

# Instituto AMATE — Fluxo Completo & Técnico (Plataforma Completa)

**Versão:** Plataforma completa (Clube de Saúde Pet + Gestão Clínica + Rede de Apoio)

Formato: mesmo padrão do *Game List* — telas, autorização, API, modelo de dados, regras de negócio, planos e roadmap.

---

## 1 — Visão geral rápida

Plataforma para gestão de associados (tutores), rede de parceiros (clínicas, veterinários, lojas) e administração do Instituto AMATE. Funciona como **plano de saúde social + clube de benefícios + hub de ações sociais** (vacinação, microchip, voluntariado, suporte em luto). Arquitetura pensada para Next.js (frontend), Supabase/Postgres (backend), storage (S3/Supabase Storage), e PSP (Pagamentos).

---

## 2 — Perfis / Roles (RBAC)

- `tutor` — Tutor Associado (pessoa física).
- `clinica` — Clínica (pj) — painel clínico.
- `veterinario` — Veterinário (pf) — vinculado a `clinica`.
- `loja` — Pet shop / Farmácia parceira.
- `admin` — Administrador operacional.
- `admin_master` — Superadmin (infra / integração / DPO).

Cada role tem permissões finas claras (read/write/approve/settle). Usar RBAC com claims JWT.

---

## 3 — Planos & Regras Comerciais (resumo)

Planos:

- **Free** — R\$0 — acesso à informação, mapa, campanhas.
- **Associado AMATE** — R\$29,90/mês — benefícios principais.
- **Premium Tutor** — R\$49,90/mês — prioridade em agendamentos, menor coparticipação.
- **Parceiro Loja** — R\$49,90/mês — destaque marketplace.
- **Profissional / Clínica** — R\$69,90–R\$89,90/mês — relatórios avançados e destaque.

Regras chave:

- Carência 15 dias para serviços eletivos. Urgências sem carência.
  - Limites iniciais (configuráveis): 2 entregas grátis/mês; 2 consultas por semestre/pet; 1 procedimento odontológico/ano por pet.
  - No-show: 1 advertência → 2º no-show = bloqueio 30 dias (serviço específico).
  - Coparticipação em cirurgias/procedimentos conforme tabela AMATE.
  - Pagamento: recorrência via PSP (Pix/Cartão/Boleto). Webhooks para status.
- 

## 4 — Fluxos de autenticação & onboarding (telas)

### 4.1 Telas de Auth

- Landing — CTA: Entrar / Conhecer Planos / Doe
- EscolhaPerfil — selecione: Tutor / Clínica / Veterinário / Loja
- Signup (por perfil) — campos específicos:
  - Tutor: nome, CPF, telefone, e-mail, senha, endereço (CEP), aceita termos.
  - Clínica/Loja: razão social, CNPJ, endereço, contato, upload KYC (alvará, contrato).
  - Veterinário: nome, CPF, CRMV, clínica vinculada.
- Login:
  - Email + senha
  - SMS/WhatsApp OTP
  - Google Sign-In / Facebook / Apple ID
- KYCFlow (para parceiros): upload docs → verificação manual (admin) → selo `verified`.
- Onboarding Tutor:
  - Cadastro de pets (nome, espécie, raça, data nascimento, microchip ID opcional, alergias, observações)
  - Escolha de plano → pagamento → carteira digital liberada (QR / QR Dinâmico)

### 4.2 Fluxo rápido de agendamento (tutor)

1. Dashboard Tutor → “Agendar Consulta”
2. Busca por clínicas (filtro: distância, especialidade, disponibilidade)
3. Seleciona serviço (consulta/exame/cirurgia) → escolhe data/hora → confirmar

4. Sistema aplica elegibilidade (rules engine verifica carência/limite)
  5. Confirmação → lembretes 48h/12h via push/WhatsApp/SMS
  6. Check-in na clínica via QR/ID → atendimento → prontuário + receipts.
- 

## 5 — Telas / Painéis por perfil (detalhado)

### Tutor (App Web + Mobile)

- Dashboard (assinatura, próximos agend., últimos atendimentos)
- Meus Pets (editar, microchip, prontuário)
- Carteira Digital (QR CODE, ID de associado, validade, limites, plano)
- Agendar (consulta/exame/cirurgia)
- Teletriagem (chat inicial + triagem automática + chat com equipe )
- Minhas Vacinas (carteira de vacinação digital com lembretes)
- Pedidos (ração/Medicamento ) — tracking + elegibilidade
- Histórico & Prontuário resumido
- Cursos & Campanhas (inscrever-se, completar módulos)
- Voluntariado / Doações (ver eventos, inscrever-se, doar)
- Perfil & Pagamentos (mudar plano, cancelar)
- Suporte / Luto (ticket + encaminhamentos)

### Clínica / Veterinário

- Perfil da Clínica: serviços, catalogo de serviços, dados, tabela AMATE (preços com desconto)
- Agenda (slots, encaixes, prioridade AMATE)
- Pacientes do dia (acesso ao prontuário do pet)
- Registrar atendimento (diagnóstico, procedimentos, prescrição, vacina)
- Emitir e enviar resultados de exame (upload imagem/pdf)
- Comodato (emprestar equipamentos e registrar retorno)
- Relatórios (atendimentos AMATE, faturamento, repasses)
- Configurar tabela AMATE (preços sociais)
- Financeiro (solicitação de repasse / conciliação)

## Loja / Farmácia

- Catálogo e estoque
- Criar promoções / cupons exclusivos AMATE
- Receber pedidos e integrar com logística (fulfillment)
- Relatórios de vendas por cupom / associado

## Admin

- Gestão Usuários / Perfis / KYC
- Regras Engine (limites, carências, coparticipações configuráveis)
- Gestão de Planos / Preços
- Painel KPIs (assinantes, churn, volume atendimentos, entregas)
- Moderação (content, denúncias)
- Relatórios financeiros / export CSV
- Configuração de integrações (PSP, WhatsApp, Logistic)

## / Admin\_master

- Gestão Usuários / Perfis / KYC
- Regras Engine (limites, carências, coparticipações configuráveis)
- Gestão de Planos / Preços / Repasses
- Painel KPIs (assinantes, churn, volume atendimentos, entregas)
- Moderação (content, denúncias)
- Relatórios financeiros / export CSV
- Configuração de integrações (PSP, WhatsApp, Logistic)
- Auditoria e Logs (compliance LGPD)
- Finance Dashboard (assinaturas, repasses, conciliações)
- BI: KPIs, churn, NPS, volume por serviço
- ADMIN MASTER ESTA ACIMA DE TODOS OS PERFIL.

---

## 6 — Modelo de Dados (ERD simplificado — tabelas e campos principais)

users

id, nome, email, phone, cpf, role, status, created\_at, updated\_at

#### `pets`

`id, user_id, nome, especie, raca, data_nasc, peso, microchip_id, observacoes`

#### `memberships`

`id, user_id, plan, price, status, start_date, end_date, carencia_ends, canceled_at`

#### `appointments`

`id, pet_id, user_id, clinic_id, vet_id, service_code, date_time, status, checkin_code, notes`

#### `services`

`code, name, category, default_price, ama_te_price_code`

#### `partners`

`id, type (clinic|shop), cnpj, name, address, geo_lat, geo_lng, verified, price_table_id`

#### `price_tables`

`id, partner_id, service_code, price_full, price_amate, effective_from`

#### `orders`

`id, user_id, items(json), total, shipping_fee, status, tracking_code, partner_id`

#### `usage_ledger`

`id, user_id, pet_id, benefit_id, partner_id, amount, date, status`

#### `vaccinations`

`id, pet_id, vaccine_name, dose_number, date_administered, clinic_id, lot_number, next_due`

#### `microchips`

`id, microchip_code, pet_id, implanted_by, date_implanted`

#### `tickets`

`id, user_id, type, subject, message, status, assigned_to`

#### `audit_logs`

`id, actor_id, action, target_type, target_id, payload(json), timestamp`

#### `rules`

`id, name, json_rule, active, created_by, created_at`

Observação: use UUIDs para `id` e índices em `user_id`, `pet_id`, `clinic_id`, `date_time`.

## 7 — API (endpoints principais — esboço REST)

### Auth

- POST /auth/signup
- POST /auth/login → { token, user }
- POST /auth/otp(phone)
- POST /auth/refresh

### Tutores

- GET /me
- POST /pets
- GET /pets
- POST /memberships/subscribe – inicia cobrança
- GET /benefits — mostra elegibilidade
- GET /vaccinations/:pet\_id
- POST /appointments → valida via rules
- GET /appointments?user\_id=
- POST /orders (ração/med)
- GET /ledger (uso de benefícios)

### Parceiros / Clínicas

- POST /partners (KYC)
- GET /partners?lat=&lng=&type=
- POST /partners/:id/prices
- POST /partners/:id/checkin (checar QR do associado) → resposta: {eligible:true|false, reason, limit}

### Admin

- GET /admin/reports?from=&to=
- POST /admin/rules
- POST /billing/settlement

### Webhooks

- POST /webhooks/payments (PSP)
- POST /webhooks/logistics (status entrega)

### Exemplo: validar elegibilidade (checkin)

#### Request

POST /partners/123/checkin

Body: { member\_id: "uuid", pet\_id: "uuid", benefit:"consulta" }

#### Response

```
{ "eligible": true, "used": 1, "limit": 2, "coparticipation": 20.00, "carencia_ok": true }
```

## 8 — Motor de Regras (Rules Engine)

Armazenar regras como JSON para avaliação dinâmica:

Exemplo de regra (entrega grátis):

```
{
  "id": "entrega_gratis_v1",
  "benefit": "entrega_gratis",
  "conditions": {
    "plans": ["associado", "premium"],
    "min_order_value": 79.0,
    "geozone": ["01000-000", "99999-999"],
    "max_per_period": 2,
    "period": "month"
  },
  "coparticipation": 0.0
}
```

Processo de avaliação:

1. Recebe request (member\_id, benefit, partner, value).
2. Busca regras ativas para benefit.
3. Avalia condições: plano, período (contagem), geozone, carência.
4. Retorna eligible + motivo ou not\_eligible + reason.

## 9 — Regras de negócio críticas (resumo)

- **Carência** por serviço (ex.: cirurgias eletivas → 15 dias).
- **Limites** por período (mês/semestre/ano).
- **Coparticipação** por faixa de procedimento.
- **KYC obrigatório** para repasse a parceiros.
- **Microchip**: um código único → vincular pet ao banco AMATE.

- **Comodato:** empréstimo de material com contrato digital (registro de devolução).
- **No-show & Cancel:** política e penalidade configurável.

## 10 — UX / Telas (wireframe textual — fluxo principal)

### Tela: Dashboard Tutor

- Top bar: saldo/assinatura, notificações, cadastrar pets
- Cards: Próxima Consulta / Vacinas Vencidas / Benefícios Usados / Pedidos em andamento
- CTA: Agendar / Pedir Ração / Ver Carteira

### Tela: Agendamento

- Filtros (distância, especialidade, data)
- Lista de clínicas (com selo verified e tempo médio de espera)
- Selecionar serviço → selecionar slot → confirmar → pagamento se necessário

### Tela: Carteira de Vacinação

- Lista vacinas por pet → botão “Registrar vacina” (apenas clínica)
- Cronograma automático → lembrete configurável (30/7 dias)

### Tela: Clínica — Agenda

- Visual semanal / dia
- Lista pacientes com botão Abrir Prontuário
- Registrar Atendimento → inserir diagnósticos, procedimentos, receitas, exames → enviar para tutor

## 10/a. UX/UI — fluxo de telas (resumido)

1. Onboarding tutor → cadastrar pets → escolher plano → pagamento → carteira digital liberada.
2. Tutor abre app → agendar consulta → escolher clínica → escolher horário → confirmar → receber lembretes.
3. Clínica recebe notificação → aceita → faz check-in via QR → registra atendimento → gera `usage_ledger` para cobrança e repasse.
4. Loja recebe pedido de ração → confirma entrega ↔ tutor recebe tracking.
5. Admin observa KPIs / aprova parceiros / gerencia regras.



## 11 — Integrações recomendadas

- **PSP:** Stripe / PagSeguro / Asaas (recorrência + webhooks)
- **WhatsApp Business API:** OTP, lembretes, confirmações, conteúdo de campanhas
- **Logistics:** Tiny / Hub2b / integração local para entrega de ração
- **Microchip Vendor API** (se existir) ou CSV import/export
- **Supabase Storage / S3:** exames, radiografias, carteirinhas PNG/PDF

## 12 — Segurança & LGPD

- Criptografia at-rest para dados sensíveis (CPF, documentos)
- TLS em trânsito + CSP + proteção contra CSRF
- DPO contato, Política de Privacidade, DPA com parceiros
- Consentimento explícito em signup para tratamento de dados e marketing (logs)
- Anonimização para analytics quando necessário

## 13 — KPIs iniciais para acompanhar

- Ativação D7 (%)
- Churn mensal
- N° consultas por mês (AMATE)
- Tempo médio atendimento
- Volume de pedidos de ração / entregas
- N° de parceiros verificados
- NPS do tutor

## 14 — Contas / Usuários de teste (seed)

- `admin_master@amate.org` → role: `admin_master`
- `oper@amate.org` → role: `admin`
- `maria.tutor@ex.com` → role: `tutor` — pets: Thor (cao), Luna (gata)
- `clinicacentro@ex.com` → role: `clinica` — CNPJ fictício
- `dr.paula@ex.com` → role: `veterinario` — vinculado clinicacentro

- petshopbairro@ex.com → role: loja

as senhas de usuario de teste devem ser a mesma usuarioteste