

Integração Frontend - FASE 3

Resumo da Integração

Esta documentação descreve a integração completa da FASE 3 do backend ao frontend React, incluindo:

- Painel Admin com Dashboard, Gestão de Pedidos e Cupons
- Páginas Cliente (Meus Pedidos e Minhas Avaliações)
- Sistema de Notificações em tempo real
- Funcionalidade de Cupons no Checkout
- Sistema de Avaliações de Produtos

Funcionalidades Implementadas

1. Sistema de Notificações

- **Context:** `NotificacoesContext.tsx`
- **Componente:** `NotificacoesDropdown.tsx`
- **Features:**
 - Polling automático a cada 30 segundos
 - Badge com contador de notificações não lidas
 - Dropdown com lista de notificações
 - Marcar como lida (individual ou todas)
 - Deletar notificações
 - Tipos: PEDIDO, PROMOCAO, SISTEMA

2. Painel Admin

2.1 Layout Admin (`AdminLayout.tsx`)

- Sidebar com navegação:
 - Dashboard
 - Pedidos
 - Cupons
 - Produtos (link para gestão existente)
- Header com notificações
- Logout

2.2 Dashboard (`/admin/dashboard`)

Estatísticas:

- Pedidos Hoje/Semana/Mês
- Vendas Hoje/Semana/Mês
- Ticket Médio

Gráficos:

- Vendas dos últimos 30 dias (linha)
- Pedidos por Status (pizza)

Listas:

- Produtos Mais Vendidos
- Pedidos Recentes

2.3 Gestão de Pedidos (/admin/pedidos)**Filtros:**

- Status (dropdown)
- Data Início/Fim
- Busca por nome do cliente
- Paginação

Tabela:

- Número do pedido
- Cliente (nome e email)
- Data e hora
- Status (badge colorido)
- Total
- Ações: Ver Detalhes, Atualizar Status

Modais:

- Detalhes do Pedido (completo)
- Atualizar Status (todos os status disponíveis)

2.4 Gestão de Cupons (/admin/cupons)**Tabela:**

- Código
- Tipo (PERCENTUAL/VALOR_FIXO)
- Valor
- Validade (data início/fim)
- Uso (atual/máximo)
- Status (ativo/inativo)
- Ações: Editar, Deletar

Modal Criar/Editar:

- Código (uppercase automático)
- Tipo de desconto
- Valor
- Data início/fim
- Uso máximo (opcional)
- Validação completa

3. Páginas Cliente**3.1 Meus Pedidos (/meus-pedidos)****Features:**

- Lista de todos os pedidos do cliente
- Card com informações resumidas
- Status colorido
- Botões:
- Ver Detalhes (modal completo)
- Avaliar (se ENTREGUE)
- Cancelar (se PENDENTE)

3.2 Minhas Avaliações (/minhas-avaliacoes)

Features:

- Lista de todas as avaliações feitas
- Card com produto, nota e comentário
- Data da avaliação
- Botão deletar

4. Sistema de Avaliações

4.1 Modal de Avaliação (ModalAvaliacao.tsx)

- Lista todos os produtos do pedido
- Nota de 1-5 estrelas (interativo)
- Comentário opcional
- Validação: pelo menos uma nota

4.2 Componente de Avaliações (AvaliacoesProduto.tsx)

- Média de avaliações
- Total de avaliações
- Lista de avaliações com:
 - Nome do usuário
 - Nota (estrelas)
 - Comentário
 - Data

5. Integração de Cupons no Checkout

CheckoutBackend atualizado:

- Campo de cupom
- Botão “Aplicar Cupom”
- Validação em tempo real
- Exibição do desconto aplicado
- Botão remover cupom
- Recálculo automático do total



Estrutura de Arquivos Criados/Modificados

```

src/
├── services/
│   └── backendApi.ts (ATUALIZADO - novos endpoints)
├── contexts/
│   └── NotificacoesContext.tsx (NOVO)
├── components/
│   ├── NotificacoesDropdown.tsx (NOVO)
│   ├── ModalAvaliacao.tsx (NOVO)
│   └── AvaliacoesProduto.tsx (NOVO)
├── layouts/
│   └── AdminLayout.tsx (NOVO)
├── pages/
│   ├── admin/
│   │   ├── Dashboard.tsx (NOVO)
│   │   ├── Pedidos.tsx (NOVO)
│   │   └── Cupons.tsx (NOVO)
│   ├── cliente/
│   │   ├── MeusPedidos.tsx (NOVO)
│   │   └── MinhasAvaliacoes.tsx (NOVO)
│   └── storefront/
│       └── CheckoutBackend.tsx (ATUALIZADO - cupons)
├── utils/
│   └── statusColors.ts (NOVO)
├── routes/
│   └── AppRouter.tsx (ATUALIZADO - novas rotas)
└── App.tsx (ATUALIZADO - NotificacoesProvider)

```



Cores de Status

PENDENTE: **amarelo** (#FCD34D)
 CONFIRMADO: **azul** (#3B82F6)
 EM_PREPARO: **laranja** (#F97316)
 SAIU_ENTREGA: **roxo** (#8B5CF6)
 ENTREGUE: **verde** (#10B981)
 CANCELADO: **vermelho** (#EF4444)



Credenciais de Teste

Admin Pizza Express

- Email: `admin@pizza-express.com`
- Senha: `pizza123`
- Empresa: Pizza Express

Admin Burger King

- Email: `admin@burger-king.com`
- Senha: `pizza123`
- Empresa: Burger King

Cliente

- Email: `cliente@exemplo.com`
- Senha: `cliente123`

Como Testar

1. Iniciar Backend

```
cd backend  
npm run dev
```

2. Iniciar Frontend

```
cd deliverai-v1  
npm install  
npm run dev
```

3. Fluxo de Teste Admin

3.1 Login e Dashboard

1. Acesse `/login-backend`
2. Login com credenciais admin
3. Navegue para `/admin/dashboard`
4. Verifique:
 - Cards de estatísticas
 - Gráfico de vendas
 - Gráfico de pedidos por status
 - Produtos populares
 - Pedidos recentes

3.2 Gestão de Pedidos

1. Navegue para `/admin/pedidos`
2. Teste filtros:
 - Filtrar por status
 - Filtrar por data
 - Buscar por cliente
3. Clique em “Ver Detalhes” em um pedido
4. Clique em “Atualizar Status”
5. Mude o status do pedido
6. Verifique atualização na lista

3.3 Gestão de Cupons

1. Navegue para `/admin/cupons`
2. Clique em “Criar Cupom”
3. Preencha:
 - Código: `TESTE10`
 - Tipo: Percentual
 - Valor: 10
 - Datas válidas
4. Salve e verifique na lista
5. Teste editar cupom
6. Teste deletar cupom

4. Fluxo de Teste Cliente

4.1 Fazer Pedido com Cupom

1. Logout do admin
2. Login com credenciais cliente
3. Navegue para `/storefront-backend`
4. Adicione produtos ao carrinho
5. Vá para checkout
6. Preencha endereço
7. Digite código do cupom: `TESTE10`
8. Clique em “Aplicar”
9. Verifique desconto aplicado
10. Finalize pedido

4.2 Meus Pedidos

1. Navegue para `/meus-pedidos`
2. Verifique lista de pedidos
3. Clique em “Ver Detalhes”
4. Se houver pedido ENTREGUE, clique em “Avaliar”
5. Avalie produtos com nota e comentário
6. Envie avaliação

4.3 Minhas Avaliações

1. Navegue para `/minhas-avaliacoes`
2. Verifique lista de avaliações
3. Teste deletar uma avaliação

5. Testar Notificações

1. Login como admin
2. Observe o ícone de sino no header
3. Verifique badge com contador
4. Clique no sino
5. Veja lista de notificações
6. Marque uma como lida
7. Clique em “Marcar todas”
8. Delete uma notificação



Endpoints Integrados

Cupons

- `POST /cupons` - Criar cupom
- `GET /cupons` - Listar cupons
- `GET /cupons/:id` - Buscar cupom
- `PUT /cupons/:id` - Atualizar cupom
- `DELETE /cupons/:id` - Deletar cupom
- `POST /cupons/validar` - Validar cupom

Avaliações

- `POST /avaliacoes` - Criar avaliação
- `GET /avaliacoes/produto/:id` - Listar por produto
- `GET /avaliacoes/minhas` - Minhas avaliações
- `DELETE /avaliacoes/:id` - Deletar avaliação

Notificações

- `GET /notificacoes` - Listar notificações
- `GET /notificacoes/nao-lidas` - Não lidas
- `PATCH /notificacoes/:id/lida` - Marcar como lida
- `PATCH /notificacoes/marcar-todas-lidas` - Marcar todas
- `DELETE /notificacoes/:id` - Deletar notificação

Pedidos

- `GET /pedidos` - Listar com filtros
- `GET /pedidos/meus` - Meus pedidos
- `GET /pedidos/:id` - Buscar pedido
- `PATCH /pedidos/:id/status` - Atualizar status
- `PATCH /pedidos/:id/cancelar` - Cancelar pedido

Dashboard

- `GET /dashboard/estatisticas` - Estatísticas gerais
- `GET /dashboard/vendas` - Vendas por período
- `GET /dashboard/produtos-populares` - Produtos populares



Dependências Necessárias

Certifique-se de que estas dependências estão instaladas:

```
{
  "dependencies": {
    "react": "^18.x",
    "react-router-dom": "^6.x",
    "lucide-react": "^0.x",
    "date-fns": "^2.x",
    "recharts": "^2.x",
    "axios": "^1.x"
  }
}
```



Próximos Passos

1. **Testes E2E:** Implementar testes automatizados
2. **Otimizações:**
 - Lazy loading de componentes
 - Memoização de cálculos pesados
 - Debounce em filtros
3. **Melhorias UX:**
 - Skeleton loaders

- Animações de transição
- Feedback visual aprimorado

4. Features Adicionais:

- Exportar relatórios (PDF/Excel)
- Notificações push
- Chat de suporte



Notas Importantes

- **Multi-tenant:** Todas as funcionalidades respeitam o isolamento por empresa
- **Autenticação:** Rotas protegidas por role (ADMIN vs CLIENTE)
- **Responsividade:** Todas as páginas são responsivas
- **Acessibilidade:** Componentes seguem boas práticas de a11y
- **Performance:** Polling de notificações otimizado (30s)
- **Segurança:** Validações no frontend e backend



Troubleshooting

Notificações não aparecem

- Verifique se o usuário está autenticado
- Confirme que o backend está rodando
- Verifique console para erros de API

Cupom não aplica

- Verifique se o cupom está ativo
- Confirme se está dentro da validade
- Verifique se não atingiu uso máximo

Gráficos não carregam

- Instale recharts: `npm install recharts`
- Verifique se há dados no período

Erro de CORS

- Configure CORS no backend para aceitar origem do frontend
- Verifique variáveis de ambiente



Suporte

Para dúvidas ou problemas:

1. Verifique esta documentação
 2. Consulte logs do backend
 3. Verifique console do navegador
 4. Revise código dos componentes
-

Versão: 1.0.0

Data: Outubro 2025

Autor: Equipe Deliverei