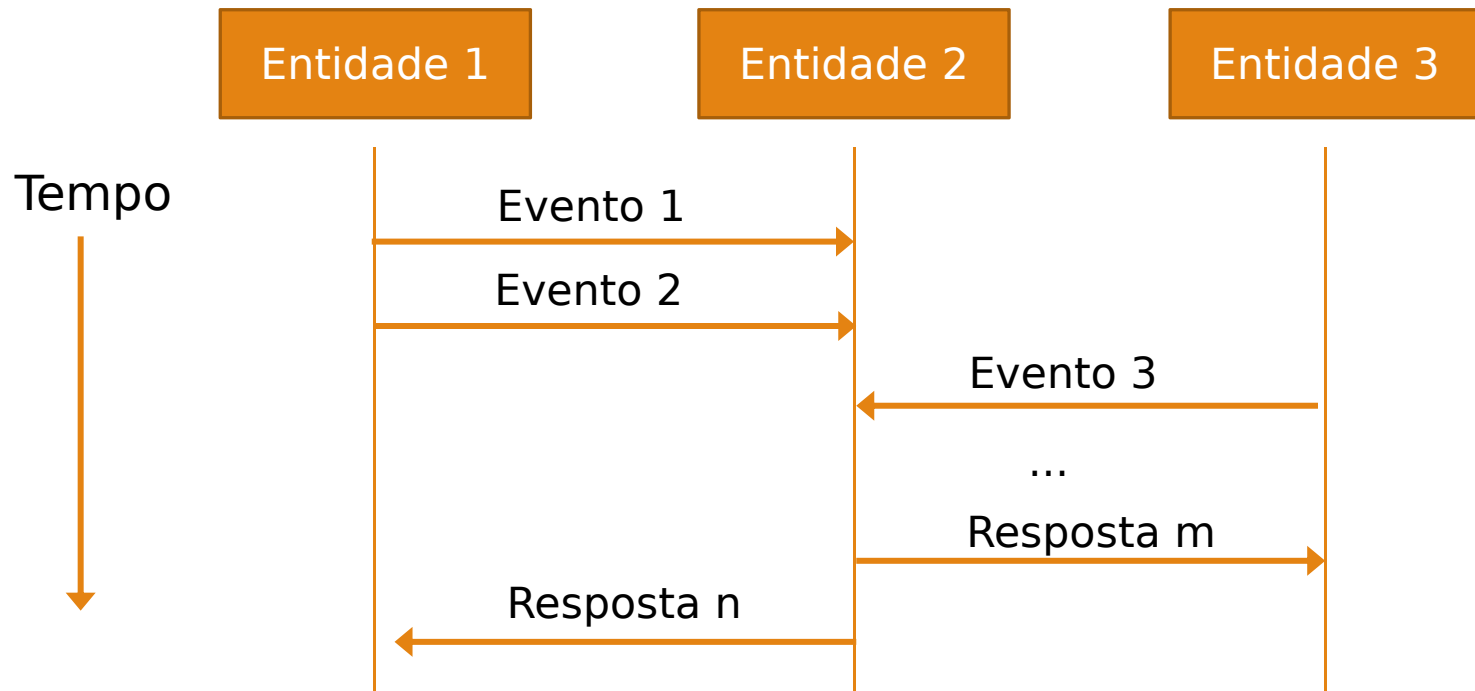
- Modelos devem incluir:
 - **perfil do usuário final do sistema**, sendo usuários **principiantes** (os que não conhecem informática), **instruídos e intermitentes** (os que usam informática esporadicamente, mas dominam o assunto) e os **instruídos e frequentes** (os que usam frequentemente e dominam o assunto)
 - **tarefas do usuário**, sendo as **tarefas que cada pessoa pode realizar** no sistema e como estas tarefas se relacionam umas com as outras

Dica:

Modelagem de interfaces

Pode ser feita por meio de cenários:

- Cenário é uma sequência de eventos/respostas que fluem entre as entidades do ambiente e o sistema, de acordo com uma finalidade
- Exemplo: modelo de interface através de um cenário:



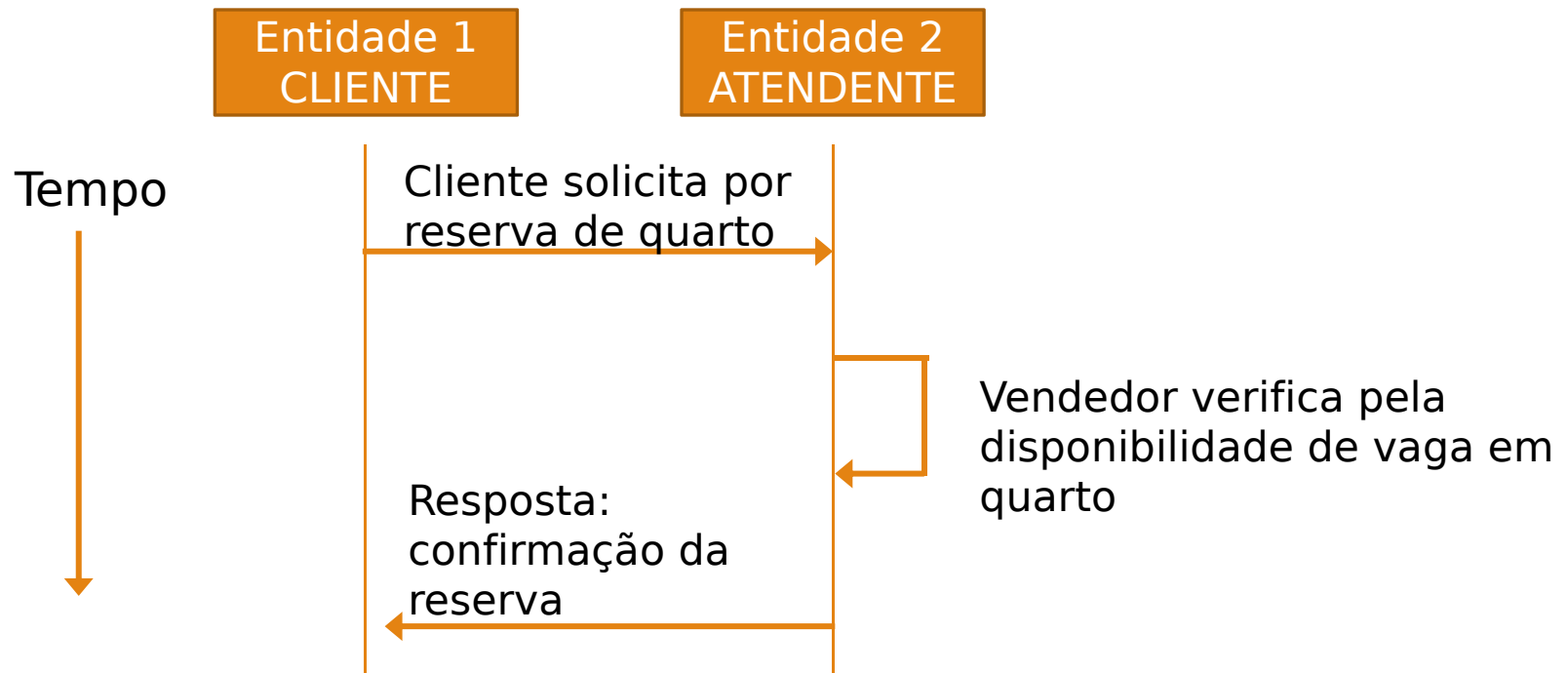
Exemplo: Reserva de um quarto em hotel

Entidade 1: Cliente

Entidade 2: Atendente

Sistema: Gerenciamento de Hotel

Eventos: Pedido, verificação de vaga, confirmação de reserva

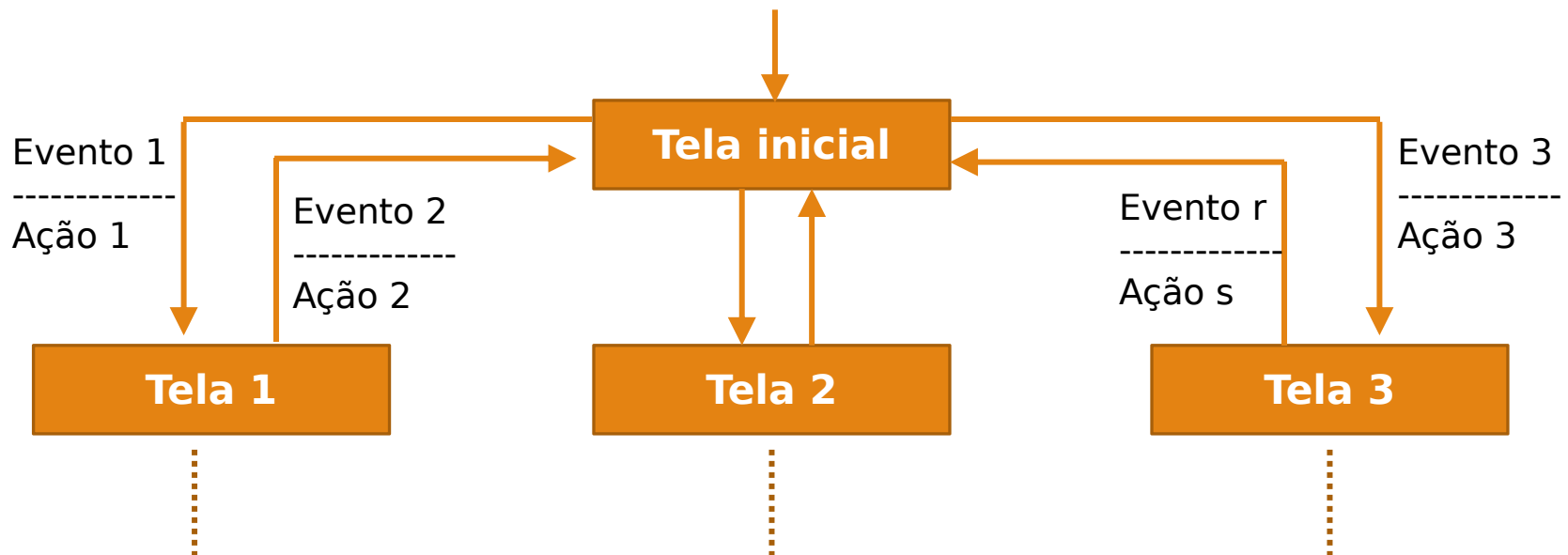


Dica:

Modelagem de navegação entre telas

Modelo de navegação (“diagrama de transição de telas”) é baseado em um diagrama de transição de estados

- Cada estado é representado pelo tipo de tela
- Transições representam eventos ou ações que ativam as funções do software



Ambiente de desenvolvimento de interfaces

Ambientes oferecem ferramentas que facilitam a criação de “janelas”, “interação de dispositivos”, “mensagens de erros”, “comandos” e muitos outros elementos de um ambiente interativo.

- Exemplo: Delphi, Visual Basic, Eclipse, Netbeans, Qt, etc.

Existem normas que podem orientar a adoção de melhores práticas de projeto de interface:

- Ex: **NBR 9241-11** (define usabilidade e explica como identificar a informação necessária a ser considerada na especificação e avaliação de usabilidade de um computador considerando “medidas de desempenho” e “satisfação do usuário”)
- **Usabilidade**: medida em que um produto pode ser usado por usuários definidos para atingir seus objetivos com eficácia, eficiência e satisfação

QUESTIONÁRIO

Descreva com suas palavras o que é “usabilidade”

Baseando-se em algum sistema que você usa regularmente, os princípios de projeto de interface com o usuário descritos foram seguidos? (apresente o sistema que você usa e sua opinião)

As tarefas dos usuários são normalmente consideradas nos seus projetos de interface ao resolver os exercícios de algoritmos? de que forma são analisados?

Elabore um modelo de cenário para “terminais de caixa-eletrônico”

Escolha uma loja de comércio eletrônico e elabore um modelo de navegação de suas páginas.

Usando o Netbeans

Como criar um formulário (janela de sistema)?

Como inserir componentes de interface nesta tela?

- Insira:

- Botões
- Botões de alternar
- Caixas de seleção
- Botões de radio (precisa de um Grupo de Botões)
- Caixa de combinação
- Listar
- Campo de texto
- Área de texto
- Barra de rolagem
- Campo deslizante
- Campo de senha
- Menus e submenus

Bibliografia

HIRAMA, Kechi. **Engenharia de Software: qualidade e produtividade com tecnologia.** Editora Campus, 2011. Páginas: 99-105.