

Universidade Federal de Sergipe
Centro de Ciências Exatas e Tecnologia
DEPARTAMENTO DE COMPUTAÇÃO - DCOMP
CIÊNCIA DA COMPUTAÇÃO

Documentos Desenvolvimento de Software

Prof. Dr. MICHEL DOS SANTOS SOARES

São Cristóvão - SE

Novembro de 2015

Componentes:

CLAUDIO MOTA OLIVEIRA - 201210007817
FELIPE MORAIS ARAGAO - 201320007330
FLORENCIO NATAN DOS SANTOS GAMA - 201210008144
FRANCISCO MOTA CABRAL FILHO - 201210009197
ÍCARO MARLEY OLIVEIRA FRAGA DA SILVA - 201210008538
JOSÉ CAIQUE OLIVEIRA DA SILVA - 201210008126
RENATO SILVEIRA NUNES JUNIOR - 201120001156

Conteúdo

| | | |
|----------|--|-----------|
| 1 | Levantamento de requisitos | 5 |
| 1.1 | Propósito do Documento | 5 |
| 1.2 | Escopo do Produto | 5 |
| 1.3 | Definições e Abreviações | 5 |
| 1.4 | Visão Geral do Restante do Documento | 5 |
| 1.5 | Descrição Geral | 6 |
| 1.5.1 | Perspectiva do Produto | 6 |
| 1.5.2 | Funções do Produto | 6 |
| 1.5.3 | Características do Usuário | 6 |
| 1.5.4 | Restrições Gerais | 6 |
| 1.5.5 | Suposições e Dependências | 6 |
| 1.6 | Requisitos específicos | 7 |
| 1.6.1 | Requisitos Funcionais | 7 |
| 2 | Plano de projeto | 11 |
| 2.1 | Problema | 11 |
| 2.2 | Equipe | 12 |
| 2.3 | Cronograma (Inclui Milestones) | 13 |
| 2.4 | Estimativa de Custo | 15 |
| 3 | Casos de Uso | 17 |
| 4 | Código Projeto | 27 |
| 4.1 | Model | 27 |
| 4.1.1 | Comanda | 27 |
| 4.1.2 | Componentes produtos | 27 |
| 4.1.3 | Fornecedor | 28 |
| 4.1.4 | Funcionario | 28 |
| 4.1.5 | Item | 28 |
| 4.1.6 | Pedido | 29 |
| 4.1.7 | Produto | 29 |
| 4.1.8 | Vende | 30 |
| 4.2 | View | 31 |
| 4.2.1 | Comanda | 31 |
| 4.2.2 | Compra | 37 |
| 4.2.3 | Fornecedor | 44 |
| 4.2.4 | Funcionario | 49 |
| 4.2.5 | Pedido | 54 |
| 4.2.6 | Produto | 57 |
| 4.3 | Controller | 65 |
| 4.3.1 | Aplicação | 65 |
| 4.3.2 | Comanda | 65 |
| 4.3.3 | Compra | 67 |
| 4.3.4 | Fornecedor | 70 |
| 4.3.5 | Funcionario | 71 |
| 4.3.6 | Pedido | 73 |
| 4.3.7 | Produto | 74 |
| 4.4 | Banco de Dados | 77 |

| | | |
|----------|---|-----------|
| 5 | Diagramas | 82 |
| 5.1 | Diagrama de Classe de Análise | 82 |
| 5.2 | Diagrama de Classe de Projeto | 84 |
| 5.3 | Diagrama de Atividade | 85 |
| 5.3.1 | Abertura e fechamento da comanda | 85 |
| 5.3.2 | Consultar informações de pedidos na comanda | 86 |
| 5.3.3 | Inclusão de produto | 87 |
| 5.4 | Diagrama de caso de uso | 89 |
| 5.5 | Diagramas de Sequência | 90 |
| 5.5.1 | Consultar informações de pedidos na comanda | 90 |
| 5.5.2 | Geração de Lista de itens por produto | 91 |
| 5.5.3 | Inclusão de comanda | 92 |
| 5.5.4 | Inclusão de pedido na comanda | 93 |
| 5.5.5 | Inclusão de produto | 94 |

1 Levantamento de requisitos

1.1 Propósito do Documento

Este documento contém a especificação de um sistema para controle e atendimento de um restaurante. Nas próximas seções, será apresentado de forma mais detalhada as características do sistema.

1.2 Escopo do Produto

O sistema destina-se a gerência de restaurantes, abrangendo os seguintes setores: financeiro, estoque, controle de funcionários e atendimento.

1.3 Definições e Abreviações

Definições

Item(ns): Cada mercadoria existente no estoque.

Produto(s): Mercadorias comercializadas pelo restaurante.

Pedidos: Requisição de produtos feitas pelo comanda.

Comanda: Lista de pedidos vinculada a uma mesa.

Comanda Fechada: Comanda que não permite a adesão de pedidos.

Comanda Ativa: Comanda que permite a adesão de pedidos.

Restaurante: Estabelecimento comercial de venda de produtos.

Funcionário: Pessoa física com vínculo empregatício com o restaurante.

Relatório: Apresentação de um conjunto de informações específicas do restaurante.

Entrada de Itens: A aquisição de itens feita pelo restaurante, e inseridos no estoque.

Saída de Itens: A retirada de um item do estoque.

Arrecadação Líquida: Valor arrecadado subtraído dos gastos.

1.3.2 Abreviações

RF: Requisitos Funcionais;

RNF: Requisitos Não Funcionais;

1.4 Referências

SOMMERVILLE, I. Engenharia de Software. Pearson/Prentice Hall

PRESSMAN, R. S. Engenharia de Software. McGraw Hill

1.4 Visão Geral do Restante do Documento

Nas próximas seções serão mostradas as características gerais do sistema e suas funcionalidades. Segue ainda, a descrição de restrições e dependências que devem ser consideradas para o devido funcionamento. Arelado às funcionalidades do sistema, é apresentada também uma lista de requisitos funcionais e não funcionais que servirão como guia para o desenvolvimento do sistema.

1.5 Descrição Geral

1.5.1 Perspectiva do Produto

Espera-se que o sistema seja desenvolvido e implementado durante as disciplinas de Desenvolvimento de Software I, II e III da Universidade Federal de Sergipe, bem como implantá-lo em algum restaurante e verificar se as necessidades do restaurante foram atendidas em todas as partes descritas.

1.5.2 Funções do Produto

O produto permite a gestão de um restaurante, possibilitando o controle de estoque, controle do atendimento, gerenciando pedidos e comandas, e permite também o controle financeiro e de funcionários.

1.5.3 Características do Usuário

Segue abaixo a definição de cada tipo de usuário do sistema:

Cozinheiro: Usuário cuja interação com o sistema é limitada a visualização dos pedidos e a notificação de quando um pedido estiver pronto.

Garçom: Usuário responsável pelo atendimento, solicitação, alteração, exclusão e entrega de pedidos, bem como o fechamento de comandas.

Gerente: Usuário com acesso a todas as funcionalidades do sistema.

1.5.4 Restrições Gerais

Os funcionários que utilizarão o sistema deverão ser treinados para que se tornem aptos para o uso do sistema.

O restaurante deverá possuir dispositivos móveis (Smartphone ou Tablet) com configuração mínima de: processador de 1GHz, memória RAM 1GB, espaço de armazenamento interno de 500 MB e sistema operacional Android 4.2 .

O restaurante deverá possuir uma rede local Wi-fi (IEEE 802.11) disponível exclusivamente para os funcionários.

O restaurante deverá possuir um computador central onde ficarão todos os dados do sistema.

1.5.5 Suposições e Dependências

Faz-se necessário para o funcionamento do software a existência de tablets ou smartphones para que os garçons utilizem no atendimento.

Rede wifi para que os tablets ou smartphones se comuniquem com o sistema do restaurante, e um computador que funcione como servidor do sistema.

1.6 Requisitos específicos

1.6.1 Requisitos Funcionais

1. Inclusão de fornecedores.

O sistema deve efetuar o cadastro dos fornecedores.

1. Alteração de fornecedores.

O sistema deve efetuar a alteração dos dados cadastrais de fornecedores.

1. Exclusão de fornecedores.

O sistema deve efetuar a exclusão de fornecedores.

1. Consulta de fornecedores.

O sistema deve efetuar a consulta dos dados dos fornecedores.

1. Listagem de fornecedores.

O sistema deve efetuar a listagem dos dados dos fornecedores.

1. Geração de relatórios de itens por fornecedor.

O sistema deve efetuar a geração de relatórios de itens por fornecedor.

1. Inclusão dos itens.

O sistema deve efetuar a inclusão dos dados dos itens fornecidos para o restaurante.

1. Alteração de itens.

O sistema deve efetuar a alteração dos dados dos itens fornecidos para o restaurante.

1. Exclusão de itens.

O sistema deve efetuar a exclusão dos itens fornecidos para o restaurante.

1. Consultar de itens.

O sistema deve efetuar a consulta dos dados dos itens.

1. Listagem de itens.

O sistema deve efetuar a listagem de itens por fornecedor.

1. Aviso de quantidade de itens abaixo do limite delimitado.

O sistema deve informar quando a quantidade de um item estiver abaixo do valor mínimo definido pelo gerente.

1. Geração de relatório de Itens em falta.

O sistema deve efetuar geração de relatórios dos itens que estão em falta, ou seja, aqueles que a quantidade é igual a zero.

1. Inclusão de comandas.

O sistema deve efetuar a inclusão de comandas, incluindo a hora de abertura.

1. Associar comanda a um funcionário.

O sistema deve associar uma comanda a um funcionário responsável.

1. Encerramento de comandas.

O sistema deve efetuar o encerramento de comandas, incluindo a hora de encerramento.

1. Alteração de comandas.

O sistema deve efetuar a alteração de comandas.

1. Listagem de comandas ativas.

O sistema deve efetuar a listagem de comandas que não possuem hora de encerramento.

1. Impressão de comanda.

O sistema deve efetuar a impressão da comanda.

1. Consulta informações dos pedidos da Comandas.

O sistema deve permitir a consulta de informações dos pedidos da comanda.

1. Inclusão de pedido de Produto na comanda.

O sistema deve efetuar a inclusão do pedido, incluindo a hora inicial no pedido.

1. Entrega do pedido de Produto.

O sistema deve efetuar inclusão da hora final no pedido.

1. Cancelamento de pedidos.

O sistema deve permitir o cancelamento de pedidos.

1. Informações de pedido.

O sistema deve permitir a consulta das informações dos pedidos.

1. Opções de pagamento.

O sistema deverá informar sobre as opções de pagamento aceitas.

1. Inclusão de produto.

O sistema deve efetuar o cadastro do produto.

1. Alteração de produto.

O sistema deve efetuar a alteração do produto.

1. Exclusão de produto.

O sistema deve efetuar a exclusão do produto.

1. Consulta de produto.

O sistema deve efetuar a consulta de informações do produto.

1. Geração de Lista de Itens por Produto.

O sistema deve efetuar a geração da lista de itens que compoem um produto.

1. Validação do produto a partir dos itens.

O sistema deve efetuar verificação da possibilidade de produção do produto a partir dos itens.

1. Inclusão de Funcionários.

O sistema deve efetuar o cadastro do funcionário.

1. Alteração de Funcionários.

O sistema deve efetuar a alteração do funcionário.

1. Demissão de Funcionários.

O sistema deve efetuar a demissão do funcionário.

1. Consulta de Funcionários.

O sistema deve efetuar consulta de funcionário.

1. Listagem de Funcionários

O sistema deve efetuar listagem dos dados do funcionário.

1. Geração de relatórios das comandas encerradas.

O sistema deve efetuar a geração de relatórios contendo as comandas encerradas do restaurante em um intervalo de tempo, em ordem de dias, definido pelo gerente.

1. Geração de relatórios de despesas.

O Sistema deve efetuar a geração de relatórios contendo as despesas do restaurante em um intervalo de tempo, em ordem de dias, definido pelo gerente.

1. Geração de relatórios da arrecadação líquida.

O Sistema deve efetuar a geração de relatórios contendo a arrecadação líquida do restaurante em um intervalo de tempo, em ordem de dias, definido pelo gerente.

3.2 Requisitos Não-Funcionais

RNF1: O sistema deve retornar as consultas, ou seja, prover a exibição dos dados, relativos aos itens em, no máximo, 6 segundos, em 90% dos casos.

RNF2: O sistema deve retornar as consultas, ou seja, prover a exibição dos dados, relativos aos fornecedores em no máximo 6 segundos, em 90% dos casos.

RNF3: O sistema deve processar a alteração de dados relativos a

fornecedores e itens em, no máximo, 8 segundos, em 90% dos casos.

RNF4: O sistema deve processar a inclusão de dados relativos a fornecedores e itens em, no máximo, 8 segundos, em 90% dos casos.

RNF5: O sistema deve processar a exclusão de dados relativos a fornecedores e itens em, no máximo, 8 segundos, em 90% dos casos.

RNF6: O sistema deve gerar relatórios em, no máximo, 6 segundos, em 90% dos casos.

RNF7: O sistema deve retornar o resultado de uma consulta ao sistema em no máximo 6s em 90% dos casos.

RNF8: A listagem de itens fora da validade referente a RF14 deve ser realizada em no máximo 10 segundos em 90% dos casos.

RNF9: A listagem com os itens referente a RF9 deve ser realizada em no máximo 10 segundos em 90% dos casos.

RNF10: O relatório de despesas do restaurante referente a RF11 deve ser realizada em no máximo 10 segundos em 90% dos casos.

2 Plano de projeto

Devido ao avanço tecnológico e a crescente necessidade de encontrar soluções mais aprimoradas para o gerenciamento de um negócio, empresas buscam cada vez mais por sistemas que auxiliem no controle e gestão de suas atividades.

Esse projeto tem como objetivo a criação de um sistema para gerenciamento de restaurantes, permitindo o controle de diversas funções como: estoque, atendimento e finanças. Esse sistema será integrado em computadores juntamente com o uso de tecnologias móveis de forma a permitir uma maior eficiência tanto no atendimento ao cliente como no controle das funções do restaurante. Devido a essa necessidade, além de computadores e uma rede de computadores, é necessário também que o estabelecimento disponha de dispositivos móveis (tablets), para o uso nas atividades que exigem o dispositivo.

O prazo necessário para conclusão das etapas de desenvolvimento e implantação do projeto foi estimado em 132 dias úteis.

2.1 Problema

As atividades realizadas em um restaurante produzem uma grande quantidade de informações como: pedidos de clientes, compras com fornecedores, contratação de funcionários e gerenciamento de estoque. Devido a isso, surgem diversas dificuldades como:

- Gerência de grande volume de papel oriundo da venda de refeições, da compra de itens com fornecedores e da administração de funcionários;
- Lentidão no atendimento aos clientes, sendo esta uma das principais causas de cancelamento dos pedidos;
- Erros no preparo de pedidos dada a má compreensão do que é anotado em comandas;
- Erros na entrega dos pedidos, onde é feita uma entrega em local errado ou ainda o pedido não corresponde à solicitação desejada pelo cliente.
- Erros de cálculo quando a conta de uma comanda é solicitada;

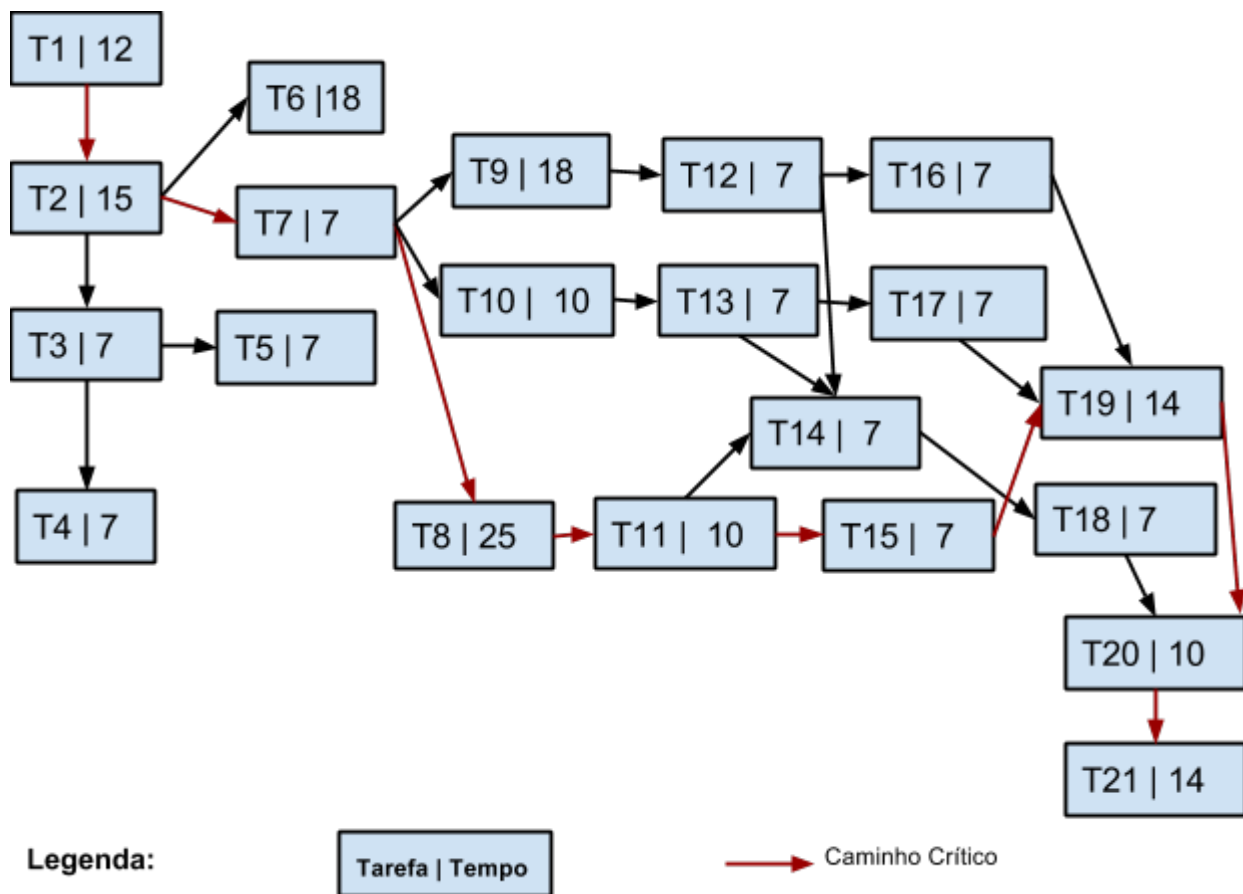
2.2 Equipe

A equipe é formada por um total de 12 pessoas distribuídas entre as diferentes tarefas do sistema.

| Profissional | Quantidade | Horas trabalhadas |
|---------------------------------|------------|-------------------|
| Administrador de Banco de Dados | 1 | 264 |
| Analista de Sistemas | 1 | 688 |
| Analista Programador Junior | 3 | 480 |
| Analista Programador Pleno | 3 | 592 |
| Analista Programador Sênior | 1 | 592 |
| Designer Pleno | 1 | 200 |
| Testador Pleno | 2 | 248 |

2.3 Cronograma (Inclui Milestones)

| Nome | Identificador | Dependências | Duração |
|--|---------------|--------------|---------|
| Levantamento de requisitos | T1 | - | 12 dias |
| Modelagem do software | T2 | T1 | 15 dias |
| Modelagem BD | T3 | T2 | 7 dias |
| Revisão dos requisitos | T4 | T3 | 7 dias |
| Implementação do BD | T5 | T3 | 7 dias |
| Desenvolvimento da interface gráfica | T6 | T2 | 18 dias |
| Implementação de componentes auxiliares | T7 | T2 | 7 dias |
| Implementação do Módulo Gerente | T8 | T7 | 25 dias |
| Implementação do Módulo Garçom | T9 | T7 | 18 dias |
| Implementação do Módulo do Estoquista | T10 | T7 | 10 dias |
| Revisão de requisitos do módulo Gerente | T11 | T8 | 10 dias |
| Revisão de requisitos do módulo Garçom | T12 | T9 | 7 dias |
| Revisão de requisitos do módulo Estoquista | T13 | T10 | 7 dias |
| Revisão de interface gráfica | T14 | T11,T12,T13 | 7 dias |
| Teste do Módulo do Gerente | T15 | T11 | 7 dias |
| Teste do Módulo do Garçom | T16 | T12 | 7 dias |
| Teste do Módulo do Estoquista | T17 | T13 | 7 dias |
| Geração de documentação do usuário | T18 | T14 | 7 dias |
| Integração dos módulos | T19 | T15,T16,T17 | 14 dias |
| Teste do sistema | T20 | T19 | 10 dias |
| Implantação do sistema | T21 | T20 | 14 dias |



A partir do caminho crítico mostrado, o tempo mínimo estimado para o projeto é 132 dias úteis.

2.4 Estimativa de Custo

1

Custo com profissionais

O custo estimado com profissionais é mostrado e detalhado na tabela a seguir.

| Profissional | Valor por hora Bruto (R\$) | Horas Mensais | Salário Mensal Bruto (R\$) | Salário Mensal Líquido (R\$) |
|---------------------------------------|----------------------------|---------------|----------------------------|------------------------------|
| Administrador de Banco de Dados Pleno | 51,4046 | 176 | 9.047,21 | 4.761,69 |
| Analista de Sistema Sênior | 62,0138 | 176 | 10.914,43 | 5.744,44 |
| Analista Programador Júnior | 26,1124 | 176 | 4.595,79 | 2.418,84 |
| Analista Programador Pleno | 41,3600 | 176 | 7.279,37 | 3.831,25 |
| Analista Programador Sênior | 58,1506 | 176 | 10.234,52 | 5.386,59 |
| Designer Pleno | 21,6219 | 176 | 3.805,47 | 2.002,88 |
| Testador Pleno | 32,7917 | 176 | 5.771,34 | 3.037,55 |

- Assumindo que o valor dos impostos acumula 90% sobre o salário bruto de cada funcionário.

Custo com equipamento²

O custo com equipamentos foi estimado de acordo com a quantidade de produtos que serão comprados e os respectivos preços na data de escrita deste documento. Esse custo compreende os seguintes equipamentos: 5 computadores de mesa, 5 computadores portáteis, 1 livro de banco de dados para consulta, 1 livro de programação mobile para consulta, 1 *tablet* para teste do módulo do garçom. Os valores desses equipamentos são detalhados na tabela a seguir.

| Produto | Valor unitário(R\$) | Quantidade | Custo (R\$) |
|---------------------|---------------------|------------|------------------|
| Computador de mesa | 1.500 | 5 | 7.500 |
| Computador portátil | 1.600 | 5 | 8.000 |
| Tablets | 800 | 1 | 800 |
| Livros | 150 | 2 | 300 |
| Total | 4.050,00 | 13 | 16.600,00 |

Custo com treinamento

Para adaptação dos programadores as tecnologias usadas no projeto, o programador sênior fará um treinamento dos analistas programadores júnior durante a fase de levantamento de requisitos e modelagem do projeto. Esse treinamento deve ser feito obrigatoriamente até o fim da tarefa 2 e somente haverá treinamento dos profissionais de nível pleno, caso este profissional seja contratado sem passar pelo mesmo cargo de nível

¹ O custo foi baseado nos valores disponíveis no seguinte endereço: <http://info.abril.com.br/noticias/carreira/2014/02/veja-o-salario-de-180-cargos-em-ti.shtml>

² Valores dos computadores retirados do site da Dell, Kabum, Saraiva

júnior na empresa. Ou seja, não haverá custo extra com o treinamento já que esse profissional que fará o treinamento estará recebendo o seu salário para essa função.

Custo com ambiente

Para o desenvolvimento do sistema, é necessária a aquisição ou aluguel de um imóvel onde a equipe trabalhará. Foi estabelecido um valor de R\$ 900 mensais para o aluguel de uma sala durante o tempo estimado no item 4, que foi de 132 dias. Com isso, o valor total estimado foi de R\$5.400,00.

Custo Total

| Tipo de custo | Custo (R\$) |
|------------------------|--------------------|
| Custo com funcionários | 222.307,7432 |
| Equipamentos | 16.600,00 |
| Ambiente | 5.400,00 |
| Total | 244.307,74 |

3 Casos de Uso

Nome: Cadastrar funcionário.

Descrição: Cadastra um funcionário no sistema.

Identificador: CDU1

Importância: 4

Ator Primário: Gerente.

Fluxo Principal:

| | |
|---|-----------------------|
| Gerente | Sistema |
| 1 - Insere os dados do funcionário no sistema | |
| | 2 - Registra os dados |
| | 3 - Confirma operação |

Nome: Alterar funcionário

Descrição: Altera as informações de um funcionário no sistema.

Identificador: CDU2

Importância: 3

Ator Primário: Gerente.

Pré-condições: O funcionário deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|---|--|
| Gerente | Sistema |
| 1 - Solicita o registro do funcionário ao sistema | |
| | 2a - Retorna o registro do funcionário. 2b - Retorna aviso que o funcionário não se encontra no sistema |
| 3 - Altera os dados do funcionário no sistema | |
| | 4 - Registra os dados do funcionário |
| | 5 - Confirma operação |

Nome: Demitir funcionário

Descrição: Remove cadastro do funcionário associado ao sistema.

Identificador: CDU3

Importância: 3

Ator Primário: Gerente.

Pré-condições: O funcionário deve estar cadastrado no sistema e deve não estar associado a qualquer escala de horário do sistema.

Fluxo Principal:

| | |
|---|---|
| Gerente | Sistema |
| 1 - Solicita o registro do funcionário ao sistema | |
| | 2a - Retorna o registro do funcionário 2b - Retorna aviso que o funcionário não se encontra no sistema |

| | |
|--|-----------------------------------|
| 3 - Solicita remoção do funcionário ao sistema | |
| | 4 - Exclui funcionário do sistema |
| | 5 - Confirma operação |

Nome: Registrar hora extra.

Descrição: Registra as horas extras dos funcionários.

Identificador: CDU4

Importância: 2

Ator Primário: Gerente.

Pré-condições: O funcionário deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|---|---|
| Gerente | Sistema |
| 1 - Solicita o registro do funcionário ao sistema | |
| | 2a - Retorna o registro do funcionário 2b - Retorna aviso que o funcionário não se encontra no sistema |
| 3 - Insere as horas extras no sistema | |
| | 4 - Registra as horas extras do funcionário |
| | 5 - Confirma operação |

Nome: Cadastrar Escala de Horários.

Descrição: Cadastra as escalas de horários dos funcionários.

Identificador: CDU5

Importância: 4

Ator Primário: Gerente.

Pré-condições: O funcionário deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|--|----------------------------------|
| Gerente | Sistema |
| 1 - Informa turno de trabalho | |
| 2 - Solicita escala de horários ao sistema | |
| | 3 - Retorna escala de horários |
| 4 - Aloca funcionário na escala de horário | |
| | 5 - Registra alocação de horário |
| | 6 - Confirma operação |

Nome: Alterar Escala de Horários

Descrição: Altera as escalas de horários dos funcionários.

Identificador: CDU6

Importância: 1

Ator Primário: Gerente.

Pré-condições: O funcionário deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|--|---------------------------------------|
| Gerente | Sistema |
| 1 -Solicita escala de horários ao sistema | |
| | 2 - Retorna escala de horários |
| 3 - Altera a escala de horários do funcionário | |
| | 4 - Registra a nova escala de horário |
| | 5 - Confirma operação |

Nome: Cadastrar desconto.

Descrição: Cadastra descontos.

Identificador: CDU7

Importância: 3

Ator Primário: Gerente.

Pré-condições: Devem existir refeições cadastradas.

Fluxo Principal:

| | |
|-------------------------------------|--------------------------|
| Gerente | Sistema |
| 1 - Consulta refeições | |
| | 2 - Retorna as refeições |
| 3 - Associa refeições a um desconto | |
| | 4 - Registra desconto |
| | 5 - Confirma operação |

Nome: Alterar desconto.

Descrição: O gerente altera o desconto.

Identificador: CDU8

Importância: 3

Ator Primário: Gerente.

Pré-condições: O desconto deve existir.

Fluxo Principal:

| | |
|------------------------------------|----------------------------------|
| Gerente | Sistema |
| 1 - Solicita registro desconto | |
| | 2 - Retorna registro do desconto |
| 3 - Altera informações do desconto | |
| | 5 - Registra desconto |
| | 6 - Confirma operação |

Nome: Excluir desconto.

Descrição: O gerente remove o desconto.

Identificador: CDU9

Importância: 3

Ator Primário: Gerente.

Pré-condições: O desconto deve existir.

Fluxo Principal:

| | |
|-----------------------------------|----------------------------------|
| Gerente | Sistema |
| 1 - Solicita registro do desconto | |
| | 2 - Retorna registro do desconto |
| 3 - Exclui desconto | |
| 4 - Confirma exclusão. | |
| | 5 - Realiza exclusão |
| | 6 - Confirma operação |

Nome: Cadastrar Item.

Descrição: O estoquista cadastra um novo item no sistema.

Identificador: CDU10

Importância: 4

Ator Principal: Estoquista.

Fluxo Principal:

| | |
|--|-----------------------|
| Estoquista | Sistema |
| 1 - Informa ao sistema o item a ser cadastrado | |
| | 2 - Registra cadastro |
| | 3 - Confirma operação |

Nome: Alterar Item.

Descrição: Altera um determinado item no sistema.

Identificador: CDU11

Importância: 3

Ator Principal: Estoquista.

Pré-condições: O Item deve existir.

Fluxo Principal:

| | |
|--------------------------------|---------------------------------|
| Estoquista | Sistema |
| 1 - Consulta o item no sistema | |
| | 2 - Retorna o registro do item. |
| 3 - Realiza alterações | |
| 4 - Confirma alterações | |
| | 5 - Registra alterações |
| | 6 - Confirma operação |

Nome: Excluir Item.

Descrição: O estoquista exclui o item no sistema.

Identificador: CDU12

Importância: 3

Ator Principal: Estoquista.

Pré-condições: O Item deve existir.

Fluxo Principal:

| | |
|-------------------------------|---------------------------------|
| Estoquista | Sistema |
| 1 - Consulta item no sistema. | |
| | 2 - Retorna o registro do item. |
| 3 - Exclui item | |
| 4 - Confirma exclusão. | |
| | 5 - Realiza exclusão |
| | 6 - Confirma operação |

Nome: Consultar Item.

Descrição: O estoquista consulta item no sistema.

Identificador: CDU13

Importância: 3

Ator Principal: Estoquista.

Fluxo Principal:

| | |
|---------------------------------------|--|
| Estoquista | Sistema |
| 1 - Informa ao sistema item escolhido | |
| | 2a - Retorna o registro do item. 2b - Retorna aviso que o item não se encontra no sistema |
| 3 - Visualiza item consultado | |

Nome: Cadastrar fornecedor.

Descrição: Cria o cadastro de um fornecedor no sistema.

Identificador: CDU14

Importância: 5

Ator Primário: Estoquista.

Fluxo Principal:

| | |
|---|-------------------------|
| Estoquista | Sistema |
| 1 - Informa ao sistema dados do fornecedor a ser cadastrado | |
| | 2 - Registra fornecedor |
| | 3 - Confirma operação |

Nome: Alterar fornecedor.

Descrição: Altera informações do cadastro de um fornecedor no sistema.

Identificador: CDU15

Importância: 3

Ator Primário: Estoquista.

Pré- Condições: O fornecedor deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|------------|---------|
| Estoquista | Sistema |
|------------|---------|

| | |
|--|---------------------------------------|
| 1 - Solicita o registro do fornecedor ao sistema | |
| | 2 - Retorna o registro do fornecedor. |
| 3 - Altera os dados do fornecedor no sistema | |
| | 4 - Registra os dados do fornecedor |
| | 5 - Confirma cadastro do fornecedor |

Nome: Excluir fornecedor.

Descrição: Exclui um fornecedor do sistema.

Identificador: CDU16

Importância: 4

Ator Primário: Estoquista.

Pré- Condições: Fornecedor deve estar cadastrado no sistema.

Fluxo Principal:

| | |
|--------------------------------------|---------------------------------------|
| Estoquista | Sistema |
| 1 - Consulta o fornecedor no sistema | |
| | 2 - Retorna o registro do fornecedor. |
| 3 - Exclui o fornecedor | |
| 4 - Confirma exclusão | |
| | 5 - Realiza exclusão |
| | 6 - Confirma operação |

Nome: Consultar fornecedor.

Descrição: Faz uma consulta nos fornecedores cadastrados no sistema.

Identificador: CDU17

Importância: 3

Ator Primário: Estoquista.

Fluxo Principal:

| | |
|---|--|
| Estoquista | Sistema |
| 1 - Informa ao sistema o fornecedor a ser consultado. | |
| | 2a - Retorna o registro do fornecedor. 2b - Retorna aviso que o fornecedor não se encontra no sistema |
| 3 - Visualiza a consulta. | |

Nome: Consultar refeição.

Descrição: O Garçom consulta no sistema as refeições disponíveis.

Identificador: CDU18

Importância: 4

Ator Principal: Garçom.

Fluxo Principal:

| | |
|--------|---------|
| Garçom | Sistema |
|--------|---------|

| | |
|--|--|
| 1 - Informa ao sistema a refeição a ser consultada | |
| | 2a - Retorna o registro da refeição. 2b - Retorna um aviso que a refeição não está cadastrada no sistema. |
| 3- Visualiza a refeição | |

Nome: Cadastrar refeição.

Descrição: Cadastra uma refeição no sistema.

Identificador: CDU19

Importância: 5

Ator Primário: Gerente.

Fluxo Principal:

| | |
|---|-----------------------|
| Gerente | Sistema |
| 1 - Informa ao sistema a refeição a ser cadastrada. | |
| | 2 - Registra refeição |
| | 3 - Confirma operação |

Nome: Alterar refeição.

Descrição: Altera o cadastro de uma refeição no sistema.

Identificador: CDU20

Importância: 4

Ator Primário: Gerente.

Pré- Condições: A refeição deve estar cadastrada no sistema.

Fluxo Principal:

| | |
|----------------------------------|-------------------------------------|
| Gerente | Sistema |
| 1 - Consulta refeição no sistema | |
| | 2 - Retorna o registro da refeição. |
| 3 - Realiza alterações | |
| 4 - Confirma alterações | |
| | 5 - Registra alterações |
| | 6 - Confirma operação |

Nome: Excluir refeição.

Descrição: Exclui o cadastro de uma refeição no sistema.

Identificador: CDU21

Importância: 3

Ator Primário: Gerente.

Pré- Condições: A refeição deve estar cadastrada no sistema.

Fluxo Principal:

| | |
|---------|---------|
| Gerente | Sistema |
|---------|---------|

| | |
|----------------------------------|-------------------------------------|
| 1 - Consulta refeição no sistema | |
| | 2 - Retorna o registro da refeição. |
| 3 - Exclui refeição | |
| 4 - Confirma exclusão | |
| | 5 - Realiza exclusão |
| | 6 - Confirma operação |

Nome: Incluir Pedido na Comanda.

Descrição: O Garçom inclui o pedido do Cliente na comanda referente à mesa deste Cliente.

Identificador: CDU22

Importância: 5

Ator Principal: Garçom

Pré- Condições: Comanda da mesa deve ter sido incluída.

Fluxo Principal:

| | |
|----------------------------------|--------------------------------|
| Garçom | Sistema |
| 1 - Consulta comanda no sistema. | |
| | 2 - Retorna comanda consultada |
| 3 - Insere o pedido no sistema. | |
| | 4 - Registra o pedido |
| 5 - Inclui o pedido na comanda. | |
| | 6 - Realiza Inclusão |
| | 7 - Confirma inclusão |

Nome: Excluir pedido da comanda.

Descrição: O Garçom remove o pedido da comanda mediante solicitação do Cliente.

Identificador: CDU23

Importância: 3

Ator Principal: Garçom.

Pré- Condições: O pedido deve ter sido incluído na comanda.

Fluxo Principal:

| | |
|---|----------------------------------|
| Garçom | Sistema |
| 1- Consulta a comanda no sistema. | |
| | 2 - Retorna a comanda consultada |
| 3- Solicita pedido a ser removido da comanda. | |
| | 4 - Retorna pedido |
| 5 - Exclui pedido da comanda | |
| | 6 - Realiza Exclusão |
| | 7 - Confirma Exclusão |

Nome: Incluir comanda

Descrição: O Garçom cria uma comanda ativa associada a um cliente.

Identificador: CDU24

Importância: 5

Ator Principal: Garçom.

Fluxo Principal:

| | |
|--|------------------------------|
| Garçom | Sistema |
| 1- Solicita ao sistema o início de uma nova comanda. | |
| | 2 - Inclui uma nova comanda. |

Nome: Pagar comanda.

Descrição: É efetuado o pagamento da comanda.

Identificador: CDU25

Importância: 5

Ator Principal: Garçom.

Pré- Condições: A comanda deve ter sido incluída e estar ativa.

Fluxo Principal:

| | |
|--|--------------------------|
| Garçom | Sistema |
| 1- Consulta a comanda no sistema. | |
| | 2 - Retorna comanda |
| 3 - Solicita finalização da comanda. | |
| | 4 - Finaliza comanda |
| | 5 - Imprime a conta |
| 6 - Informa ao sistema a forma de pagamento escolhida pelo cliente | |
| | 7 - Registra pagamento |
| | 8 - Imprime cupom fiscal |

Nome: Consultar comanda.

Descrição: O Garçom consulta no sistema por uma comanda de uma mesa específica.

Identificador: CDU26

Importância: 1

Ator Principal: Garçom.

Fluxo Principal:

| | |
|--|---|
| Garçom | Sistema |
| 1- Informa ao sistema qual mesa deseja consultar | |
| | 2 - Retorna as informações da mesa desejada |
| 3 - Recebe a informação solicitada ao sistema | |

Nome: Gerar Comandas.

Descrição: O garçom solicita a geração de comandas.

Identificador: CDU27

Importância: 5

Ator Principal: Garçom.

Fluxo Principal:

| | |
|---|-------------------|
| Garçom | Sistema |
| 1- Solicita a geração de comanda no sistema | |
| | 2- Gera a comanda |

Nome: Gerar Relatório

Descrição: O funcionário gera os relatórios .

Identificador: CDU28

Importância: 4

Ator Principal: Funcionário.

Fluxo Principal:

| Funcionário | Sistema |
|-------------------------------------|----------------------------------|
| 1 - Solicita os tipos de relatórios | |
| | 2 - Informa opções de relatórios |
| 3 - Seleciona o relatório desejado | |
| | 4 - Gera o relatório |

4 Código Projeto

4.1 Model

4.1.1 Comanda

```
1 class Comanda < ActiveRecord::Base
2   belongs_to :funcionario, primary_key: "idFuncionario", foreign_key: "idFuncionario"
3   has_many :pedidos, class_name: 'Pedido', primary_key: "idcomanda", foreign_key: "idComanda"
4
5   has_many :produtos, through: :pedidos
6   validates_presence_of :idFuncionario, :on => :create, :message => "O campo funcionario
7     n o pode ser vazio"
8   validates_presence_of :mesa, :on => :create, :message => "O campo mesa n o pode ser vazio"
9
10  validates_presence_of :dataAbertura, :on => :create, :message => "A data de abertura da
    comanda deve ser informada"
11
12    self.primary_key = :idcomanda
13 end
```

4.1.2 Componentes produtos

```
1 class ComponentesProduto < ActiveRecord::Base
2   belongs_to :item, primary_key: "idItem", foreign_key: "idItem"
3   belongs_to :produto, primary_key: "idProduto", foreign_key: "idProduto"
4   validates_presence_of :idProduto, :on => :create, :message => "O campo produto deve ser
5     informado"
6   validates_presence_of :idItem, :on => :create, :message => "O campo Item deve ser
7     informado"
8   validates_presence_of :quantidade, :on => :create, :message => "A campo quantidade deve
9     ser informada"
10 end
```

4.1.3 Fornecedor

```
1 class Fornecedor < ActiveRecord::Base
2   has_many :vende, foreign_key: "idFornecedor"
3   has_many :items, ->{uniq}, through: :vende
4
5   validates_presence_of :nome, :on => :create, :message => "0 campo nome deve ser informado"
6   validates_presence_of :cnpj, :on => :create, :message => "0 campo CNPJ deve ser informado"
7   validates_presence_of :cidade, :on => :create, :message => "0 campo cidade deve ser
8     informado"
9   validates_presence_of :estado, :on => :create, :message => "0 campo estado deve ser
10     informado"
11   validates_presence_of :telefone, :on => :create, :message => "0 campo telefone deve ser
12     informado"
13 end
```

4.1.4 Funcionario

```
1 class Funcionario < ActiveRecord::Base
2   has_many :comandas, primary_key: "idFuncionario", foreign_key: "idFuncionario"
3   validates_presence_of :nome, :on => :create, :message => "0 campo nome deve ser informado"
4   validates_presence_of :cpf, :on => :create, :message => "0 campo cpf deve ser informado"
5   validates_presence_of :cidade, :on => :create, :message => "0 campo cidade deve ser
6     informado"
7   validates_presence_of :estado, :on => :create, :message => "0 campo estado deve ser
8     informado"
9   validates_presence_of :num, :on => :create, :message => "0 campo num deve ser informado"
10  validates_presence_of :endereco, :on => :create, :message => "0 campo endere o deve ser
11    informado"
12  validates_presence_of :telefone, :on => :create, :message => "0 campo telefone deve ser
13    informado"
14  validates_presence_of :salario, :on => :create, :message => "0 campo salario deve ser
15    informado"
16  validates_presence_of :dataContratacao, :on => :create, :message => "0 campo data de
17    contrata o deve ser informado"
18  validates_presence_of :ativo, :on => :create, :message => "0 campo ativo deve ser
19    informado"
20  validates_presence_of :turno, :on => :create, :message => "0 campo turno deve ser
21    informado"
22  validates_presence_of :login, :on => :create, :message => "0 campo login deve ser
23    informado"
24  validates_presence_of :nivelAcesso, :on => :create, :message => "0 campo nivel de acesso
25    deve ser informado"
26
27  has_secure_password
28
29  validates_presence_of :password, :on => :create, :message => "0 campo senha deve ser
30    informado"
31  validates_presence_of :password_digest, :on => :create, :message => "0 campo senha deve
32    ser informado"
33  validates_presence_of :password_confirmation, :on => :create, :message => "0 campo
34    confirma o de senha deve ser informado"
35
36  validates_presence_of :password, :within => 6..20, :on => :create, :message => "0
37    campo senha deve ter entre 6 e 20 caracteres"
38  validates_presence_of :password_digest, :within => 6..20, :on => :create, :message =>
39    "0 campo senha deve ter entre 6 e 20 caracteres"
40  validates_length_of :password_confirmation, :within => 6..20, :on => :create, :message =>
41    "0 campo confirma o de senha deve entre 6 e 20 caracteres"
42 end
```

4.1.5 Item

```

1 class Item < ActiveRecord::Base
2   has_many :componentes_produtos, primary_key: "idItem", foreign_key: "idItem"
3   has_many :produtos, through: :componentes_produtos
4   has_many :vende
5
6   validates_presence_of :nome, :on => :create, :message => "0 campo nome deve ser informado"
7   validates_presence_of :marca, :on => :create, :message => "0 campo marca deve ser
8     informado"
9   validates_presence_of :descricao, :on => :create, :message => "0 campo descricao deve ser
10     informado"
11   validates_presence_of :quantidade, :on => :create, :message => "0 campo quantidade deve
12     ser informado"
13   validates_presence_of :quantidadeMinima, :on => :create, :message => "0 campo quantidade
14     minima deve ser preenchido"
15 end

```

4.1.6 Pedido

```

1 class Pedido < ActiveRecord::Base
2   belongs_to :comanda, primary_key: "idcomanda", foreign_key: "idcomanda"
3   belongs_to :produto, primary_key: "idProduto", foreign_key: "idProduto"
4
5   validates_presence_of :idFuncionario, :on => :create, :message => "0 campo funcionario
6     deve ser informado"
7   validates_presence_of :idProduto, :on => :create, :message => "0 campo produtos deve ser
8     informado"
9   validates_presence_of :idcomanda, :on => :create, :message => "0 campo comanda deve ser
10     informado"
11   validates_presence_of :dataPedido, :on => :create, :message => "0 campo data do pedido
12     deve ser informado"
13
14   def is_finalizado
15     return self.dataFinalizacao != nil
16   end
17 end

```

4.1.7 Produto

```

1 class Produto < ActiveRecord::Base
2   has_many :pedidos, primary_key: "idPedido", foreign_key: "idPedido"
3   has_many :comandas, through: :pedidos
4   has_many :componentes_produtos, primary_key: "idProduto", foreign_key: "idProduto", :
5     dependent => :delete_all
6   has_many :items, through: :componentes_produtos
7
8   validates_presence_of :nome, :on => :create, :message => "0 campo nome deve ser informado"
9   validates_presence_of :valor, :on => :create, :message => "0 campo valor deve ser
10     informado"
11   validates_presence_of :descricao, :on => :create, :message => "0 campo descricao deve ser
12     informado"
13
14   def quantidade
15     componentes = self.componentes_produtos
16
17     menor_quantidade = -1;
18     componentes.each do |componente|
19       if(menor_quantidade == -1)
20         menor_quantidade = componente.item.quantidade / componente.quantidade
21       else
22         quantidade_temporaria = componente.item.quantidade / componente.quantidade
23         menor_quantidade = menor_quantidade > quantidade_temporaria ? quantidade_temporaria
24           : menor_quantidade;
25       end
26     end
27   end
28 end

```

```
23     return menor_quantidade > -1 ? menor_quantidade : 0
24   end
25
26 end
```

4.1.8 Vende

```
1 class Vende < ActiveRecord::Base
2   belongs_to :fornecedor, foreign_key: :idFornecedor
3   belongs_to :item, foreign_key: :idItem
4
5   validates_presence_of :idFornecedor, :on => :create, :message => "0 campo fornecedor deve
   ser informado"
6   validates_presence_of :idItem, :on => :create, :message => "0 campo item deve ser
   informado"
7   validates_presence_of :data, :on => :create, :message => "0 campo data deve ser informado"
8   validates_presence_of :valor, :on => :create, :message => "0 campo valor deve ser
   informado"
9
10   self.primary_key = :idVende, :idFornecedor, :idItem
11 end
```

4.2 View

4.2.1 Comanda

4.2.1.1 Form

```
1 <%= form_for(@comanda) do |f| %>
2   <% if @comanda.errors.any? %>
3     <div id="error_explanation">
4       <h2><%= pluralize(@comanda.errors.count, "error") %> prohibited this comanda from
        being saved:</h2>
5
6       <ul>
7         <% @comanda.errors.full_messages.each do |message| %>
8           <li><%= message %></li>
9         <% end %>
10      </ul>
11    </div>
12  <% end %>
13
14  <div class="field">
15    <%= f.label :idcomanda %><br>
16    <%= f.number_field :idcomanda %>
17  </div>
18  <div class="field">
19    <%= f.label :idFuncionario %><br>
20    <%= f.number_field :idFuncionario %>
21  </div>
22  <div class="field">
23    <%= f.label :mesa %><br>
24    <%= f.number_field :mesa %>
25  </div>
26  <div class="field">
27    <%= f.label :dataAbertura %><br>
28    <%= f.datetime_select :dataAbertura %>
29  </div>
30  <div class="field">
31    <%= f.label :dataEncerramento %><br>
32    <%= f.datetime_select :dataEncerramento %>
33  </div>
34  <div class="field">
35    <%= f.label :valorFinal %><br>
36    <%= f.text_field :valorFinal %>
37  </div>
38  <div class="actions">
39    <%= f.submit %>
40  </div>
41 <% end %>
```

4.2.1.2 Editar

```
1 <h1>Editing Comanda</h1>
2
3 <%= render 'form' %>
4
5 <%= link_to 'Show', @comanda %> |
6 <%= link_to 'Back', comandas_path %>
```

4.2.1.3 Index

```
1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Comandas Ativas</span>
5     </div>
6   </div>
7   <div class="actions">
8     <a class="btn green" data-toggle="modal" href="#nova-comanda">
9       <i class="fa fa-plus-circle"></i> Nova Comanda
```

```

10     </a>
11 </div>
12 </div>
13 <div class="portlet-body">
14     <div class="row">
15
16         <% @comandas.each do |comanda|
17             if comanda.dataEncerramento == nil
18                 %>
19                     <div class="col-lg-3 col-md-3 col-sm-6 col-xs-12">
20                         <div class="dashboard-stat blue">
21                             <div class="visual">
22                                 <i class="fa fa-cutlery"></i>
23                             </div>
24                             <div class="details">
25                                 <div class="number">
26                                     <span data-counter="counterup" data-value="<%= comanda.mesa %>"><%=
27                                         comanda.mesa %></span>
28                                 </div>
29                                 <div class="desc"> Mesa</div>
30                             </div>
31                             <a class="more" href="/comandas/<%= comanda.idcomanda %>"> Mais Informa es
32                                 <i class="m-icon-swapright m-icon-white"></i>
33                             </a>
34                         </div>
35                     </div>
36                 <% end
37                 end%>
38
39             </div>
40         </div>
41     </div>
42
43 <div class="portlet light">
44     <div class="portlet-title">
45         <div class="caption font-blue-sunglo">
46             <span class="caption-subject bold uppercase">Comandas Encerradas</span>
47         </div>
48     </div>
49     <div class="portlet-body">
50         <div class="row">
51
52             <% @comandas.each do |comanda|
53                 if comanda.dataEncerramento != nil
54                     %>
55                         <div class="col-lg-3 col-md-3 col-sm-6 col-xs-12">
56                             <div class="dashboard-stat blue">
57                                 <div class="visual">
58                                     <i class="fa fa-cutlery"></i>
59                                 </div>
60                                 <div class="details">
61                                     <div class="number">
62                                         <span data-counter="counterup" data-value="<%= comanda.mesa %>"><%=
63                                             comanda.mesa %></span>
64                                     </div>
65                                     <div class="desc"> Mesa</div>
66                                 </div>
67                                 <a class="more" href="/comandas/<%= comanda.idcomanda %>"> Mais
68                                     Informa es
69                                     <i class="m-icon-swapright m-icon-white"></i>
70                                 </a>
71                             </div>
72                         </div>
73                     <% end
74                     end%>

```



```

75     </div>
76   </div>
77 </div>
78
79 <div id="nova-comanda" class="modal fade" tabindex="-1" data-width="760" style="display:
    none; width: 760px; margin-left: -379px; margin-top: 0px;" aria-hidden="true">
80
81   <%= form_tag('/comandas/create', method: "post", class:'form-horizontal') do |f| %>
82     <div class="modal-header">
83       <button type="button" class="close" data-dismiss="modal" aria-hidden="true"></button>
84       <h4 class="modal-title">Cadastrar Comanda</h4>
85     </div>
86     <div class="modal-body">
87       <div class="row">
88         <div class="form-body">
89           <div class="col-md-12">
90             <div class="form-group">
91               <label class="col-md-3 control-label">Mesa</label>
92               <div class="col-md-9">
93                 <input name="comanda[mesa]" class="form-control"/>
94               </div>
95             </div>
96           </div>
97         </div>
98       </div>
99     </div>
100     <div class="modal-footer">
101       <input type="submit" class="btn blue" value="Cadastrar"/>
102     </div>
103   <%end%>
104 </div>

```

4.2.1.4 Index json

```

1 json.array!(@comandas) do |comanda|
2   json.extract! comanda, :idcomanda, :idFuncionario, :mesa, :dataAbertura, :dataEncerramento
3   , :valorFinal
4   json.url comanda_url(comanda, format: :json)
5 end

```

4.2.1.5 Novo

```

1 <h1>New Comanda</h1>
2
3 <%= render 'form' %>
4
5 <%= link_to 'Back', comandas_path %>

```

4.2.1.6 Visualizar

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-red-sunglo">
4       <span class="caption-subject bold uppercase">Comanda #<%= @comanda.idcomanda %></span>
5     </div>
6     <div class="actions">
7       <a class="btn red" data-toggle="modal" href="#encerrar">
8         <i class="fa fa-times"></i>Encerrar
9       </a>
10    </div>
11  </div>
12  <div class="portlet-body">
13    <form class="form-horizontal" role="form">
14      <div class="form-body">
15
16        <div class="row">

```

```

17     <div class="col-md-6">
18         <div class="form-group">
19             <label class="col-md-3 control-label">Mesa</label>
20
21             <div class="col-md-9">
22                 <input type="text" readonly class="form-control" value="%= @comanda.mesa
23                     %>"/>
24             </div>
25         </div>
26         <div class="form-group">
27             <label class="col-md-3 control-label">Funcion rio</label>
28
29             <div class="col-md-9">
30                 <input type="text" readonly class="form-control" value="%= @comanda.
31                     funcionario.nome %>"/>
32             </div>
33         </div>
34     </div>
35     <div class="col-md-6">
36         <div class="form-group">
37             <label class="col-md-3 control-label">Abertura</label>
38
39             <div class="col-md-9">
40                 <div class="input-icon">
41                     <i class="fa fa-clock-o"></i>
42                     <input readonly type="text" class="form-control timepicker timepicker-
43                         default" value="%= @comanda.dataAbertura %>"/>
44                 </div>
45             </div>
46         </div>
47         <div class="form-group">
48             <label class="col-md-3 control-label">Encerramento</label>
49
50             <div class="col-md-9">
51                 <div class="input-icon">
52                     <i class="fa fa-clock-o"></i>
53                     <input readonly type="text" class="form-control timepicker timepicker-
54                         default" value="%= @comanda.dataEncerramento %>"/>
55                 </div>
56             </div>
57         </div>
58     </div>
59 </div>
60 <div class="portlet-title">
61     <div class="caption">
62         <i class="fa fa-coffee font-red"></i>
63         <span class="caption-subject font-red bold uppercase">Pedidos (<%= @comanda.pedidos.
64             length %>)</span>
65     </div>
66     <div class="actions">
67         <a class="btn blue" data-toggle="modal" href="#novo-pedido">
68             <i class="fa fa-plus-circle"></i> Novo Pedido
69         </a>
70     </div>
71 </div>
72 <div class="portlet-body">
73     <% if @pedido.errors.any? %>
74     <div class="alert alert-danger display-hide" style="display: block;">
75         <ul>
76             <% @pedido.errors.full_messages.each do |message| %>
77                 <li><%= message %></li>
78             <% end %>
79         </ul>
80     </div>

```

```

80 <% end %>
81 <div class="table-scrollable">
82   <table class="table table-hover">
83     <thead>
84       <tr>
85         <th class="col-md-1"> #</th>
86         <th class="col-md-7"> Nome</th>
87         <th class="col-md-1"> Pre o</th>
88         <th class="col-md-1"> Situa o</th>
89         <th class="col-md-2"> A o</th>
90       </tr>
91     </thead>
92     <tbody>
93       <%
94         i = 0
95         total = 0
96         @comanda.pedidos.each do |pedido|
97           situacao = pedido.cancelado == true ? 2 : (pedido.dataFinalizacao != nil ? 1
98             : 0)
99           labelText = situacao == 2 ? "Cancelado" : (situacao == 0 ? "Em Espera" : "
100             Finalizado")
101           labelSituacao = situacao == 2 ? "danger" : (situacao == 0 ? "warning" : "
102             success")
103           i = i+1
104           if situacao != 2
105             total += pedido.produto.valor
106           end
107         %>
108       <tr>
109         <td> <%= i %></td>
110         <td> <%= pedido.produto.nome %></td>
111         <td> <%= situacao == 2 ? "-" : "R$ "+pedido.produto.valor.to_s %></td>
112         <td> <span class="label label-sm label-<%= labelSituacao %>"><%= labelText
113           %></span></td>
114         <td>
115           <%if situacao != 2%>
116             <a href="/comandas/<%= pedido.id %>/finalizar_pedido" class="label
117               label-sm label-success">Finalizar</a>
118             <a href="/comandas/<%= pedido.id %>/cancelar_pedido" class="label
119               label-sm label-danger">Cancelar</a>
120           <% else %>
121             -
122           <% end %>
123         </td>
124       </tr>
125     </tbody>
126   </table>
127 </div>
128
129 <div id="novo-pedido" class="modal fade" tabindex="-1" data-width="760" style="display:
130   none; width: 760px; margin-left: -379px; margin-top: 0px;" aria-hidden="true">
131
132   <%= form_tag(new_pedido_comanda_url, method: "post", class:'form-horizontal') do |f|
133     %>
134     <div class="modal-header">
135       <button type="button" class="close" data-dismiss="modal" aria-hidden="true"></
136         button>
137       <h4 class="modal-title">Novo Pedido</h4>
138     </div>
139     <div class="modal-body">

```

```

139     <div class="row">
140       <div class="form-body">
141         <div class="col-md-12">
142           <div class="form-group">
143             <label class="col-md-3 control-label">Produto</label>
144             <div class="col-md-9">
145               <select type="text" name="pedido[idProduto]" class="form-control">
146                 <% produtos_array = Produto.all.map { |produto| [produto.nome.to_s()
147                   + ' ('+produto.quantidade.to_s()+')', produto.idProduto] } %>
148                 <%= options_for_select(produtos_array) %>
149               </select>
150             </div>
151           </div>
152         </div>
153       </div>
154     </div>
155     <div class="modal-footer">
156       <input type="submit" class="btn green" value="Adicionar"/>
157     </div>
158   <%end%>
159 </div>
160
161 <div id="encerrar" class="modal fade" tabindex="-1" data-width="760" style="display:
162   none; width: 760px; margin-left: -379px; margin-top: 0px;" aria-hidden="true">
163
164   <%= form_tag(encerrar_comanda_path, method: "post", class:'form-horizontal') do |f| %>
165   <div class="modal-header">
166     <button type="button" class="close" data-dismiss="modal" aria-hidden="true"></button>
167
168     <h4 class="modal-title">Encerrar Comanda</h4>
169   </div>
170   <div class="modal-body">
171     <div class="row">
172       <div class="form-body">
173         <div class="col-md-12">
174           <div class="form-group">
175             <label class="col-md-3 control-label">Valor Consumido</label>
176             <div class="col-md-9">
177               <input type="text" min="0" step="0.05" id="comanda-valor-inicial" readonly
178                 class="form-control" value="<%= total %>" />
179             </div>
180           </div>
181         </div>
182       <div class="form-group">
183         <label class="col-md-3 control-label">Descontos</label>
184         <div class="col-md-9">
185           <input type="number" min="0" step="0.05" id="comanda-descontos" class="
186             form-control" value="0" />
187         </div>
188       </div>
189     </div>
190     <div class="form-group">
191       <label class="col-md-3 control-label">Gar om</label>
192       <div class="col-md-9">
193         <input type="number" min="0" step="0.05" id="comanda-garcom" class="form-
194           control" value="0" />
195       </div>
196     </div>
197     <div class="form-group">
198       <label class="col-md-3 control-label">Valor Pago</label>
199       <div class="col-md-9">
200         <input type="number" min="0" step="0.05" id="comanda-valor-final" readonly
201           name="comanda[valorFinal]" class="form-control" value="<%= total
202             %>" />
203       </div>
204     </div>
205   </div>
206 </div>

```

```

199     </div>
200     <div class="modal-footer">
201       <input type="submit" class="btn red" value="Encerrar"/>
202     </div>
203     <%end%>
204   </div>
205 </div>
206 </div>

```

4.2.1.7 Visualizar json

```

1 json.extract! @comanda, :idcomanda, :idFuncionario, :mesa, :dataAbertura, :dataEncerramento,
  :valorFinal, :created_at, :updated_at

```

4.2.2 Compra

4.2.2.1 Form

```

1 <%I18n.locale = "br"%>
2 <p id="notice"><%= notice %></p>
3 <%= form_for(@vende , url: vende_show_path(@vende.idVende, @vende.idFornecedor, @vende.
  idItem)) do |f| %>
4   <% if @vende.errors.any? %>
5     <div id="error_explanation">
6       <h2><%= pluralize(@vende.errors.count, "erro") %> impediram o sucesso da operacao:</h2>
7       <
8       <ul>
9         <% @vende.errors.full_messages.each do |message| %>
10          <li><%= message %></li>
11        <% end %>
12      </ul>
13    </div>
14    <% end %>
15    <%if(@print != nil)%>
16      <%=@print%>
17    <% end %>
18    <div class="field">
19      <%=Fornecedor%><br>
20      <%@sid = -1%>
21      <%=select("selecao", "fornecedor", @fornecedores.collect{|fornecedor| [fornecedor.
        nome, @sid = @sid+1]}, {include_blank: false})%>
22    </div>
23    <div class="field">
24      <%=Item%><br>
25      <%@sid = -1%>
26      <%=select("selecao", "item", @item.collect{|item| [item.nome, @sid = @sid+1]}, {
        include_blank: false})%>
27    </div>
28    <div class="field">
29      <%= f.label :data %><br>
30      <%= f.date_select :data%>
31    </div>
32    <br>
33    <% if(not @modo_edicao) %>
34      <div class="field">
35        <%= "Quantidade de itens"%><br>
36        <%= number_field("numero", "item", options = {min: 1, :class => "form-control input-
          small"}) %>
37      </div>
38    <% end%>
39    <div class="field">
40      <%= f.label :valor %><br>
41      <%= f.number_field :valor ,:class => "form-control input-small"%>
42    </div>
43    <br>
44    <div class="actions">
45      <%= f.submit "Salvar", :class => "btn green" %>

```

```

46 </div>
47 <% end %>

```

4.2.2.2 Item

```

1 <%= form_for(@item, url: item_show_path(@item)) do |f| %>
2 <% if @item.errors.any? %>
3   <div id="error_explanation">
4     <h2><%= pluralize(@item.errors.count, "erro") %> impediram o sucesso da operacao:</h2>
5
6     <ul>
7       <% @item.errors.full_messages.each do |message| %>
8         <li><%= message %></li>
9       <% end %>
10    </ul>
11  </div>
12 <% end %>
13 <div class="field">
14   <%= f.label :nome %><br>
15   <%= f.text_field :nome, :class => "form-control input-medium" %>
16 </div>
17 <div class="field">
18   <%= f.label :marca %><br>
19   <%= f.text_field :marca, :class => "form-control input-medium" %>
20 </div>
21 <div class="field">
22   <%= f.label :descricao %><br>
23   <%= f.text_field :descricao, :class => "form-control input-medium" %>
24 </div>
25 <div class="field">
26   <%= f.label :quantidade %><br>
27   <%= f.number_field :quantidade, :class => "form-control input-small" %>
28 </div>
29 <div class="field">
30   <%= f.label "Quantidade M nima" %><br>
31   <%= f.number_field :quantidadeMinima, :class => "form-control input-small"%>
32 </div>
33 <br>
34 <div class="actions">
35   <%= f.submit "Salvar", :class => 'btn green' %>
36 </div>
37 <% end %>

```

4.2.2.3 Tabela Itens

```

1 <table class="table table-hover">
2   <thead>
3     <tr>
4       <th>Nome</th>
5       <th>Marca</th>
6       <th>Quantidade</th>
7       <th>Alerta</th>
8       <th>Opções</th>
9       <th colspan="3"></th>
10    </tr>
11  </thead>
12  <br>
13  <tbody>
14
15    <% @items.each do |item| %>
16
17      <tr>
18        <td><%= item.nome %></td>
19        <td><%= item.marca %></td>
20        <td><%= item.quantidade %></td>
21        <td align = "center">

```

```

22         <% if(item.quantidade < item.quantidadeMinima) %><span class="glyphicon
           glyphicon-exclamation-sign" title="A quantidade de itens esta baixa
           "></span><% end %>
23     </td>
24     <td align = "right"> <a href = <%= item_show_path(item)%>> <span class="glyphicon
           glyphicon-log-in" title="Detalhes"> </a></td>
25     <td align = "center"> <a href = <%= item_edit_path(item)%>> <span class="glyphicon
           glyphicon-edit" title="Modificar"> </a></td>
26     <td align = "center"> <a href = <%= item_delete_path(item)%>> <span class="glyphicon
           glyphicon-remove" title="Remover"> </a></td>
27 </tr>
28 <% end %>
29 </tbody>
30 </table>

```

4.2.2.4 Editar

```

1 <div class="portlet light">
2     <div class="portlet-title">
3         <div class="caption font-blue-sunglo">
4             <span class="caption-subject bold uppercase">Modificando compra</
              span>
5         </div>
6     </div>
7     <%= render 'form' %>
8
9     <%= button_to 'Voltar', vende_index_path, :class => "btn btn-default", :method => :
       get %><br>
10 </div>

```

4.2.2.5 Index

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4     <div class="portlet-title">
5         <div class="caption font-blue-sunglo">
6             <span class="caption-subject bold uppercase">Lista de compras de
              itens</span>
7         </div>
8     </div>
9
10    <table class="table table-hover">
11        <thead>
12            <tr>
13                <th>C digo </th>
14                <th>Data</th>
15                <th>Valor</th>
16                <th>Op es</th>
17                <th colspan="3"></th>
18            </tr>
19        </thead>
20
21        <tbody>
22            <% @vendes.each do |vende| %>
23                <tr>
24                    <td><%= vende.idVende %></td>
25                    <td><%= vende.data %></td>
26                    <td><%= vende.valor %></td>
27                    <td align = "right"> <a href = <%= vende_show_path(
                        vende.idVende, vende.idFornecedor, vende.idItem)
                        %>> <span class="glyphicon glyphicon-log-in"
                        title="Detalhes"> </a></td>
28                    <td align = "center"> <a href = <%= vende_edit_path(
                        vende.idVende, vende.idFornecedor, vende.idItem)
                        %>> <span class="glyphicon glyphicon-edit" title
                        ="Modificar"> </a></td>

```

```

29         <td align = "center"> <a href = <%=
            vende_delete_path(vende.idVende, vende.
            idFornecedor, vende.idItem)%>> <span class="
            glyphicon glyphicon-remove" title="Remover"> </a
            ></td>
30     </tr>
31     <% end %>
32 </tbody>
33 </table>
34 <br>
35 <br>
36 <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
    get %><br>
37 </div>

```

4.2.2.6 Index json

```

1 json.array!(@vendes) do |vende|
2   json.extract! vende, :id, :idVende, :idFornecedor, :idItem, :data, :valor
3   json.url vende_url(vende, format: :json)
4 end

```

4.2.2.7 Editar Item

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Modificando item</span>
5     </div>
6   </div>
7
8   <%= render 'item_form' %>
9
10  <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
    get %><br>
11 </div>

```

4.2.2.8 Item index

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="row">
5     <div class="col-sm-5">
6       <%= link_to item_new_path, class: "btn green", :method => :get do %>
7         <i class="fa fa-plus-circle"></i>
8         Cadastrar novo Item
9       <% end %>
10    </div>
11    <div class="col-sm-5"><%= button_to "Relat rio dos Itens em falta",
        itens_falta_relatorio_path, :class => "btn btn-default", :method => :get
        %></div>
12  </div>
13  <br>
14  <div class="row">
15    <div class="col-sm-5">
16      <%= link_to vende_new_path, class: "btn green", :method => :get do
        %>
17        <i class="fa fa-plus-circle"></i>
18        Registrar compra de Itens
19      <% end %>
20    </div>
21    <div class="col-sm-5"><%= button_to "Listar compras de itens",
        vende_index_path, :class => "btn btn-default", :method => :get %></div>
22  </div>
23 </div>
24 </div>

```



```

25
26 <div class="portlet light">
27   <div class="portlet-title">
28     <div class="caption font-blue-sunglo">
29       <span class="caption-subject bold uppercase">Lista de Itens</span>
30     </div>
31   </div>
32   <br>
33   <%= form_for(item_index_path) do |f| %>
34     <div class="col-md-4">
35       <%= f.text_field :nome, placeholder: 'Digite o nome do item', class:
36         "form-control" %>
37     </div>
38     <div class="actions">
39       <%= f.submit "Consulta", class: "btn btn-default"%>
40     </div>
41   <%end%>
42
43   <%= render partial: 'item_table', locals: {items: @items } %>
44 </div>
45 <br>
46 <br>

```

4.2.2.9 Item index json

```

1 json.array!(@items) do |item|
2   json.extract! item, :id, :nome, :marca, :descricao, :quantidade, :quantidadeMinima
3   json.url item_url(item, format: :json)
4 end

```

4.2.2.10 Novo Item

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Novo item</span>
5     </div>
6   </div>
7
8   <%= render 'item_form' %>
9
10  <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
    get %><br>
11 </div>

```

4.2.2.11 Consultar Item

```

1 <p id="notice"><%= notice %></p>
2
3 <h1>Consultando Itens</h1>
4
5 <%= form_for(item_search_path) do |f| %>
6   <div class="col-md-6">
7     <%= f.label 'Nome:' %><br>
8     <%= f.text_field :nome, class: "form-control" %>
9   </div>
10  <br/>
11  <div class="actions">
12    <%= f.submit "Consulta", class: "btn green"%>
13  </div>
14
15 <% end %>
16
17 <%= render partial: 'item_table', locals: {items: @items } %>
18
19 <br>

```

4.2.2.12 Visualizar Item

```
1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="portlet-title">
5     <div class="caption font-blue-sunglo">
6       <span class="caption-subject bold uppercase">Detalhes do item</span>
7     </div>
8   </div>
9
10  <div class="field">
11    <strong>Nome:</strong>
12    <%= @item.nome %>
13  </div>
14
15  <div class="field">
16    <strong>Marca:</strong>
17    <%= @item.marca %>
18  </div>
19
20  <div class="field">
21    <strong>Descrição:</strong>
22    <%= @item.descricao %>
23  </div>
24
25  <div class="field">
26    <strong>Quantidade:</strong>
27    <%= @item.quantidade %>
28  </div>
29
30  <div class="field">
31    <strong>Quantidade mínima:</strong>
32    <%= @item.quantidadeMinima %>
33  </div>
34  <br>
35  <%= button_to 'Modificar', item_edit_path(@item), :class => "btn blue", :method => :
36    get %><br>
37  <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
    get %><br>
38 </div>
```

4.2.2.13 Item show json

```
1 json.extract! @item, :id, :nome, :marca, :descricao, :quantidade, :quantidadeMinima, :
  created_at, :updated_at
```

4.2.2.14 Novo

```
1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Registrar compra de
        itens</span>
5     </div>
6   </div>
7
8   <%= render 'form' %>
9
10  <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
    get %><br>
11 </div>
```

4.2.2.15 Relatório itens em falta

```
1 <% content_for :js do%>
2   <%= javascript_tag do %>
3     function printDiv(divName) {
```

```

4      var printContents = document.getElementById(divName).innerHTML;
5      var originalContents = document.body.innerHTML;
6
7      document.body.innerHTML = printContents;
8
9      window.print();
10
11     document.body.innerHTML = originalContents;
12 }
13 <% end %>
14 <% end %>
15
16 <div class="portlet light">
17     <div class="portlet-title">
18         <div class="caption font-blue-sunglo">
19             <span class="caption-subject bold uppercase">Lista de itens em falta
20         </span>
21     </div>
22 </div>
23 <%if @item.length == 0%>
24     <p id="notice"><%= 'N o h itens em falta.' %></p>
25 <%else%>
26     <div id="print">
27         <table class="table table-hover">
28             <thead>
29                 <tr>
30                     <th>Nome</th>
31                     <th>Marca</th>
32                     <th>Quantidade</th>
33                     <th>Quantidade m nima</th>
34                     <th colspan="3"></th>
35                 </tr>
36             </thead>
37             <tbody>
38
39                 <% @item.each do |item| %>
40                     <tr>
41                         <td><%= item.nome %></td>
42                         <td><%= item.marca %></td>
43                         <td><%= item.quantidade %></td>
44                         <td><%= item.quantidadeMinima %></td>
45                     </tr>
46                 <% end %>
47             </tbody>
48         </table>
49     </div>
50
51     <br>
52     <%= link_to 'Imprimir', '', class: "btn btn-default" ,onclick: 'printDiv("
53         print")' %> <br>
54
55     <%= button_to 'Voltar', item_index_path, :class => "btn btn-default", :method => :
56         get %><br>
57 </div>

```

4.2.2.16 Visualizar

```

1     <p id="notice"><%= notice %></p>
2 <div class="portlet light">
3     <div class="portlet-title">
4         <div class="caption font-blue-sunglo">
5             <span class="caption-subject bold uppercase">Detalhes de compra</
6             span>
7         </div>
8     </div>

```

```

9      <br>
10
11
12      <p>
13          <strong>Codigo: </strong>
14          <%= @vende.idVende %>
15      </p>
16
17      <p>
18          <strong>Fornecedor: </strong>
19          <%= Fornecedor.find(@vende.idFornecedor).nome %>
20      </p>
21
22      <p>
23          <strong>Item: </strong>
24          <%= (item = Item.find(@vende.idItem)).nome + " (" + item.marca + ")" %>
25      </p>
26
27      <p>
28          <strong>Data: </strong>
29          <%= @vende.data %>
30      </p>
31
32      <p>
33          <strong>Valor: </strong>
34          <%= @vende.valor %>
35      </p>
36      <br>
37
38      <%= button_to 'Modificar', vende_edit_path(@vende.idVende, @vende.idFornecedor,
39          @vende.idItem), :class => "btn blue", :method => :get %><br>
40      <%= button_to 'Voltar', vende_index_path, :class => "btn btn-default", :method => :
41          get %><br>
42  </div>

```

4.2.2.17 Visualizar json

```

1  json.extract! @vende, :id, :idVende, :idFornecedor, :idItem, :data, :valor, :created_at, :
    updated_at

```

4.2.3 Fornecedor

4.2.3.1 Form

```

1  <%= form_for(@fornecedor) do |f| %>
2      <% if @fornecedor.errors.any? %>
3          <div id="error_explanation">
4              <h2><%= pluralize(@fornecedor.errors.count, "error") %> prohibited this fornecedor from
                being saved:</h2>
5
6              <ul>
7                  <% @fornecedor.errors.full_messages.each do |message| %>
8                      <li><%= message %></li>
9                  <% end %>
10             </ul>
11         </div>
12         <% end %>
13         <div class="row">
14             <div class="col-md-6" >
15                 <%= f.label :nome %><br>
16                 <%= f.text_field :nome, class: "form-control" %>
17             </div>
18         </div>
19         <div class="row">
20             <div class="col-md-2">
21                 <%= f.label :cnpj %><br>
22                 <%= f.text_field :cnpj, class: "form-control" %>

```

```

23   </div>
24 </div>
25 <div class="row">
26   <div class="col-md-4">
27     <%= f.label :cidade %><br>
28     <%= f.text_field :cidade, class: "form-control" %>
29   </div>
30
31   <div class="col-md-4">
32     <%= f.label :estado %><br>
33     <%= f.text_field :estado, class: "form-control" %>
34   </div>
35
36   <div class="col-md-4">
37     <%= f.label :num %><br>
38     <%= f.text_field :num, class: "form-control" %>
39   </div>
40 </div>
41 <div class="row">
42   <div class="col-md-6">
43     <%= f.label :endereco %><br>
44     <%= f.text_field :endereco, class: "form-control" %>
45   </div>
46 </div>
47 <div class="row">
48   <div class="col-md-2">
49     <%= f.label :telefone %><br>
50     <%= f.text_field :telefone, class: "form-control" %>
51   </div>
52 </div>
53 <br/>
54 <div class="form-actions">
55   <%= f.submit class: "btn green" %>
56 </div>
57 <% end %>

```

4.2.3.2 Lista de itens

```

1 <table class="table table-hover">
2   <thead>
3     <tr>
4       <th>Nome</th>
5       <th>Marca</th>
6       <th>Quantidade</th>
7       <th>Alerta</th>
8       <!-- <th>0p es </th> -->
9       <th colspan="3"></th>
10    </tr>
11  </thead>
12  <br>
13  <tbody>
14
15    <% items.each do |item| %>
16
17      <tr>
18        <td><%= item.nome %></td>
19        <td><%= item.marca %></td>
20        <td><%= item.quantidade %></td>
21        <td align = "center">
22          <% if(item.quantidade <= item.quantidadeMinima) %><span class="glyphicon
                glyphicon-exclamation-sign" title="A quantidade de itens esta baixa
                "></span><% end %>
23
24          </td>
25        <!-- <td align = "right"> <a href = <%= item_show_path(item)%>> <span class="
                glyphicon glyphicon-log-in" title="Detalhes"> </a></td>
26        <td align = "center"> <a href = <%= item_edit_path(item)%>> <span class="glyphicon
                glyphicon-edit" title="Modificar"> </a></td>

```

```

26         <td align = "center"> <a href = <%= item_delete_path(item)%>> <span class="glyphicon
27             glyphicon-remove" title="Remover"> </a></td> -->
28     </tr>
29     <% end %>
30 </tbody>
</table>

```

4.2.3.3 Tabela

```

1 <table class="table table-hover">
2   <thead>
3     <tr>
4       <th>Nome</th>
5       <th>Cnpj</th>
6       <th>Telefone</th>
7       <th colspan="3"></th>
8     </tr>
9   </thead>
10
11  <tbody>
12    <% fornecedors.each do |fornecedor| %>
13      <tr>
14        <td><%= fornecedor.nome %></td>
15        <td><%= fornecedor.cnpj %></td>
16        <td><%= fornecedor.telefone %></td>
17        <td><%= link_to fornecedor do %>
18          <span class="glyphicon glyphicon-log-in" title="Detalhes">
19            <% end %></td>
20        <td><%= link_to edit_fornecedor_path(fornecedor) do %>
21          <span class="glyphicon glyphicon-edit" title="Modificar">
22            <% end %></td>
23        <td><%= link_to fornecedor, method: :delete, data: { confirm: 'Are you sure?' } do
24          %>
25          <span class="glyphicon glyphicon-remove" title="Remover">
26            <% end %></td>
27      </tr>
28    <% end %>
29  </tbody>
</table>

```

4.2.3.4 Editar

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Editando Fornecedor</span>
5     </div>
6   </div>
7   <div class="portlet-body">
8
9     <%= render 'form' %>
10
11     <%= link_to 'Show', @fornecedor %> |
12     <%= link_to 'Back', fornecedores_path %>
13
14   </div>
15 </div>

```

4.2.3.5 Index

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="portlet-title">
5     <div class="caption font-blue-sunglo">
6       <span class="caption-subject bold uppercase">Lista de Fornecedores</span>
7     </div>

```

```

8   </div>
9   <div class="portlet-body">
10     <%= render partial: 'table', locals: {fornecedores: @fornecedores } %>
11
12     <br>
13
14     <%= link_to new_fornecedor_path, class: "btn green" do %>
15       <i class="fa fa-plus-circle"></i>
16       Novo Fornecedor
17     <% end %>
18
19     <%= link_to fornecedores_search_path, class: "btn green" do %>
20       <i class="fa fa-search"></i>
21       Consulta de fornecedor
22     <% end %>
23
24   </div>
25 </div>

```

4.2.3.6 Index json

```

1 json.array!(@fornecedores) do |fornecedor|
2   json.extract! fornecedor, :id, :nome, :cnpj, :cidade, :estado, :num, :endereco, :telefone
3   json.url fornecedor_url(fornecedor, format: :json)
4 end

```

4.2.3.7 Novo

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Novo Fornecedor</span>
5     </div>
6   </div>
7   <div class="portlet-body">
8
9     <%= render 'form' %>
10
11     <%= link_to 'Back', fornecedores_path %>
12
13   </div>
14 </div>

```

4.2.3.8 Relatório

```

1 <% content_for :js do%>
2   <%= javascript_tag do %>
3     function printDiv(divName) {
4       var printContents = document.getElementById(divName).innerHTML;
5       var originalContents = document.body.innerHTML;
6
7       document.body.innerHTML = printContents;
8
9       window.print();
10
11       document.body.innerHTML = originalContents;
12     }
13   <% end %>
14 <% end %>
15
16 <div class="portlet light">
17   <div class="portlet-title">
18     <div class="caption font-blue-sunglo">
19       <span class="caption-subject bold uppercase">Relatorio do fornecedor: <%= @fornecedor.
20         nome %></span>
21     </div>
22   </div>

```

```

22 <div class="portlet-body">
23
24   <%= link_to 'Imprimir', '', onclick: 'printDiv("print")' %>
25   <div id="print">
26     <%= render partial: 'list_items.html.erb', locals: {items: @fornecedor.items} %>
27   </div>
28 </div>
29 </div>

```

4.2.3.9 Buscar

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="portlet-title">
5     <div class="caption font-blue-sunglo">
6       <span class="caption-subject bold uppercase">Consulta de Fornecedor pelo nome</span>
7     </div>
8   </div>
9   <div class="portlet-body">
10
11     <%= form_for(fornecedores_search_path) do |f| %>
12     <div class="col-md-6">
13       <%= f.label 'Nome:' %><br>
14       <%= f.text_field :nome, class: "form-control" %>
15     </div>
16     <br/>
17     <div class="actions">
18       <%= f.submit "Consulta", class: "btn green"%>
19     </div>
20
21   <% end %>
22
23   <%= render partial: 'table', locals: {fornecedores: @fornecedores } %>
24
25   <br>
26 </div>
27 </div>

```

4.2.3.10 Visualizar

```

1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Exibindo fornecedor: <%= @fornecedor.nome
5       %></span>
6     </div>
7   </div>
8   <div class="portlet-body">
9
10     <p id="notice"><%= notice %></p>
11
12     <p>
13       <strong>Nome:</strong>
14       <%= @fornecedor.nome %>
15     </p>
16
17     <p>
18       <strong>Cnpj:</strong>
19       <%= @fornecedor.cnpj %>
20     </p>
21
22     <p>
23       <strong>Cidade:</strong>
24       <%= @fornecedor.cidade %>
25     </p>
26
27     <p>

```



```

27 <strong>Estado:</strong>
28 <%= @fornecedor.estado %>
29 </p>
30
31 <p>
32 <strong>Num:</strong>
33 <%= @fornecedor.num %>
34 </p>
35
36 <p>
37 <strong>Endereco:</strong>
38 <%= @fornecedor.endereco %>
39 </p>
40
41 <p>
42 <strong>Telefone:</strong>
43 <%= @fornecedor.telefone %>
44 </p>
45
46 <%= link_to 'Edit', edit_fornecedor_path(@fornecedor) %> |
47 <%= link_to 'Back', fornecedores_path %>
48 </div>
49 <br/><br/>
50 <div class="portlet-title">
51   <div class="caption font-blue-sunglo">
52     <span class="caption-subject bold uppercase">Itens desse fornecedor</span>
53   </div>
54 </div>
55
56 <div class="portlet-body">
57
58   <%= render partial: 'list_items.html.erb', locals: {items: @fornecedor.items} %>
59
60   <%= link_to 'Gerar relatorio', fornecedores_relatorio_path(@fornecedor) %>
61 </div>
62 </div>

```

4.2.3.11 Visualizar json

```

1 json.extract! @fornecedor, :id, :nome, :cnpj, :cidade, :estado, :num, :endereco, :telefone,
  :created_at, :updated_at

```

4.2.4 Funcionario

4.2.4.1 Form

```

1 <% content_for :css do %>
2 <%= stylesheet_link_tag "/assets/global/plugins/bootstrap-datepicker/css/bootstrap-
  datepicker.css" %>
3 <% end %>
4
5 <% content_for :js do %>
6 <%= javascript_include_tag "/assets/global/plugins/bootstrap-datepicker/js/bootstrap-
  datepicker.js" %>
7 <% end %>
8
9 <%= form_for(@funcionario) do |f| %>
10 <% if @funcionario.errors.any? %>
11 <div id="error_explanation">
12 <h2><%= pluralize(@funcionario.errors.count, "error") %> prohibited this funcionario from
  being saved:</h2>
13
14 <ul>
15 <% @funcionario.errors.full_messages.each do |message| %>
16 <li><%= message %></li>
17 <% end %>
18 </ul>
19 </div>

```

```

20     <% end %>
21
22 <div class="row">
23     <div class="col-md-6">
24         <%= f.label :nome %><br>
25         <%= f.text_field :nome, class: "form-control" %>
26     </div>
27 </div>
28 <div class="row">
29     <div class="col-md-2">
30         <%= f.label :cpf %><br>
31         <%= f.text_field :cpf, class: "form-control" %>
32     </div>
33 </div>
34 <div class="row">
35     <div class="col-md-4">
36         <%= f.label :cidade %><br>
37         <%= f.text_field :cidade, class: "form-control" %>
38     </div>
39     <div class="col-md-4">
40         <%= f.label :estado %><br>
41         <%= f.text_field :estado, class: "form-control" %>
42     </div>
43     <div class="col-md-4">
44         <%= f.label :num %><br>
45         <%= f.text_field :num, class: "form-control" %>
46     </div>
47 </div>
48 <div class="row">
49     <div class="col-md-6">
50         <%= f.label :endereco %><br>
51         <%= f.text_field :endereco, class: "form-control" %>
52     </div>
53 </div>
54 <div class="row">
55     <div class="col-md-2">
56         <%= f.label :telefone %><br>
57         <%= f.text_field :telefone, class: "form-control" %>
58     </div>
59 </div>
60 <div class="row">
61     <div class="col-md-1">
62         <%= f.label :salario %><br>
63         <%= f.text_field :salario, class: "form-control" %>
64     </div>
65 </div>
66 <div class="row">
67     <div class="col-md-3">
68         <%= f.label :dataContratacao %><br>
69         <%= f.text_field :dataContratacao, class: "form-control", data: {provide: "datepicker"} %>
70     </div>
71 </div>
72 <div class="row">
73     <div class="col-md-6">
74         <%= f.label :ativo %><br>
75         <%= f.check_box :ativo, class: "form-control" %>
76     </div>
77 </div>
78 <div class="row">
79     <div class="col-md-4">
80         <%= f.label :turno %><br>
81         <%= f.select :turno, [["Manh ", 0], ["Tarde", 1], ["Noite", 2]], {}, class: "form-control" %>
82     </div>
83 </div>
84 <div class="row">
85     <div class="col-md-6">

```

```

86     <%= f.label :login %><br>
87     <%= f.text_field :login, class: "form-control" %>
88   </div>
89 </div>
90 <div class="row">
91   <div class="col-md-6">
92     <%= f.label :password %><br>
93     <%= f.password_field :password, class: "form-control" %>
94   </div>
95   <div class="col-md-6">
96     <%= f.label :password_confirmation %><br>
97     <%= f.password_field :password_confirmation, class: "form-control" %>
98   </div>
99 </div>
100 <div class="row">
101   <div class="col-md-4">
102     <%= f.label :nivelAcesso %><br>
103     <%= f.select :nivelAcesso, [["Gar on", 0], ["cozinheiro", 1], ["Gerente", 2]], {},
104       class: "form-control" %>
105   </div>
106 </div>
107 <br/><br/>
108 <div class="actions">
109   <%= f.submit class: "btn green"%>
110 </div>
<% end %>

```

4.2.4.2 Tabela

```

1 <table class="table table-hover">
2   <thead>
3     <tr>
4       <th>Nome</th>
5       <th>Cpf</th>
6       <th>Telefone</th>
7       <th>Salario</th>
8       <th>Ativo</th>
9       <th>Turno</th>
10      <th colspan="3"></th>
11    </tr>
12  </thead>
13
14  <tbody>
15    <% funcionarios.each do |funcionario| %>
16      <tr>
17        <td><%= funcionario.nome %></td>
18        <td><%= funcionario.cpf %></td>
19        <td><%= funcionario.cidade %></td>
20        <td><%= funcionario.telefone %></td>
21        <td><%= funcionario.salario %></td>
22        <td><%= funcionario.ativo %></td>
23        <td><%= funcionario.turno %></td>
24        <td><%= link_to funcionario do %>
25          <span class="glyphicon glyphicon-log-in" title="Detalhes">
26        <% end %></td>
27        <td><%= link_to edit_funcionario_path(funcionario) do %>
28          <span class="glyphicon glyphicon-edit" title="Modificar">
29        <% end %></td>
30        <td><%= link_to funcionarios_demitir_path(funcionario), method: :get,
31          data: { confirm: 'Are you sure?' } do %>
32          <span class="glyphicon glyphicon-remove" title="Demitir">
33        <% end %></td>
34      </tr>
35    <% end %>
36  </tbody>
37 </table>

```

4.2.4.3 Editar

```
1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Editando Funcionario: <%= @funcionario.
5         nome %></span>
6     </div>
7   </div>
8   <div class="portlet-body">
9     <%= render 'form' %>
10
11     <%= link_to 'Show', @funcionario %> |
12     <%= link_to 'Back', funcionarios_path %>
13
14   </div>
15 </div>
```

4.2.4.4 Index

```
1 <p id="notice"><%= notice %></p>
2 <div class="portlet light">
3   <div class="portlet-title">
4     <div class="caption font-blue-sunglo">
5       <span class="caption-subject bold uppercase">Listando Funcionarios</span>
6     </div>
7   </div>
8   <div class="portlet-body">
9
10     <%= render partial: 'table', locals: {funcionarios: @funcionarios } %>
11
12     <br>
13
14     <%= link_to new_funcionario_path, class: "btn green" do %>
15       <i class="fa fa-plus-circle"></i>
16       New Funcionario
17     <% end %>
18
19     <%= link_to funcionarios_search_path, class: "btn green" do %>
20       <i class="fa fa-search"></i>
21       Consulta de funcionarios
22     <% end %>
23   </div>
24 </div>
```

4.2.4.5 Index json

```
1 json.array!(@funcionarios) do |funcionario|
2   json.extract! funcionario, :id, :nome, :cpf, :cidade, :estado, :num, :endereco, :telefone,
3     :salario, :dataContratacao, :ativo, :turno, :login, :password_digest, :nivelAcesso
4   json.url funcionario_url(funcionario, format: :json)
5 end
```

4.2.4.6 Novo

```
1 <div class="portlet light">
2   <div class="portlet-title">
3     <div class="caption font-blue-sunglo">
4       <span class="caption-subject bold uppercase">Novo Funcionario <%= @funcionario.nome
5         %></span>
6     </div>
7   </div>
8   <div class="portlet-body">
9     <%= render 'form' %>
10
11     <%= link_to 'Back', funcionarios_path %>
```

```

12 </div>
13 </div>

```

4.2.4.7 Buscar

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="portlet-title">
5     <div class="caption font-blue-sunglo">
6       <span class="caption-subject bold uppercase">Busca de Funcionarios pelo nome</span>
7     </div>
8   </div>
9   <div class="portlet-body">
10
11     <%= form_for(funcionario_search_path) do |f| %>
12       <div class="col-md-6">
13         <%= f.label 'Nome:' %><br>
14         <%= f.text_field :nome, class: "form-control" %>
15       </div>
16       <br/>
17       <div class="actions">
18         <%= f.submit "Consulta", class: "btn green"%>
19       </div>
20
21     <% end %>
22
23     <%= render partial: 'table', locals: {funcionarios: @funcionarios } %>
24
25     <br>
26   </div>
27 </div>

```

4.2.4.8 Visualizar

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet light">
4   <div class="portlet-title">
5     <div class="caption font-blue-sunglo">
6       <span class="caption-subject bold uppercase">Mostrando Funcionario: <%= @funcionario.
7         nome %></span>
8     </div>
9   </div>
10   <div class="portlet-body">
11
12     <p>
13       <strong>Nome:</strong>
14       <%= @funcionario.nome %>
15     </p>
16
17     <p>
18       <strong>Cpf:</strong>
19       <%= @funcionario.cpf %>
20     </p>
21
22     <p>
23       <strong>Cidade:</strong>
24       <%= @funcionario.cidade %>
25     </p>
26
27     <p>
28       <strong>Estado:</strong>
29       <%= @funcionario.estado %>
30     </p>
31
32     <p>
33       <strong>Num:</strong>

```

```

33 <%= @funcionario.num %>
34 </p>
35
36 <p>
37 <strong>Endereco:</strong>
38 <%= @funcionario.endereco %>
39 </p>
40
41 <p>
42 <strong>Telefone:</strong>
43 <%= @funcionario.telefone %>
44 </p>
45
46 <p>
47 <strong>Salario:</strong>
48 <%= @funcionario.salario %>
49 </p>
50
51 <p>
52 <strong>Datacontratacao:</strong>
53 <%= @funcionario.dataContratacao %>
54 </p>
55
56 <p>
57 <strong>Ativo:</strong>
58 <%= @funcionario.ativo %>
59 </p>
60
61 <p>
62 <strong>Turno:</strong>
63 <%= @funcionario.turno %>
64 </p>
65
66 <p>
67 <strong>Login:</strong>
68 <%= @funcionario.login %>
69 </p>
70
71 <p>
72 <strong>Password digest:</strong>
73 <%= @funcionario.password_digest %>
74 </p>
75
76 <p>
77 <strong>Nivelacesso:</strong>
78 <%= @funcionario.nivelAcesso %>
79 </p>
80
81 <%= link_to 'Edit', edit_funcionario_path(@funcionario) %> |
82 <%= link_to 'Back', funcionarios_path %>
83
84 </div>
85 </div>

```

4.2.4.9 Visualizar json

```

1 json.extract! @funcionario, :id, :nome, :cpf, :cidade, :estado, :num, :endereco, :telefone,
  :salario, :dataContratacao, :ativo, :turno, :login, :password_digest, :nivelAcesso, :
  created_at, :updated_at

```

4.2.5 Pedido

4.2.5.1 Form

```

1 <%= form_for(@pedido) do |f| %>
2   <% if @pedido.errors.any? %>
3     <div id="error_explanation">

```

```

4      <h2><%= pluralize(@pedido.errors.count, "error") %> prohibited this pedido from being
      saved:</h2>
5
6      <ul>
7        <% @pedido.errors.full_messages.each do |message| %>
8          <li><%= message %></li>
9        <% end %>
10      </ul>
11    </div>
12    <% end %>
13
14    <div class="field">
15      <%= f.label :idPedido %><br>
16      <%= f.number_field :idPedido %>
17    </div>
18    <div class="field">
19      <%= f.label :idFuncionario %><br>
20      <%= f.number_field :idFuncionario %>
21    </div>
22    <div class="field">
23      <%= f.label :idProduto %><br>
24      <%= f.number_field :idProduto %>
25    </div>
26    <div class="field">
27      <%= f.label :idcomanda %><br>
28      <%= f.number_field :idcomanda %>
29    </div>
30    <div class="field">
31      <%= f.label :dataPedido %><br>
32      <%= f.datetime_select :dataPedido %>
33    </div>
34    <div class="field">
35      <%= f.label :dataFinalizacao %><br>
36      <%= f.datetime_select :dataFinalizacao %>
37    </div>
38    <div class="field">
39      <%= f.label :cancelado %><br>
40      <%= f.check_box :cancelado %>
41    </div>
42    <div class="actions">
43      <%= f.submit %>
44    </div>
45    <% end %>

```

4.2.5.2 Editar

```

1    <h1>Editing Pedido</h1>
2
3    <%= render 'form' %>
4
5    <%= link_to 'Show', @pedido %> |
6    <%= link_to 'Back', pedidos_path %>

```

4.2.5.3 Index

```

1    <p id="notice"><%= notice %></p>
2
3    <h1>Listing Pedidos</h1>
4
5    <table>
6      <thead>
7        <tr>
8          <th>Idpedido</th>
9          <th>Idfuncionario</th>
10         <th>Idproduto</th>
11         <th>Idcomanda</th>
12         <th>Datapedido</th>
13         <th>Datafinalizacao</th>

```

```

14     <th>Cancelado</th>
15     <th colspan="3"></th>
16 </tr>
17 </thead>
18
19 <tbody>
20   <% @pedidos.each do |pedido| %>
21     <tr>
22       <td><%= pedido.idPedido %></td>
23       <td><%= pedido.idFuncionario %></td>
24       <td><%= pedido.idProduto %></td>
25       <td><%= pedido.idcomanda %></td>
26       <td><%= pedido.dataPedido %></td>
27       <td><%= pedido.dataFinalizacao %></td>
28       <td><%= pedido.cancelado %></td>
29       <td><%= link_to 'Show', pedido %></td>
30       <td><%= link_to 'Edit', edit_pedido_path(pedido) %></td>
31       <td><%= link_to 'Destroy', pedido, method: :delete, data: { confirm: 'Are you sure?'
32         } %></td>
33     </tr>
34   <% end %>
35 </tbody>
36 </table>
37
38 <br>
39 <%= link_to 'New Pedido', new_pedido_path %>

```

4.2.5.4 Index json

```

1 json.array!(@pedidos) do |pedido|
2   json.extract! pedido, :id, :idPedido, :idFuncionario, :idProduto, :idcomanda, :dataPedido,
3     :dataFinalizacao, :cancelado
4   json.url pedido_url(pedido, format: :json)
5 end

```

4.2.5.5 Novo

```

1 <h1>New Pedido</h1>
2
3 <%= render 'form' %>
4
5 <%= link_to 'Back', pedidos_path %>

```

4.2.5.6 Visualizar

```

1 <p id="notice"><%= notice %></p>
2
3 <p>
4   <strong>Idpedido:</strong>
5   <%= @pedido.idPedido %>
6 </p>
7
8 <p>
9   <strong>Idfuncionario:</strong>
10  <%= @pedido.idFuncionario %>
11 </p>
12
13 <p>
14   <strong>Idproduto:</strong>
15   <%= @pedido.idProduto %>
16 </p>
17
18 <p>
19   <strong>Idcomanda:</strong>
20   <%= @pedido.idcomanda %>
21 </p>

```



```

22
23 <p>
24   <strong>Datapedido:</strong>
25   <%= @pedido.dataPedido %>
26 </p>
27
28 <p>
29   <strong>Datafinalizacao:</strong>
30   <%= @pedido.dataFinalizacao %>
31 </p>
32
33 <p>
34   <strong>Cancelado:</strong>
35   <%= @pedido.cancelado %>
36 </p>
37
38 <%= link_to 'Edit', edit_pedido_path(@pedido) %> |
39 <%= link_to 'Back', pedidos_path %>

```

4.2.5.7 Form

```

1 json.extract! @pedido, :id, :idPedido, :idFuncionario, :idProduto, :idcomanda, :dataPedido,
  :dataFinalizacao, :cancelado, :created_at, :updated_at

```

4.2.6 Produto

4.2.6.1 Form

```

1 <%= form_for(@produto, url: produto_show_path(@produto), id: "new_produto") do |f| %>
2   <% if @produto.errors.any? %>
3     <div id="error_explanation">
4       <h2><%= pluralize(@produto.errors.count, "error") %> prohibited this produto from
        being saved:</h2>
5
6       <ul>
7         <% @produto.errors.full_messages.each do |message| %>
8           <li><%= message %></li>
9         <% end %>
10      </ul>
11    </div>
12    <% end %>
13
14    <div class="portlet light ">
15      <div class="portlet-body form">
16        <form role="form">
17          <div class="form-body">
18            <div class="form-group">
19              <div>
20                <%= f.label :nome, class: "control-label" %>
21                <%= f.text_field :nome, class: "form-control input-xlarge" %>
22              </div>
23              <br>
24              <div>
25                <%= f.label :valor, class: "control-label" %><br>
26                <%= f.text_field :valor, class: "form-control input-xlarge" %>
27              </div>
28              <br>
29              <div>
30                <%= f.label :descricao, class: "control-label" %><br>
31                <%= f.text_field :descricao, class: "form-control input-xlarge" %>
32              </div>
33            <br>
34
35            <table id="tabela-selecionados" class="table table-hover">
36              <thead>
37                <tr>
38                  <th style="width: 130px;">Nome</th>
39                  <th style="width: 130px;">Marca</th>

```

```

40         <th style="width: 70px;">Quantidade</th>
41         <th style="width: 70px;"><th>
42     </tr>
43 </thead>
44 <br>
45 <tbody>
46
47     <% @items.each do |item| %>
48
49         <tr style="display: none">
50             <td><%= item.nome %></td>
51             <td><%= item.marca %></td>
52             <td></td>
53         </tr>
54         <% end %>
55     </tbody>
56 </table>
57
58 <div class="actions">
59     <a class="btn green" data-toggle="modal" href="#nova-comanda">
60         <i class="fa fa-plus-circle"></i>Adicionar Item
61     </a>
62 </div>
63 <!--
64     <div class="field">
65         <%= f.label :idCompoe %><br>
66         <%= f.number_field :idCompoe %>
67     </div>
68     <div class="field">
69         <%= f.label :idProduto %><br>
70         <%= f.number_field :idProduto %>
71     </div>
72     <div class="field">
73         <%= f.label :idItem %><br>
74         <%= f.number_field :idItem %>
75     </div>
76     <div class="field">
77         <%= f.label :quantidade %><br>
78         <%= f.number_field :quantidade %>
79     </div>
80     <div class="actions">
81         <%= f.submit %>
82     </div>
83 -->
84 <br><br>
85
86
87 <div class="actions">
88     <%= submit_tag "Cadastrar Produto", :class => "btn green", id: "cdstr" %><br>
89     <%= link_to "Teste", "#", class: "btn green", onclick: "cadastrar()", remote:
90         true %>
91 </div>
92 <hr>
93 </div>
94 </form>
95
96 </div>
97 </div>
98
99 <% end %>
100
101
102
103
104 <div id="nova-comanda" class="modal fade" tabindex="-1" data-width="600" style="display:
105     none; width: 600px; margin-left: -379px; margin-top: 0px;" aria-hidden="true">

```

```

106 <%= form_tag('/produtos/create', method: "post", class: 'form-horizontal') do |f| %>
107   <div class="modal-header">
108     <button type="button" class="close" data-dismiss="modal" aria-hidden="true"></button>
109     <h2 class="modal-title">Lista de Itens</h2>
110   </div>
111   <div class="modal-body">
112     <div class="row">
113       <div class="form-body">
114         <div class="col-md-12">
115           <div class="form-group">
116             <div id="alert_container" class="custom-alerts alert alert-danger fade in"
              style="display: none;"><button type="button" class="close" data-dismiss="
              alert" aria-hidden="true"></button>Voce precisa definir a quantidade dos
              itens selecionados</div>
117
118             <table id="lista-de-itens" class="table table-hover">
119               <thead>
120                 <tr>
121                   <th>#</th>
122                   <th>Nome</th>
123                   <th>Marca</th>
124                   <th>Quantidade</th>
125                 </tr>
126               </thead>
127               <tbody>
128                 <% @items.each do |item| %>
129                   <tr>
130                     <td>
131                       <input type="checkbox" onchange="pdtItemAlterado(this)" value="<%=
                        item.id %>">
132                     </td>
133                     <td><%= item.nome %></td>
134                     <td><%= item.marca %></td>
135                     <td style="width: 70px;"><input type="number" min="0" name="quantidade"
                        class="form-control"/></td>
136                   </tr>
137                 <% end %>
138               </tbody>
139             </table>
140
141           </div>
142         </div>
143       </div>
144     </div>
145   </div>
146   <div class="modal-footer">
147     <input type="button" class="btn green" onclick="pdtChecarItensSelecionados()" aria-
      hidden="true" value="Adicionar"/>
148   </div>
149 <%end%>
150 </div>

```

4.2.6.2 Editar Componentes

```

1 <h1>Editar Componentes do Produto</h1>
2
3 <%= render 'form' %>
4
5 <%= link_to 'Mostrar', @componentes_produto %> |
6 <%= link_to 'Voltar', componentes_produtos_path %>

```

4.2.6.3 Index Componentes

```

1 <p id="notice"><%= notice %></p>
2
3 <h1>Lista de Componentes Produtos</h1>
4

```

```

5 <table>
6   <thead>
7     <tr>
8       <th>Idcompoe</th>
9       <th>Idproduto</th>
10      <th>Iditem</th>
11      <th>Quantidade</th>
12      <th colspan="3"></th>
13    </tr>
14  </thead>
15
16  <tbody>
17    <% @componentes_produtos.each do |componentes_produto| %>
18      <tr>
19        <td><%= componentes_produto.idCompoe %></td>
20        <td><%= componentes_produto.idProduto %></td>
21        <td><%= componentes_produto.idItem %></td>
22        <td><%= componentes_produto.quantidade %></td>
23        <td><%= link_to 'Mostrar', componentes_produto %></td>
24        <td><%= link_to 'Editar', edit_componentes_produto_path(componentes_produto) %></td>
25        <td><%= link_to 'Deletar', componentes_produto, method: :delete, data: { confirm: '
26          Voc est certo?' } %></td>
27      </tr>
28    <% end %>
29  </tbody>
30 </table>
31
32 <br>
33 <%= link_to 'Novo Componente', new_componentes_produto_path %>

```

4.2.6.4 Componentes index json

```

1 json.array!(@componentes_produtos) do |componentes_produto|
2   json.extract! componentes_produto, :id, :idCompoe, :idProduto, :idItem, :quantidade
3   json.url componentes_produto_url(componentes_produto, format: :json)
4 end

```

4.2.6.5 Novo Componente

```

1 <h1>New Componentes Produto</h1>
2
3 <%= render 'form' %>
4
5 <%= link_to 'Voltar', componentes_produtos_path %>

```

4.2.6.6 Visualizar Componentes

```

1 <p id="notice"><%= notice %></p>
2
3 <p>
4   <strong>Idcompoe:</strong>
5   <%= @componentes_produto.idCompoe %>
6 </p>
7
8 <p>
9   <strong>Idproduto:</strong>
10  <%= @componentes_produto.idProduto %>
11 </p>
12
13 <p>
14   <strong>Iditem:</strong>
15   <%= @componentes_produto.idItem %>
16 </p>
17
18 <p>
19   <strong>Quantidade:</strong>

```

```

20 <%= @componentes_produto.quantidade %>
21 </p>
22
23 <%= link_to 'Editar', edit_componentes_produto_path(@componentes_produto) %> |
24 <%= link_to 'Voltar', componentes_produtos_path %>

```

4.2.6.7 Visualizar Componentes json

```

1 json.extract! @componentes_produto, :id, :idCompoe, :idProduto, :idItem, :quantidade, :
  created_at, :updated_at

```

4.2.6.8 Editar Componentes

```

1 <div class="portlet-title">
2   <div class="caption">
3     <i class=" font-green-sharp"></i>
4     <span class="caption-subject font-green-sharp bold uppercase"><h3>Editar Produto</h3>
      </span>
5   </div>
6 </div>
7
8 <%= render 'form' %>
9
10
11 <div class="portlet-title">
12   <div class="caption">
13     <i class=" font-green-sharp"></i>
14     <span class="caption-subject font-green-sharp bold uppercase"><h3>Componentes
      Produto</h3></span>
15   </div>
16 </div>
17 <div class="portlet light ">
18   <div class="portlet-body form">
19     <form role="form">
20       <div class="form-body">
21         <div class="form-group">
22           <table class="table table-hover">
23             <thead>
24               <tr>
25                 <th>Nome</th>
26                 <th>Marca</th>
27                 <th>Quantidade</th>
28                 <th>Remover</th>
29               </tr>
30             </thead>
31             <tbody>
32               <%= @produto.componentes_produtos.each do |componente| %>
33                 <tr>
34                   <td><%= componente.item.nome %></td>
35                   <td><%= componente.item.marca %></td>
36                   <td><%= componente.quantidade %></td>
37                   <td align = "left"> <a href = <%=
      componentes_delete_path(componente)%>> <span class="
      glyphicon glyphicon-remove" title="Remover"> </a></
      td>
38                 </tr>
39               <%= end %>
40             </tbody>
41           </table>
42         </div>
43       </div>
44     </form>
45   </div>
46 </div>
47
48
49 <div class="actions">

```

```

50     <a class="btn btn-default" data-toggle="modal" href=<%=produto_show_path(@produto)%>>
51         Mostrar
52     </a>
53
54     <a class="btn btn-default" data-toggle="modal" href= <%=produtos_path%>>
55         Voltar
56     </a>
57 </div>

```

4.2.6.9 Index

```

1 <p id="notice"><%= notice %></p>
2
3 <div class="portlet-title">
4     <div class="caption">
5         <i class=" font-green-sharp"></i>
6         <span class="caption-subject font-green-sharp bold uppercase"><h3>Lista de Produtos
7             </h3> </span>
8     </div>
9 </div>
10
11 <div class="portlet light ">
12     <div class="portlet-body form">
13         <form role="form">
14             <div class="form-body">
15                 <div class="form-group">
16
17                     <table class="table table-hover">
18                         <thead>
19                             <tr>
20                                 <th>Nome</th>
21                                 <th>Valor</th>
22                                 <th>Descricao</th>
23                                 <th colspan="3"></th>
24                             </tr>
25                         </thead>
26
27                         <tbody>
28                             <% @produtos.each do |produto| %>
29                             <tr>
30                                 <td><%= produto.nome %></td>
31                                 <td><%= produto.valor %></td>
32                                 <td><%= produto.descricao %></td>
33
34                                 <td align = "right"> <a href = <%= produto_show_path(produto)%>> <span class="
35                                     glyphicon glyphicon-log-in" title="Detalhes"> </a></td>
36                                 <td align = "center"> <a href = <%= produto_edit_path(produto)%>> <span class
37                                     ="glyphicon glyphicon-edit" title="Modificar"> </a></td>
38                                 <td align = "center"> <a href = <%= produto_delete_path(produto)%>> <span
39                                     class="glyphicon glyphicon-remove" title="Remover"> </a></td>
40                             </tr>
41                             <% end %>
42                         </tbody>
43                     </table>
44                 </div>
45             </div>
46         </form>
47     </div>
48 </div>
49
50 <br>
51 <div class="actions">
52     <a class="btn green" data-toggle="modal" href= <%=produto_new_path%>>
53         <i class="fa fa-plus-circle"></i> Novo Produto
54     </a>
55 </div>

```

4.2.6.10 Index json

```
1 json.array!(@produtos) do |produto|
2   json.extract! produto, :id, :nome, :valor, :descricao
3   json.url produto_url(produto, format: :json)
4 end
```

4.2.6.11 Novo produto

```
1 <div class="portlet-title">
2   <div class="caption">
3     <i class=" font-green-sharp"></i>
4     <span class="caption-subject font-green-sharp bold uppercase"><h3>Novo Produto</h3>
5       </span>
6   </div>
7 </div>
8 <%= render 'form' %>
9
10 <div class="actions">
11   <a class="btn btn-default" data-toggle="modal" href= <%=produto_index_path%> >
12     Voltar
13   </a>
14 </div>
```

4.2.6.12 Visualizar Produto

```
1 <p id="notice"><%= notice %></p>
2
3
4 <div class="portlet-title">
5   <div class="caption">
6     <i class=" font-green-sharp"></i>
7     <span class="caption-subject font-green-sharp bold uppercase"><h3>Exibir Produto</h3>
8       </span>
9   </div>
10 </div>
11 <div class="portlet light ">
12   <div class="portlet-body form">
13     <form role="form">
14       <div class="form-body">
15         <div class="form-group">
16           <div>
17             <label class=" control-label">Nome</label>
18             <input type="text" readonly class="form-control input-xlarge" value="<%= @produto.
19               nome %>" />
20           </div>
21           <br>
22           <div>
23             <label class="control-label">Valor</label>
24             <input type="text" readonly class="form-control input-xlarge" value="<%= @produto.
25               valor %>" />
26           </div>
27           <br>
28           <div>
29             <label class="control-label">Descrição</label>
30             <input type="text" readonly class="form-control input-xlarge" value="<%= @produto.
31               descricao %>" />
32           </div>
33         </div>
34       </div>
35     </form>
36   </div>
37 </div>
38 <div class="portlet-title">
39   <div class="caption">
40     <i class=" font-green-sharp"></i>
```

```

39         <span class="caption-subject font-green-sharp bold uppercase"><h3>Exibir
          Componentes</h3> </span>
40     </div>
41 </div>
42
43 <div class="portlet light ">
44     <div class="portlet-body form">
45         <form role="form">
46             <div class="form-body">
47                 <div class="form-group">
48                     <table class="table table-hover">
49                         <thead>
50                             <tr>
51                                 <th>Nome</th>
52                                 <th>Marca</th>
53                                 <th>Quantidade</th>
54                             </tr>
55                         </thead>
56                         <br>
57                         <tbody>
58                             <% @produto.componentes_produtos.each do |componente| %>
59                             <tr>
60                                 <td><%= componente.item.nome %></td>
61                                 <td><%= componente.item.marca %></td>
62                                 <td><%= componente.quantidade %></td>
63                             </tr>
64                             <% end %>
65                         </tbody>
66                     </table>
67                 </div>
68             </div>
69         </form>
70     </div>
71 </div>
72
73
74 <div class="actions">
75     <a class="btn btn-default" data-toggle="modal" href=<%= produto_edit_path(@produto) %>>
76         Editar
77     </a>
78
79     <a class="btn btn-default" data-toggle="modal" href= <%= produtos_path %>>
80         Voltar
81     </a>
82 </div>

```

4.2.6.13 Visualizar produto json

```

1 json.extract! @produto, :id, :nome, :valor, :descricao, :created_at, :updated_at

```


4.3 Controller

4.3.1 Aplicação

```
1 class ApplicationController < ActionController::Base
2   # Prevent CSRF attacks by raising an exception.
3   # For APIs, you may want to use :null_session instead.
4   protect_from_forgery with: :exception
5   include SessionsHelper
6 end
```

4.3.2 Comanda

```
1 class ComandasController < ApplicationController
2   before_action :set_comanda, only: [:show, :edit, :update, :encerrar, :new_pedido]
3   before_action :set_pedido, only: [:cancelar_pedido, :finalizar_pedido]
4
5   # GET /comandas
6   # GET /comandas.json
7   def index
8     @comandas = Comanda.all
9     @comanda = Comanda.new
10  end
11
12  # GET /comandas/1
13  # GET /comandas/1.json
14  def show
15    @pedido = Pedido.new
16  end
17
18  # GET /comandas/new
19  def new
20    @comanda = Comanda.new
21  end
22
23  # GET /comandas/1/cancelar_pedido
24  def finalizar_pedido
25    comanda = @pedido.comanda
26    @pedido.dataFinalizacao = Time.now.strftime("%Y-%m-%d %H:%M:%S")
27    @pedido.save
28    respond_to do |format|
29      format.html { redirect_to comanda, notice: 'Pedido concluído com Sucesso' }
30      format.json { head :no_content }
31    end
32  end
33  # GET /comandas/1/cancelar_pedido
34  def cancelar_pedido
35    comanda = @pedido.comanda
36    @pedido.cancelado = true
37
38    if !@pedido.is_finalizado
39      componentes = Produto.find(@pedido.idProduto).componentes_produtos
40      componentes.each do |componente|
41        componente.item.quantidade += componente.quantidade
42        componente.item.save
43      end
44    end
45
46    @pedido.save
47    respond_to do |format|
48      format.html { redirect_to comanda, notice: 'Pedido cancelado com Sucesso' }
49      format.json { head :no_content }
50    end
51  end
52
53  # GET /comandas/1/edit
```

```

54 def edit
55 end
56
57
58 # POST /comandas/1/new_pedido
59 def new_pedido
60   @pedido = Pedido.new
61   @pedido.idcomanda = @comanda.idcomanda
62   @pedido.idFuncionario = @comanda.idFuncionario
63   @pedido.idProduto = params[:pedido][:idProduto]
64   @pedido.dataPedido = Time.now.strftime("%Y-%m-%d %H:%M:%S")
65
66   allOkay = true
67   componentes = Produto.find(@pedido.idProduto).componentes_produtos
68   componentes.each do |componente|
69     allOkay = allOkay && (componente.quantidade <= componente.item.quantidade)
70   end
71   if allOkay
72     componentes.each do |componente|
73       componente.item.quantidade -= componente.quantidade
74       componente.item.save
75     end
76   end
77
78   respond_to do |format|
79     if allOkay && @pedido.save
80       format.html { redirect_to @comanda, notice: 'Pedido adicionado com Sucesso' }
81       format.json { render :show, status: :created, location: @comanda }
82     else
83       format.html { redirect_to @comanda, notice: @pedido.errors }
84       format.json { render json: @pedido.errors, status: :unprocessable_entity }
85     end
86   end
87 end
88
89
90 # POST /comandas
91 # POST /comandas.json
92 def create
93   @comanda = Comanda.new
94   @comanda.idFuncionario = current_funcionario.id
95   @comanda.mesa = params[:comanda][:mesa]
96   @comanda.dataAbertura = Time.now.strftime("%Y-%m-%d %H:%M:%S")
97
98   respond_to do |format|
99     if @comanda.save
100       format.html { redirect_to @comanda, notice: 'Comanda criada com Sucesso' }
101       format.json { render :show, status: :created, location: @comanda }
102     else
103       format.html { render :new }
104       format.json { render json: @comanda.errors, status: :unprocessable_entity }
105     end
106   end
107 end
108
109 # PATCH/PUT /comandas/1
110 # PATCH/PUT /comandas/1.json
111 def update
112   respond_to do |format|
113     if @comanda.update(comanda_params)
114       format.html { redirect_to @comanda, notice: 'Comanda was successfully updated.' }
115       format.json { render :show, status: :ok, location: @comanda }
116     else
117       format.html { render :edit }
118       format.json { render json: @comanda.errors, status: :unprocessable_entity }
119     end
120   end
121 end

```

```

122
123 # DELETE /comandas/1
124 # DELETE /comandas/1.json
125 def encerrar
126   @comanda.dataEncerramento = Time.now.strftime("%Y-%m-%d %H:%M:%S")
127   @comanda.valorFinal = params[:comanda][:valorFinal]
128   @comanda.save
129   respond_to do |format|
130     format.html { redirect_to comandas_url, notice: 'Comanda encerrada com sucesso.' }
131     format.json { head :no_content }
132   end
133 end
134
135 private
136 # Use callbacks to share common setup or constraints between actions.
137 def set_comanda
138   @comanda = Comanda.find(params[:id])
139 end
140
141 def set_pedido
142   @pedido = Pedido.find(params[:id])
143 end
144
145 # Never trust parameters from the scary internet, only allow the white list through.
146 def comanda_params
147   params.require(:comanda).permit(:idcomanda, :idFuncionario, :mesa, :dataAbertura, :
148     dataEncerramento, :valorFinal)
149 end
150 end

```

4.3.3 Compra

```

1 class CompraController < ApplicationController
2   before_action :verificar_login
3   before_action :set_vende, only: [:show, :edit, :update, :destroy]
4   before_action :set_info, only: [:new, :show, :edit, :update, :create]
5   before_action :set_item, only: [:item_show, :item_edit, :item_update, :item_destroy]
6
7   # GET /vendes
8   # GET /vendes.json
9   def index
10     @vendes = Vende.all
11   end
12
13   # GET /vendes/1
14   # GET /vendes/1.json
15   def show
16     @vendes = Vende.find([params[:id1], params[:id2], params[:id3]])
17   end
18
19   # GET /vendes/new
20   def new
21     @modo_edicao = false
22     @vende = Vende.new
23   end
24
25   # GET /vendes/1/edit
26   def edit
27     @modo_edicao = true
28     @vende = Vende.find([params[:id1], params[:id2], params[:id3]])
29   end
30
31   # POST /vendes
32   # POST /vendes.json
33   def create

```

```

34     if(params[:numero][:item].length <= 0)
35         problema = '    necess rio definir uma quantidade para a compra.'
36         redirect_to vende_new_path, notice: problema
37         return
38     end
39     @vende = Vende.new(vende_params)
40     @vende.fornecedor = @fornecedores[params[:selecao][:fornecedor].to_i]
41     @vende.item = @item[params[:selecao][:item].to_i]
42
43     respond_to do |format|
44         if @vende.save
45             @vende.item.quantidade += params[:numero][:item].to_i
46             @vende.item.save
47             format.html { redirect_to item_index_path, notice: 'A compra foi registrada
48                             com sucesso.' }
49             format.json { render :show, status: :created, location: @vende }
50         else
51             format.html { render :new }
52             format.json { render json: @vende.errors, status: :unprocessable_entity }
53         end
54     end
55
56     # PATCH/PUT /vendes/1
57     # PATCH/PUT /vendes/1.json
58     def update
59         @vende = Vende.find([params[:id1], params[:id2], params[:id3]])
60         respond_to do |format|
61             if @vende.update(vende_params)
62                 format.html { redirect_to vende_show_path(@vende.idVende, @vende.idFornecedor,
63                     @vende.idItem), notice: 'A compra foi atualizada com sucesso.' }
64                 format.json { render :show, status: :ok, location: @vende }
65             else
66                 format.html { render :edit }
67                 format.json { render json: @vende.errors, status: :unprocessable_entity }
68             end
69         end
70     end
71
72     # DELETE /vendes/1
73     # DELETE /vendes/1.json
74     def destroy
75         @vende = Vende.find([params[:id1],params[:id2],params[:id3]])
76         @vende.destroy
77         respond_to do |format|
78             format.html { redirect_to vende_index_path, notice: 'A compra foi removida com sucesso
79                             .' }
80             format.json { head :no_content }
81         end
82     end
83
84     # GET /items
85     # GET /items.json
86     def item_index
87         #@items = Item.all
88
89         if (params[item_index_path].nil? ||
90             params[item_index_path][:nome].empty?)
91             nome = nil
92         else
93             nome = params[item_index_path][:nome]
94         end
95         if nome != nil
96             @items = Item.where("nome like '%#{nome}%'")
97         else
98             @items = Item.all
99         end
100     end

```

```

99
100 # GET /items/1
101 # GET /items/1.json
102 def item_show
103   @item = Item.find(params[:id])
104 end
105
106 # GET /items/new
107 def item_new
108   @item = Item.new
109 end
110
111 # GET /items/1/edit
112 def item_edit
113   @item = Item.find(params[:id])
114 end
115
116 def relatorio_itens_falta
117   @item = Item.where("quantidade < quantidadeMinima")
118 end
119
120 # POST /items
121 # POST /items.json
122 def item_create
123   @item = Item.new(item_params)
124   respond_to do |format|
125     if @item.save
126       format.html { redirect_to item_index_path, notice: 'O item foi criado com sucesso.' }
127       format.json { render :show, status: :created, location: @item }
128     else
129       format.html { render :item_new }
130       format.json { render json: @item.errors, status: :unprocessable_entity }
131     end
132   end
133 end
134
135 # PATCH/PUT /items/1
136 # PATCH/PUT /items/1.json
137 def item_update
138   @item = Item.find(params[:id])
139   respond_to do |format|
140     if @item.update(item_params)
141       format.html { redirect_to item_show_path(@item), notice: 'O item foi atualizado com sucesso.' }
142       format.json { render :show, status: :ok, location: @item }
143     else
144       format.html { render :edit }
145       format.json { render json: @item.errors, status: :unprocessable_entity }
146     end
147   end
148 end
149
150 # DELETE /items/1
151 # DELETE /items/1.json
152 def item_destroy
153   @item = Item.find(params[:id])
154   @item.destroy
155   respond_to do |format|
156     format.html { redirect_to item_index_path, notice: 'O item foi removido com sucesso.' }
157     format.json { head :no_content }
158   end
159 end
160
161 private
162 # Use callbacks to share common setup or constraints between actions.
163 def set_vende

```

```

164     @vende = Vende.find([params[:id1], params[:id2], params[:id3]])
165   end
166
167   # Never trust parameters from the scary internet, only allow the white list through.
168   def vende_params
169     params.require(:vende).permit(:idFornecedor, :idItem, :data, :valor)
170   end
171
172   def verificar_login
173     if(not logged_in?)
174       redirect_to login_path
175     end
176   end
177
178   def set_info()
179     @fornecedores = Fornecedor.order(:nome)
180     @item = Item.order(:nome)
181   end
182
183   # Use callbacks to share common setup or constraints between actions.
184   def set_item
185     @item = Item.find(params[:id])
186   end
187
188   # Never trust parameters from the scary internet, only allow the white list through.
189   def item_params
190     params.require(:item).permit(:nome, :marca, :descricao, :quantidade, :quantidadeMinima
191     )
192   end
193 end

```

4.3.4 Fornecedor

```

1 class FornecedoresController < ApplicationController
2   before_action :set_fornecedor, only: [:show, :edit, :update, :destroy, :relatorio]
3
4   # GET /fornecedores
5   # GET /fornecedores.json
6   def index
7     @fornecedores = Fornecedor.all
8   end
9
10  # GET /fornecedores/1
11  # GET /fornecedores/1.json
12  def show
13  end
14
15  # GET /fornecedores/new
16  def new
17    @fornecedor = Fornecedor.new
18  end
19
20  # GET /fornecedores/1/edit
21  def edit
22  end
23
24  def relatorio
25
26  end
27
28  def search
29    if (params[fornecedores_search_path].nil? ||
30        params[fornecedores_search_path][:nome].empty?)
31      nome = nil
32    else
33      nome = params[fornecedores_search_path][:nome]

```

```

34   end
35   if nome != nil
36     @fornecedores = Fornecedor.where("nome like '#{nome}')"
37   else
38     @fornecedores = Fornecedor.all
39   end
40 end
41
42 # POST /fornecedores
43 # POST /fornecedores.json
44 def create
45   @fornecedor = Fornecedor.new(fornecedor_params)
46
47   respond_to do |format|
48     if @fornecedor.save
49       format.html { redirect_to @fornecedor, notice: 'Fornecedor was successfully created' }
50       format.json { render :show, status: :created, location: @fornecedor }
51     else
52       format.html { render :new }
53       format.json { render json: @fornecedor.errors, status: :unprocessable_entity }
54     end
55   end
56 end
57
58 # PATCH/PUT /fornecedores/1
59 # PATCH/PUT /fornecedores/1.json
60 def update
61   respond_to do |format|
62     if @fornecedor.update(fornecedor_params)
63       format.html { redirect_to @fornecedor, notice: 'Fornecedor was successfully updated' }
64       format.json { render :show, status: :ok, location: @fornecedor }
65     else
66       format.html { render :edit }
67       format.json { render json: @fornecedor.errors, status: :unprocessable_entity }
68     end
69   end
70 end
71
72 # DELETE /fornecedores/1
73 # DELETE /fornecedores/1.json
74 def destroy
75   @fornecedor.destroy
76   respond_to do |format|
77     format.html { redirect_to fornecedores_url, notice: 'Fornecedor was successfully destroyed.' }
78     format.json { head :no_content }
79   end
80 end
81
82 private
83 # Use callbacks to share common setup or constraints between actions.
84 def set_fornecedor
85   @fornecedor = Fornecedor.find(params[:id])
86 end
87
88 # Never trust parameters from the scary internet, only allow the white list through.
89 def fornecedor_params
90   params.require(:fornecedor).permit(:nome, :cnpj, :cidade, :estado, :num, :endereco, :
91     telefone)
92 end

```

4.3.5 Funcionario

```

1 class FuncionariosController < ApplicationController
2   include ApplicationHelper
3   before_action only: [:new, :create, :index, :update, :destroy] do redirecionar_privilegio
4     (0) end
5   before_action :privilegio_funcionario, only: [:show, :edit]
6   before_action :set_funcionario, only: [:show, :edit, :update, :destroy]
7
8   # GET /funcionarios
9   # GET /funcionarios.json
10  def index
11    @funcionarios = Funcionario.all
12  end
13
14  # GET /funcionarios/1
15  # GET /funcionarios/1.json
16  def show
17  end
18
19  # GET /funcionarios/new
20  def new
21    @funcionario = Funcionario.new
22  end
23
24  # GET /funcionarios/1/edit
25  def edit
26  end
27
28  def demitir
29    @funcionario = Funcionario.find(params[:id])
30    @funcionario.ativo = false
31    respond_to do |format|
32      if @funcionario.update(@funcionario.attributes)
33        format.html { redirect_to @funcionario, notice: 'funcionario was successfully fired' }
34        format.json { render :show, status: :ok, location: @funcionario }
35      else
36        format.html { render :edit }
37        format.json { render json: @funcionario.errors, status: :unprocessable_entity }
38      end
39    end
40  end
41
42  def search
43    if (params[funcionarios_search_path].nil? ||
44        params[funcionarios_search_path][:nome].empty?)
45      nome = nil
46    else
47      nome = params[funcionarios_search_path][:nome]
48    end
49    if nome != nil
50      @funcionarios = Funcionario.where("nome like '#{nome}%'")
51    else
52      @funcionarios = Funcionario.all
53    end
54  end
55
56  # POST /funcionarios
57  # POST /funcionarios.json
58  def create
59    @funcionario = Funcionario.new(funcionario_params)
60
61    respond_to do |format|
62      if @funcionario.save
63        format.html { redirect_to @funcionario, notice: 'Funcionario was successfully created.' }
64        format.json { render :show, status: :created, location: @funcionario }
65      else
66        format.html { render :new }
67      end
68    end
69  end
70 end

```



```

66         format.json { render json: @funcionario.errors, status: :unprocessable_entity }
67     end
68 end
69 end
70
71 # PATCH/PUT /funcionarios/1
72 # PATCH/PUT /funcionarios/1.json
73 def update
74     respond_to do |format|
75         if @funcionario.update(funcionario_params)
76             format.html { redirect_to @funcionario, notice: 'funcionario was successfully
77                 updated.' }
78             format.json { render :show, status: :ok, location: @funcionario }
79         else
80             format.html { render :edit }
81             format.json { render json: @funcionario.errors, status: :unprocessable_entity }
82         end
83     end
84 end
85
86 # DELETE /funcionarios/1
87 # DELETE /funcionarios/1.json
88 def destroy
89     @funcionario.destroy
90     respond_to do |format|
91         format.html { redirect_to funcionarios_url, notice: 'Funcionario was successfully
92             destroyed.' }
93         format.json { head :no_content }
94     end
95 end
96
97 private
98 def privilegio_funcionario
99     redirecionar_privilegio(0) if(Funcionario.find(params[:id]) !=
100         current_funcionario)
101 end
102
103 # Use callbacks to share common setup or constraints between actions.
104 def set_funcionario
105     @funcionario = Funcionario.find(params[:id])
106 end
107
108 # Never trust parameters from the scary internet, only allow the white list through.
109 def funcionario_params
110     params.require(:funcionario).permit(:nome, :cpf, :cidade, :estado, :num, :endereco, :
111         telefone, :salario, :dataContratacao, :ativo, :turno, :login, :password, :
112         password_confirmation, :nivelAcesso)
113 end
114 end

```

4.3.6 Pedido

```

1 class PedidosController < ApplicationController
2     before_action :set_pedido, only: [:show, :edit, :update, :destroy]
3
4     # GET /pedidos
5     # GET /pedidos.json
6     def index
7         @pedidos = Pedido.all
8     end
9
10    # GET /pedidos/1
11    # GET /pedidos/1.json
12    def show
13    end
14
15    # GET /pedidos/new

```

```

16 def new
17   @pedido = Pedido.new
18 end
19
20 # GET /pedidos/1/edit
21 def edit
22 end
23
24 # POST /pedidos
25 # POST /pedidos.json
26 def create
27   @pedido = Pedido.new(pedido_params)
28
29   respond_to do |format|
30     if @pedido.save
31       format.html { redirect_to @pedido, notice: 'Pedido was successfully created.' }
32       format.json { render :show, status: :created, location: @pedido }
33     else
34       format.html { render :new }
35       format.json { render json: @pedido.errors, status: :unprocessable_entity }
36     end
37   end
38 end
39
40 # PATCH/PUT /pedidos/1
41 # PATCH/PUT /pedidos/1.json
42 def update
43   respond_to do |format|
44     if @pedido.update(pedido_params)
45       format.html { redirect_to @pedido, notice: 'Pedido was successfully updated.' }
46       format.json { render :show, status: :ok, location: @pedido }
47     else
48       format.html { render :edit }
49       format.json { render json: @pedido.errors, status: :unprocessable_entity }
50     end
51   end
52 end
53
54 # DELETE /pedidos/1
55 # DELETE /pedidos/1.json
56 def destroy
57   @pedido.destroy
58   respond_to do |format|
59     format.html { redirect_to pedidos_url, notice: 'Pedido was successfully destroyed.' }
60     format.json { head :no_content }
61   end
62 end
63
64 private
65 # Use callbacks to share common setup or constraints between actions.
66 def set_pedido
67   @pedido = Pedido.find(params[:id])
68 end
69
70 # Never trust parameters from the scary internet, only allow the white list through.
71 def pedido_params
72   params.require(:pedido).permit(:idPedido, :idFuncionario, :idProduto, :idcomanda, :
73     dataPedido, :dataFinalizacao, :cancelado)
74 end
75 end

```

4.3.7 Produto

```

1 # coding: utf-8
2 require 'json'
3 class ProdutosController < ApplicationController

```

```

4   include ApplicationHelper
5   before_action only: [:new, :create, :update, :edit, :destroy] do redirecionar_privilegio
      (0) end
6   before_action do redirecionar_privilegio(2) end
7   before_action :set_produto, only: [:show, :edit, :update, :destroy]
8   before_action :set_componentes_produto, only: [:show, :edit, :update, :destroy]
9
10  # GET /produtos
11  # GET /produtos.json
12  def index
13    @produtos = Produto.all.order :nome
14  end
15
16  # GET /produtos/1
17  # GET /produtos/1.json
18  def show
19  end
20
21  # GET /produtos/new
22  def new
23    @produto = Produto.new
24    @componentes_produto = ComponentesProduto.new
25    @items = Item.all
26  end
27
28  # GET /produtos/1/edit
29  def edit
30  end
31
32  # POST /produtos
33  # POST /produtos.json
34  def create
35    Produto.transaction do
36      @produto = Produto.new(produto_params)
37      items = JSON.parse(params["items"])
38      respond_to do |format|
39        if @produto.save
40          items.each do |it|
41            componentes_produto = ComponentesProduto.new
42            componentes_produto.idProduto = @produto.idProduto
43            componentes_produto.idItem = it["id"]
44            componentes_produto.quantidade = it["quantidade"]
45            componentes_produto.save
46          end
47          format.html { redirect_to @produto, notice: 'Produto criado com sucesso.' }
48          format.json { render :show, status: :created, location: @produto }
49        else
50          format.html { render :new }
51          format.json { render json: @produto.errors, status: :unprocessable_entity }
52        end
53      end
54    end
55  end
56
57  # PATCH/PUT /produtos/1
58  # PATCH/PUT /produtos/1.json
59  def update
60    respond_to do |format|
61      if @produto.update(produto_params)
62        format.html { redirect_to @produto, notice: 'Produto foi atualizado com sucesso.' }
63        format.json { render :show, status: :ok, location: @produto }
64      else
65        format.html { render :edit }
66        format.json { render json: @produto.errors, status: :unprocessable_entity }
67      end
68    end
69  end
70

```

```

71 # DELETE /produtos/1
72 # DELETE /produtos/1.json
73 def destroy
74
75   @produto.destroy
76   respond_to do |format|
77     format.html { redirect_to produtos_url, notice: 'Produto foi deletado com sucesso.' }
78     format.json { head :no_content }
79   end
80 end
81
82
83 # Controlador Componente Produto.
84
85 # GET /componentes_produtos
86 # GET /componentes_produtos.json
87 def componentes_index
88   @componentes_produtos = ComponentesProduto.all
89 end
90
91 # GET /componentes_produtos/1
92 # GET /componentes_produtos/1.json
93 def componentes_show
94 end
95
96 # GET /componentes_produtos/new
97 def componentes_new
98   @componentes_produto = ComponentesProduto.new
99 end
100
101 # GET /componentes_produtos/1/edit
102 def componentes_edit
103 end
104
105 def componentes_adicionar
106 end
107
108
109 # POST /componentes_produtos
110 # POST /componentes_produtos.json
111 def componentes_create
112   puts "Come a"
113   puts params
114   teste = JSON.parse(params["items"])
115   # @lista_componentes.push({idItem: params['componentes_produto']['idItem'],
116   #                           quantidade: params['componentes_produto']['quantidade']})
117   # @lista_componentes.each do |x|
118   #   puts "Valor: #{x} ."
119   # end
120   puts "termina"
121
122   respond_to do |format|
123     format.html { redirect_to :back, notice: 'Componente criado com sucesso.' }
124     #format.json { render :componentes_show, status: :created, location:
125                   @componentes_produto }
126   end
127
128   # @componentes_produto = ComponentesProduto.new(componentes_produto_params)
129
130   # respond_to do |format|
131   #   if @componentes_produto.save
132   #     format.html { redirect_to :back, notice: 'Componente criado com sucesso.' }
133   #     format.json { render :show, status: :created, location: @componentes_produto }
134   #   else
135   #     format.html { render :new }
136   #     format.json { render json: @componentes_produto.errors, status: :
137                       unprocessable_entity }
138   #   end

```

```

137     # end
138 end
139
140 # PATCH/PUT /componentes_produtos/1
141 # PATCH/PUT /componentes_produtos/1.json
142 def componentes_update
143   respond_to do |format|
144     if @componentes_produto.update(componentes_produto_params)
145       format.html { redirect_to @componentes_produto, notice: 'Componente foi atualizado
146         com sucesso' }
147       format.json { render :show, status: :ok, location: @componentes_produto }
148     else
149       format.html { render :edit }
150       format.json { render json: @componentes_produto.errors, status: :
151         unprocessable_entity }
152     end
153   end
154 end
155
156 # DELETE /componentes_produtos/1
157 # DELETE /componentes_produtos/1.json
158 def componentes_destroy
159   @componentes_produto = ComponentesProduto.find(params[:id])
160   @componentes_produto.destroy
161   respond_to do |format|
162     format.html { redirect_to :back, notice: 'Componente foi deletado com sucesso.' }
163     format.json { head :no_content }
164   end
165 end
166
167 private
168 # Use callbacks to share common setup or constraints between actions.
169 def set_componentes_produto
170   #@componentes_produto = ComponentesProduto.find(params[:id])
171 end
172
173 # Never trust parameters from the scary internet, only allow the white list through.
174 def componentes_produto_params
175   params.require(:componentes_produto).permit(:idCompoe, :idProduto, :idItem, :
176     quantidade)
177 end
178
179 def set_produto
180   @produto = Produto.find(params[:id])
181 end
182
183 # Never trust parameters from the scary internet, only allow the white list through.
184 def produto_params
185   params.require(:produto).permit(:nome, :valor, :descricao)
186 end
187 end

```

4.4 Banco de Dados

```

1  --
2  -- Tabela 'comanda'
3  --
4
5  DROP TABLE IF EXISTS 'comanda';
6  CREATE TABLE 'comanda' (
7    'idcomanda' int(11) NOT NULL AUTO_INCREMENT,
8    'idFuncionario' int(11) NOT NULL,
9    'mesa' int(11) NOT NULL,
10   'dataAbertura' datetime NOT NULL,
11   'dataEncerramento' datetime DEFAULT NULL,
12   'valorFinal' float DEFAULT NULL,

```

```

13 PRIMARY KEY ('idcomanda'),
14 KEY 'fk_comanda_Funcionario1_idx' ('idFuncionario'),
15 CONSTRAINT 'fk_comanda_Funcionario1' FOREIGN KEY ('idFuncionario') REFERENCES 'funcionario'
16 ) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8;
17
18 --
19 -- Tabela 'componentes_produto'
20 --
21
22 DROP TABLE IF EXISTS 'componentes_produto';
23 CREATE TABLE 'componentes_produto' (
24 'idCompoe' int(11) NOT NULL AUTO_INCREMENT,
25 'idProduto' int(11) NOT NULL,
26 'idItem' int(11) NOT NULL,
27 'quantidade' int(11) DEFAULT '1',
28 PRIMARY KEY ('idCompoe'),
29 KEY 'fk_Produto_has_Item_Item1_idx' ('idItem'),
30 KEY 'fk_Produto_has_Item_Produto1_idx' ('idProduto'),
31 CONSTRAINT 'fk_Produto_has_Item_Item1' FOREIGN KEY ('idItem') REFERENCES 'item' ('idItem')
32 ON DELETE CASCADE ON UPDATE NO ACTION,
33 CONSTRAINT 'fk_Produto_has_Item_Produto1' FOREIGN KEY ('idProduto') REFERENCES 'produto'
34 ('idProduto') ON DELETE NO ACTION ON UPDATE NO ACTION
35 ) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=utf8;
36
37 --
38 -- Tabela 'fornecedor'
39 --
40
41 DROP TABLE IF EXISTS 'fornecedor';
42 CREATE TABLE 'fornecedor' (
43 'idFornecedor' int(11) NOT NULL AUTO_INCREMENT,
44 'nome' varchar(45) NOT NULL,
45 'cnpj' varchar(11) NOT NULL,
46 'cidade' varchar(45) NOT NULL,
47 'estado' varchar(45) NOT NULL,
48 'num' varchar(45) DEFAULT NULL,
49 'endereco' varchar(45) NOT NULL,
50 'telefone' varchar(11) NOT NULL,
51 PRIMARY KEY ('idFornecedor')
52 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
53
54 --
55 -- Tabela table 'funcionario'
56 --
57
58 DROP TABLE IF EXISTS 'funcionario';
59 CREATE TABLE 'funcionario' (
60 'idFuncionario' int(11) NOT NULL AUTO_INCREMENT,
61 'nome' varchar(45) NOT NULL,
62 'cpf' varchar(9) NOT NULL,
63 'cidade' varchar(45) NOT NULL,
64 'estado' varchar(45) NOT NULL,
65 'num' varchar(5) NOT NULL,
66 'endereco' varchar(45) NOT NULL,
67 'telefone' varchar(11) NOT NULL,
68 'salario' float NOT NULL,
69 'dataContratacao' date NOT NULL,
70 'ativo' tinyint(1) NOT NULL,
71 'turno' int(11) NOT NULL,
72 'login' varchar(45) NOT NULL,
73 'password_digest' varchar(60) NOT NULL,
74 'nivelAcesso' int(1) NOT NULL,
75 PRIMARY KEY ('idFuncionario'),
76 UNIQUE KEY 'login_UNIQUE' ('login'),
77 UNIQUE KEY 'cpf_UNIQUE' ('cpf')
78 ) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;

```

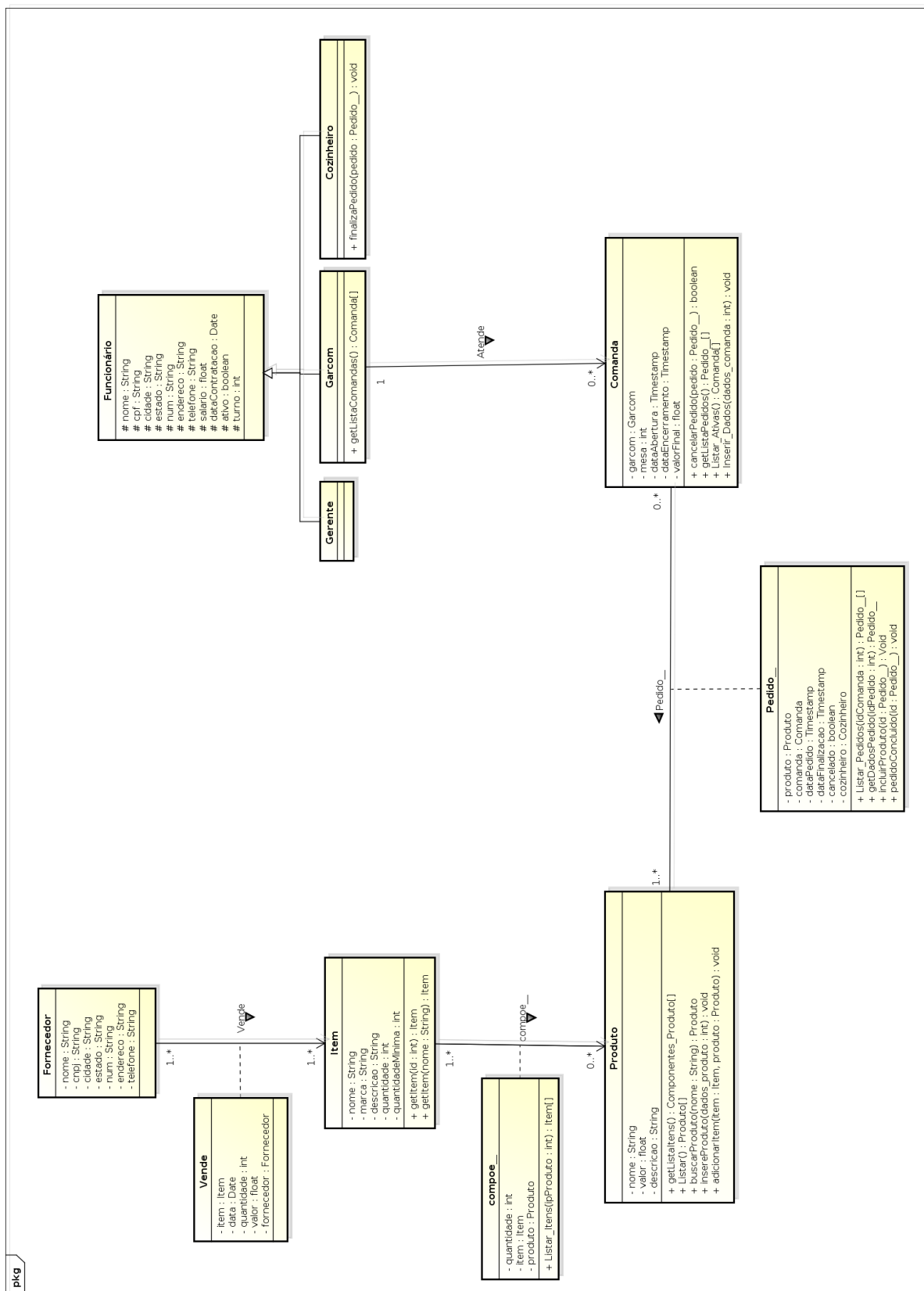
```

78 --
79 -- Tabela 'item'
80 --
81
82 DROP TABLE IF EXISTS 'item';
83 CREATE TABLE 'item' (
84     'idItem' int(11) NOT NULL AUTO_INCREMENT,
85     'nome' varchar(45) NOT NULL,
86     'marca' varchar(45) DEFAULT NULL,
87     'descricao' varchar(120) DEFAULT NULL,
88     'quantidade' int(11) DEFAULT '1',
89     'quantidadeMinima' int(11) DEFAULT NULL,
90     PRIMARY KEY ('idItem')
91 ) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8;
92
93 --
94 -- Tabela 'pedido'
95 --
96
97 DROP TABLE IF EXISTS 'pedido';
98 CREATE TABLE 'pedido' (
99     'idPedido' int(11) NOT NULL AUTO_INCREMENT,
100     'idFuncionario' int(11) NOT NULL,
101     'idProduto' int(11) NOT NULL,
102     'idcomanda' int(11) NOT NULL,
103     'dataPedido' timestamp DEFAULT CURRENT_TIMESTAMP,
104     'dataFinalizacao' datetime DEFAULT NULL,
105     'cancelado' tinyint(1) DEFAULT NULL,
106     PRIMARY KEY ('idPedido'),
107     KEY 'fk_Produto_has_comanda_comanda1_idx' ('idcomanda'),
108     KEY 'fk_Produto_has_comanda_Produto1_idx' ('idProduto'),
109     KEY 'fk_Pedido_Funcionario1_idx' ('idFuncionario'),
110     CONSTRAINT 'fk_Pedido_Funcionario1' FOREIGN KEY ('idFuncionario') REFERENCES 'funcionario'
        ('idFuncionario') ON DELETE CASCADE ON UPDATE NO ACTION,
111     CONSTRAINT 'fk_Produto_has_comanda_Produto1' FOREIGN KEY ('idProduto') REFERENCES 'produto'
        ('idProduto') ON DELETE CASCADE ON UPDATE NO ACTION,
112     CONSTRAINT 'fk_Produto_has_comanda_comanda1' FOREIGN KEY ('idcomanda') REFERENCES 'comanda'
        ('idcomanda') ON DELETE CASCADE ON UPDATE NO ACTION
113 ) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8;
114
115 --
116 -- Tabela 'produto'
117 --
118
119 DROP TABLE IF EXISTS 'produto';
120 CREATE TABLE 'produto' (
121     'idProduto' int(11) NOT NULL AUTO_INCREMENT,
122     'nome' varchar(45) NOT NULL,
123     'valor' float NOT NULL,
124     'descricao' varchar(120) DEFAULT NULL,
125     PRIMARY KEY ('idProduto')
126 ) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8;
127
128 --
129 -- Tabela 'vende'
130 --
131
132 DROP TABLE IF EXISTS 'vende';
133 CREATE TABLE 'vende' (
134     'idVende' int(11) NOT NULL AUTO_INCREMENT,
135     'idFornecedor' int(11) NOT NULL,
136     'idItem' int(11) NOT NULL,
137     'data' date NOT NULL,
138     'valor' float NOT NULL,
139     PRIMARY KEY ('idVende'),
140     UNIQUE KEY 'idVende' ('idVende'),
141     KEY 'fk_Fornecedor_has_Item_Item1_idx' ('idItem'),
142     KEY 'fk_Fornecedor_has_Item_Fornecedor_idx' ('idFornecedor'),

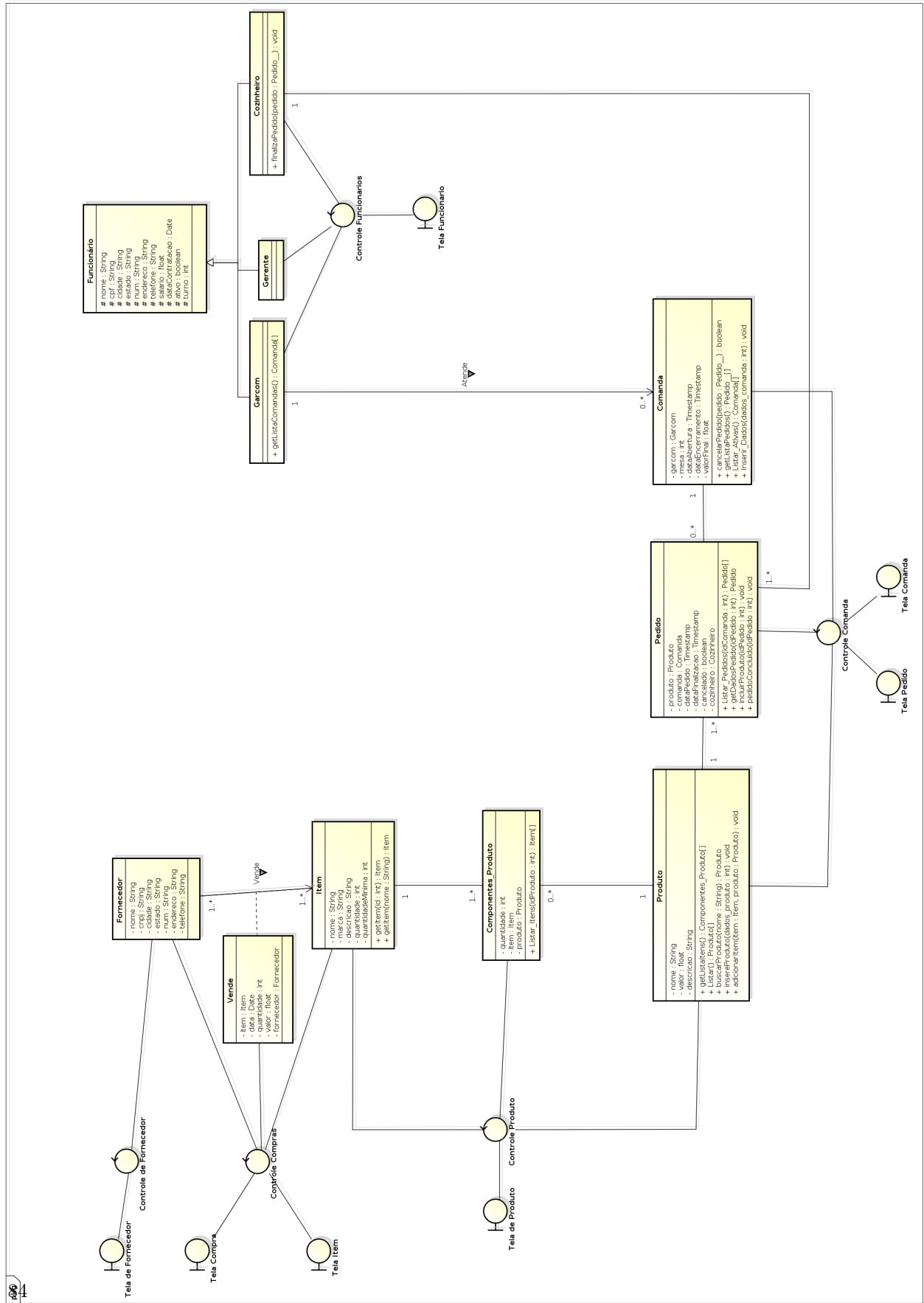
```

```
143 CONSTRAINT 'fk_Fornecedor_has_Item_Fornecedor' FOREIGN KEY ('idFornecedor') REFERENCES '
    fornecedor' ('idFornecedor') ON DELETE NO ACTION ON UPDATE NO ACTION,
144 CONSTRAINT 'fk_Fornecedor_has_Item_Item1' FOREIGN KEY ('idItem') REFERENCES 'item' ('
    idItem') ON DELETE CASCADE ON UPDATE NO ACTION
145 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```


5.1 Diagrama de Classe de Análise

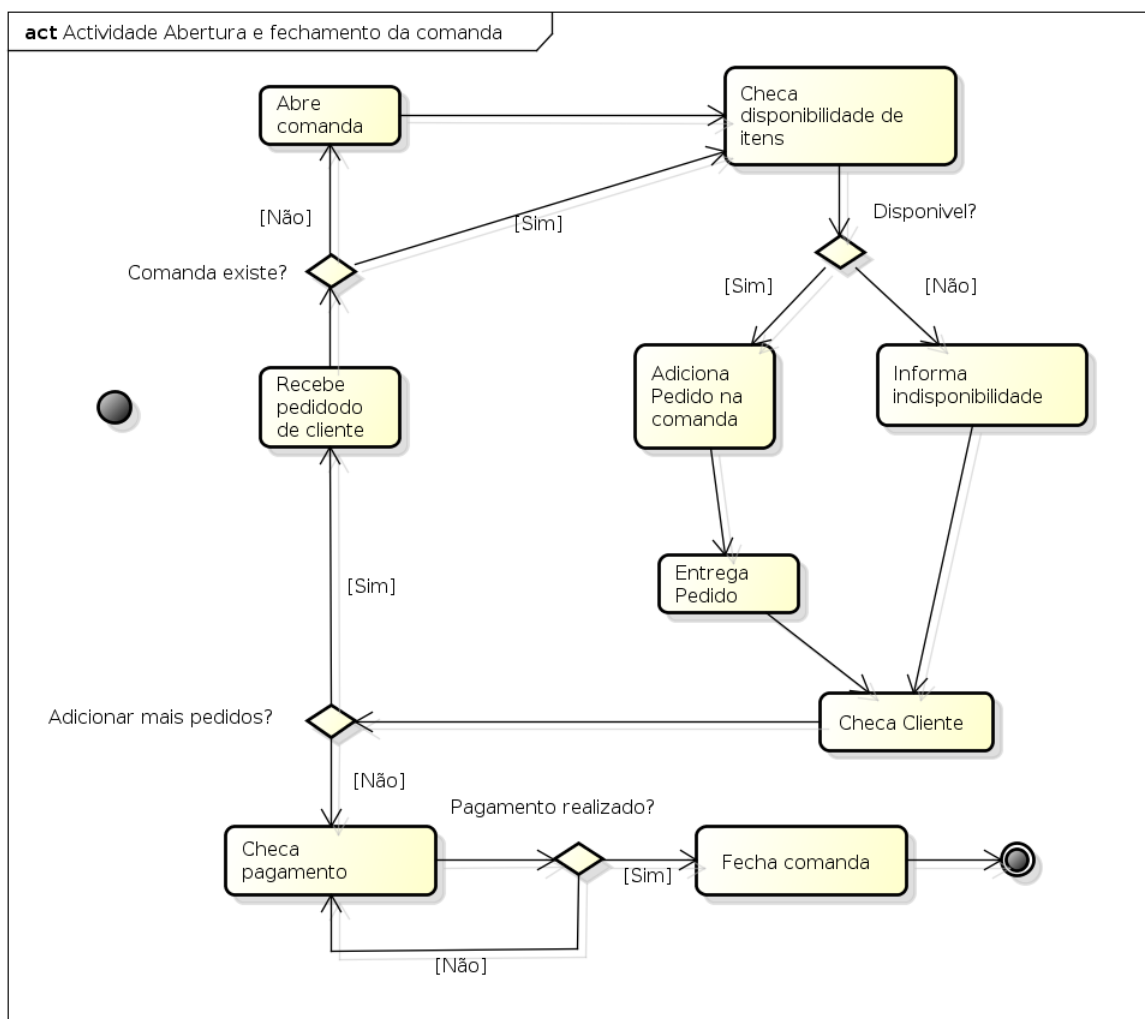


5.2 Diagrama de Classe de Projeto



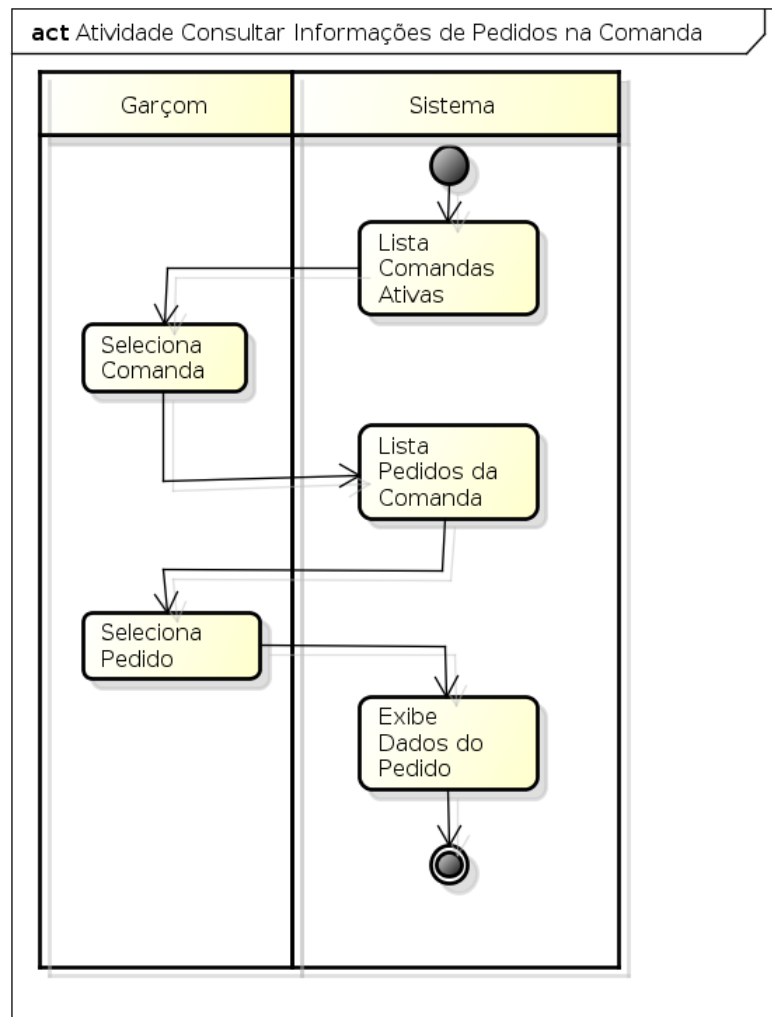
5.3 Diagrama de Atividade

5.3.1 Abertura e fechamento da comanda



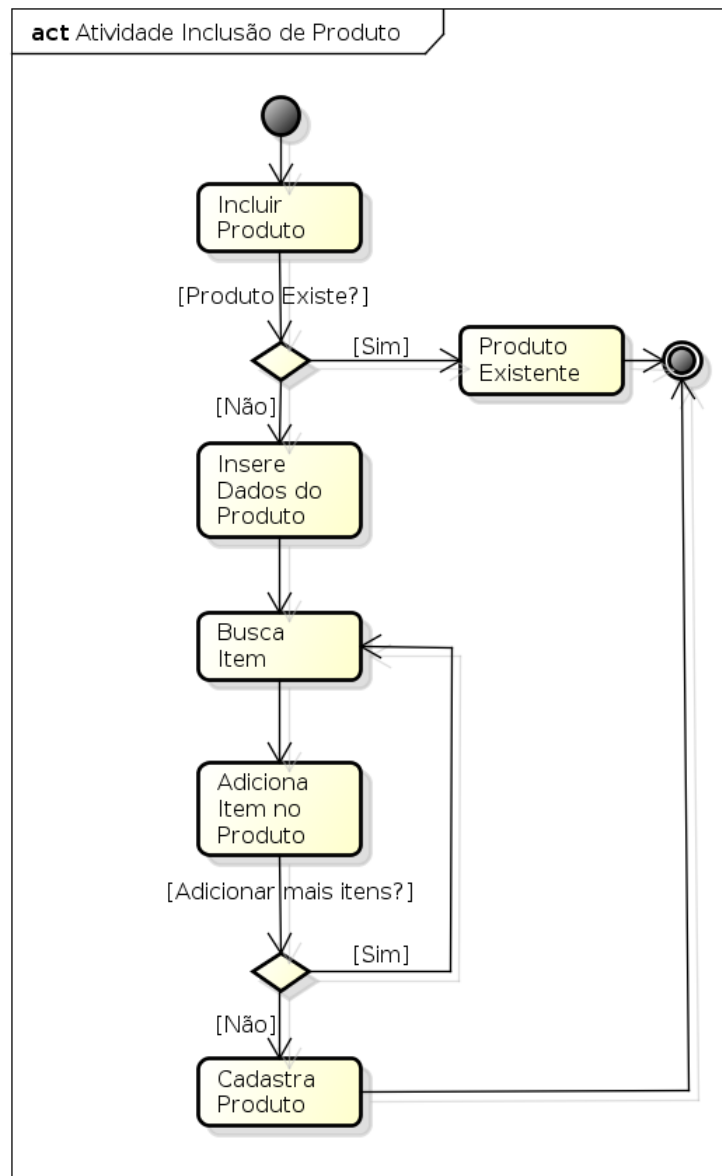
powered by Astah

5.3.2 Consultar informações de pedidos na comanda



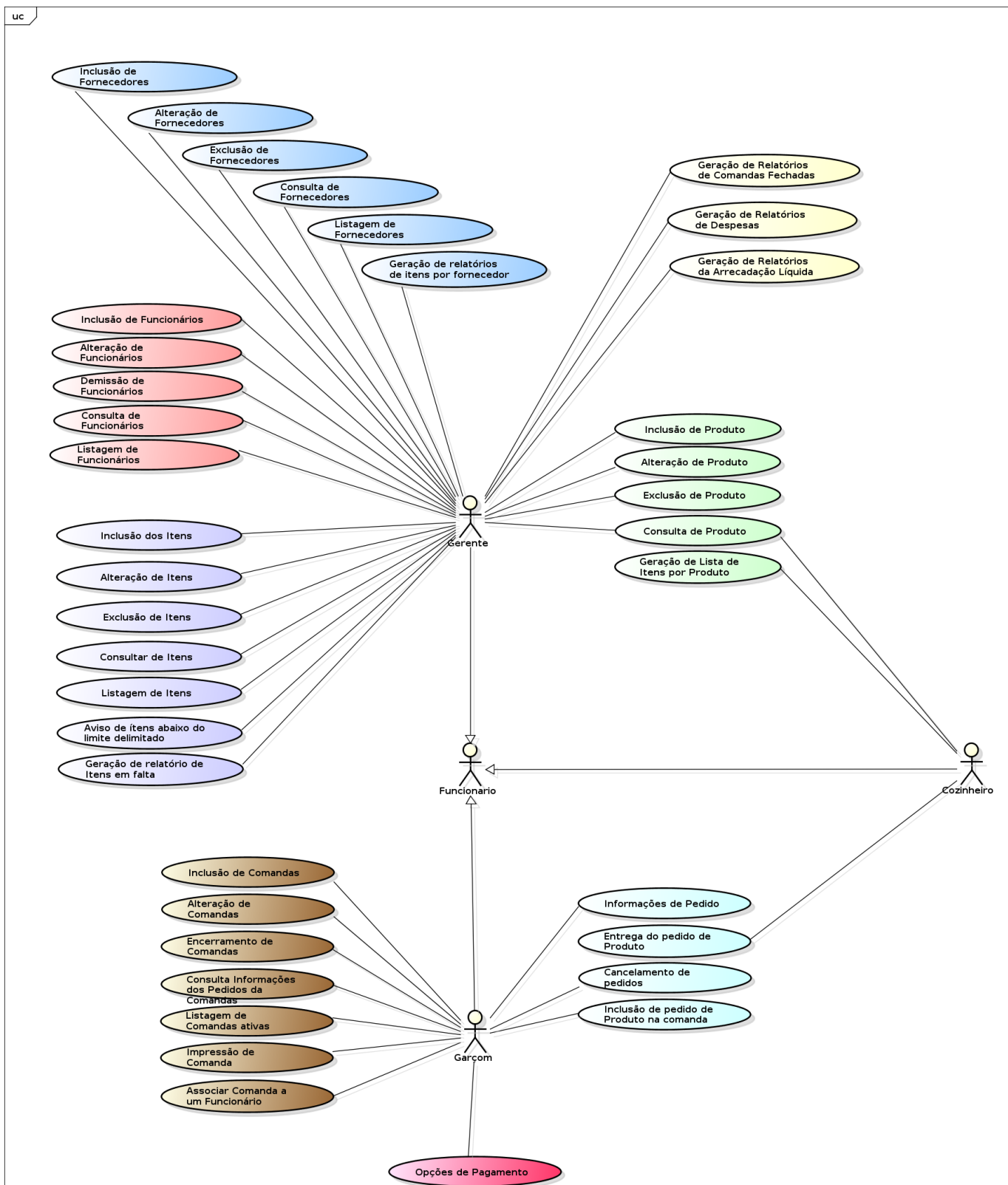
powered by Astah

5.3.3 Inclusão de produto



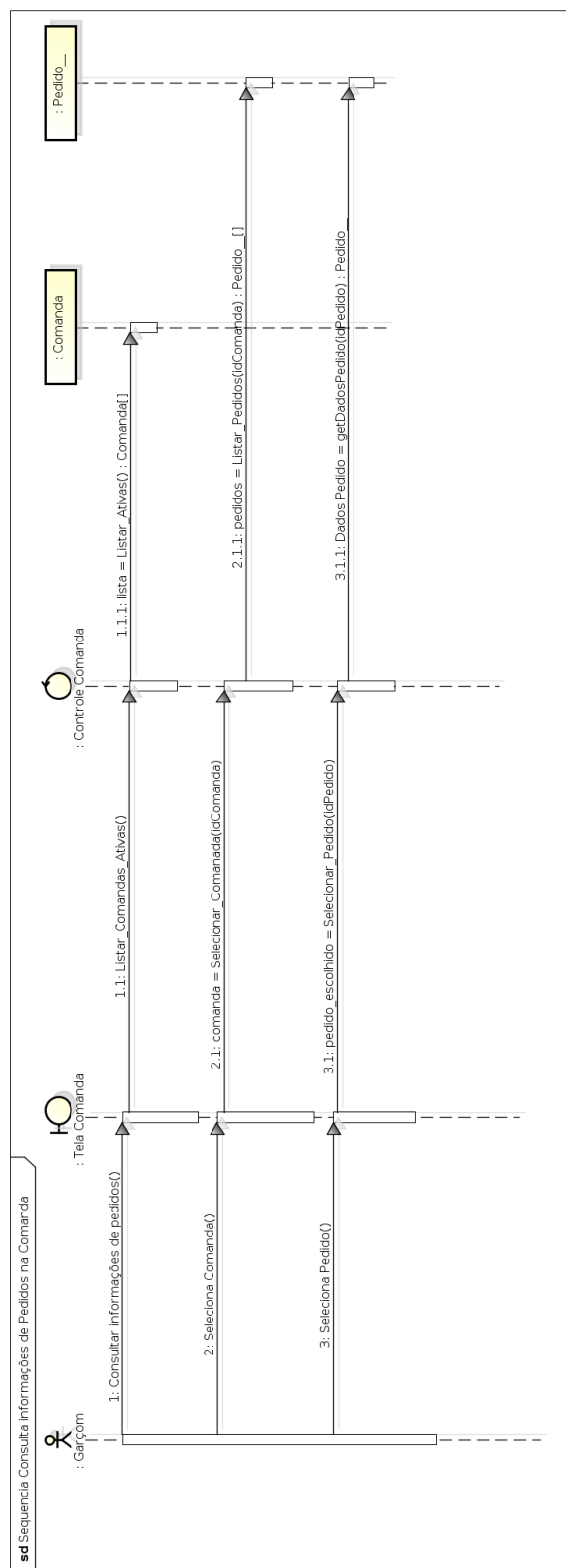
powered by Astah

5.4 Diagrama de caso de uso

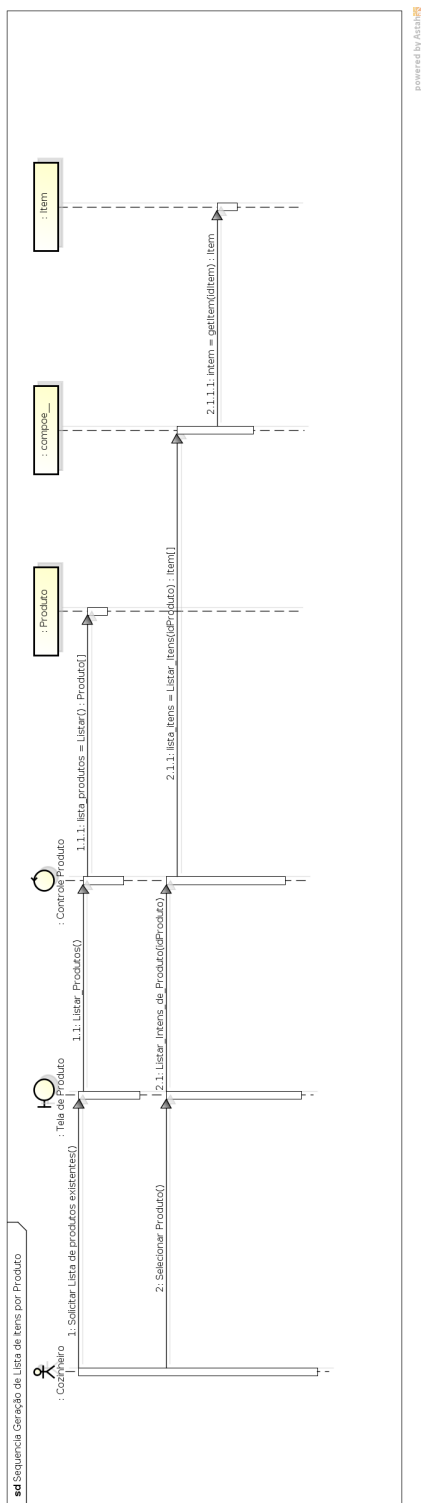


5.5 Diagramas de Sequência

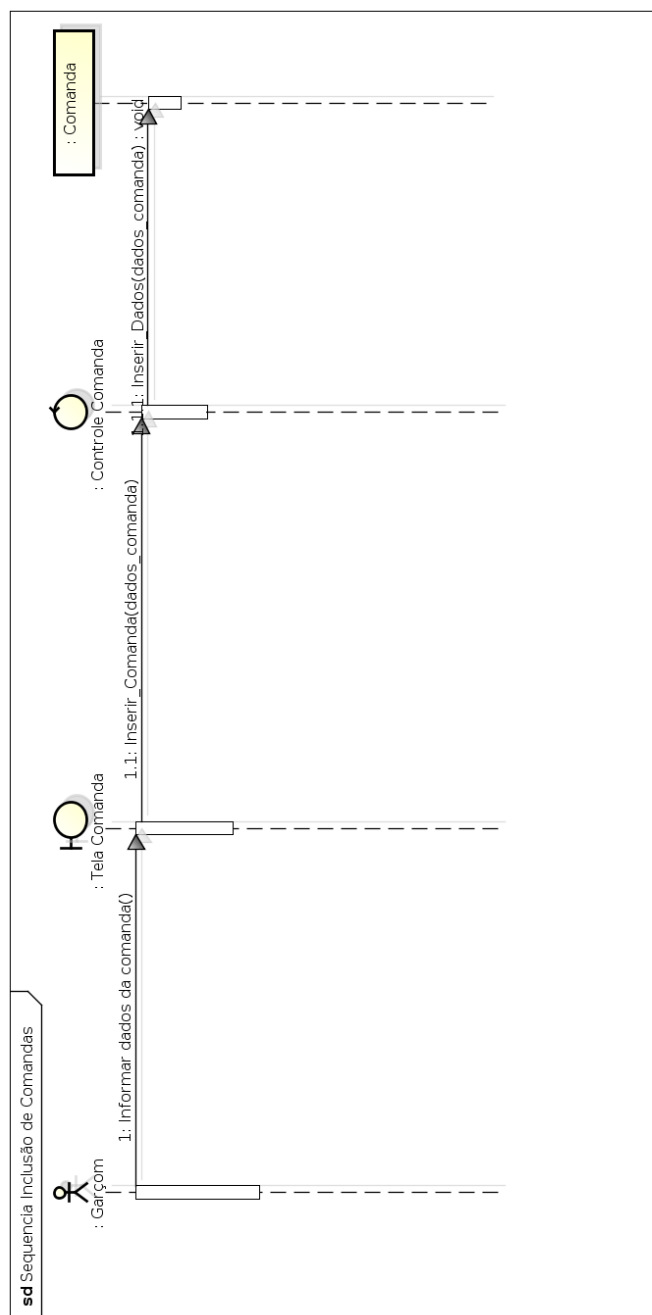
5.5.1 Consultar informações de pedidos na comanda



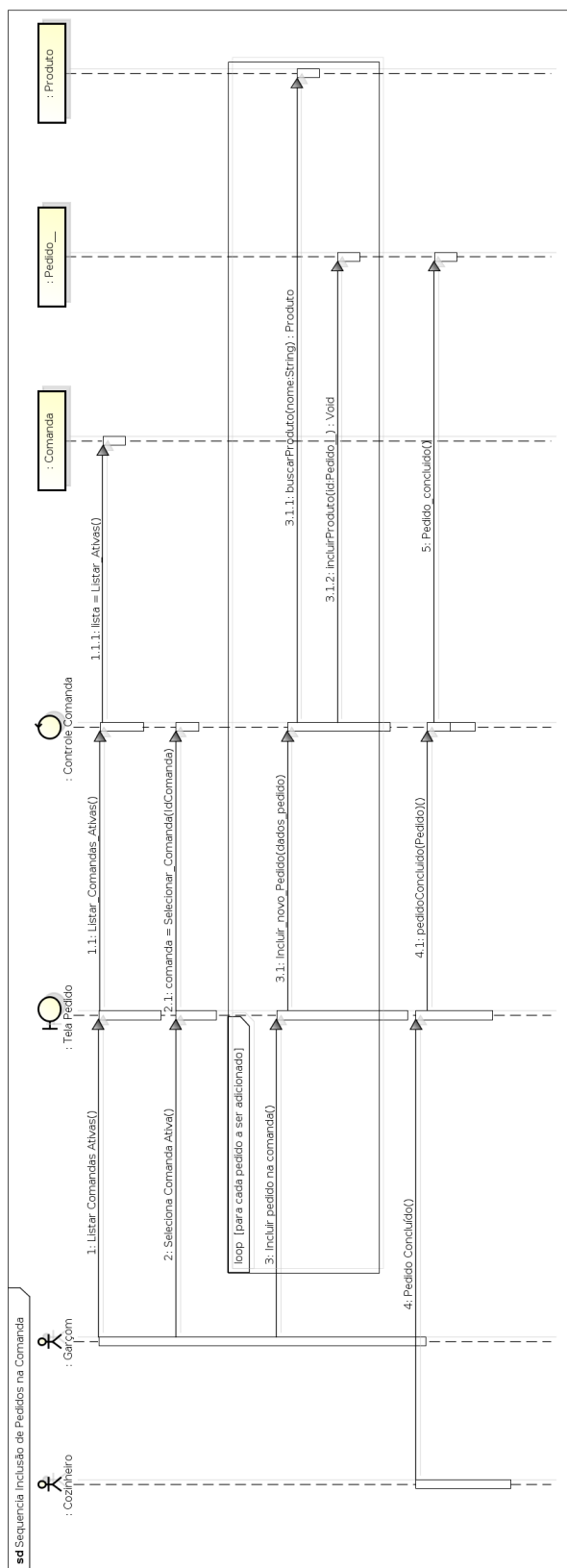
5.5.2 Geração de Lista de itens por produto



5.5.3 Inclusão de comanda



5.5.4 Inclusão de pedido na comanda



5.5.5 Inclusão de produto

