

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

TOMAS MATTIA

**Software de Suporte para Empresas de  
Representação Comercial Focado em  
Mobilidade e Integração**

Trabalho de Graduação.

Prof. Dr. Marcelo Soares Pimenta  
Orientador

Porto Alegre, dezembro de 2011

# SUMÁRIO

<b>LISTA DE ABREVIATURAS E SIGLAS</b> . . . . .	3
<b>LISTA DE FIGURAS</b> . . . . .	4
<b>RESUMO</b> . . . . .	5
<b>ABSTRACT</b> . . . . .	6
<b>1 INTRODUÇÃO</b> . . . . .	7
1.1 Contextualização . . . . .	7
1.2 Motivação . . . . .	8
1.3 Objetivo . . . . .	8
1.4 Estrutura do Trabalho . . . . .	8
<b>2 ESTUDO DE CASO</b> . . . . .	9
2.1 Modo de Operação Atual . . . . .	9
2.1.1 Principais Dificuldades . . . . .	10
2.2 Modo de Operação Proposto . . . . .	11
<b>3 FUNDAMENTOS E CONCEITOS BÁSICOS</b> . . . . .	12
3.1 <i>Responsive Web Design e Mobile First</i> . . . . .	12
3.2 RESTful Web Services . . . . .	13
3.3 Modelagem Ágil . . . . .	14
<b>4 DESENVOLVIMENTO</b> . . . . .	15
4.1 Exploração . . . . .	15
4.1.1 Modelagem Inicial dos Requisitos . . . . .	15
4.1.2 Modelagem Inicial de Arquitetura . . . . .	17
4.2 Planejamento . . . . .	19
4.3 Iterações . . . . .	21
4.3.1 Modelagem . . . . .	21
4.3.2 Programação e Teste . . . . .	23
4.3.3 Integração . . . . .	25
<b>5 PROTÓTIPO</b> . . . . .	27
5.1 Mobilidade . . . . .	27
5.2 Integração . . . . .	32
5.3 Funcionamento Passo-a-Passo . . . . .	33
5.3.1 Cadastro de Pedidos . . . . .	34

5.3.2	Mural	41
<b>6</b>	<b>CONCLUSÃO</b>	<b>43</b>
	<b>REFERÊNCIAS</b>	<b>44</b>
	<b>APÊNDICE A TESTES DE ACEITAÇÃO DE HISTÓRIAS</b>	<b>46</b>
	<b>APÊNDICE B RESTFUL WEB SERVICES IMPLEMENTADOS</b>	<b>61</b>
<b>B.1</b>	<b>Clientes</b>	<b>61</b>
<b>B.2</b>	<b>Pedidos</b>	<b>62</b>
B.2.1	Itens de Pedido	63
<b>B.3</b>	<b>Produtos</b>	<b>64</b>
<b>B.4</b>	<b>Representadas</b>	<b>65</b>
<b>B.5</b>	<b>Transportadoras</b>	<b>66</b>

## **LISTA DE ABREVIATURAS E SIGLAS**

BDD	Behaviour Driven Development
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
MA	Modelagem Ágil
REST	REpresentational State Transfer
URI	Uniform Resource Identifier
XML	eXtensible Markup Language
XP	Programação eXtrema

## LISTA DE FIGURAS

Figura 2.1:	Exemplo de tela do software de gestão utilizado atualmente pelo cliente-alvo . . . . .	10
Figura 4.1:	Versão inicial do modelo de dados conceitual . . . . .	16
Figura 4.2:	Modelo inicial da interface do usuário para dispositivos com telas grandes . . . . .	16
Figura 4.3:	Modelo inicial da interface do usuário para dispositivos com telas pequenas . . . . .	17
Figura 4.4:	Modelo inicial da arquitetura do sistema . . . . .	18
Figura 4.5:	Mapa de histórias . . . . .	20
Figura 4.6:	Exemplo de protótipo de tela desenhado em papel para uma história .	21
Figura 4.7:	Exemplo de diagrama de estados para uma história . . . . .	22
Figura 4.8:	Modelo do esquema físico do banco de dados ao final da quarta iteração	23
Figura 4.9:	Exemplo de um cenário de teste de aceitação de uma história . . . . .	24
Figura 4.10:	Exemplo de execução de cenário não implementado ou com falhas . .	24
Figura 4.11:	Exemplo de definição de um passo de cenário . . . . .	24
Figura 4.12:	Exemplo de execução de cenário implementado corretamente . . . .	25
Figura 4.13:	Exemplo de execução da suíte de testes completa . . . . .	25
Figura 5.1:	Exemplo de visualização em dispositivo com tela de 1024px de largura	27
Figura 5.2:	Exemplo de visualização em dispositivo com tela de 768px de largura	28
Figura 5.3:	Exemplo de visualização em dispositivo com tela de 480px de largura	29
Figura 5.4:	Exemplo de visualização em dispositivo com tela de 320px de largura	30
Figura 5.5:	Menu de navegação primária expandido em dispositivos com telas pequenas . . . . .	31
Figura 5.6:	Exemplo de visualização sem renderização de estilos . . . . .	32
Figura 5.7:	Exemplo de resposta de requisição <i>GET</i> sobre uma coleção de registros	33
Figura 5.8:	Exemplo de resposta de requisição <i>GET</i> sobre um único registro . . .	33
Figura 5.9:	Cadastro de Pedido - Tela Inicial . . . . .	34
Figura 5.10:	Cadastro de Pedido - Adicionar Item . . . . .	35
Figura 5.11:	Cadastro de Pedido - Adicionar Item - Mensagens de Erro . . . . .	36
Figura 5.12:	Cadastro de Pedido - Rascunho . . . . .	37
Figura 5.13:	Cadastro de Pedido - Informações de Pagamento, Comissão e Frete .	38
Figura 5.14:	Cadastro de Pedido - Aguardando Confirmação . . . . .	39
Figura 5.15:	Lista de pedidos organizados por data . . . . .	40
Figura 5.16:	Menu para filtrar pedidos por situação . . . . .	40
Figura 5.17:	Página Inicial - Mural . . . . .	41

## RESUMO

Este trabalho descreve a concepção e o desenvolvimento de um sistema que visa auxiliar no gerenciamento das informações de empresas de representação comercial. Uma grande dificuldade enfrentada por empresas de representação comercial é a diversidade e falta de compatibilidade dos meios de armazenamento e transferência de informação utilizados pelas partes envolvidas. Além disso, boa parte dos acessos a essas informações são feitos longe da sede da empresa, o que torna a mobilidade um fator fundamental para o sucesso do negócio.

O objetivo deste trabalho é aprimorar a gerência de informação de empresas de representação comercial, através do desenvolvimento de um projeto ágil de software junto a uma empresa real, utilizada como cliente-alvo. O sistema desenvolvido é focado em duas características principais, as quais não são atendidas pelas soluções utilizadas atualmente: integração e mobilidade.

Inicialmente, é descrita a situação atual da empresa utilizada como cliente-alvo, apresentando os principais papéis e atividades envolvidas, suas deficiências e sugestões para melhorias. A seguir, são discutidos os requisitos para um sistema que visa auxiliar nessas atividades. Também são apresentados os conceitos básicos das tecnologias empregadas no desenvolvimento deste trabalho. Na descrição da solução proposta, são apresentadas todas as etapas do projeto, desde o levantamento dos requisitos até a implementação, passando pela análise e os modelos que foram úteis durante o projeto. Por fim, uma descrição de funcionamento passo a passo das principais funcionalidades do sistema é apresentada.

**Palavras-chave:** Modelagem ágil, mobilidade, integração.

## **Support Software for Sales Representative Companies Focused on Mobility and Integration**

### **ABSTRACT**

This paper describes the design and development of a system designed to help manage the information of sales representative companies. A major difficulty faced by sales representative companies is the diversity and lack of compatibility of the tools used to store and transmit information between the business participants. Moreover, most of the accesses to the information are made away from the company's office, which makes mobility a key factor for the business success.

The objective of this work is to improve the management of information of sales representative companies, through the development of an agile software project for a real company, used as a target customer. The system developed is focused on two main features which are not present in the solutions used today by the company: integration and mobility.

Initially, we describe the current situation of the company used as a target customer, presenting the main roles and activities involved, their shortcomings and suggestions for improvements. Then we discuss the requirements for a system that aims to assist in these activities. Basic concepts of the technologies used in developing this project are also outlined. The description of the proposed solution states all stages of the project, from requirements elicitation to implementation, through the analysis and the models that were useful during the project. Finally, a step-by-step demonstration of the main features of the system is presented.

**Keywords:** agile modeling, mobile, integration.

# 1 INTRODUÇÃO

Este capítulo apresenta o negócio de representação comercial através das principais atividades e cargos envolvidos, seguido da motivação, o objetivo e a estrutura geral deste trabalho.

## 1.1 Contextualização

O representante comercial atua como facilitador de relacionamentos entre as empresas para as quais trabalha, chamadas de representadas, e seus clientes. As representadas são produtoras e/ou fornecedoras de um ou mais produtos, enquanto os clientes são estabelecimentos comerciais que negociam o produto da representada com o consumidor final. São tarefas do representante comercial:

- Promover os produtos da representada, através de visitas regulares aos clientes;
- Negociar propostas e pedidos;
- Acompanhar o processo de venda, controlando questões como prazos de entrega, quantidade e qualidade das mercadorias;
- Intermediar as chamadas de suporte para os produtos vendidos, como instruções de instalação e uso e assistências técnicas;
- Auxiliar na cobrança de valores devidos;
- Apresentar relatórios de vendas para a representada.

Geralmente, o representante conta com o apoio de uma ou mais pessoas, constituindo uma empresa de representação comercial. Os principais cargos de uma empresa de representação comercial são:

- *Representante*: gerencia o negócio como um todo, tomando as decisões mais importantes da empresa. É o representante quem define quais são as representadas e a região de atuação da empresa, negocia comissões dos vendedores e condições especiais para os clientes, etc.;
- *Secretária*: executa tarefas administrativas, como gerenciamento de cadastros e atendimento de telefonemas. Trabalha no escritório da empresa;
- *Vendedores*: responsáveis pelas atividades de promoção e venda dos produtos. Devido às necessidades de comunicação e bom relacionamento, estão frequentemente viajando para visitar pessoalmente antigos e potenciais clientes.



A separação entre os cargos não costuma ser rígida. Dependendo da situação, é comum que uma mesma pessoa assuma mais de uma função. Por exemplo: o representante também visita clientes, como um vendedor, e também trabalha no escritório da empresa, realizando tarefas em conjunto com a secretária. Por sua vez, a secretária também ajuda nas atividades de promoção e venda, inclusive podendo receber comissões por isso.

## **1.2 Motivação**

Uma grande dificuldade enfrentada por empresas de representação comercial é a diversidade e falta de compatibilidade dos meios de armazenamento e comunicação de informação utilizados pelas partes envolvidas. Por estarem frequentemente viajando, os funcionários precisam de acesso remoto aos dados da representada. Esse acesso nem sempre é possível, já que os documentos geralmente ficam armazenados localmente em um único computador no escritório da empresa.

## **1.3 Objetivo**

O objetivo deste trabalho é descrever a concepção e o desenvolvimento de um sistema que visa auxiliar no gerenciamento das informações de empresas de representação comercial. O sistema foi realizado através da adoção de princípios ágeis de desenvolvimento de software junto a uma empresa real, utilizada como cliente-alvo. O sistema desenvolvido é focado em duas características principais que não são atendidas pelas soluções utilizadas atualmente pelo cliente-alvo: integração e mobilidade.

## **1.4 Estrutura do Trabalho**

Este trabalho está estruturado da seguinte forma: o capítulo 2 descreve a situação atual da empresa utilizada como cliente-alvo, apontando as principais dificuldades de comunicação enfrentadas e discutindo os requisitos para um sistema que visa auxiliar nessas dificuldades. O capítulo 3 apresenta os fundamentos e conceitos básicos aplicados no desenvolvimento deste trabalho. O capítulo 4 apresenta a solução proposta, descrevendo todas as etapas do projeto, desde o levantamento de requisitos até a implementação, passando pela análise e os principais modelos utilizados. Por fim, no capítulo 5, uma descrição passo-a-passo das principais funcionalidades do sistema é apresentada.

## 2 ESTUDO DE CASO

Este capítulo descreve a situação atual da empresa utilizada como cliente-alvo, apontando as principais dificuldades de comunicação enfrentadas e discutindo os requisitos para um sistema que visa auxiliar nessas dificuldades.

### 2.1 Modo de Operação Atual

Este trabalho foi realizado no contexto de uma empresa de representação comercial real, a qual foi utilizada como fonte de informações e cliente-alvo do projeto. Entretanto, a maioria das características aqui apresentadas também são encontradas em grande parte das empresas de representação comercial.

A empresa é composta por apenas dois funcionários: um representante e uma secretária. O representante é o responsável pelas viagens e contato pessoal com os clientes, enquanto que a secretária fica no escritório da empresa. A empresa trabalha principalmente com o setor moveleiro, sendo que sua área de atuação envolve boa parte da região norte do estado do Rio Grande do Sul.

A empresa conta com aproximadamente 200 clientes e 10 representadas de diferentes linhas de produtos, desde cozinhas até estofados e salas de jantar. Uma única representada chega a ter mais de 30 modelos de produtos, cada qual com diferentes combinações de características, como tamanho e cor.

Os principais meios de armazenamento de dados e comunicação utilizados atualmente são:

- Anotações em papel, transmitidas por telefone ou FAX;
- Documentos de texto ou planilhas eletrônicas, transferidos por email ou programas de mensagens instantâneas;
- Arquivos exportados do sistema de gestão da empresa, compartilhados através de softwares de armazenamento online.

O software de gestão utilizado pela empresa foi desenvolvido há mais de 10 anos, e executa somente sobre o sistema operacional Microsoft Disk Operating System (WIKIPEDIA, 2011). O único meio de comunicação com sistemas externos do software atual é através da exportação e importação de arquivos de texto. A Figura 2.1 apresenta a tela de cadastro de pedidos do software atual.



Figura 2.1: Exemplo de tela do software de gestão utilizado atualmente pelo cliente-alvo

Recentemente, o representante da empresa adquiriu um tablet com acesso à internet para manter-se conectado com a empresa em suas viagens. Porém, para poder utilizar o software de gestão da empresa, o representante ainda necessita de um computador portátil pessoal, o qual também é levado nas viagens.

### 2.1.1 Principais Dificuldades

As principais dificuldades encontradas pela empresa são o isolamento do atual sistema de gestão e a grande quantidade de tarefas repetitivas que consomem mais tempo do que deveriam.

Por exemplo, ao receber um novo pedido, seja por telefone, FAX ou email, a secretária deve:

1. Digitar o pedido no software da empresa;
2. Enviar o pedido para o cliente e a representada, de duas maneiras:
  - Exportar o pedido como arquivo de texto e enviar o mesmo por email; ou
  - Digitar o pedido novamente, desta vez diretamente no software de gestão da própria representada.
3. Aguardar confirmação de recebimento do pedido pela representada (novamente, ou por email ou diretamente pelo software da representada);
4. Anotar a confirmação do recebimento do pedido pela representada. As anotações são feitas em papel, pois o software atual não suporta esta funcionalidade.

Esse processo é aceitável para poucos pedidos, mas quando o volume de trabalho aumenta ele se torna repetitivo, lento e sujeito a erros.

Além disso, uma série de tarefas que poderiam ser automatizadas são feitas de maneira manual, por exemplo:

- Instalação do software de gestão em novos computadores;

- Sincronização dos dados entre os computadores que possuem o software de gestão instalado;
- Backups diários dos dados da empresa; etc.

## **2.2 Modo de Operação Proposto**

Através de automação e reorganização de tarefas, pretendemos agilizar o dia-a-dia da empresa, reduzindo custos com processos ineficientes e sujeitos a erros. Para isso, focamos em dois pontos principais: mobilidade e integração.

Devido à grande diversidade de plataformas utilizadas no meio corporativo, o software desenvolvido deve ser o mais acessível possível. Especificamente, três contextos de uso são prioritários: computadores de mesa, tablets e smartphones, independentemente de fabricante de hardware ou sistema operacional.

Também é necessário que o software forneça uma interface de comunicação padronizada para que outros sistemas sejam capazes de acessar e receber dados da empresa. Essa interface deve ser amplamente suportada pelos principais ambientes de desenvolvimento utilizados em aplicações modernas de gestão.

## 3 FUNDAMENTOS E CONCEITOS BÁSICOS

Este capítulo resume os fundamentos e conceitos básicos necessários para o entendimento deste trabalho: *Responsive Web Design* e *Mobile First*, focados nas questões de mobilidade, e *RESTful Web Services* para a parte de integração. Também é apresentada brevemente a Modelagem Ágil, utilizada ao longo do desenvolvimento do projeto de *software* junto do cliente-alvo.

### 3.1 *Responsive Web Design* e *Mobile First*

Com o surgimento de novos dispositivos com acesso à internet, a maioria dos websites, antes projetados para serem usados em computadores de mesa, passou a sofrer com problemas de usabilidade quando acessados em dispositivos com telas menores (NIELSEN, 2009).

*Responsive Web Design* (MARCOTTE, 2011) é um termo criado recentemente para descrever a aplicação de uma série de técnicas de programação de interfaces web a fim de obter uma interface que se adapta automaticamente a diferentes tamanhos de tela. As três ferramentas agrupadas sob este termo são:

- Um leiaute baseado em uma grade flexível;
- Imagens e mídias flexíveis; e
- *Media queries*, um módulo da especificação CSS3 (ETEMAD, 2010).

Porém, não basta que a interface seja adaptável. Para que bons índices de usabilidade sejam alcançados, a interface deve ser projetada especificamente para dispositivos móveis, por diversos motivos:

- Dificilmente todo o conteúdo de uma interface convencional cabe no pouco espaço disponível nas telas dos dispositivos móveis (NIELSEN, 2011a), mesmo quando adaptado. Interfaces para dispositivos móveis devem ser focadas na execução de poucas e simplificadas tarefas;
- O contexto de uso é diferente. Quando utilizando um dispositivo móvel, muitas vezes o usuário está em movimento ou dedicando apenas parte de sua atenção para a tarefa que está tentando realizar;
- O modo de interação é diferente. Com computadores de mesa, os principais meios de interação são teclado e mouse. Já nos dispositivos móveis, existe uma grande variedade de meios, como teclados numéricos, teclados QWERTY e telas sensíveis ao toque.

O conceito de *Mobile First* (WROBLEWSKI, 2011) lida diretamente com essas questões. O principal objetivo é planejar e desenvolver um sistema pensando primeiramente em como será o seu uso em dispositivos móveis. Há três motivações principais para se fazer isso:

- Prepara-se para lidar com o grande crescimento do mercado de dispositivos móveis nos últimos anos;
- As diversas restrições impostas pelos dispositivos móveis são vistas como vantagens que restringem o escopo do sistema em desenvolvimento, evitando a adição de funcionalidades desnecessárias e focando somente naquilo que realmente agrega valor;
- Apesar das limitações, dispositivos móveis também possuem uma série de novas capacidades que não são comumente encontradas em computadores de mesa, como ferramentas de geolocalização.

Porém, uma série de cuidados devem ser tomados ao desenvolver pensando primeiramente em dispositivos móveis. Alguns pontos principais merecem destaque:

- Deve-se entender qual o principal contexto de uso do software: procurar e encontrar informações, explorar ou passar o tempo, criar ou editar registros, etc.;
- O conteúdo deve ser exibido como a informação mais importante, e a navegação como secundária. A navegação deve ser relevante e bem posicionada para que o usuário possa aprofundar-se em uma determinada seção ou pular diretamente para outra seção com igual facilidade;
- Para dispositivos com telas sensíveis ao toque, deve-se dimensionar e posicionar os elementos de forma organizada, a fim de evitar ativações acidentais causadas por “dedos-gordos” (NIELSEN, 2011b);
- Facilitar a entrada de dados através da utilização de campos especiais e formatação automática de valores.

### 3.2 RESTful Web Services

Por ser um sistema distribuído, uma maneira simples e eficiente de transferir dados entre as partes envolvidas é necessária. Os *RESTful Web Services* têm uma grande aceitação pelas suas características como simplicidade, escalabilidade e versatilidade herdadas do estilo de arquitetura *REpresentational State Transfer* (REST), muitas vezes considerado como uma filosofia de design.

O estilo REST busca alinhar o desenvolvimento de aplicações com a arquitetura da Web (FIELDING, 2000). É descrito como uma rede de *websites* (um estado virtual), onde o usuário progride com uma aplicação selecionando as ligações (transições de estado), tendo como resultado a página seguinte (que representa o estado seguinte). O estilo REST aplica algumas restrições sobre os elementos da arquitetura, como:

- Separação de conceitos, interface e responsabilidades entre cliente e servidor;

- Sem estado: não é permitido estado de sessão entre cliente e servidor. Toda requisição do cliente deve conter todos os dados necessários para ser entendida pelo servidor;
- Cache: o resultado de uma requisição individual pode ser armazenado para uso posterior; e
- Interface uniforme: a interface de acesso entre os componentes deve ser uniforme.

Os *RESTful Web Services* são implementados usando os padrões que fizeram da Web o maior sistema distribuído existente. Eles utilizam-se da definição dos métodos de acesso e códigos de status do padrão HTTP para trabalhar sobre recursos. Um recurso é qualquer coisa importante o suficiente para ser referenciado por si só. Todo recurso deve ter um endereço único para identificá-lo, ou URI. Um exemplo de endereço para um pedido pode ser *http://www.exemplo.com/pedidos/34*, onde 34 é o código identificador do pedido. Clientes HTTP manipulam os recursos através do envio de mensagens específicas para a URI do recurso, como GET (para ler um recurso), PUT (para atualizar um recurso), POST (para criar um novo recurso) e DELETE (para excluir um recurso).

*RESTful Web Services* são um padrão de fato, mas não são um padrão definido por algum organismo. Há, no entanto, um consenso sobre diversas práticas que se tornaram modelos por se apresentarem adequadas à filosofia que acompanha a arquitetura REST. Entre estas práticas, incluem-se: o uso correto dos métodos HTTP (também chamado de verbos); a semântica adequada do endereçamento dos recursos; e a manutenção de um padrão homogêneo, público e bem documentado, em toda aplicação, em todos recursos, em todas as situações.

Mais detalhes sobre REST podem ser encontrados na literatura: (FIELDING, 2000), (RICHARDSON; RUBY, 2000), (HE, 2004) e (EDIGER, 2008).

### 3.3 Modelagem Ágil

A Modelagem Ágil (AMBLER, 2002a) é um processo baseado na prática para modelagem e documentação eficazes de sistemas baseados em *software*. A metodologia MA é um conjunto de práticas guiado por princípios e valores para profissionais de *software* aplicarem em seu dia a dia. A MA não é um processo prescritivo, ou seja, ela não define procedimentos detalhados sobre como criar um determinado tipo de modelo. Em vez disso, fornece conselhos sobre como utilizar modelos para aprimorar a comunicação entre as pessoas envolvidas em projetos de *software*.

Os principais valores da modelagem ágil são: comunicação, simplicidade, retorno, coragem e humildade. A esses valores, é somada uma série de princípios derivados da Aliança Ágil (AGILE ALLIANCE, 2011), como a importância de adotar simplicidade na criação de modelos e encampar a mudança para refletir a frequente variação dos requisitos com o tempo.

Para modelar de maneira ágil, utiliza-se uma série de práticas da MA, como a criação de diversos modelos em paralelo e a participação ativa dos clientes. A MA deve ser utilizada juntamente com outras metodologias, como a Programação eXtrema (XP) (BECK, 1999). Maiores detalhes sobre a modelagem ágil podem ser encontrados na literatura: (AMBLER, 2002a) e (AMBLER, 2011).

## 4 DESENVOLVIMENTO

O projeto foi desenvolvido em um contexto ágil, utilizando uma empresa real como cliente-alvo. Apesar de não ter sido seguida nenhuma metodologia específica, procurou-se trabalhar em um processo adaptativo, baseado principalmente nos princípios da modelagem ágil (AMBLER, 2002a) e práticas de desenvolvimento da XP (BECK, 1999). Neste capítulo, apresentamos as principais etapas do projeto: exploração, planejamento e iterações.

### 4.1 Exploração

A fase de exploração fornece uma base para o desenvolvimento do projeto, através de duas atividades principais: a modelagem inicial dos requisitos e a modelagem inicial da arquitetura. O principal objetivo é ter uma base de requisitos para desenvolver uma boa primeira versão do software.

#### 4.1.1 Modelagem Inicial dos Requisitos

Durante a modelagem inicial dos requisitos, foram desenvolvidos três modelos: modelo de uso, modelo de dados conceitual e um modelo para a interface do usuário.

##### 4.1.1.1 *Modelo de Uso*

O primeiro passo na definição dos requisitos foi visitar pessoalmente os locais de trabalho do cliente-alvo e observá-lo em ação (NIELSEN, 1993). As visitas foram feitas tanto no escritório da empresa quanto em viagens junto a um vendedor. Em seguida, foi feita uma análise das principais funcionalidades do software de gestão utilizado pela empresa.

A partir deste contato inicial, foi feita uma entrevista informal com o representante. O resultado desta entrevista foi um modelo de uso para o sistema (RIEDEMANN; FREITAG, 2009), representado no formato de histórias de usuário (COHN, 2004). Abaixo, apresentamos três exemplos de histórias escritas:

- Secretária cadastra produto;
- Vendedor inicia cadastro de pedido;
- Cliente confirma recebimento de assistência.

A lista completa das histórias de usuário escritas nesta etapa pode ser vista no mapa de histórias apresentado na figura 4.5.



#### 4.1.1.2 Modelo de Dados Conceitual

Após a elaboração do modelo de uso, foi desenvolvido um modelo de dados conceitual (AMBLER, 2002b). O modelo é bastante enxuto, e foi usado apenas para facilitar a visualização dos principais conceitos envolvidos no negócio e as relações entre eles. Por motivos de clareza, a Figura 4.1 apresenta o modelo na sua versão digitalizada. O modelo original, desenhado em papel, sofreu diversas alterações ao longo do desenvolvimento do projeto.

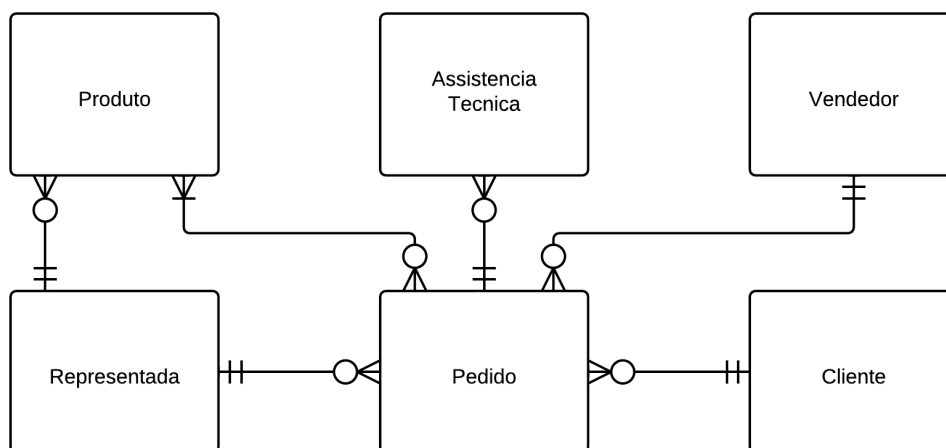


Figura 4.1: Versão inicial do modelo de dados conceitual

#### 4.1.1.3 Modelo da Interface do Usuário

Para o modelo inicial da interface do usuário, foram utilizados rascunhos em papel (SNYDER, 2003). Foram desenvolvidos dois modelos, com ênfase no posicionamento dos blocos de navegação e conteúdo do sistema (KALBACH, 2007). A Figura 4.2 apresenta o modelo inicial da interface do usuário para dispositivos com telas grandes, como *notebooks*, computadores de mesa ou *tablets* com telas de 10 polegadas ou mais.

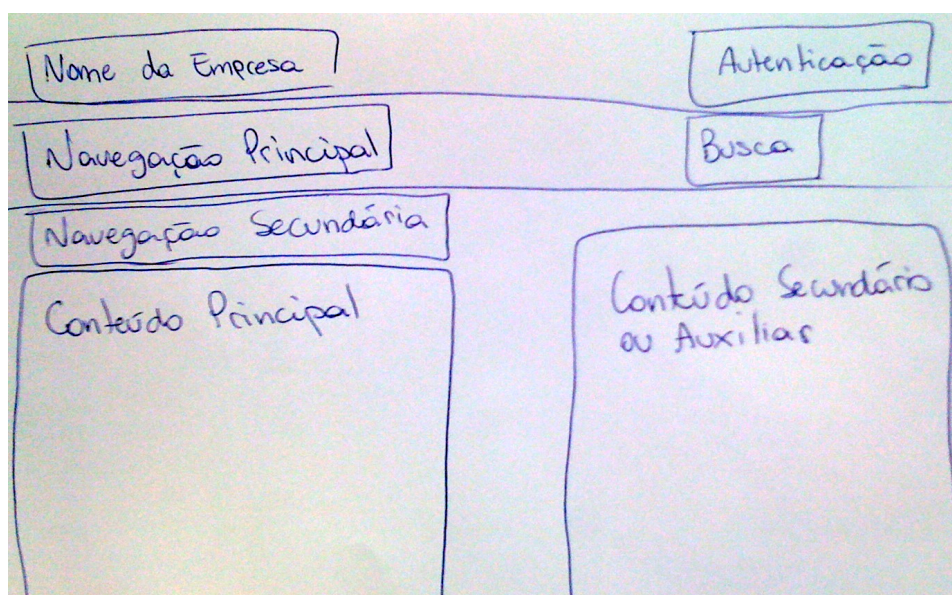


Figura 4.2: Modelo inicial da interface do usuário para dispositivos com telas grandes

A Figura 4.3 apresenta o modelo inicial da interface para dispositivos com telas pequenas, como *smartphones* ou *tablets* com telas de até 10 polegadas.

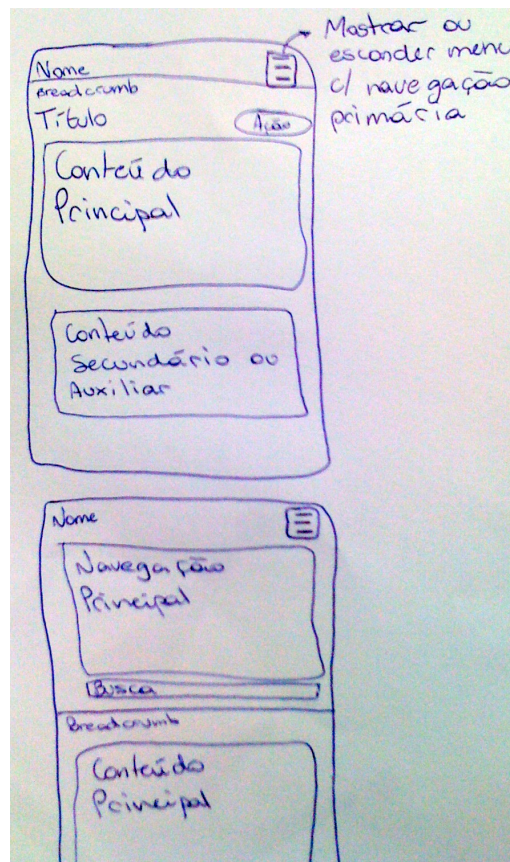


Figura 4.3: Modelo inicial da interface do usuário para dispositivos com telas pequenas

#### 4.1.2 Modelagem Inicial de Arquitetura

O objetivo da modelagem inicial de arquitetura é tentar identificar uma arquitetura que tenha uma boa chance de funcionar. Por ser um sistema distribuído, optou-se pela utilização de uma arquitetura cliente-servidor. A Figura 4.4 apresenta o modelo inicial de arquitetura para o sistema.

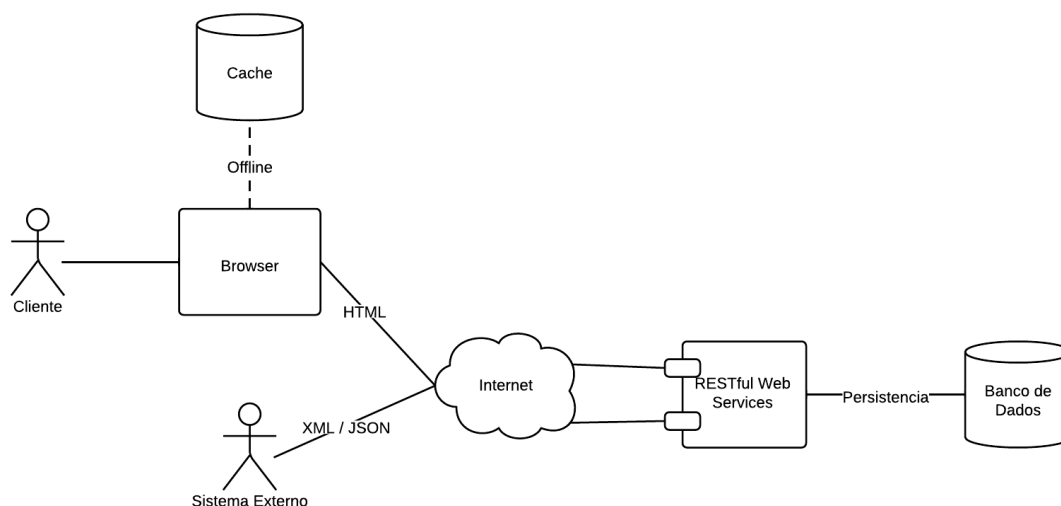


Figura 4.4: Modelo inicial da arquitetura do sistema

#### 4.1.2.1 Servidor

No lado do servidor, decidiu-se utilizar *RESTful Web Services* operando sobre um banco de dados relacional. Os motivos foram a facilidade de implementação e relativa simplicidade do protocolo quando comparado aos *Web Services* “clássicos”. (PAUTASSO; ZIMMERMANN; LEYMANN, 2008). Desta forma, separamos a implementação da lógica do negócio em uma série de serviços independentes, o que agiliza o desenvolvimento e a manutenção do sistema. Além disso, por utilizar tecnologias padronizadas, a integração com outros sistemas também é facilitada.

#### 4.1.2.2 Cliente

No lado do cliente, foram comparadas três abordagens, sempre buscando atender o requisito inicial de que o software deve rodar em três plataformas distintas: celulares, tablets e computadores de mesa. Para a comparação, foram desenvolvidas uma série de *Spike Solutions* (WELLS, 1999). *Spike Solutions* implementam uma funcionalidade mínima que exercite todos os elementos da arquitetura do sistema. No nosso caso, implementamos um simples aplicativo para gerenciamento de tarefas a fazer. As três abordagens estudadas foram:

- **Código nativo:** desenvolver um software para cada tipo de plataforma a ser suportada pelo sistema. A principal vantagem é ter acesso à todas funcionalidades da plataforma, além de ganhos notáveis de performance. Porém, existe um alto custo associado ao desenvolvimento e manutenção de várias bases de código. As duas plataformas exploradas foram: *Android* (GOOGLE, 2011) e *iOS* (APPLE, 2011).
- **Geradores de código:** neste caso, o software é desenvolvido sobre uma única plataforma, a qual compila o código original em código nativo para outras plataformas. A principal vantagem é ainda ter acesso à maioria das funcionalidades de baixo nível da plataforma, além de ser possível manter apenas uma base de código. No entanto, os geradores de código estudados suportavam apenas dispositivos móveis. A versão para computadores de mesa ainda teria que ser desenvolvida à parte. Os geradores explorados foram *Titanium* (APPCELERATOR, 2011) e *Rhodes* (RHOMOBILE, 2011).

- **Padrões Web:** desenvolvendo o software como uma página da internet, basta que a plataforma possua um navegador para acessar o software. As principais dificuldades são manter o software funcionando offline, e personalizar o software para os diferentes tamanhos de dispositivo. Ambos problemas podem ser resolvidos com as tecnologias emergentes apresentadas no Capítulo 3.

Por ser a solução com menor custo de desenvolvimento, e que suportaria o maior número de plataformas, optou-se por um cliente baseado em padrões Web.

## 4.2 Planejamento

O propósito da fase de planejamento é entrar em acordo com o cliente em uma data na qual o menor e melhor conjunto de histórias de usuários será implementado. (AMBLER, 2002a).

Na fase de planejamento, as histórias de usuário foram priorizadas por valor e frequência de uso. A fim de melhor visualizar os processos do negócio e as relações entre eles, as histórias foram dispostas em um Mapa de Histórias (PATTON, 2005), conforme apresentado na Figura 4.5.

Ao dispor as histórias em dois eixos (criticidade e sequência de uso), surge uma organização natural para as diferentes etapas envolvidas no negócio. As etapas são separadas por linhas tracejadas verticais e identificadas por um nome, localizado na parte superior do mapa.

Cada história é representada por um retângulo amarelo, contendo título (em negrito), ator principal, frequência de ocorrência e prioridade.

Os pequenos retângulos vermelhos dentro de cada história representam a frequência de ocorrência da história, sendo: um retângulo para semestral, dois para mensal, três para semanal, quatro para diária, e cinco retângulos para frequência horária.

Os pequenos retângulos azuis representam a criticidade (ou valor, ou prioridade) da história, sendo um retângulo para prioridade baixa, dois retângulos para prioridade média e três retângulos para prioridade alta.

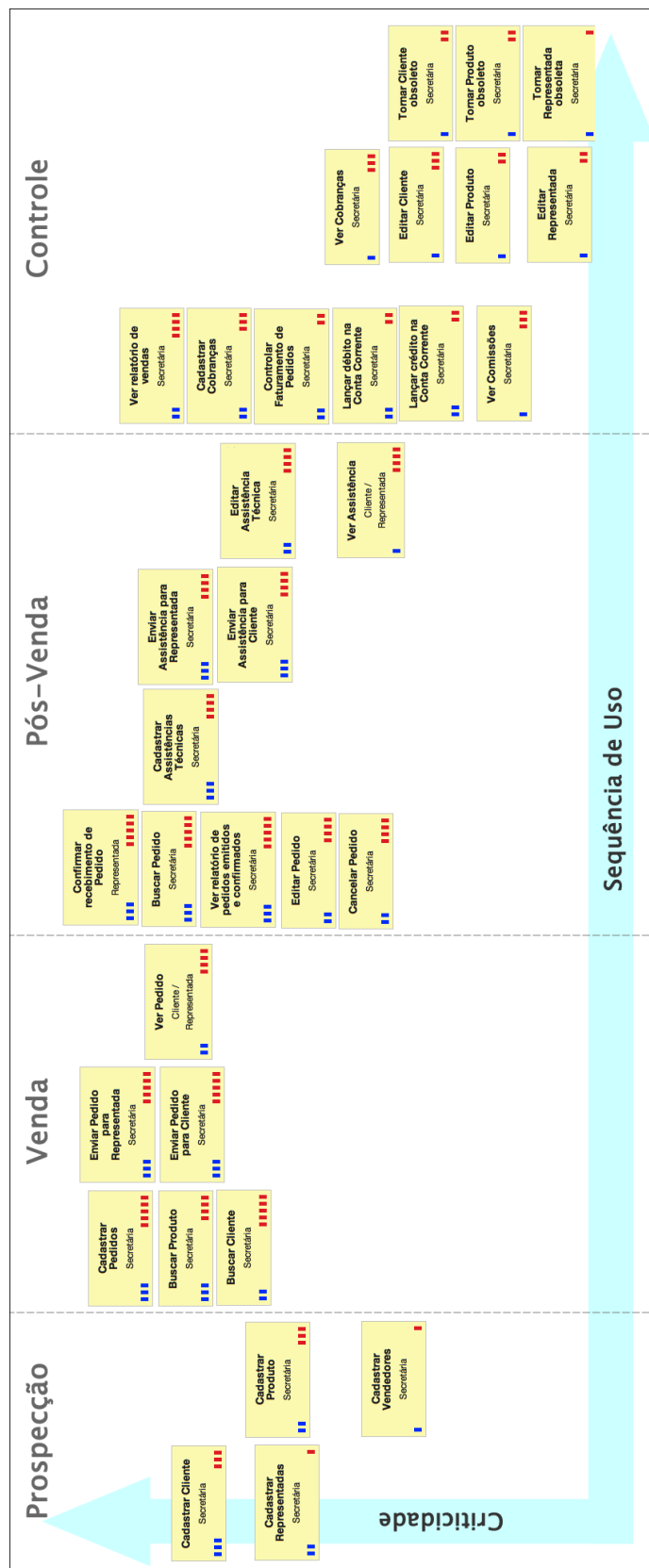


Figura 4.5: Mapa de histórias

Após a elaboração do mapa de histórias, as histórias foram estimadas em *story points* (COHN, 2004). Foi decidido que a primeira parte do projeto, focada na implementação de um protótipo com um conjunto mínimo de funcionalidades, teria quatro iterações de

duas semanas cada, além da primeira iteração que ocorreu durante a fase de exploração.

### 4.3 Iterações

A fase de Iterações inclui o principal trabalho do projeto. É nesta fase que ocorrem a maioria das tarefas de desenvolvimento, incluindo modelagem, programação, teste e integração (AMBLER, 2002a).

#### 4.3.1 Modelagem

O objetivo da modelagem nesta etapa é esclarecer os requisitos de mais alto-nível retirados da lista de histórias do projeto. Foram utilizados três modelos principais para isso: protótipos de telas desenhados em papel, diagramas de estado e o modelo físico do banco de dados.

##### 4.3.1.1 Protótipos de Telas Desenhados em Papel

Para cada história selecionada, inicialmente foram desenhados protótipos das telas da história em papel. A Figura 4.6 apresenta um exemplo de protótipo de tela em papel para a história Listar Pedidos por Data.

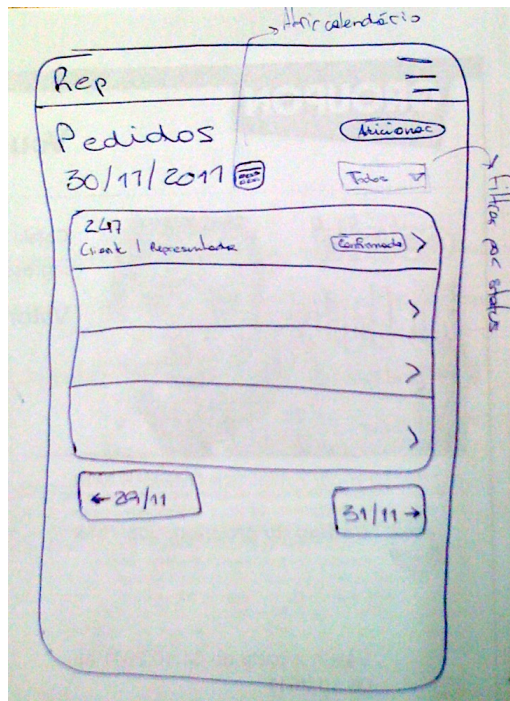


Figura 4.6: Exemplo de protótipo de tela desenhado em papel para uma história

##### 4.3.1.2 Diagramas de Estado

Para histórias envolvendo lógica de negócio mais complexa, foram desenhados diagramas de estado para as entidades envolvidas na história. A Figura 4.7 apresenta um diagrama de estados para a história Cadastrar Pedido.



Figura 4.7: Exemplo de diagrama de estados para uma história

#### 4.3.1.3 Modelo do Esquema Físico do Banco de Dados

A cada término de iteração, era gerado um diagrama do modelo físico de dados para refletir os novos requisitos e garantir que todas as informações necessárias estavam sendo armazenadas. A Figura 4.8 apresenta o modelo físico de dados ao final da quarta iteração (o modelo é gerado automaticamente a partir do código fonte, por isso está em inglês).

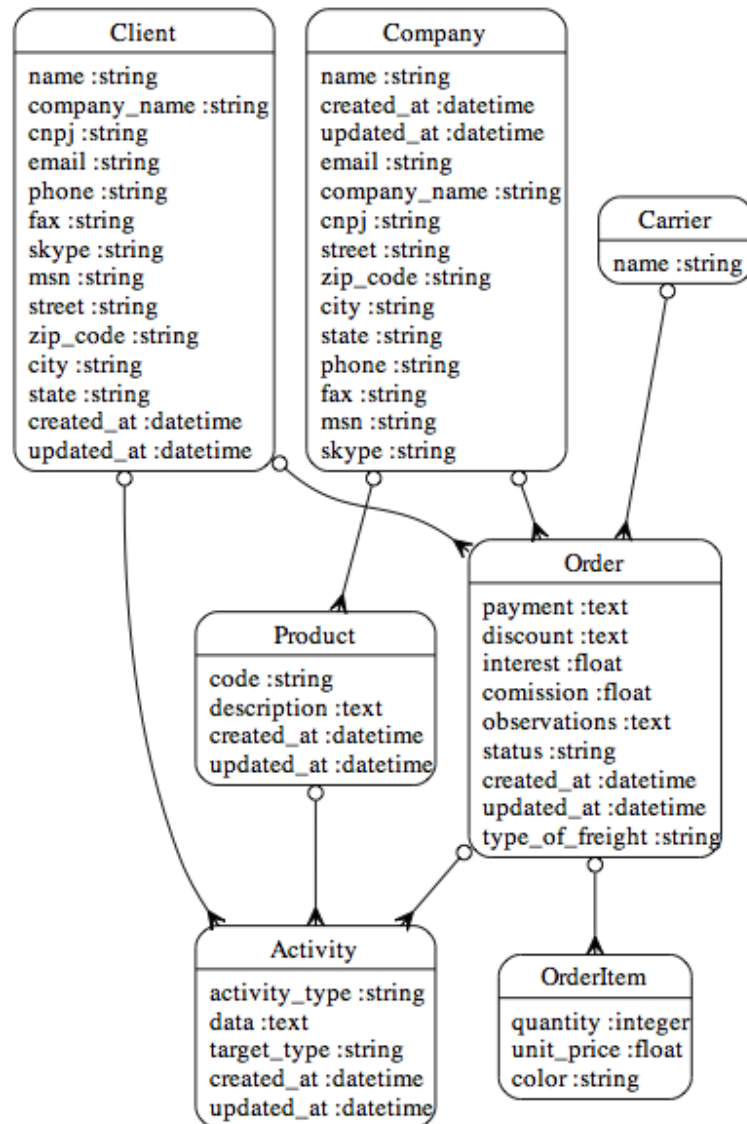


Figura 4.8: Modelo do esquema físico do banco de dados ao final da quarta iteração

Nota-se que o modelo possui um nível de detalhe maior em relação ao modelo de dados conceitual (Figura 4.1). Por outro lado, faltam entidades importantes, como Assis-tências Técnicas. Isso destaca a natureza iterativa do desenvolvimento ágil, onde os modelos evoluem junto com o software.

### 4.3.2 Programação e Teste

Somente na fase de programação e teste que o código do sistema começou a ser escrito. A implementação seguiu o ciclo de *Behaviour-Driven Development* (BDD) (CHELIMSKY, 2010), onde testes de aceitação para cada história descrevem o comportamento esperado do sistema na visão dos *stakeholders* do projeto. Os testes são escritos em linguagem natural, a qual é interpretada e executada automaticamente como uma suíte de testes. Cada história foi implementada seguindo os passos descritos a seguir:

1. Começar com um cenário para a história. O cenário deve estar claro para todos, e os resultados esperados da sua execução também, inclusive como a interface do usuário suportará a interação do usuário com o sistema. A Figura 4.9 apresenta um



exemplo de cenário escrito para a história Editar Itens de Pedido.

```

1 # Language: pt
2
3 Funcionalidade: Editar itens de um pedido
4 Para que um pedido não seja enviado com itens errados
5 Como Vendedor
6 Eu quero editar itens de um pedido
7
8 Contexto:
9 Dado que um pedido existe
10 E que o pedido possui pelo menos um item
11
12 Cenário: Sucesso
13 Dado que eu estou na página de editar um item do pedido
14 Quando eu preencher o formulário do item com dados válidos
15 E eu clicar em "Salvar"
16 Então eu devo ver uma mensagem de sucesso
17 E eu devo ver o pedido

```

Figura 4.9: Exemplo de um cenário de teste de aceitação de uma história

2. Executar o cenário para revelar quais passos do cenário ainda não foram implementados. A maioria dos passos estará indefinido ou pendente. A Figura 4.10 apresenta a primeira execução do cenário.

```

$ bundle exec cucumber --tags @working
Using the default profile...
# language: pt
Funcionalidade: Editar itens de um pedido
Para que um pedido não seja enviado com itens errados
Como Vendedor
Eu quero editar itens de um pedido

Contexto:                                     # features/orders/edit_order_items.feature:8
Dado que um pedido existe                     # features/step_definitions/orders_steps.rb:8
E que o pedido possui pelo menos um item     # features/step_definitions/orders_steps.rb:38

@working
Cenário: Sucesso                             # features/orders/edit_order_items.feature:13
Dado que eu estou na página de editar um item do pedido # features/orders/edit_order_items.feature:14
  Undefined step: "que eu estou na página de editar um item do pedido" (Cucumber::Undefined)
  features/orders/edit_order_items.feature:14:in `Dado que eu estou na página de editar um item do pedido'
Quando eu preencher o formulário do item com dados válidos # features/step_definitions/orders_steps.rb:97
E eu clicar em "Salvar" # features/step_definitions/web_steps.rb:3
Então eu devo ver uma mensagem de sucesso # features/step_definitions/web_steps.rb:7
E eu devo ver o pedido # features/step_definitions/orders_steps.rb:149

1 cenário (1 undefined)
7 steps (4 skipped, 1 undefined, 2 passed)
0m0.942s

You can implement step definitions for undefined steps with these snippets:

Dado /^que eu estou na página de editar um item do pedido$/ do
  pending # express the regexp above with the code you wish you had
end

```

Figura 4.10: Exemplo de execução de cenário não implementado ou com falhas

3. Escrever uma descrição para o primeiro passo, executar o cenário novamente e vê-lo falhar. A Figura 4.11 apresenta um exemplo de descrição para o primeiro passo do cenário.

```

70 Dado /^que eu estou na página de editar um item do pedido$/ do
71   @order_item = @order.order_items.first
72   visit(edit_order_order_item_path(@order, @order_item))
73 end

```

Figura 4.11: Exemplo de definição de um passo de cenário

4. Especificar e implementar somente o código necessário para que o passo seja executado com sucesso.
5. Voltar para a etapa 2 até que todos passos estejam executando corretamente;
6. Após ter especificado e implementado todos os objetos e métodos descobertos ao longo do desenvolvimento da história, executar o cenário novamente para garantir ele está passando. A figura 4.12 apresenta a execução do cenário implementado corretamente.

```

$ bundle exec cucumber --tags @working
Using the default profile...
# language: pt
Funcionalidade: Editar itens de um pedido
  Para que um pedido não seja enviado com itens errados
  Como Vendedor
  Eu quero editar itens de um pedido

Contexto:
  Dado que um pedido existe # features/orders/edit_order_items.feature:8
  E que o pedido possui pelo menos um item # features/step_definitions/orders_steps.rb:38

@working
Cenário: Sucesso # features/orders/edit_order_items.feature:13
  Dado que eu estou na página de editar um item do pedido # features/step_definitions/orders_steps.rb:70
  Quando eu preencher o formulário do item com dados válidos # features/step_definitions/orders_steps.rb:97
  E eu clicar em "Salvar" # features/step_definitions/web_steps.rb:3
  Então eu devo ver uma mensagem de sucesso # features/step_definitions/web_steps.rb:7
  E eu devo ver o pedido # features/step_definitions/orders_steps.rb:149

1 cenário (1 passed)
7 steps (7 passed)
0m2.318s

```

Figura 4.12: Exemplo de execução de cenário implementado corretamente

Todos os testes de aceitação de histórias escritos até o final da quarta iteração podem ser encontrados no Apêndice A.

### 4.3.3 Integração

Ao terminar o desenvolvimento de uma história, executa-se a suíte de testes completa do sistema para garantir que nenhuma modificação feita no código quebrou uma funcionalidade já implementada. A Figura 4.13 apresenta, de forma resumida, um exemplo de execução de toda a suíte de testes do sistema.

```

$ bundle exec cucumber --format progress; bundle exec rspec --format progress
Using the default profile...
.....
.....
.....
.....
.....

62 cenários (62 passed)
309 steps (309 passed)
0m18.884s
.....
.....
.....
.....

Finished in 35.21 seconds
332 examples, 0 failures

```

Figura 4.13: Exemplo de execução da suíte de testes completa

Somente após a execução da suíte de testes é que a implementação da nova história é integrada ao sistema de controle de versão de código e ao ambiente de produção.

Por ser um projeto ainda em desenvolvimento no momento de escrita deste texto, as fases de Produção e Descontinuidade ficaram de fora do escopo deste trabalho.

## 5 PROTÓTIPO

Este capítulo descreve o protótipo funcional que foi desenvolvido durante as quatro primeiras iterações do projeto. As características principais (mobilidade e integração) são apresentadas, seguidas de um exemplo de funcionamento passo-a-passo.

### 5.1 Mobilidade

Seguindo as técnicas de *Responsive Web Design*, apresentadas no Capítulo 2, foram desenvolvidas quatro versões de folhas de estilo para o sistema, cada uma especializada para um determinado tamanho de tela. Os tamanhos de tela foram determinados em função das seguintes categorias de dispositivos:

- *Tablets* grandes em orientação paisagem ou monitores de alta resolução, que geralmente possuem telas com 1024px de largura ou mais (Figura 5.1).

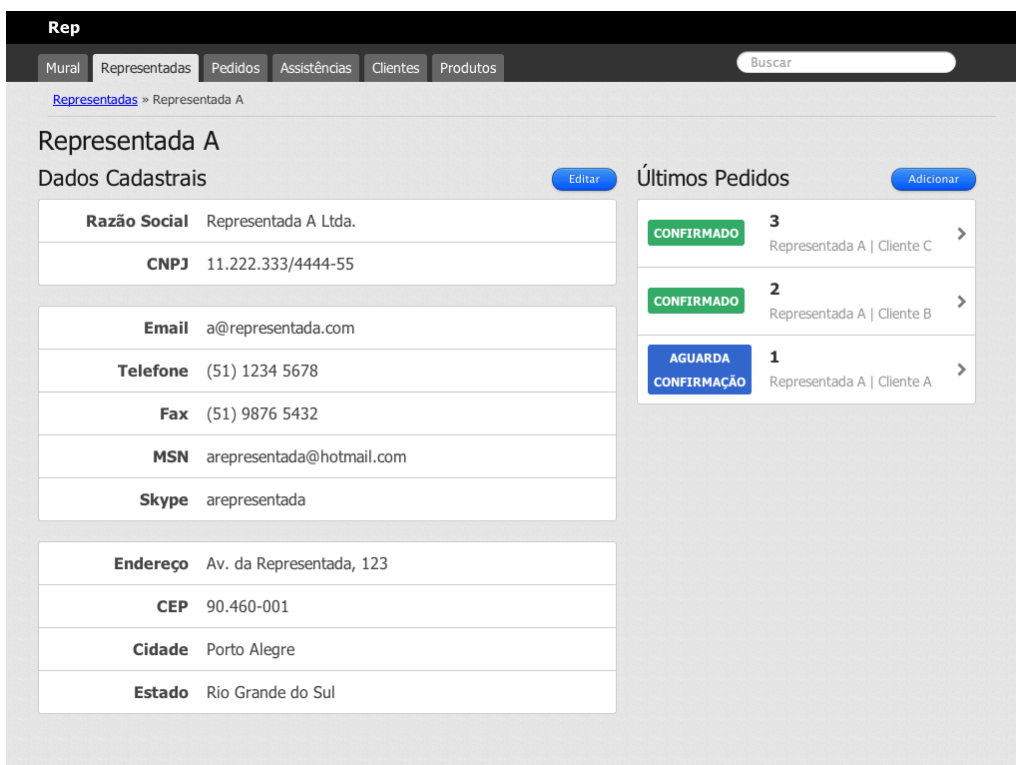


Figura 5.1: Exemplo de visualização em dispositivo com tela de 1024px de largura

- *Tablets* grandes em orientação retrato ou monitores de baixa resolução, que geralmente possuem telas com até 1024px de largura (Figura 5.2). Exceto pela diminuição da largura das duas colunas principais da página, nada foi alterado, já que *websites* tradicionais funcionam relativamente bem em tablets grandes. (NIELSEN, 2010).

The screenshot shows a web application interface on a tablet. At the top, there is a navigation bar with the title 'Rep' and several menu items: 'Mural', 'Representadas', 'Pedidos', 'Assistências', 'Clientes', and 'Produtos'. A search bar labeled 'Buscar' is located on the right side of the navigation bar. Below the navigation bar, the breadcrumb 'Representadas > Representada A' is visible. The main content area is divided into two columns. The left column is titled 'Representada A' and contains a section for 'Dados Cadastrais' with an 'Editar' button. This section lists various contact details: Razão Social (Representada A Ltda.), CNPJ (11.222.333/4444-55), Email (a@representada.com), Telefone ((51) 1234 5678), Fax ((51) 9876 5432), MSN (arepresentada@hotmail.com), and Skype (arepresentada). Below this is an 'Endereço' section with details: Av. da Representada, 123, CEP 90.460-001, Cidade Porto Alegre, and Estado Rio Grande do Sul. The right column is titled 'Últimos Pedidos' with an 'Adicionar' button. It displays a list of three orders: Order 3 (CONFIRMADO) for Representada A | Cliente C, Order 2 (CONFIRMADO) for Representada A | Cliente B, and Order 1 (AGUARDA CONFIRMAÇÃO) for Representada A | Cliente A.

Figura 5.2: Exemplo de visualização em dispositivo com tela de 768px de largura

- *Smartphones* grandes ou em modo paisagem e *tablets* pequenos, que geralmente possuem telas com até 768px de largura (Figura 5.3). Neste caso, as alterações em relação a figura 5.2 são notáveis: as colunas de conteúdo agora ocupam a largura inteira da tela, com o conteúdo secundário (ou auxiliar) sendo reposicionado abaixo do conteúdo principal. Além disso, a navegação é resumida para os *breadcrumbs* e o botão de menu no topo da página.

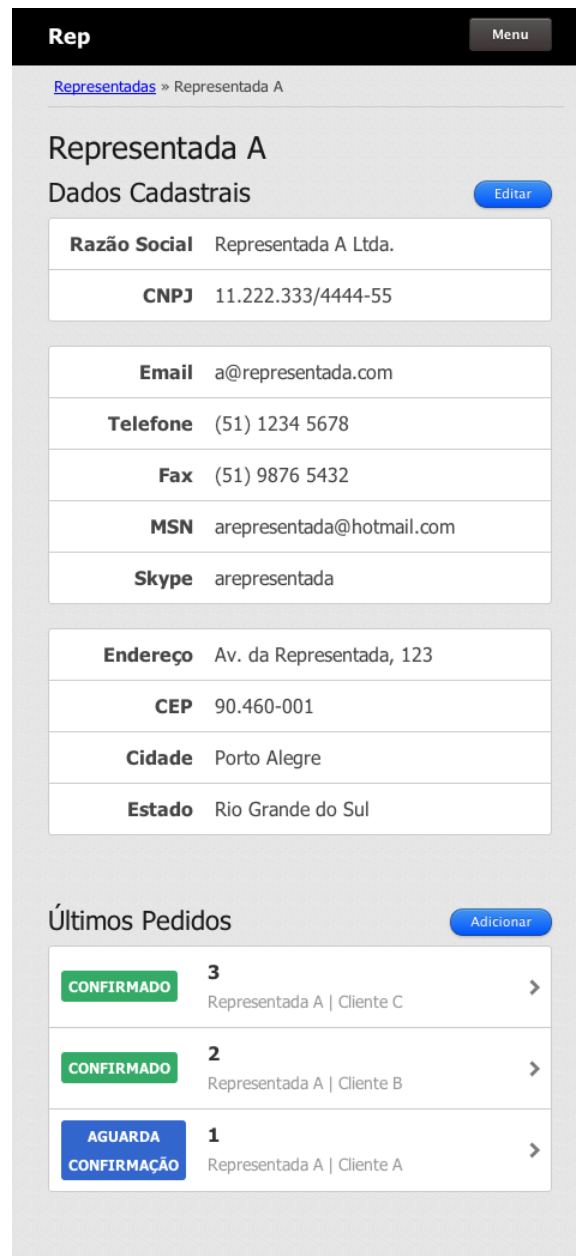


Figura 5.3: Exemplo de visualização em dispositivo com tela de 480px de largura

- *Smartphones* pequenos ou em orientação retrato, que geralmente possuem telas de até 480px de largura (Figura 5.4). Mais uma vez, apenas larguras foram adaptadas em comparação com a Figura 5.3.



Figura 5.4: Exemplo de visualização em dispositivo com tela de 320px de largura

É interessante ressaltar que a estrutura de navegação segue a mesma organização daquela apresentada no protótipo inicial da interface do usuário (Figuras 4.2 e 4.3). Nas telas pequenas, a navegação primária fica escondida, a fim de economizar espaço da tela. Quando necessário, o usuário pode acessá-la clicando no botão “Menu”, localizado no canto superior direito da tela (Figura 5.5).

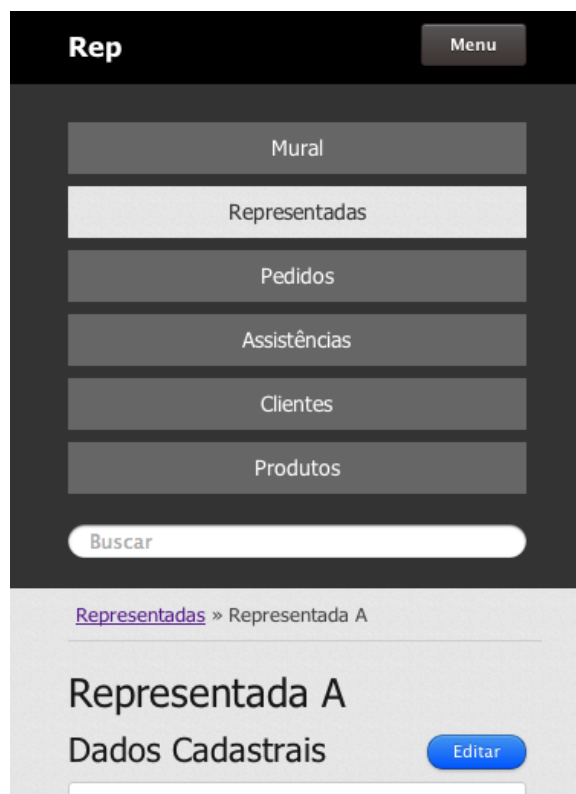


Figura 5.5: Menu de navegação primária expandido em dispositivos com telas pequenas

As características principais dos estilos, como cores, tamanho das letras e a separação entre os blocos de conteúdos, não mudam entre as quatro versões a fim de manter consistência entre as versões. Ações importantes que podem ser realizadas sobre o conteúdo principal da página, como “Adicionar” ou “Editar”, são destacadas como botões finos e compridos em alto-relevo.

Dispositivos limitados ou com *browsers* antigos, que não possuem capacidade de renderização de folhas de estilo, também podem acessar o sistema. Neste caso, a visualização é simplificada, porém bastante funcional, conforme apresentado na Figura 5.6. Isso se deve à utilização de marcações semanticamente corretas no código do sistema.



# Rep

## [Menu](#)

- [Mural](#)
- [Representadas](#)
- [Pedidos](#)
- [Assistências](#)
- [Clientes](#)
- [Produtos](#)

[Representadas](#) » Representada A

## Representada A

### Dados Cadastrais

#### [Editar](#)

**Razão Social** Representada A Ltda.

**CNPJ** 11.222.333/4444-55

**Email** a@representada.com

**Telefone** (51) 1234 5678

**Fax** (51) 9876 5432

**MSN** arep@hotmail.com

**Skype** arepresentada

Figura 5.6: Exemplo de visualização sem renderização de estilos

## 5.2 Integração

A interface apresentada na seção anterior não é a única maneira de interagir com o sistema. Softwares de terceiros podem querer acessar os dados da empresa de forma automática, sem ter que passar por processos manuais de exportação e importação de dados.

Para isso, foram desenvolvidos uma série de *RESTful Web Services* para as entidades do sistema. Por exemplo, para obter os dados de todos os produtos cadastrados, basta fazer uma requisição *GET* no endereço da coleção de produtos. Um exemplo de resposta desta requisição, no formato XML, é apresentado na Figura 5.7.

```

▼<products type="array">
  ▼<product>
    <id type="integer">3</id>
    <code>123</code>
    <description nil="true"/>
    <created-at type="datetime">2011-12-08T02:21:06Z</created-at>
    <updated-at type="datetime">2011-12-08T02:21:06Z</updated-at>
    <company-id type="integer">1</company-id>
  </product>
  ▼<product>
    <id type="integer">4</id>
    <code>12345</code>
    <description nil="true"/>
    <created-at type="datetime">2011-12-08T02:22:22Z</created-at>
    <updated-at type="datetime">2011-12-08T02:22:22Z</updated-at>
    <company-id type="integer">1</company-id>
  </product>
  ▶<product>...</product>
  ▶<product>...</product>
</products>

```

Figura 5.7: Exemplo de resposta de requisição *GET* sobre uma coleção de registros

Para obter os dados de uma representada, deve-se realizar uma requisição *GET* sobre o endereço da representada, conforme apresentado na Figura 5.8.

```

▼<company>
  <id type="integer">1</id>
  <name>Representada A</name>
  <created-at type="datetime">2011-12-07T21:12:48Z</created-at>
  <updated-at type="datetime">2011-12-07T22:04:04Z</updated-at>
  <email>a@representada.com</email>
  <company-name>Representada A Ltda.</company-name>
  <cnj>11.222.333/4444-55</cnj>
  <street>Av. da Representada, 123</street>
  <zip-code>90.460-001</zip-code>
  <city>Porto Alegre</city>
  <state>RS</state>
  <phone>(51) 1234 5678</phone>
  <fax>(51) 9876 5432</fax>
  <msn>arep@hotmail.com</msn>
  <skype>arepresentada</skype>
</company>

```

Figura 5.8: Exemplo de resposta de requisição *GET* sobre um único registro

O Apêndice B apresenta a lista completa dos serviços oferecidos pelo sistema.

### 5.3 Funcionamento Passo-a-Passo

Nesta seção apresentamos duas das principais funcionalidades do sistema: cadastro de pedidos e a página inicial do sistema, a qual representa o conceito de um mural com uma visão resumida do negócio. Nota-se a aplicação das boas práticas para desenvolvimento de interfaces para dispositivos móveis apresentadas no Capítulo 3, como o foco em pequenas tarefas, a disposição e dimensionamento de elementos de ação (como botões e campos de formulário) e o esquema de navegação.

### 5.3.1 Cadastro de Pedidos

Por ser a tarefa mais importante e a mais frequentemente realizada no sistema (ver Figura 4.5), apresentamos o funcionamento passo-a-passo do cadastro de um pedido. Para facilitar o acesso, foram criados diversos atalhos para iniciar o cadastro de um pedido:

- A partir da página inicial do sistema;
- A partir da página de uma representada. Neste caso, o novo pedido já é cadastrado para a representada selecionada;
- A partir da página de um cliente. Neste caso, o novo pedido já é cadastrado para o cliente selecionado;
- A partir da lista de pedidos.

A Figura 5.9 apresenta a tela inicial do cadastro de pedidos quando acessada a partir da página de uma representada.



Figura 5.9: Cadastro de Pedido - Tela Inicial

Após selecionar a representada e o cliente, o usuário tem duas opções: adicionar as informações de pagamento, comissão e frete, ou adicionar itens ao pedido. A segunda opção foi criada para situações em que os itens do pedido precisam ser anotados rapidamente, como quando recebendo um pedido por telefone. A Figura 5.10 apresenta a tela de adicionar item ao pedido.

Rep Menu

[Pedidos](#) » [6](#) » Adicionar Item

Pedido cadastrado(a) com sucesso.

### Adicionar Item

\* Produto  
Selecione...

Cor

\* Valor Unitário

\* Quantidade

OU

Figura 5.10: Cadastro de Pedido - Adicionar Item

Após preencher o formulário do item, o usuário pode escolher se deseja adicionar mais um item ou simplesmente adicionar o item atual e retornar para o pedido. Caso o usuário não preencha os dados do formulário corretamente, uma mensagem de erro é apresentada, e os campos que contêm erros são destacados (Figura 5.11). Cada campo preenchido incorretamente também recebe uma mensagem para facilitar na correção do erro de preenchimento.

The screenshot shows a mobile application interface with a black header containing the text "Rep" and a "Menu" button. Below the header, the breadcrumb "Pedidos » 6 » Adicionar Item" is visible. A red-bordered box contains the error message: "Ocorreu um erro ao cadastrar Item. Por favor, verifique se todos os campos foram preenchidos corretamente." The main title is "Adicionar Item". There are four required fields, each with a red asterisk and a red error message below it: 1. "Produto" is a dropdown menu with "Selecione..." selected and the error "não pode ficar em branco". 2. "Cor" is an empty text input field. 3. "Valor Unitário" is an empty text input field with the error "não pode ficar em branco". 4. "Quantidade" is a text input field containing "-1.00" with a numeric keypad icon and the error "deve ser maior que 0". At the bottom, there are two blue buttons: "Salvar" and "Salvar e Adicionar Outro Item", separated by the word "OU".

Figura 5.11: Cadastro de Pedido - Adicionar Item - Mensagens de Erro

Após corrigir os erros e adicionar o(s) item(ns), o usuário é redirecionado para a tela do pedido (Figura 5.12), onde pode rever os dados cadastrados e realizar ações sobre o mesmo.

**Rep** Menu

[Pedidos](#) » 6

Item cadastrado(a) com sucesso.

**Pedido 6**  
08/12/2011 RASCUNHO

**Cliente** Cliente C >

**Representada** Representada A >

**Pagamento, Comissão e Frete** Editar

<b>Prazo de Pagamento</b>	Não informado
<b>Desconto</b>	Não informado
<b>Juros</b>	0.0%
<b>Comissão</b>	0.0%
<b>Tipo do Frete</b>	Não informado

**Itens** Adicionar

<b>Poltrona 400</b> R\$700,00   Vermelha	<b>2</b> >
---	------------

<b>Total de Comissão</b>	R\$0,00
<b>Total do Pedido</b>	R\$700,00

**Enviar Pedido**

Figura 5.12: Cadastro de Pedido - Rascunho

Na parte superior da tela ficam as informações mais importantes do pedido: data de cadastro, cliente, representada e situação do mesmo (de acordo com os estados apresentados na Figura 4.7). Logo abaixo ficam as informações de pagamento, comissão e frete do pedido, seguidas pelos itens. Cada item é apresentado de forma resumida, contendo apenas as informações mais importantes: código, valor total, cor e quantidade solicitada. Depois dos itens ficam as informações calculadas sobre os dados do pedido, como valor da comissão e total do pedido. Por fim, ficam as ações que podem ser realizadas sobre um

pedido, como enviar, confirmar recebimento ou cancelar. O posicionamento das ações ao final da tela é proposital para que o usuário revise todas as informações do pedido antes de realizar alguma ação sobre o mesmo.

Para completar o cadastro do pedido, acessamos a tela de informações de pagamento, comissão e frete do pedido (Figura 5.13).

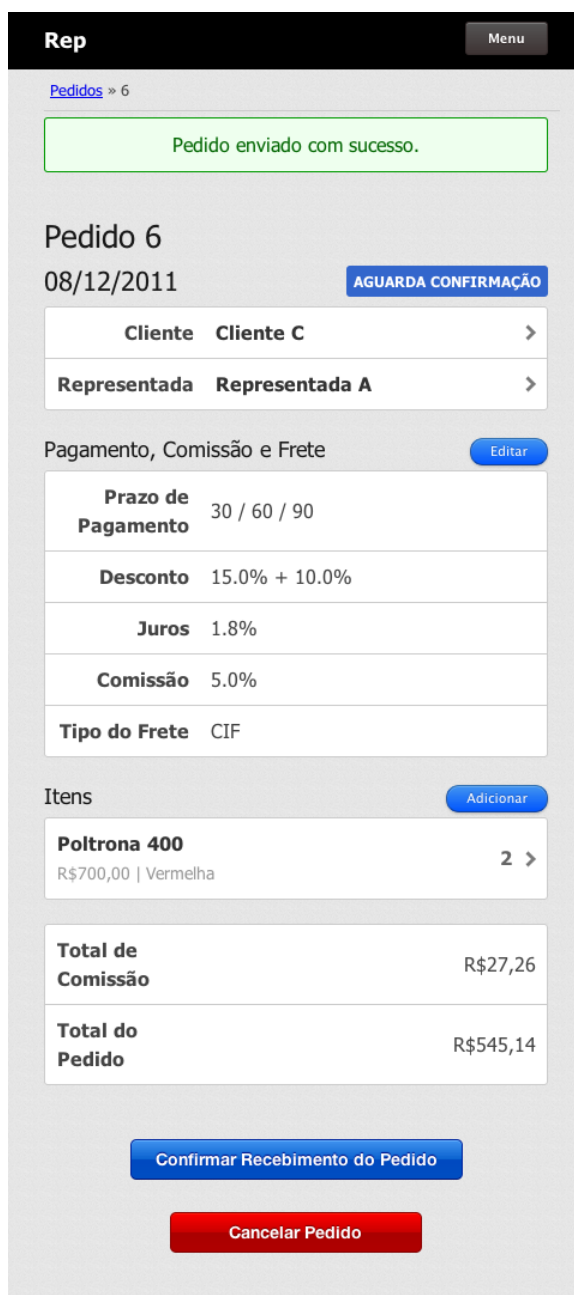
The screenshot shows a web interface for editing a request. At the top, there is a header with 'Rep' on the left and a 'Menu' button on the right. Below the header, a breadcrumb trail reads 'Pedidos » 6 » Editar Pedido'. The main title is 'Editar Pedido'. The form contains several sections:

- \* Prazo de Pagamento:** A row of five input fields with up/down arrows. The first three contain '30', '60', and '90'. The last two are empty.
- \* Desconto:** A row of five input fields with up/down arrows and plus signs between them. The first two contain '15' and '10'. The last three are empty.
- Juros:** A single input field containing '1.80'.
- Comissão:** A single input field containing '5.00'.
- Tipo do Frete:** A dropdown menu with 'CIF' selected.
- Transportadora:** A dropdown menu with 'Selezione...' selected.
- Observações:** A large empty text area.

At the bottom left of the form is a blue button labeled 'Salvar'.

Figura 5.13: Cadastro de Pedido - Informações de Pagamento, Comissão e Frete

Quando um pedido está completo, podemos enviá-lo para o cliente e a representada. Ao clicar no botão "Enviar Pedido"(Figura 5.12), uma cópia do pedido é enviada por email para o cliente e para a representada, o usuário recebe uma notificação de que o pedido foi entregue e a situação do pedido é atualizada para *Aguardando Confirmação* (Figura 5.14). As outras ações, como cancelar ou confirmar o recebimento de um pedido, procedem da mesma forma. Botões de ações utilizadas menos frequentemente, como cancelar um pedido, são destacados em vermelho para diferenciá-los das ações mais comuns.



**Rep** Menu

[Pedidos](#) » 6

Pedido enviado com sucesso.

**Pedido 6**  
08/12/2011 AGUARDA CONFIRMAÇÃO

**Cliente** Cliente C >

**Representada** Representada A >

Pagamento, Comissão e Frete Editar

<b>Prazo de Pagamento</b>	30 / 60 / 90
<b>Desconto</b>	15.0% + 10.0%
<b>Juros</b>	1.8%
<b>Comissão</b>	5.0%
<b>Tipo do Frete</b>	CIF

Itens Adicionar

<b>Poltrona 400</b> R\$700,00   Vermelha	2 >
---	-----

<b>Total de Comissão</b>	R\$27,26
<b>Total do Pedido</b>	R\$545,14

Confirmar Recebimento do Pedido

Cancelar Pedido

Figura 5.14: Cadastro de Pedido - Aguardando Confirmação



Todos os pedidos cadastrados podem ser visualizados na página de listar pedidos (Figura 5.15). Apenas as informações mais importantes do pedido são exibidas inicialmente, com destaque para a situação e o código do pedido. Cores são utilizadas para facilitar o reconhecimento da situação de cada pedido. Os pedidos são organizados por data, e a navegação entre os dias é feita de forma linear através dos botões para o dia anterior ou o próximo dia.

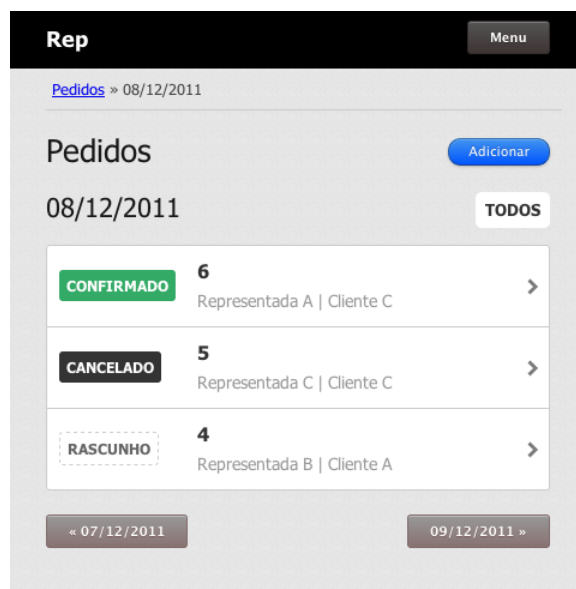


Figura 5.15: Lista de pedidos organizados por data

Os pedidos também podem ser filtrados de acordo com sua situação, conforme apresentado na Figura 5.16. Ao selecionar uma situação, somente os pedidos que encontram-se naquela situação são listados.



Figura 5.16: Menu para filtrar pedidos por situação

### 5.3.2 Mural

Na página inicial do sistema foi criada uma analogia de um painel administrativo, ou de um mural de atividades (Figura 5.17). No mural, ficam as últimas atividades realizadas no sistema, como criação e atualização de cadastros, envio ou cancelamento de pedidos, etc. Além disso, informações importantes, como quais pedidos ainda estão pendentes, também tem um espaço de destaque.

The screenshot displays a dashboard titled 'Mural' with a 'Rep' header and a 'Menu' button. It is divided into two main sections: 'Atividades Recentes' and 'Pedidos Pendentes'.

**Atividades Recentes**

8/12	<b>Pedido 6 enviado</b>	14h55	Representada A   Cliente C	>
8/12	<b>Pedido 6 cadastrado</b>	14h30	Representada A   Cliente C	>
8/12	<b>Pedido 5 cadastrado</b>	14h24	Representada C   Cliente C	>
8/12	<b>Pedido 4 cadastrado</b>	3h48	Representada B   Cliente A	>
8/12	<b>Produto 12345 cadastrado</b>	2h22	Representada A	>
8/12	<b>Produto 123 cadastrado</b>	2h21	Representada A	>
7/12	<b>Pedido 2 confirmado</b>	21h37	Representada A   Cliente B	>
7/12	<b>Pedido 2 enviado</b>	21h37	Representada A   Cliente B	>
7/12	<b>Pedido 3 confirmado</b>	21h35	Representada A   Cliente C	>
7/12	<b>Pedido 3 enviado</b>	21h35	Representada A   Cliente C	>

**Pedidos Pendentes** Adicionar

<b>AGUARDA CONFIRMAÇÃO</b>	<b>6</b>	Representada A   Cliente C	>
<b>RASCUNHO</b>	<b>5</b>	Representada C   Cliente C	>
<b>RASCUNHO</b>	<b>4</b>	Representada B   Cliente A	>
<b>AGUARDA CONFIRMAÇÃO</b>	<b>1</b>	Representada A   Cliente A	>

Figura 5.17: Página Inicial - Mural

À medida que o projeto evoluir e novas funcionalidades forem implementadas, outras informações podem entrar no mural, como resumos de relatórios de vendas ou assistências técnicas pendentes.

Por ser um projeto em desenvolvimento, aspectos como testes completos de usabilidade e integrações reais com outros sistemas ficaram de fora do escopo deste trabalho, mas serão essenciais para a validação do sistema proposto.

## 6 CONCLUSÃO

Este trabalho apresentou a concepção e o desenvolvimento de um sistema que visa auxiliar no gerenciamento de informações de empresas de representação comercial. Através da adoção de princípios ágeis de desenvolvimento de software junto à uma empresa real, utilizada como cliente-alvo, um protótipo funcional foi desenvolvido. O sistema focou em dois pontos principais que não eram atendidos pelas ferramentas utilizadas pelo cliente-alvo: mobilidade e integração.

O sistema foi projetado com uma abordagem *Mobile First*, pensado primeiramente em como seria seu uso em dispositivos móveis. Através das técnicas de *Responsive Web Design*, foi desenvolvida uma interface do usuário adaptável para dispositivos com diferentes tamanhos de tela. Foram desenvolvidas versões especializadas da interface do usuário para quatro categorias de dispositivos: *smartphones* pequenos; *smartphones* grandes ou *tablets* pequenos; *tablets* grandes ou computadores portáteis pequenos; e computadores de mesa. Além disso, uma série de *RESTful Web Services* foram desenvolvidos para fornecer uma interface de comunicação padronizada com outros sistemas de *software*.

O desenvolvimento do projeto foi apresentado em três etapas principais: exploração, planejamento e iterações. Cada etapa foi descrita com o apoio dos principais modelos utilizados, como: Mapa de Histórias de Usuário, rascunhos de interface do usuário em papel, modelo de dados conceitual e modelo do esquema físico do banco de dados. Também foi apresentado o desenvolvimento do software seguindo o ciclo de *Behaviour Driven Development*.

A continuidade do desenvolvimento do projeto deve resolver primeiramente as questões mais importantes do cliente-alvo, mas generalizações poderão ser feitas para outros modelos de negócio similares. Questões que ficaram de fora do escopo deste trabalho devem ser exploradas em maior profundidade, como o funcionamento do sistema quando não há conexão de rede, testes completos de usabilidade e integrações com outros sistemas reais.

## REFERÊNCIAS

- AGILE ALLIANCE. **Página da Aliança Ágil.** Disponível em: <<http://www.agilealliance.org/>>. Acesso em: dezembro 2011.
- AMBLER, S. W. **Agile Modeling**, John Wiley and Sons, 2002.
- AMBLER, S. W. **Data Modeling 101.** Disponível em: <<http://www.agiledata.org/essays/dataModeling101.html>>. Acesso em: dezembro 2011.
- AMBLER, S. W. **Página da Modelagem Ágil.** Disponível em: <<http://www.agilemodeling.com/>>. Acesso em: dezembro 2011.
- APPCCELERATOR INC. **Página do Titanium.** Disponível em: <<http://www.appcelerator.com/products/titanium-cross-platform-application-development/>>. Acesso em: dezembro 2011.
- APPLE INC. **Página do iOS.** Disponível em: <<http://www.apple.com/ios/>>. Acesso em: dezembro 2011.
- BECK, K. **eXtreme Programming Explaied.** Addison-Wesley Professional, 1999.
- CHELIMSKY, et al. **The RSpec Book**, Pragmatic Bookshelf, 2010.
- COHN, M. **User Stories Applied: For Agile Software Development**, Addison Wesley, 2004.
- COHN, M. **Agile Estimating and Planning**, Upper Saddle River: Prentice Hall, 2006.
- EDIGER, B. **Advanced Rails**, Sebastopool: O'Reilly, 2008.
- ETEMAD, E. J. **Cascading Style Sheets (CSS) Snapshot 2010.** Disponível em: <<http://www.w3.org/TR/2011/NOTE-css-2010-20110512/>>. Acesso em: dezembro 2011.
- FIELDING, R. **Architectural Styles and the Design of Network-based Software Architectures.** University of California, Irvine, 2000.
- GOOGLE INC. **Página do Android.** Disponível em: <<http://www.android.com/>>. Acesso em: dezembro 2011.
- HE, H. **Implementing REST Web Services: Best Practices and Guidelines.** Disponível em: <<http://www.xml.com/pub/a/2004/08/11/rest.html>>. Acesso em: dezembro 2011.

- HYATT, D.; HICKINSON, I., , **HTML5**. W3C Working Draft, 2009.
- KALBACH, J. **Designing Web Navigation**, O'Reilly Media, 2007.
- MARCOTTE, E. **Responsive Web Design**. New York, USA: A Book Apart, 2011.
- NIELSEN, J. **Usability Engineering**. San Francisco, USA: Morgan Kaufmann, 1993.
- NIELSEN, J. **Guerrilla HCI: Using Discount Usability Engineering to Penetrate the Intimidation Barrier**, 1994.
- NIELSEN, J. **Mobile Usability, First Findings**. Disponível em: <<http://www.useit.com/alertbox/mobile-usability-study-1.html>>. Acesso em: dezembro 2011.
- NIELSEN, J. **iPad Usability: First Findings From User Testing**. Disponível em: <<http://www.useit.com/alertbox/ipad-1st-study.html>>. Acesso em: dezembro 2011.
- NIELSEJ, J. **Utilize Available Screen Space**. Disponível em: <<http://www.useit.com/alertbox/screen-space-use.html>>. Acesso em: dezembro 2011.
- NIELSEN, J. **Mobile Usability Update**. Disponível em: <<http://www.useit.com/alertbox/mobile-usability.html>>. Acesso em: dezembro 2011.
- PATTON, J. **It's All In How You Slice It**. Better Software, January 2005.
- PAUTASSO, C.; ZIMMERMANN, O.; LEYMAN, F. RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision. In: INTERNATIONAL WORLD WIDE WEB CONFERENCE, WWW, 17., 2008. **Proceedings...**, Beijing, China, 2008.
- RHOMOBILE INC. **Página do Rhodes**. Disponível em: <<http://rhomobile.com/>>. Acesso em: dezembro 2011.
- RIEDEMANN, C.; Freitag, R. **Modeling Usage: Techniques and Tools**, IEEE Software 26(2), 2009, p. 20-24.
- RICHARDSON, L.; RUBY, S. **RESTful Web Services**. Sebastopol, USA: O'Reilly Media, Inc., 2007.
- SOMMERVILE, I.; KOTONYA, G. **Requirements Engineering**, John Wiley and Sons, 1997.
- SNYDER, C. **Paper Prototyping**, Morgan Kaufmann, 2003.
- WROBLEWSKI, L. **Mobile First**. New York, USA: A Book Apart, 2011.
- WELLS, D. **Spike Solution**. Disponível em: <<http://www.extremeprogramming.org/rules/spike.html>>. Acesso em: dezembro 2011.
- WIKIPEDIA. **Página do MS-DOS na Wikipedia**. Disponível em: <<http://en.wikipedia.org/wiki/MS-DOS>>. Acesso em: dezembro 2011.

## APÊNDICE A TESTES DE ACEITAÇÃO DE HISTÓRIAS

### Testes de Aceitação de História A.1: Registrar movimentações de clientes

# language: pt

**Funcionalidade:** Registrar movimentações de clientes

**Para** revisar rapidamente o que aconteceu no dia

**Como** Representante

**Eu** quero que o sistema registre as movimentações de clientes

Cenário: Novo cliente

**Quando** um cliente for cadastrado

Então eu devo ver o cliente como "cadastrado" na lista de atividades

Cenário: Editar cliente

**Quando** um cliente for editado

Então eu devo ver o cliente como "editado" na lista de atividades

---

### Testes de Aceitação de História A.2: Registrar movimentações de pedidos

# language: pt

**Funcionalidade:** Registrar movimentações de pedidos

**Para** revisar rapidamente o que aconteceu no dia

**Como** Representante

**Eu** quero que o sistema registre as movimentações de pedidos

Cenário: Novo pedido

**Quando** um pedido for cadastrado

Então eu devo ver o pedido como "cadastrado" na lista de atividades

Cenário: Pedido enviado

**Quando** um pedido for enviado

Então eu devo ver o pedido como "enviado" na lista de atividades

Cenário: Pedido confirmado

**Quando** um pedido for confirmado

Então eu devo ver o pedido como "confirmado" na lista de atividades

Cenário: Pedido cancelado

**Quando** um pedido for cancelado

Então eu devo ver o pedido como "cancelado" na lista de atividades

---

### Testes de Aceitação de História A.3: Registrar movimentações de produtos

# language: pt

**Funcionalidade:** Registrar movimentações de produtos

**Para** revisar rapidamente o que aconteceu no dia

**Como** Representante

**Eu** quero que o sistema registre as movimentações de produtos

Cenário: Novo produto

**Quando** um produto for cadastrado

Então eu devo ver o produto como "cadastrado" na lista de atividades

Cenário: Editar produto

**Quando** um produto for editado

Então eu devo ver o produto como "editado" na lista de atividades

---

### Testes de Aceitação de História A.4: Editar Transportadora

# language: pt

**Funcionalidade:** Editar Transportadora

**Como** Secretária

**Eu** quero editar uma Transportadora

**Contexto:**

**Dado** que eu estou na página de editar Transportadora

Cenário: Sucesso

**Quando** eu preencher o formulário da Transportadora com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver a Transportadora

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de uma Transportadora que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

---

### Testes de Aceitação de História A.5: Listar Transportadoras

# language: pt



**Funcionalidade:** Listar Transportadoras

**Como** Secretária

**Eu** quero ver uma lista das Transportadoras cadastrados

**Contexto:**

**Dado** que pelo menos uma Transportadora esteja cadastrada

Cenário: Sucesso

**Quando** eu estou na página de listar Transportadoras

Então eu devo ver a lista das Transportadoras cadastradas

---

#### Testes de Aceitação de História A.6: Cadastrar Transportadora

# language: pt

**Funcionalidade:** Cadastrar Transportadora

**Como** Secretária

**Eu** quero cadastrar uma Transportadora

**Contexto:**

**Dado** que eu estou na página de adicionar Transportadora

Cenário: Sucesso

**Quando** eu preencher o formulário da Transportadora com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver a Transportadora

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de uma Transportadora que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

---

#### Testes de Aceitação de História A.7: Ver Transportadora

# language: pt

**Funcionalidade:** Ver Transportadora

**Como** Secretária

**Eu** quero ver os dados de uma Transportadora cadastrado

Cenário: Sucesso

**Dado** que eu estou na página de uma Transportadora

Então eu devo ver os dados da Transportadora

---

#### Testes de Aceitação de História A.8: Editar Cliente

# language: pt

**Funcionalidade:** Editar Cliente

**Como** Secretária

**Eu** quero editar um cliente

**Contexto:**

**Dado** que eu estou na página de editar cliente

Cenário: Sucesso

**Quando** eu preencher o formulário do cliente com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver o cliente

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de um cliente que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

Cenário: CNPJ duplicado

**Quando** eu preencher o formulário com o CNPJ de um cliente que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "CNPJ" com o erro "já está em uso"

---

### Testes de Aceitação de História A.9: Buscar Cliente por Código

# language: pt

**Funcionalidade:** Buscar Cliente por Código

**Para** acessar um cliente rapidamente

**Como** Secretária

**Eu** quero buscar um cliente por código

**Contexto:**

**Dado** que um cliente existe

E que eu estou na página de listar clientes

Cenário: Sucesso

**Quando** eu buscar pelo código do cliente

Então eu devo ver o cliente

---

### Testes de Aceitação de História A.10: Listar Clientes

# language: pt

**Funcionalidade:** Listar Clientes

**Como** Secretária

**Eu** quero ver uma lista dos Clientes cadastrados

**Contexto:**

**Dado** que pelo menos um cliente esteja cadastrado

Cenário: Sucesso

**Quando** eu estou na página de listar clientes  
Então eu devo ver a lista dos clientes cadastrados

---

### Testes de Aceitação de História A.11: Cadastrar Cliente

# language: pt

**Funcionalidade:** Cadastrar Cliente

**Como** Secretária

**Eu** quero cadastrar um cliente

**Contexto:**

**Dado** que eu estou na página de adicionar cliente

Cenário: Sucesso

**Quando** eu preencher o formulário do cliente com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver o cliente

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de um cliente que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

Cenário: CNPJ duplicado

**Quando** eu preencher o formulário com o CNPJ de um cliente que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "CNPJ" com o erro "já está em uso"

---

### Testes de Aceitação de História A.12: Ver Cliente

# language: pt

**Funcionalidade:** Ver Cliente

**Como** Secretária

**Eu** quero ver os dados de um Cliente cadastrado

Cenário: Sucesso

**Dado** que eu estou na página de um cliente

Então eu devo ver os dados do cliente

E eu devo ver uma lista de pedidos recentes

E eu devo ver um link para ver o cliente no mapa

---

### Testes de Aceitação de História A.13: Editar Representada

# language: pt

**Funcionalidade:** Editar Representada

**Como** Secretária

**Eu** quero editar uma representada

**Contexto:**

**Dado** que eu estou na página de editar representada

Cenário: Sucesso

**Quando** eu preencher o formulário da representada com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver a representada

Cenário: Dados inválidos

**Quando** eu preencher o formulário da representada com dados inválidos

Então eu devo ver uma mensagem de erro

E eu devo ver os campos que contêm erros

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de uma representada que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

---

### Testes de Aceitação de História A.14: Listar Representadas

# language: pt

**Funcionalidade:** Listar Representadas

**Como** Secretária

**Eu** quero ver uma lista das Representadas cadastradas

**Contexto:**

**Dado** que pelo menos uma representada esteja cadastrada

Cenário: Sucesso

**Quando** eu estou na página de listar representadas

Então eu devo ver a lista das representadas cadastradas

---

### Testes de Aceitação de História A.15: Cadastrar Representada

# language: pt

**Funcionalidade:** Cadastrar Representada

**Como** Secretária

**Eu** quero cadastrar uma representada

**Contexto:**

**Dado** que eu estou na página de adicionar representada

Cenário: Sucesso

**Quando** eu preencher o formulário da representada com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver a representada

Cenário: Dados inválidos

**Quando** eu preencher o formulário da representada com dados inválidos

Então eu devo ver uma mensagem de erro

E eu devo ver os campos que contêm erros

Cenário: Nome duplicado

**Quando** eu preencher o formulário com o nome de uma representada que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Nome" com o erro "já está em uso"

---

### Testes de Aceitação de História A.16: Ver Representada

# language: pt

**Funcionalidade:** Ver Representada

**Como** Secretária

**Eu** quero ver os dados de uma Representada cadastrada

Cenário: Sucesso

**Dado** que eu estou na página de uma representada

Então eu devo ver os dados da representada

E eu devo ver uma lista de pedidos recentes

---

### Testes de Aceitação de História A.17: Adicionar itens a um Pedido

# language: pt

**Funcionalidade:** Adicionar itens a um Pedido

**Para** promover os produtos das representadas

**Como** Vendedor

**Eu** quero adicionar itens a um Pedido

**Contexto:**

**Dado** que um pedido existe

E que eu estou na página de adicionar um item ao pedido

Cenário: Adicionar um item

**Quando** eu preencher o formulário do item com dados válidos

E eu clicar em "Salvar"

Então eu devo ver uma mensagem de sucesso

E eu devo ver o pedido

Cenário: Adicionar mais de um item

**Quando** eu preencher o formulário do item com dados válidos

E eu clicar em "Salvar e Adicionar Outro Item"

Então eu devo ver uma mensagem de sucesso

E eu devo ver a página de Adicionar Item ao pedido

Cenário: Dados inválidos

**Quando** eu preencher o formulário do item com dados inválidos  
E eu clicar em "Salvar"  
Então eu devo ver uma mensagem de erro  
E eu devo ver os campos que contêm erros

---

## Testes de Aceitação de História A.18: Adicionar informações de Pagamento, Comissão e Frete a um pedido

# language: pt

**Funcionalidade:** Adicionar informações de Pagamento, Comissão e Frete a um pedido

**Para** garantir a cobrança e entrega de um pedido

**Como** Vendedor

**Eu** quero adicionar informações de Pagamento, Comissão e Frete a um pedido

### Contexto:

**Dado** que um pedido existe

E que uma transportadora existe

E que eu estou na página de Pagamento, Comissão e Frete do pedido

Cenário: Frete CIF

**Quando** eu selecionar frete do tipo "CIF"  
E eu enviar o formulário de pagamento e comissão com dados válidos  
Então eu devo ver uma mensagem de sucesso  
E eu devo ver o pedido

Cenário: Frete FOB com transportadora

**Quando** eu selecionar frete do tipo "FOB"  
E eu selecionar uma transportadora  
E eu enviar o formulário de pagamento e comissão com dados válidos  
Então eu devo ver uma mensagem de sucesso  
E eu devo ver o pedido

Cenário: Frete FOB sem transportadora

**Quando** eu selecionar frete do tipo "FOB"  
Mas eu não selecionar uma transportadora  
E eu enviar o formulário de pagamento e comissão com dados válidos  
Então eu devo ver uma mensagem de erro  
E eu devo ver o campo "Transportadora" com o erro "não pode ficar em branco"

Cenário: Dados inválidos

**Quando** eu não selecionar uma transportadora  
E eu enviar o formulário de pagamento e comissão com

dados inválidos  
Então eu devo ver uma mensagem de erro  
E eu devo ver os campos que contêm erros

---

### Testes de Aceitação de História A.19: Cancelar Pedido

# language: pt

**Funcionalidade:** Cancelar Pedido

**Para** que um pedido não seja fabricado por engano

**Como** Secretária

**Eu** quero cancelar um pedido

Cenário: Pedido Aguardando Confirmação

**Dado** que um pedido aguardando confirmação existe

E que eu estou na página do pedido

**Quando** eu clicar em "Cancelar Pedido"

Então eu devo ver uma mensagem de sucesso

E eu devo ver o pedido na situação "Cancelado"

Cenário: Pedido Confirmado

**Dado** que um pedido confirmado existe

E que eu estou na página do pedido

**Quando** eu clicar em "Cancelar Pedido"

Então eu devo ver uma mensagem de sucesso

E eu devo ver o pedido na situação "Cancelado"

---

### Testes de Aceitação de História A.20: Confirmar Recebimento de Pedido

# language: pt

**Funcionalidade:** Confirmar Recebimento de Pedido

**Para** que todos saibam que o pedido será produzido

**Como** Cliente

**Eu** quero confirmar o recebimento de um pedido

**Contexto:**

**Dado** que um pedido aguardando confirmação existe

E que eu estou na página do pedido

Cenário: Sucesso

**Quando** eu clicar em "Confirmar Recebimento do Pedido"

Então eu devo ver uma mensagem de sucesso

E eu devo ver o pedido na situação "Confirmado"

---

### Testes de Aceitação de História A.21: Editar itens de um pedido

# language: pt

**Funcionalidade:** Editar itens de um pedido

**Para** que um pedido não seja enviado com itens errados

**Como** Vendedor

**Eu** quero editar itens de um pedido

**Contexto:**

**Dado** que um pedido existe  
E que o pedido possui pelo menos um item

Cenário: Sucesso

**Dado** que eu estou na página de editar um item do pedido  
**Quando** eu preencher o formulário do item com dados válidos  
E eu clicar em "Salvar"  
Então eu devo ver uma mensagem de sucesso  
E eu devo ver o pedido

Cenário: Dados inválidos

**Dado** que eu estou na página de editar um item do pedido  
**Quando** eu preencher o formulário do item com dados  
inválidos  
Então eu devo ver uma mensagem de erro  
E eu devo ver os campos que contêm erros

Cenário: Remover item

**Dado** que eu estou na página de editar um item do pedido  
**Quando** eu clicar em "Excluir"  
Então eu devo ver uma mensagem de sucesso  
E eu devo ver o pedido  
E eu não devo ver o item

---

**Testes de Aceitação de História A.22: Buscar Pedido por Código**

# language: pt

**Funcionalidade:** Buscar Pedido por Código

**Para** acessar um pedido rapidamente  
**Como** Secretária  
**Eu** quero buscar um pedido por código

**Contexto:**

**Dado** que um pedido existe  
E que eu estou na página de listar pedidos

Cenário: Sucesso

**Quando** eu buscar pelo código do pedido  
Então eu devo ver o pedido

---

**Testes de Aceitação de História A.23: Finalizar Rascunho de Pedido**

# language: pt

**Funcionalidade:** Finalizar Rascunho de Pedido

**Para** que um pedido possa ser enviado para representada e  
cliente  
**Como** Secretária  
**Eu** quero finalizar um rascunho de pedido



Cenário: Sucesso

**Dado** que um rascunho completo de pedido existe

**Quando** eu estou na página do pedido

E eu clicar em "Enviar Pedido"

Então eu devo ver uma mensagem de sucesso

E eu devo ver o pedido na situação "Aguarda Confirmação"

Cenário: Pedido Incompleto

**Dado** que um rascunho incompleto de pedido existe

**Quando** eu estou na página do pedido

E eu clicar em "Enviar Pedido"

Então eu devo ver uma mensagem de erro

E eu devo ver o pedido na situação "Rascunho"

### Testes de Aceitação de História A.24: Listar Pedidos por Data

# language: pt

**Funcionalidade:** Listar Pedidos por Data

**Para** encontrar pedidos facilmente

**Como** Secretária

**Eu** quero ver uma lista de pedidos cadastrados em um determinado dia

**Contexto:**

**Dado** que um pedido existe

E que eu estou na página de listar pedidos

Cenário: Sucesso

**Quando** eu selecionar uma data que tenha pelo menos um pedido cadastrado

Então eu devo ver uma lista de pedidos

Cenário: Nenhum pedido

**Quando** eu selecionar uma data que não tenha pedidos cadastrados

Então eu não devo ver uma lista de pedidos

### Testes de Aceitação de História A.25: Listar Pedidos por Status

# language: pt

**Funcionalidade:** Listar Pedidos por Status

**Para** encontrar facilmente os pedidos que não foram enviados ou confirmados

**Como** Secretária

**Eu** quero ver uma lista de pedidos que estão em uma determinada situação

**Contexto:**

**Dado** que um rascunho completo de pedido existe

E que um rascunho incompleto de pedido existe  
 E que um pedido aguardando confirmação existe  
 E que um pedido confirmado existe  
 E que um pedido cancelado existe  
 E que um pedido recusado existe  
 E que eu estou na página de listar pedidos

Cenário: Rascunhos

**Quando** eu selecionar a situação "Rascunho"  
 Então eu devo ver somente os rascunhos de pedido

Cenário: Aguardando Confirmação

**Quando** eu selecionar a situação "Aguarda Confirmação"  
 Então eu devo ver somente os pedidos aguardando confirmação

Cenário: Confirmados

**Quando** eu selecionar a situação "Confirmado"  
 Então eu devo ver somente os pedidos confirmados

Cenário: Cancelados

**Quando** eu selecionar a situação "Cancelado"  
 Então eu devo ver somente os pedidos cancelados

Cenário: Recusados

**Quando** eu selecionar a situação "Recusado"  
 Então eu devo ver somente os pedidos recusados

## Testes de Aceitação de História A.26: Listar Pedidos Pendentes

# language: pt

**Funcionalidade:** Listar Pedidos Pendentes

**Para** encontrar facilmente todos os pedidos que não foram enviados ou confirmados

**Como** Secretária

**Eu** quero ver uma lista de pedidos pendentes no Mural

Cenário: Pedido não enviado

**Dado** que um rascunho completo de pedido existe  
**Quando** eu estou na página do Mural  
 Então eu devo ver uma lista de pedidos

Cenário: Pedido aguardando confirmação

**Dado** que um pedido aguardando confirmação existe  
**Quando** eu estou na página do Mural  
 Então eu devo ver uma lista de pedidos

Cenário: Nenhum pedido não confirmado

**Quando** eu estou na página do Mural  
 Então eu não devo ver uma lista de pedidos

### Testes de Aceitação de História A.27: Ver Pedido

# language: pt

**Funcionalidade:** Ver Pedido

**Para** conferir a situação de um pedido antes de executar ações sobre ele

**Como** Secretária

**Eu** quero ver um pedido

Cenário: Rascunho de Pedido

**Dado** que um rascunho completo de pedido existe

**Quando** eu estou na página do pedido

Então eu devo ver o pedido

E eu devo poder enviar o pedido

E eu devo ver o pedido na situação "Rascunho"

Cenário: Pedido Aguardando Confirmação

**Dado** que um pedido aguardando confirmação existe

**Quando** eu estou na página do pedido

Então eu devo ver o pedido

E eu devo poder confirmar o pedido

E eu devo poder recusar o pedido

E eu devo ver o pedido na situação "Aguarda Confirmação"

Cenário: Pedido Confirmado

**Dado** que um pedido confirmado existe

**Quando** eu estou na página do pedido

Então eu devo ver o pedido

E eu devo poder cancelar o pedido

E eu devo ver o pedido na situação "Confirmado"

Cenário: Pedido Cancelado

**Dado** que um pedido cancelado existe

**Quando** eu estou na página do pedido

Então eu devo ver o pedido

E eu devo ver o pedido na situação "Cancelado"

Cenário: Pedido Recusado

**Dado** que um pedido recusado existe

**Quando** eu estou na página do pedido

Então eu devo ver o pedido

E eu devo ver o pedido na situação "Recusado"

---

### Testes de Aceitação de História A.28: Iniciar cadastro de Pedido

# language: pt

**Funcionalidade:** Iniciar cadastro de Pedido

**Para** que eu possa anotar rapidamente os dados mais importantes de um pedido

**Como** Vendedor

**Eu** quero iniciar o cadastro de um pedido

**Contexto:**

**Dado** que um cliente existe  
 E que uma representada existe  
 E que eu estou na página de adicionar pedido

Cenário: Sucesso

**Quando** eu selecionar um cliente  
 E eu selecionar uma representada  
 E eu clicar em "Adicionar Itens >"  
 Então eu devo ver uma mensagem de sucesso  
 E eu devo ver a página de Adicionar Item ao pedido

Cenário: Vendedor esqueceu de selecionar representada

**Quando** eu selecionar um cliente  
 Mas eu esquecer de selecionar uma representada  
 E eu clicar em "Adicionar Itens >"  
 Então eu devo ver uma mensagem de erro  
 E eu devo ver o campo "Representada" com o erro "não pode ficar em branco"

Cenário: Vendedor esqueceu de selecionar cliente

**Quando** eu selecionar uma representada  
 Mas eu esquecer de selecionar um cliente  
 E eu clicar em "Adicionar Itens >"  
 Então eu devo ver uma mensagem de erro  
 E eu devo ver o campo "Cliente" com o erro "não pode ficar em branco"

### Testes de Aceitação de História A.29: Editar Produto

# language: pt

**Funcionalidade:** Editar Produto

**Como** Secretária

**Eu** quero editar um produto

**Contexto:**

**Dado** que pelo menos uma representada esteja cadastrada  
 E que eu estou na página de editar produto

Cenário: Sucesso

**Quando** eu preencher o formulário do produto com dados válidos  
 Então eu devo ver uma mensagem de sucesso  
 E eu devo ver o produto

Cenário: Código duplicado

**Quando** eu preencher o formulário com o código de um produto que já existe  
 Então eu devo ver uma mensagem de erro  
 E eu devo ver o campo "Código" com o erro "já está em uso"

### Testes de Aceitação de História A.30: Adicionar Produto

# language: pt

**Funcionalidade:** Adicionar Produto

**Como** Secretária

**Eu** quero adicionar um produto

**Contexto:**

**Dado** que pelo menos uma representada esteja cadastrada  
E que eu estou na página de adicionar produto

Cenário: Sucesso

**Quando** eu preencher o formulário do produto com dados válidos

Então eu devo ver uma mensagem de sucesso

E eu devo ver o produto

Cenário: Código duplicado

**Quando** eu preencher o formulário com o código de um produto que já existe

Então eu devo ver uma mensagem de erro

E eu devo ver o campo "Código" com o erro "já está em uso"

---

### Testes de Aceitação de História A.31: Ver Produto

# language: pt

**Funcionalidade:** Ver Produto

**Como** Secretária

**Eu** quero ver os dados de um Produto cadastrado

Cenário: Sucesso

**Dado** que eu estou na página de um produto

Então eu devo ver os dados do produto

---

## APÊNDICE B *RESTFUL WEB SERVICES* IMPLEMENTADOS

### B.1 Clientes

Tabela B.1: Modelo de dados do recurso tipo *clientes*

Propriedade	Descrição	Caráter
<i>name</i>	Nome resumido	Obrigatório
<i>company_name</i>	Razão Social	Opcional
<i>cnpj</i>	CNPJ	Obrigatório
<i>email</i>	E-mail	Opcional
<i>phone</i>	Telefone	Opcional
<i>fax</i>	Fax	Opcional
<i>skype</i>	Skype	Opcional
<i>msn</i>	MSN	Opcional
<i>street</i>	Logradouro	Opcional
<i>zip_code</i>	CEP	Opcional
<i>city</i>	Cidade	Opcional
<i>state</i>	Estado	Opcional

Tabela B.2: Serviços oferecidos para o recurso tipo *clientes*

GET	<i>/clientes/id</i>	Recuperação de um cliente.
GET	<i>/clientes</i>	Recuperação de todos clientes.
POST	<i>/clientes</i>	Criação de um cliente.
PUT	<i>/clientes/id</i>	Atualização de um cliente.
DELETE	<i>/clientes/id</i>	Exclusão de um cliente.

## B.2 Pedidos

Tabela B.3: Modelo de dados do recurso tipo *pedidos*

Propriedade	Descrição	Caráter
<i>client_id</i>	Identificador de <i>cliente</i>	Obrigatório
<i>company_id</i>	Identificador de <i>representada</i>	Obrigatório
<i>type_of_freight</i>	Tipo do Frete (CIF ou FOB)	Obrigatório
<i>carrier_id</i>	Identificador de <i>transportadora</i>	Opcional
<i>payment</i>	Forma de Pagamento	Obrigatório
<i>comission</i>	Comissão	Obrigatório
<i>discount</i>	Desconto	Obrigatório
<i>interest</i>	Juros	Obrigatório
<i>observations</i>	Observações	Opcional

Tabela B.4: Serviços oferecidos para o recurso tipo *pedidos*

Método	URI	Descrição
GET	/pedidos/ <b>id</b>	Recuperação de um pedido.
GET	/pedidos/ <b>dia/mês/ano</b>	Recuperação de todos pedidos de um determinado dia.
GET	/pedidos/pendentes	Recuperação de todos pedidos pendentes.
POST	/pedidos	Criação de um pedido.
PUT	/pedidos/ <b>id</b>	Atualização de um pedido.
PUT	/pedidos/ <b>id</b> /enviar	Envio de um pedido.
PUT	/pedidos/ <b>id</b> /confirmar	Confirmação de recebimento de um pedido.
PUT	/pedidos/ <b>id</b> /cancelar	Cancelamento de um pedido.
DELETE	/pedidos/ <b>id</b>	Exclusão de um pedido.

### B.2.1 Itens de Pedido

Tabela B.5: Modelo de dados do recurso tipo *itens de pedido*

Propriedade	Descrição	Caráter
<i>order_id</i>	Identificador de <i>pedido</i>	Obrigatório
<i>product_id</i>	Identificador de <i>produto</i>	Obrigatório
<i>color</i>	Cor	Opcional
<i>unit_price</i>	Valor unitário	Obrigatório
<i>quantity</i>	Quantidade	Obrigatório

Tabela B.6: Serviços oferecidos para o recurso tipo *itens de pedido*

Método	URI	Descrição
GET	/pedidos/ <b>pid</b> /itens/ <b>id</b>	Recuperação de um item de pedido.
GET	/pedidos/ <b>pid</b> /itens	Recuperação de todos itens de um pedido.
POST	/pedidos/ <b>pid</b> /itens	Criação de um item de um pedido.
PUT	/pedidos/ <b>pid</b> /itens/ <b>id</b>	Atualização de um item de um pedido.
DELETE	/pedidos/ <b>pid</b>	Exclusão de um item de um pedido.



### B.3 Produtos

Tabela B.7: Modelo de dados do recurso tipo *produtos*

Propriedade	Descrição	Caráter
<i>company_id</i>	Identificador de representada	Obrigatório
<i>code</i>	Código	Obrigatório
<i>description</i>	Descrição	Opcional

Tabela B.8: Serviços oferecidos para o recurso tipo *produtos*

Método	URI	Descrição
GET	/produtos/ <b>id</b>	Recuperação de um produto.
GET	/produtos	Recuperação de todos produtos.
POST	/produtos	Criação de um produto.
PUT	/produtos/ <b>id</b>	Atualização de um produto.
DELETE	/produtos/ <b>id</b>	Exclusão de um produto.

## B.4 Representadas

Tabela B.9: Modelo de dados do recurso tipo *representadas*

Propriedade	Descrição	Caráter
<i>name</i>	Nome resumido	Obrigatório
<i>company_name</i>	Razão Social	Opcional
<i>cnpj</i>	CNPJ	Opcional
<i>email</i>	E-mail	Obrigatório
<i>phone</i>	Telefone	Opcional
<i>fax</i>	Fax	Opcional
<i>skype</i>	Skype	Opcional
<i>msn</i>	MSN	Opcional
<i>street</i>	Logradouro	Opcional
<i>zip_code</i>	CEP	Opcional
<i>city</i>	Cidade	Opcional
<i>state</i>	Estado	Opcional

Tabela B.10: Serviços oferecidos para o recurso tipo *representadas*

Método	URI	Descrição
GET	/representadas/ <b>id</b>	Recuperação de uma representada.
GET	/representadas	Recuperação de todas representadas.
POST	/representadas	Criação de uma representada.
PUT	/representadas/ <b>id</b>	Atualização de uma representada.
DELETE	/representadas/ <b>id</b>	Exclusão de uma representada.

## B.5 Transportadoras

Tabela B.11: Modelo de dados do recurso tipo *transportadoras*

Propriedade	Descrição	Caráter
<i>name</i>	Nome resumido	Obrigatório

Tabela B.12: Serviços oferecidos para o recurso tipo *transportadoras*

Método	URI	Descrição
GET	/transportadoras/ <b>id</b>	Recuperação de uma transportadora.
GET	/transportadoras	Recuperação de todas transportadoras.
POST	/transportadoras	Criação de uma transportadora.
PUT	/transportadoras/ <b>id</b>	Atualização de uma transportadora.
DELETE	/transportadoras/ <b>id</b>	Exclusão de uma transportadora.