



# Regras de negócio e diretrizes de desenvolvimento

## – Projeto Omni.AI

Este documento descreve as regras de negócio, os atores envolvidos e os passos para orientar o desenvolvimento do sistema **Omni.AI**. O objetivo é entregar uma plataforma escalável de venda e suporte omnichannel, permitindo que empresas criem um agente de IA que venda, atenda e fidelize clientes de forma empática, sem exigir conhecimento técnico.

**Observação:** As stacks técnicas e detalhes de implementação ficam a cargo do time de desenvolvimento. Este documento foca em regras de negócio, fluxos e responsabilidades. Informações de exemplo (mock) são fornecidas para orientação.

## 1. Visão geral do sistema

O Omni.AI é uma plataforma SaaS composta por três interfaces principais:

1. **Painel Admin (operador da plataforma)** – Gerencia contas de empresas, planos de assinatura, políticas globais, métricas agregadas e conformidade.
2. **Painel do Usuário (empresa cliente)** – Permite que cada empresa registre-se, configure seu agente, gerencie produtos, acompanhe pedidos, analise dados e personalize o atendimento.
3. **ChatWeb (cliente final)** – Ponto de contato com o consumidor. Inclui chat web público e integrações com canais (WhatsApp, Instagram, e-mail) onde o agente atende, recomenda produtos, efetua vendas e oferece suporte pós-venda.

## 2. Papéis e permissões

### 2.1. Administrador da plataforma

- **Gerenciar empresas:** criar, editar, suspender ou excluir contas de empresas.
- **Configurar planos e limites:** definir número máximo de produtos, conversas, canais disponíveis e features por plano (e.g., teste gratuito, Básico, Premium).
- **Ver métricas globais:** acessar dashboards agregados (número total de empresas, volume de chats, conversões médias, receitas por plano).
- **Moderar conteúdos:** revisar e remover instruções ou catálogos que violem termos de uso.
- **Gerenciar billing:** ajustar preços, emitir faturas, controlar comissões sobre vendas, integrar com gateway de pagamentos para cobranças.
- **Suporte:** receber chamados de empresas e realizar handoff com engenheiros de suporte.

### 2.2. Usuário da empresa (dono/gestor)

- **Cadastro e onboarding:** fornecer dados da empresa (nome, CNPJ/ID fiscal opcional, e-mail, telefone), nome do agente, tom de voz e upload do catálogo.

- **Gerenciar produtos:** adicionar, editar e remover itens do catálogo. Cada produto deve conter ID, nome, descrição, preço, categoria, imagem e link de compra. Regra: o preço deve ser positivo e o ID único dentro da empresa.
- **Configurar agente:** definir instruções personalizadas (ex. “seja simpático e use emojis”), escolher canais (WhatsApp, Instagram, Webchat) e gerar link/QR Code da loja conversacional.
- **Visualizar chats:** acessar histórico de conversas para entender dúvidas e feedbacks. Baixar transcripts se necessário (respeitando legislação de privacidade).
- **Gerenciar pedidos:** ver lista de pedidos gerados pelo agente, atualizar status (Processando, Enviado, Entregue, Cancelado) e enviar mensagens ao cliente.
- **Acesso a métricas:** ver gráficos de conversão (chats x pedidos), valor médio de pedido (AOV), produtos mais vendidos, horários de pico e sentimento do cliente.
- **Gestão de equipe:** convidar colaboradores (sub-usuários) com permissões específicas (ex. apenas editar catálogo ou apenas ver pedidos). Cada sub-usuário usa login próprio.
- **Upgrade/Downgrade de plano:** escolher plano de assinatura, alterar ou cancelar. O sistema deve aplicar limites conforme o plano ativo.

### 2.3. Cliente final (ChatWeb)

- **Iniciar conversa:** o cliente acessa a loja via link ou QR, sem necessidade de cadastro. Pode se identificar por nome opcional.
- **Consultar produtos:** fazer perguntas (“quero sapatos tamanho 38”), receber respostas e recomendações com foto, preço e botão de compra.
- **Efetuar pedido:** selecionar itens e confirmar compra. O agente solicita informações necessárias (nome, telefone, endereço) e gera resumo do pedido. Após confirmação, o pedido é registrado e um link de pagamento (mock ou real) é enviado.
- **Acompanhar status:** perguntar “onde está meu pedido?” e receber atualização conforme status definido pela empresa.
- **Reclamar ou elogiar:** enviar feedback ou reclamação. O agente identifica sentimento e, caso necessário, encaminha para atendimento humano.

## 3. Regras de negócio por módulo

### 3.1. Onboarding

1. Ao criar conta, a empresa informa pelo menos **nome, e-mail e senha**. O e-mail deve ser único na plataforma. Um e-mail de verificação pode ser enviado (opcional no MVP).
2. O fluxo de onboarding terá etapas obrigatórias: **(a) Empresa, (b) Agente, (c) Catálogo, (d) Canais**.
3. **Empresa:** a empresa define nome e segmento. Campo de documento fiscal é opcional. Caso pertença a um país da UE, deve aceitar termos GDPR.
4. **Agente:** a empresa escolhe um **nome** para o agente (string sem caracteres especiais), seleciona um **tom de voz** pré-definido (Empático, Divertido, Profissional) e pode escrever **instruções personalizadas** para influenciar o comportamento.
5. **Catálogo:** é possível importar um arquivo CSV/Excel com colunas **ID, Nome, Descrição, Preço, Imagem (URL ou upload), Link externo**. O sistema valida formatos e tamanho dos arquivos, e rejeita se houver linhas sem nome ou com preço negativo. Produtos duplicados (mesmo ID) são sobrescritos.

6. **Canais:** o usuário marca quais canais pretende utilizar (ChatWeb obrigatoriamente; WhatsApp/Instagram opcionais). Se escolher WhatsApp, o sistema gera instruções para conectar via API oficial ou número verificado (integração real para fases posteriores).
7. Ao finalizar o onboarding, a empresa recebe um **link único** para seu chat (e.g., `https://app.omni.ai/{slug}`). Esse slug é gerado a partir do nome da empresa e precisa ser único na plataforma.

### 3.2. Catálogo e produtos

1. Cada produto pertence a exatamente uma empresa.
2. Campos obrigatórios: `nome`, `preco` (decimal positivo), `descricao` (texto livre), `imagem` (URL ou arquivo), `id_externo` (string ou numérico). Campo opcional: `categoria`.
3. Quando o usuário altera preço ou estoque, a mudança deve refletir imediatamente nas recomendações do agente.
4. O upload de imagem deve aceitar apenas formatos comuns (JPEG, PNG), com tamanho máximo de 2 MB (mock). Imagens são armazenadas em repositório cloud ou pasta local no ambiente de teste.
5. A plataforma pode limitar o número de produtos com base no plano de assinatura (ex. 20 produtos no plano gratuito). Ao atingir o limite, a empresa recebe mensagem indicando a necessidade de upgrade.

### 3.3. Configuração do agente

1. O agente utiliza um modelo de linguagem (LLM) para gerar respostas; as instruções fornecidas pela empresa são concatenadas com um template padrão. O LLM deve respeitar essas instruções.
2. O tom de voz selecionado adiciona atributos de estilo (ex. “use emojis”, “respostas formais”, etc.). Isso é aplicado em cada resposta.
3. O agente tem acesso somente aos dados da empresa (catálogo, perguntas frequentes, políticas) e não deve vaziar dados de outras empresas.
4. Para perguntas que não podem ser respondidas com base nos dados, o agente deve seguir regra de fallback: pedir mais contexto ou encaminhar para humano, registrando ticket com resumo da conversa.

### 3.4. Conversa e atendimento

1. Cada sessão de chat recebe um **ID de conversa** vinculado a `empresa_id` e `cliente_id` (cliente anônimo ou pseudônimo).
2. A conversa mantém contexto local até expirar (ex. 24 horas sem interação). Após expirar, o contexto é descartado, preservando histórico para analytics.
3. **Identificação de intenção:** a IA classifica a mensagem em tipos (consulta de produto, dúvida geral, compra, rastreo, reclamação). Cada tipo aciona um fluxo:
4. **Consulta de produto:** recuperar lista de produtos relevantes e exibir top 3.
5. **Compra:** solicitar itens, quantidades, endereço e confirmar. Criar registro de pedido.
6. **Rastreo:** ler status do pedido e retornar resposta amigável.
7. **Dúvida geral/reclamação:** usar base de conhecimento; se não resolvido, abrir ticket para humano.
8. **Recomendações:** a IA sugere produtos com base em palavras-chave, histórico de navegação (se disponível) e preferências. Deve sempre listar preço e oferecer botão “Adicionar ao carrinho” ou “Comprar agora”.

9. **Consentimento:** antes de coletar dados sensíveis (nome, telefone, endereço), o agente deve informar ao cliente para que fim serão usados e solicitar confirmação.
10. **Idiomas:** o agente tenta detectar idioma e responder na língua do cliente. Caso não consiga, responde no idioma padrão configurado pela empresa. O usuário pode digitar “/lang PT” ou “/lang EN” para mudar manualmente.
11. **Transferência para humano:** se o cliente escrever “quero falar com atendente” ou a IA detectar frustração (via sentimento negativo), a conversa é encaminhada. A plataforma envia notificação para o painel do usuário, incluindo resumo e link para intervir.

### 3.5. Pedidos e pagamentos

1. Quando o cliente confirma a compra, o sistema cria um **pedido** com: `pedido_id` (incremental ou UUID), `empresa_id`, `itens` (array de produto/quantidade/preço), `cliente_info` (nome, telefone, endereço), `total`, `status` (padrão = “Pendente Pagamento”).
2. Se houver integração de pagamento (fase futura), o sistema gera link ou QR code conforme método disponível (ex. Pix, cartão, PayPal). Se não, a compra fica registrada como “pedido manual” para a empresa processar offline.
3. O pedido muda de status conforme ação da empresa: **Pendente, Pago, Processando, Enviado, Entregue, Cancelado**. Cada mudança gera notificação para o cliente via chat.
4. O histórico de pedidos pode ser exportado em CSV pela empresa. Admin não acessa detalhes de pedidos, apenas métricas agregadas (respeitando privacidade).

### 3.6. Analytics e relatórios

1. Métricas principais por empresa:
2. **Volume de conversas** (total de chats iniciados por período).
3. **Pedidos gerados** (contagem e valor).
4. **Taxa de conversão** = pedidos / chats \* 100.
5. **Ticket médio (AOV)** = valor total / número de pedidos.
6. **Produtos mais consultados e mais vendidos**.
7. **Tempo médio de resposta do agente**.
8. **Sentimento do cliente** (positivo, neutro, negativo) por período.
9. O admin visualiza métricas globais e pode filtrar por plano ou segmento. Dados pessoais ou transacionais detalhados não são exibidos ao admin.
10. Os gráficos e relatórios devem permitir filtragem por data (dia, semana, mês) e exportação básica (CSV ou PDF). Para o MVP, mockar com dados fictícios de forma a demonstrar a feature.

### 3.7. Planos e cobrança

1. O sistema oferece pelo menos dois planos: **Gratuito** (limite de produtos, chats e sem integração de WhatsApp) e **Premium** (limites maiores, integração de canais adicionais, analytics avançados). Um plano intermediário pode ser criado conforme estratégia comercial.
2. A assinatura é mensal ou anual, com renovação automática. O usuário fornece dados de pagamento no ato da assinatura.
3. Ao atingir um limite do plano (ex. número máximo de chats), o sistema bloqueia novos chats até o próximo ciclo ou sugere upgrade.
4. O admin define preços e regras de cada plano no painel, sem necessidade de intervenção técnica.

### 3.8. Segurança e conformidade

1. Dados pessoais e credenciais devem ser armazenados de forma criptografada (mock no MVP). A comunicação entre sistemas deve ser via HTTPS.
2. O sistema deve ser compatível com **GDPR** e **LGPD**. Isso implica fornecer aos clientes finais (ChatWeb) informações sobre coleta e uso de dados, bem como opções para solicitar a exclusão de seus dados.
3. Logs de acessos e atividades devem ser armazenados por um período determinado (ex. 6 meses) para auditoria.
4. A plataforma não deve compartilhar dados de empresas entre si. Cada empresa possui um **tenant** isolado.
5. Backup periódico de banco de dados e medidas de recuperação de desastre devem ser definidos (não detalhados no MVP).

### 3.9. Internacionalização

1. A interface do painel pode ser traduzida para vários idiomas (inicialmente Português e Inglês). Todas as strings de UI devem ser externalizadas para arquivos de tradução.
2. O chat deve ser capaz de responder em diferentes idiomas, seja via modelo multilíngue ou tradução automática.
3. Formatos de data, hora, moeda e endereço devem respeitar a localidade do usuário.

### 3.10. Testes e qualidade

1. Fornecer um **ambiente de testes** (sandbox) para que empresas experimentem o agente com dados de teste antes de ativar em produção.
2. Criar **conjuntos de dados mock** contendo:
3. Produtos fictícios (por exemplo: Camiseta Clássica, Caneca Personalizada, Poster Decorativo) com preços e descrições.
4. Pedidos simulados (IDs, status, datas) para preencher dashboards.
5. Conversas simuladas (cliente pergunta, agente responde) para demonstração.
6. Definir **cenários de teste** para cada regra de negócio, incluindo casos de erro (ex. preço negativo, conexão de canal falha, upgrade de plano sem pagamento). O agente de desenvolvimento deve implementar testes automatizados para estes cenários.
7. Garantir **performance**: o tempo de resposta do chat deve ser inferior a 2 segundos na média. Em caso de atraso, exibir mensagem de processamento para o cliente.

## 4. Roteiro de desenvolvimento (macro passos)

1. **Preparação e modelagem**: Definir os modelos de dados (empresa, usuário, produto, pedido, plano, conversa), criar wireframes iniciais para os painéis (admin e usuário) e chat. Mapear APIs externas para canais (WhatsApp, pagamentos).
2. **Implementar base do sistema**: Construir autenticação, autorização e multi-tenant (isolamento por empresa). Criar estrutura do banco de dados e classes de serviço.
3. **Desenvolver Painel Admin**: Incluir gerenciamento de empresas, planos, métricas globais e moderação. Utilizar dados mockados para gráficos e tabelas.
4. **Desenvolver Painel do Usuário**: Implementar onboarding (wizard), cadastro de produtos, configuração do agente e visualização de pedidos e métricas. Inicialmente, usar mocks de pedidos e conversas.

5. **Construir ChatWeb:** Criar interface de chat para cliente final, conectar ao serviço de IA e ao motor de recomendação. Implementar flows de compra e rastreamento com dados de teste.
6. **Integração de canais:** Adicionar suporte a WhatsApp/Instagram (mesmo que simulado) e testes de conexão. Permitir geração de link e QR code.
7. **Adicionar gerenciamento de pedidos:** Criar rotas e interface para acompanhamento e atualização de pedidos. Enviar notificações no chat conforme mudança de status.
8. **Incluir analytics e relatórios:** Gerar dashboards com dados reais e/ou mock. Permitir download em CSV. Criar indicadores para admin e usuário.
9. **Implementar planos e billing:** Definir limites por plano, criar UI para upgrade/downgrade, registrar pagamentos (mock). Encerrar funcionalidades quando limites forem atingidos.
10. **Revisar segurança e compliance:** Adicionar camadas de proteção, logs, e fluxos de consentimento. Implementar opção de exclusão de dados pessoais (direito de esquecimento).
11. **Testes e validação:** Executar testes unitários e de integração para cada módulo. Conduzir testes com usuários-alvo (empresários sem conhecimento técnico) para validar usabilidade.
12. **Lançamento e iteração:** Disponibilizar versão beta para alguns clientes. Coletar feedback, priorizar melhorias e planejar integração com canais reais e pagamentos no pós-MVP.

## 5. Exemplo de dados mock (para referência)

### 5.1. Catálogo fictício

ID	Nome	Descrição	Preço	Categoria
101	Camiseta Clássica	Camiseta de algodão, várias cores (P, M, G).	59,90	Roupas
102	Caneca Personalizada	Caneca de cerâmica com estampa exclusiva.	39,90	Utensílios
103	Poster Decorativo	Poster A3 com arte minimalista.	29,90	Decoração
104	E-book de Receitas	Livro digital com 50 receitas veganas.	19,90	Digital
105	Hoodie Premium	Blusa de moletom com capuz, tamanho único.	89,90	Roupas

### 5.2. Exemplo de conversa

**Cliente:** Olá! Quero uma camiseta tamanho M, vocês têm?

**Agente:** Olá! Temos sim! Recomendo nossa **Camiseta Clássica** (R\$ 59,90) disponível em três cores. Deseja ver outras opções ou adicionar essa ao carrinho?

**Cliente:** Vou querer a clássica mesmo. Qual o valor do frete?

**Agente:** O frete para sua região é R\$ 12,00 e a entrega ocorre em até 5 dias úteis. Confirma a compra?

**Cliente:** Sim. Pode mandar!

**Agente:** Perfeito! Para finalizar, preciso do seu nome e endereço completo.

### 5.3. Exemplo de pedido

```
{
  "pedido_id": "12345",
  "empresa_id": "loja123",
  "itens": [
    {
      "produto_id": 101,
      "nome": "Camiseta Clássica",
      "quantidade": 1,
      "preco_unitario": 59.90
    }
  ],
  "total": 71.90,
  "cliente": {
    "nome": "Maria Silva",
    "telefone": "+55 11 98765-4321",
    "endereco": "Rua das Flores, 123, São Paulo-SP"
  },
  "status": "Processando",
  "criado_em": "2025-10-25T14:23:00Z"
}
```

## 6. Considerações finais

Este documento fornece um guia de regras de negócio e fluxos essenciais para desenvolvimento do Omni.AI. O foco é garantir que a plataforma atenda tanto **empresas sem experiência técnica**, que desejam vender e atender clientes por chat, quanto **consumidores finais** que buscam comprar de forma rápida e fluida. As equipes de engenharia podem usar estas regras como base para definir arquitetura, tecnologias e priorização de funcionalidades. Ajustes podem ser feitos conforme feedback dos usuários e expansão das necessidades.

---