**GFauto/Projeto\_GFauto.docx e GFauto/Projeto\_GFauto.md**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Clique para seguir o link:** |  |  |
| [Arquitetura/Tecnologias](#arquitetura) | [Fale Conosco](#fale) |  | [Nota Importante](#nota) |  |
| **NOVO** | [Fluxo visitantes](#visitantes)/fluxo\_app  [FluxoVisitante Integração](#FluxoVisitante) |  | [Objetivo GFauto](#objetivo) |  |
|  | [Fluxo cliente/anunciante](#anunciante) |  |  |  |
| [Convenções](#convencao) | [Fontes de Receita](#receita) |  | [Pág Vendas Escolha Plano](#vendas) |  |

**## Objetivo do Projeto GFauto:**“RESUMO DO PROJETO GFauto”

O Projeto GFauto atualmente na página <https://www.gfauto.com.br> está sendo reestruturado e implementado no novo projeto iniciando e <https://gfauto.vercel.app/> que tem como principal objetivo expor para visitantes da web anunciantes de produtos e serviços automotivos no Brasil.  
**O Site do Projeto GFauto busca**:  
- Conectar internautas e usuários de redes sociais, visitantes no site do Projeto GFauto proprietários de veículos automotores (como carros, motos, caminhões, etc.) **a** fornecedores das mais variadas especialidades do ramo automotivo englobando fabricação, vendas e distribuição, manutenção e serviços especializados, tecnologia e inovação, mobilidade e transporte, na área em seus Estados e suas cidades com o intuito de facilitar aos internautas/visitantes a pesquisa de serviços automotivos, e de o Cliente/Anunciante ser encontrado.

Essa conexão se dará quando o internauta estiver procurando na sua cidade um fornecedor para um problema a ser resolvido no veículo dele. Ele vai informar o “Estado” e a “Cidade” onde ele está e “O que procura?” e então será levado para uma página de resultados da especialidade na cidade dele onde estarão figurando os fornecedores para o caso “O que procura?” que ele informou. Por outro lado teremos os Clientes/Anunciantes que serão divididos em duas categorias: "Cortesia" e "Premium" o "Cortesia" poderá figurar na "página de resultados" sem pagar com uma exposição simples, o "Premium" vai ser convidado a efetuar um pagamento escolhido e terá uma exposição privilegiada na "página de resultados".

**## Quem está trabalhando no Projeto GFauto:**

1. O Desenvolvedor: Agente(AI) que deve assumir o papel, obter conhecimento e se comportar como um Profissional Sênior com as habilidades de Engenheiro de Software, Analista de Sistemas, Desenvolvedor de Softwares, Programador de Sistemas. Focado no ambiente de desenvolvimento do Projeto GFauto.

2. O Weber: É o dono/administrador do projeto que vai estar interagindo com o Desenvolvedor, recebendo do Desenvolvedor a orientação para enviar os códigos para o GitHub via Git Hash, verificando os Deploys na Vercel e fazendo as checagens via navegador web do resultado obtidos na Vercel.com

**## Definição do Ambiente de Desenvolvimento e fluxo: O "Ambiente de Trabalho" consta de:**

1. Sandbox do Desenvolvedor;

2. "Meu Local" a máquina windows do Weber; W:\GFauto

3. Repositório GigHub público: https://github.com/RLuizWeber/GFauto.git

4. Hospedagem Vercel.com <https://gfauto.vercel.app/>

**Importante Obs.:** Todas as páginas do domínio [www.gfauto.com.br](http://www.gfauto.com.br) mostradas nesse documento são apenas ilustrativas.

**## Estrutura do site: (fazer os ajustes necessários)**

- GFauto

- .git

- .mercadoPago

- .next

- .recend

- .vercel

- pag\_painelAdmin  
 -

Fluxos / Módulos (Além do README\_geral na raíz, Cada fluxo\_/Módulo contém o seu README.md específico onde estarão informadas as dependências, precedências e outras informações pertinentes ao módulo)

**- GFauto**

- .git

- .next

- .vercel  
 - app

- admin

- anuncio

- api

- pagina-correta

- resultados

README\_app.md

- backups

-

-

README\_backups.md

- componentes

- visitantes

- docs

-

-

- fluxo\_app

- componentes

- images

- styles

README\_app.md

- fluxo\_cliente\_anunciante

-

-

- fluxo\_pag\_de\_resultados

-

-

- fluxo\_pagto

-

-

- fluxo\_painel\_admin

-

-

- fluxo\_plano

- components

- styles

- types

- utils

- fluxo\_visitante

- app

- anuncio

- api

- resultados

- componentes

- visitante

- docs

- lib

- prisma

- utils

- lib

README\_lib.md

- memoria

- node\_modules

- prisma

- migrations

README\_prisma.md

- public

- fluxo\_plano

- images

-

- scripts

- db

- backups

popular\_banco\_via\_api.html

popular\_fornecedores\_autoeletricas.js  
 popular\_fornecedores\_autoeletricas.sql

README.md

README\_API.md

- tmp

-

-

- utils

**Organização dos arquivos README.md**

Objetivo: é que o Desenvolvedor perca o menor tempo possível buscando o que precisa lendo eles.

Índice: Todos devem conter um índice no início informativo do conteúdo facilitando a localização da busca. As informações contidas no README obrigatoriamente devem estar no índice.

- README\_geral.md (GFauto/): que passa informações gerais e aponta para READMEs específicos (salvo em seu próprio fluxo/módulo) por página.

- README\_Dependencias\_entre\_arquivos.md: referenciado no README\_geral.md pode ser salvo em (W:\A\_Weber\Pai\Hostmachine\gfauto\githubVercel\GFauto\docs) os endereços informados devem ser precisos e completos partindo da página principal, no caso GFauto.

**Plano de Modernização do Site GFauto.com.br**

Análise da Situação Atual

O site GFauto.com.br apresenta uma estrutura e design a ser melhorado. O site é voltado para o setor automotivo. Propõe uma abordagem "ganha-ganha" para proprietários de veículos e fornecedores automotivos.

**Pontos Críticos Identificados:**

* Falta de estratégias de monetização eficientes
* Experiência do usuário comprometida

Plano de Modernização

1. Redesenho Completo da Plataforma

1.1 Tecnologia Base

* Framework Recomendado: Next.js
* Justificativa: Oferece renderização do lado do servidor (SSR) e geração estática (SSG), melhorando o SEO e a performance
* Suporte nativo a API Routes para funcionalidades dinâmicas
* Integração perfeita com React e Tailwind CSS

1.2 Design e Interface

* Implementação de design responsivo mobile-first
* Paleta de cores moderna mantendo a identidade visual automotiva
* Tipografia clara e legível
* Áreas de herói impactantes com imagens de alta qualidade de veículos
* Microinterações para melhorar o engajamento do usuário
* Navegação intuitiva e simplificada

1.3 Estrutura do Site

* Página inicial com destaque para os principais serviços
* Páginas de serviços e parcerias
* Área de contato e sobre

2. Estratégias de Monetização

2.1 Publicidade Contextual

* Implementação de anúncios para anunciantes
* Espaços publicitários cortesia e premium para anunciantes
* Anúncios nativos integrados ao conteúdo

2.2 Marketing de Afiliados

Parcerias com toda a área automotiva, ex.:

* Lojas de autopeças
* Seguradoras
* Prestadores de Serviços, oficinas, etc.
* Concessionárias
* Aplicativos de serviços automotivos
* Lojas e Fabricantes de acessórios

2.3 Publicidade Cortesia ou Premium

Disponibiliza:

* Espaço para anúncio do estabelecimento

2.4 Marketplace

Plataforma:

* Onde [visitantes](#visitantes) encontram [clientes/anunciantes](#anunciante)

3. Otimização para SEO

* Implementação de URLs amigáveis
* Estrutura de dados schema.org para conteúdo automotivo
* Otimização de meta tags
* Conteúdo otimizado para palavras-chave relevantes do setor
* Melhoria na velocidade de carregamento
* Implementação de sitemap.xml e robots.txt

4. Integração com Redes Sociais e Plataformas

* Botões de compartilhamento em conteúdos relevantes
* Feed de redes sociais na página inicial (a estudar)
* Possibilidade de login social (a estudar)
* Integração com WhatsApp para suporte
* Integração com Google Maps para localização de serviços

5. Recursos Avançados

5.1 Aplicação de IA

* Chatbot para atendimento e dúvidas sobre multas e legislação
* Recomendações personalizadas de serviços baseadas no perfil do usuário

5.2 Recursos Interativos (a estudar)

* Calculadoras interativas (financiamento, consumo de combustível, etc.)
* Simuladores de customização de veículos
* Avaliação de preço de veículos usados

5.3 Aplicativo Mobile

* Versão simplificada do site como PWA (Progressive Web App)
* Notificações push para alertas de radares próximos (a estudar)
* Funcionalidades offline para consulta de informações essenciais (a estudar)

**Cronograma de Implementação**

Fase 1: Planejamento e Design

* Wireframes e protótipos
* Definição da arquitetura técnica
* Planejamento de conteúdo
* Definição de estratégias de monetização

Fase 2: Desenvolvimento

* Configuração do ambiente Next.js
* Desenvolvimento do front-end responsivo
* Implementação das funcionalidades principais
* Integração com APIs de terceiros

Fase 3: Conteúdo e SEO

* Migração e criação de conteúdo
* Otimização para mecanismos de busca
* Implementação de estratégias de monetização

Fase 4: Testes e Lançamento

* Testes de usabilidade
* Testes de performance
* Lançamento da versão beta
* Coleta de feedback e ajustes

Fase 5: Marketing e Crescimento (contínuo)

* Campanhas de divulgação
* Parcerias estratégicas
* Análise de métricas e otimização contínua

Estimativa de Investimento

Custos Recorrentes

* Hospedagem e infraestrutura: R$ 300 - R$ 500/mês
* Manutenção e atualizações: R$ 1.000 - R$ 2.000/mês
* Marketing digital: R$ 2.000 - R$ 5.000/mês (variável conforme objetivos)

Retorno Esperado

Fontes de Receita

* Publicidade: 10-20% da receita total
* Assinaturas premium: 25-35% da receita total
* Comissões de afiliados: 15-25% da receita total
* Marketplace e outros: 30-40% da receita total

Projeção de Crescimento

* Aumento de tráfego: 200-300% no primeiro ano
* Conversão para assinantes: 2-5% dos visitantes
* Receita mensal estimada após 12 meses: R$ 10.000 - R$ 30.000

**Requisitos de Hospedagem**

* Hospedagem recomendada: Vercel (otimizada para Next.js) com deploy automático via GitHub
* Alternativas: AWS, Google Cloud, Microsoft Azure, Netlify ou DigitalOcean
* Especificações mínimas: 2 vCPUs, 4GB RAM, 20GB SSD, 1TB de transferência mensal
* Escalabilidade: Capacidade de aumentar recursos conforme o crescimento do tráfego

**Requisitos de Software**

* Node.js: Versão 18.x LTS ou superior
* Certificado SSL: HTTPS obrigatório
* Banco de dados: MySQL/PostgreSQL (relacional)
* CDN: Cloudflare, Vercel Edge Network ou similar para otimização de entrega

**Requisitos de Domínio e DNS**

* Manutenção do domínio atual (gfauto.com.br)
* Acesso ao gerenciamento de DNS para configurações necessárias

**Requisitos de Segurança**

* Firewall e WAF para proteção contra ataques
* Sistema de backup automatizado (depois de implementado)
* Autenticação de dois fatores para acesso administrativo

**Integrações Necessárias**

* Gateway de pagamento (para assinaturas e marketplace) (Mercado Pago)
* Serviços de email marketing (resend.com)
* Google Analytics ou similar
* Redes de anúncios e afiliados

**Performance e Monitoramento**

* Tempo de carregamento abaixo de 3 segundos
* Conformidade com Core Web Vitals do Google
* Monitoramento 24/7 de disponibilidade
* Sistema de logs e alertas

**Conformidade**

* LGPD (Lei Geral de Proteção de Dados)
* Acessibilidade WCAG 2.1 nível AA
* Gerenciamento adequado de consentimento de cookies

**Implantação Contínua (CI/CD)**

* Repositório Git para versionamento
* Pipeline de implantação automatizada
* Ambientes para desenvolvimento, produção (nesse desenvolvimento iremos trabalhar na base quente, sem ambiente de desenvolvimento teste)

A recomendação final é utilizar a plataforma Vercel, que oferece otimização nativa para Next.js, implantação simplificada, CDN global, escalabilidade automática e certificados SSL gratuitos, proporcionando o melhor equilíbrio entre facilidade, performance e custo para seu projeto.

**Vercel - Plataforma para Next.js**

**Site oficial**: [https://vercel.com](https://vercel.com/)

**Principais vantagens da Vercel:**

* Otimizada especificamente para Next.js (integração perfeita)
* Implantação simplificada diretamente do GitHub/GitLab
* CDN global para entrega rápida de conteúdo
* Certificados SSL gratuitos e automáticos
* Previews automáticos para cada atualização
* Escalabilidade automática conforme o tráfego
* Ferramentas de análise de performance integradas

1. **Plano Pro ($36/mês)**
   * Recomendado
   * Domínios personalizados ilimitados
   * Equipes de até 10 membros
   * Melhor performance e recursos
   * Suporte prioritário
   * Análises avançadas

**Arquitetura e Tecnologias**

1. **Arquitetura e Tecnologias:**
   * Framework**:** Next.js (App Router).
   * Banco de Dados: Vercel Postgres com Prisma ORM.
   * Hospedagem: Vercel (com deploy automático via GitHub).
   * APIs Externas Integradas:
   * Resend: Para envio de e-mails transacionais (domínio gfauto.com.br verificado).
   * Mercado Pago: Para processamento de pagamentos (Ok).
2. **Progresso Atual:**
   * Projeto Next.js no GitHub (RLuizWeber/GFauto) com deploy na Vercel (https://gfauto.vercel.app) .
   * Conexão com Vercel Postgres via Prisma estabelecida.
   * Modelo Advertiser definido no schema.prisma e migrações aplicadas.
   * Rotas de API funcionais:
   * /api/send-email (Resend).
   * /api/**create-payment** (Mercado Pago).
   * Variáveis de ambiente para chaves de API e URLs base configuradas e funcionando na Vercel.
3. **Segurança (Considerações Iniciais):**
   * Uso de PATs para GitHub e variáveis de ambiente seguras na Vercel.
   * SSL fornecido pela Vercel.
   * Pontos para atenção futura: Autenticação de usuários na aplicação, autorização detalhada, proteção contra vulnerabilidades web comuns (XSS, CSRF), rate limiting.

**Fontes de Receita**

3. Uma das principais fontes de monetização, além das outras, é os anunciantes pagando para figurar nas páginas das cidades dentro de suas especialidades. Esse valor é bem pequeno mas existindo muitas possibilidades por cidade resultaria numa arrecadação significativa.

**Objetivo do Projeto GFauto**

Ação que irá aplicar as técnicas mais modernas da análise, desenvolvimento e engenharia de softwares, marketing e administração visando melhorar e aperfeiçoar esse serviço. Transformando-se em uma aplicação web com Next.js, hospedada na Vercel, com funcionalidades de gerenciamento e integrações com serviços de terceiros, visando uma plataforma robusta e escalável.

**Fluxo\_app (antigo do** **Visitante)**

O primeiro foco da **página inicial(01)**: <https://www.gfauto.com.br> para o visitante é em "Começar Agora - Escolha o Estado" ou seja o “**Fluxo\_app**” o que vai ao encontro do slogam; que consta na figura: “logo.png” no cabeçalho “Pesquise o melhor local para o seu carro”. Ao preencher os campos “Estado” (26 opções), “Cidade” (5.500 opções), e “O que Procura?” (opções indeterminadas inicialmente) ele vai clicar no botão “Buscar Serviços”

- Atualmente hospedado na hostmach.com está funcionando assim:

Em [www.gfauto.com.br](http://www.gfauto.com.br) na sessão “Começar Agora” ao Clicar em “Escolha o Estado” abre um DropDown e o visitante clica no Estado desejado, no caso de escolher RS é mostrado para ele esta **página(02)**: <https://www.gfauto.com.br/estado/es_rs.htm> onde vai estar na página do RS e deverá escolher uma cidade desejada por ele (ex. Passo Fundo) e então ao clicar na cidade desejada é levado para esta **página(03)**: <https://www.gfauto.com.br/estado/passo_fundo.htm> onde lhe é mostrado algumas opções de Especialidades Automotivas na cidade de Passo Fundo que entre as quais ele pode escolher aquela especialidade que ele está buscando, caso ele clique em qualquer uma delas vai encontrar uma disposição tipo esta **página(04)**: **Página de Resultados** <https://www.gfauto.com.br/autoeletricas/autoeletricas.htm> caso tenha clicado em “Auto Elétricas” na página anterior. E isso ocorre com todas as outras especialidade.

- Mas como estamos trabalhando com banco de dados precisamos mudar isso:  
Em vez dos DropDown para o Estado, depois para a Cidade e depois clicar na Especialidade vamos fazer ele chegar na página 04 desejada de uma forma muito mais inteligente, eficiente e rápida:  
Então O “Começar Agora” será seguido de:

**Informe o Estado:** (campo para o Estado). **Informe a Cidade:** (campo para a Cidade). **O que procura?** (campo para a busca)

-01 Informe o Estado: pode ser digitado (e um dispositivo autocompletar) as opções serão as siglas dos estados e/ou os nomes completos aí deixo por sua conta definir como. Lembrando que precisamos que seja muito fácil para o Visitante. Nesse campo só será possível avançar para o próximo campo de o valor conferir com os constantes (estados do Brasil) no BD.  
-02 Informe a Cidade: mais ou menos a mesma lógica (autocompletar) no entanto a cidade tem que corresponder ao estado solicitado, considerando que temos o mesmo nome de cidade replicado em mais de um Estado. Aqui também só será possível avançar para o próximo campo se o valor conferir com os constantes (cidades por estado no Brasil) no BD.  
-03 O que procura?: Então ao saber o Estado e a Cidade o sistema vai buscar a Especialidade que o Visitante está buscando na cidade dele. E isso abre uma discussão interessante que precisamos fazer que é o tipo de busca. Ex.: O Visitante pode informar(digitar com ajuda do autocompletar): farol quebrado, o que vamos entregar para ele? Auto Elétricas, Revendas de Auto Peças? Ele pode também digitar direto Auto Elétrica e isso ficará mais fácil. Aqui só será possível avançar para o próximo campo (botão) se o valor conseguir identificar uma “Especialidade” que confira com as constantes no BD.

Por isso precisamos:

- Inicialmente Popular o banco de dados com termos e especialidades, ex.: as encontradas na pág. <https://www.gfauto.com.br/estado/passo_fundo.htm> e mais algumas que lembrar-mos.

- E na sequência   
Ex.: Uma busca inteligente enquanto o site for sendo utilizado até porque o “Visitante” é imprevisível então será necessário identificar o que ele busca e entregar a “página de resultados” mais próxima do que ele deseja.

- A lógica será chegar até a especialidade desejada (base que crescerá à medida que for sendo implementada com as informações digitadas pelo usuário no campo “Especialidade” da página(06) digitando ela própria diretamente ou as palavras a ela mais diretamente relacionadas frequentemente. A base dessas palavras relacionadas a cada especialidade crescerá também através das informações registradas no campo “descrição” da página(06) abaixo.

Nota: Precisamos pensar que essa busca será feita para pequenas, médias e grandes cidades o que quando for para pequenas cidades e médias cidades será tranquilo pois são poucos estabelecimentos por cidade, mas quando vai para grandes cidades mais de 1.500.000 de habitantes seria interessante a busca ser nos locais mais próximos da localização do Visitante, cerca de 10 Km. Caso não tenha ninguém no perímetro mais próximo ele aumenta o perímetro. Precisamos discutir a viabilidade de implementar isso.

Estas três opções tem que retornar do **BD Postgres Vercel**.: E como o BD será alimentado:

01 – Inicialmente podemos incluir todos os estados e estará pronto, só necessário alterar se o país criar um novo estado;

02 – Também será alimentado de forma definitiva (e cidades de acordo com seu respectivo estado) e alterado se for criada nova cidade.

03 – Esse nós iremos popular através de um estudo inicial entre as Especialidades e suas características e incluir um número significativo de possíveis opções que os usuário imaginem. Depois ele irá sendo alimentado conforme os usuário pesquisem por novas palavras / termos.

**Complementação da informação**

- "Autocompletar": É mais ou menos isso: O usuário consegue informar duas letras, ex RS ou outra Sigla dos outros Estados; (mas o sistema deveria deixar e sugerir, enquanto o usuário vai digitando, todas as siglas dos Estados do Brasil e ainda o nome completo seguindo o desejo do usuário.);

- "O que procura?": também só uma letra. (mas o sistema deveria deixar e sugerir as descrições contidas no campo central dos anúncios contidos nas "páginas de resultado" Obs.: Acho que uma idéia interessante aqui seria criarmos uma série de "Títulos" iniciais como vemos neste exemplo: print https://www.gfauto.com.br/estado/passo\_fundo.htm por cidade / Estado e que iria sendo implementado baseado nos novos anunciantes de cada cidade. Ex.: Esta seria a lista inicial para todas as cidades quando não há incidência de anúncio para uma dessas opções da lista inicial para uma das cidades o sistema responderá "não há anunciantes na especialidade "x" na cidade"; Que tal já popularmos o BD com essa lista inicial para todas as cidades do Brasil? Analise qual a melhor forma de implementar isso no sistema, ok?

**Nota importante**

BD Postgres, na Vercel. repositório GitHub GFauto. Serviço de e-mail resend.com, meio de pagamento Mercado Pago

1. **Interface de Cadastro de Anunciantes (MVP - Produto Mínimo Viável):**
   * Objetivo: Permitir o cadastro de novos anunciantes e o início do processo de pagamento.
   * Ações Chave: Criar página de “Cadastro do Cliente/Anunciante” (/fluxo\_cliente\_anunciante) com formulário, API para salvar dados, integração com API de pagamento existente, páginas de feedback (/pagamento/sucesso, /pagamento/falha).

**Autenticação e Gerenciamento Básico**

1. **Autenticação de Usuários (Anunciantes):**
   * Objetivo: Permitir que anunciantes criem contas seguras e acessem áreas restritas ao próprio Cliente/Anunciante permitindo a ele gerenciar o conteúdo do anúncio dele.
   * Ações Chave: Escolher e implementar solução (ex: NextAuth.js), proteger rotas/APIs, criar páginas de login/registro/recuperação de senha.
2. **Painel Básico do Anunciante:**
   * Objetivo: Fornecer aos anunciantes logados uma visão de seus dados, status de assinatura/pagamento e gerenciar o conteúdo do anúncio dele.
   * Ações Chave: Criar página de painel, exibir informações do banco de dados (atualizadas pelo webhook).

**Fluxo do Cliente/****Anunciante:**

O processo deverá acontecer praticamente sem intervenção administrativa, ou seja: O Cliente/Anunciante a partir do botão “Anunciar Agora” vai escolher o plano > informa seus dados (cadastro e login) (página(06) > cria seu login > informa dados do anúncio > efetua o pagamento > e publica seu anúncio. Essa publicação estará condicionada para o “Premium” ao “pagamento aprovado”. Quando o “Cortesia” ou o “Premium” quiserem fazer qualquer alteração no anúncio, basta logar e efetuá-las. Ele receberá um e-mail a cada alteração que ele efetuar.

**Cortesia** vai para > informa dados do anúncio > publicar (após publicar ele vai receber um e-mail parabenizando pelo sucesso na publicação mostrando o link para ele conferir (confira como ficou) e sugerindo para ele que pode fazer o upgrade pelo valor de R$ 36,00/1 ano – R$ 60,00/2 anos – R$ 75,00/3 anos, mostrando um exemplo de anúncio Premium e logo abaixo “Escolha seu Plano” se ele clicar vai para a página Plano. Onde irá logar e seguirá para o pagamento e restante do fluxo Premium >

**Premium** vai para pagto > informa dados do anúncio e publicar Recebe o e-mail parabenizando e com o link para visualizar o anúncio dele na página da especialidade dele na cidade dele.

Anúncios publicados. Essa publicação estará condicionada para o “Premium” ao “pagamento aprovado”. Quando o “Cortesia” ou o “Premium” quiserem fazer qualquer alteração no anúncio, basta logar e efetuá-las. Ele receberá um e-mail a cada alteração que ele efetuar.   
Quando o cliente faz o cadastro e login (que deve ser validado pelo e-mail dele), ao final recebe um e-mail de confirmação.

Quando o cliente efetua o pagamento ele receberá e-mails confirmando o pagamento (pagamento recebido) ou recusado, etc..

Da página(01) Como destacamos o botão “Anunciar Agora” ao clicar, ele é levado para a **página(05)** [clique para ver a escolha do Plano](#plano) que ainda não temos hospedada, precisamos construí-la e hospedá-la. Então na página(05) ele vai escolher o Plano e clicar no plano escolhido (registrar no BD) que o levará para página(06).

Se ele clicou no “Plano Cortesia” será levado para a página(06):

- a **página(06)**: <https://www.gfauto.com.br/aa_anuncio/form_anuncio.html> onde ele irá informar seus dados e do anúncio que serão inseridos no nosso Banco de Dados postgres vercel, (Nota: que no exemplo da página(06) faltam alguns dados como: CNPJ, Pessoa Responsável, CPF, Cel de contato.) Nessa mesma página ele deverá criar um usuário e senha para acessar a área dele. Assim que ele preencher tudo poderá clicar em “publicar” e o anúncio já aparece on-line para ele conferir (tipo essa: <https://www.gfauto.com.br/autopecas/autopecas.htm> ).

**Nesta página(06) Acho que devemos separar em duas partes:**1. Cadastro Básico com os seguintes campos de preenchimento: Nome / Razão Social, Nome de Fantasia, CNPJ, Pessoa Responsável, CPF, Cel. de Contato, Endereço da Empresa, Bairro, CEP, Cidade, Estado, Seu Cargo, Seu e-mail,

2. Dados do Seu anúncio:

Essas informações irão para o BD imediatamente pois esse anúncio deverá figurar na “Página de Resultado” junto com os outros anunciantes da mesma especialidade e mesma cidade.

Itens obrigatórios (\*)  
Preencha todos os Campos

|  |  |  |
| --- | --- | --- |
| Nome da Empresa: | Auto Peças Carrão | \*(caixa para o Nome) |

|  |  |  |
| --- | --- | --- |
| “Campo Especialidade:”(Identificador de campo para o desenvolvedor, não aparece na página)  Em qual **Especialidade** do Ramo automotivo você prefere figurar: Ex.: Auto Elétricas, Revenda Semi Novos, Mecânica, etc. | Acessórios, Alarmes e Som | \*(caixa para a Especialidade) |

|  |  |  |
| --- | --- | --- |
| Slogan: | Sua Garantia de Qualidade | \*(Caixa para o Slogam) |

|  |  |  |
| --- | --- | --- |
| Descrição: | Peças e Serviços / Todas as Marcas  Especializada em Desmontagem e Alto Aproveitamento Auto Peças Novas e Usadas  Compra-se Carros Acidentados | \*(Caixa para a Descrição) |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Celular | (11)9 9999 8888 e (11)9 8877 7788 | (Pode ser só 1) WhatsApp? | Sim |  | Não |  |  |

|  |  |  |
| --- | --- | --- |
| Endereço: | Rua Francisco de Lima Figueiredo, 345, centro  São Paulo/SP | \*(Caixa para Endereço) |

|  |  |  |
| --- | --- | --- |
| Inserir Imagem:  Arte, Foto do Cartão de Visita ou fachada da Empresa. (se não tiver a imagem agora, pode inserir através do botão “Atualizar Dados” no final do anúncio à direita a qualquer tempo). | Anexar arquivo Ou arraste e solte aqui | Anexar |

***(Obs. Para o Desenvolvedor.: Seria ideal o Cliente/Anunciante ao ir preenchendo os campos acima, Nome da Empresa, Slogan, etc... e conforme ele vai preenchendo já vai aparecendo no espaço do meio do anúncio, inclusive a imagem. Assim que ele preencher tudo poderá clicar em “publicar” e o anúncio já aparece on-line para ele conferir (tipo essa:*** [***https://www.gfauto.com.br/autopecas/autopecas.htm***](https://www.gfauto.com.br/autopecas/autopecas.htm) ***).***

Como ficará o Seu anúncio:

|  |  |  |
| --- | --- | --- |
|  | **Auto Peças Carrão**  Sua Garantia de Qualidade  Peças e Serviços / Todas as Marcas  Especializada em Desmontagem e Alto Aproveitamento Auto Peças Novas e Usadas  Compra-se Carros Acidentados  Fone/Whats (11) 9 9999 8888 e (11)9 8877 7788  **Rua Francisco de Lima Figueiredo, 345, centro**  **São Paulo/SP** | **Mapa**  (Localizar no google pelo endereço)  **Atualizar Dados** |

Confira na web:   
- Para “Cortesia”: “URL” onde ele poderá ver como estará figurando nos anunciantes da mesma especialidade dele na cidade dele.  
- Para “Premium”: Efetue o pagamento na página(06) e depois acesse: “URL” onde ele poderá ver como estará figurando nos anunciantes da mesma especialidade dele na cidade dele.

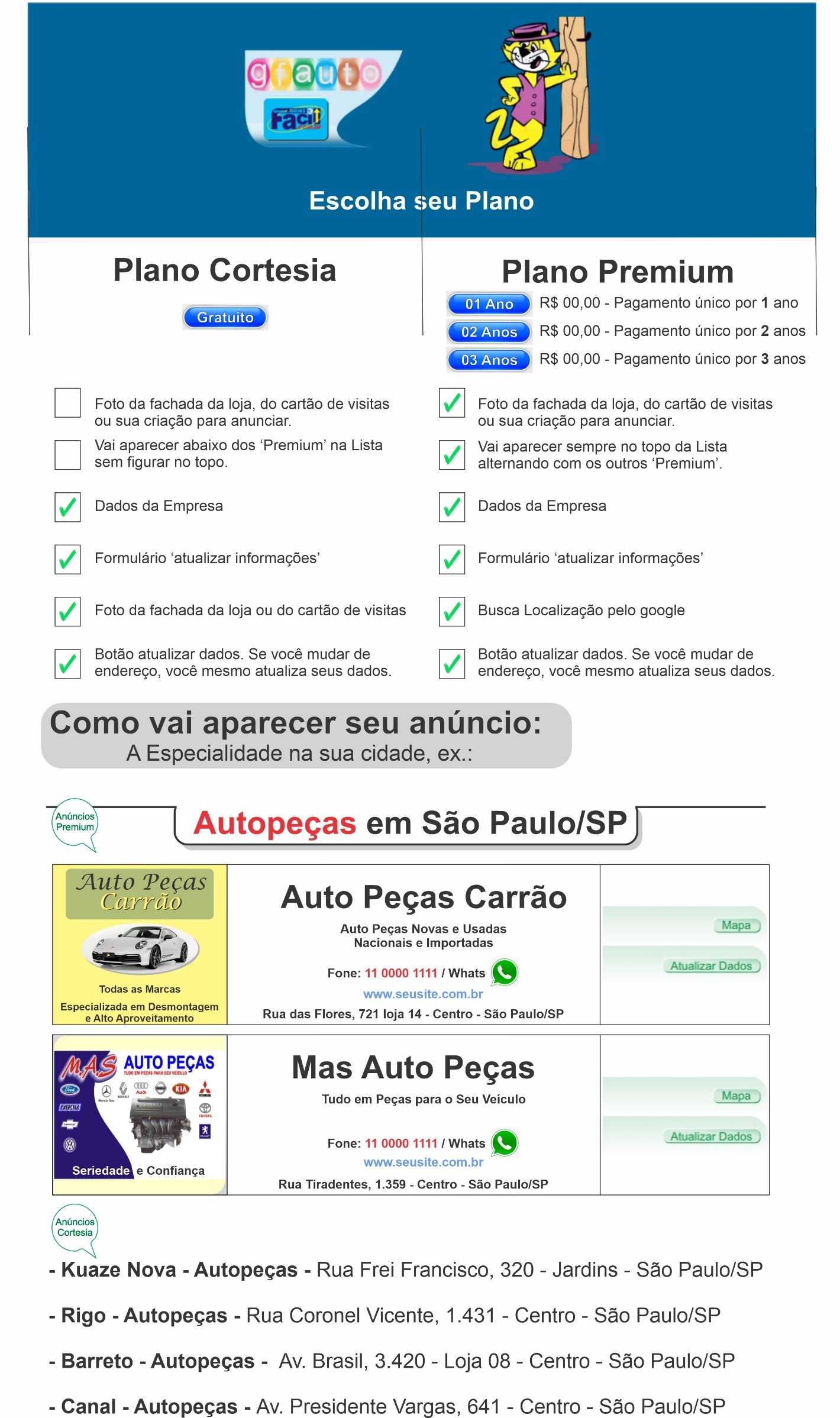
Esses dados do anúncio deverão ser armazenados no Banco de Dados de forma que sejam recuperados sempre que for necessário mostrar o anúncio. E o anúncio será mostrado dependendo do status de pagamento em dia.

- a **página(07)**: do pagto via Mercado Pago (que no momento não temos o front end dela mas o webhook do Mercado Pago já está testado e funcionando);

Para pagamentos aprovados, o cliente receberá um e-mail confirmando o pagamento e o status no BD ficará “Em dia”; Para pagamentos não aprovados, o cliente também receberá um e-mail informando e solicitando que refaça. E o status no BD ficará “Não Aprovado”

Página de Vendas – Ex. de Planos: Cortesia/Premium

Página do Plano – fluxo\_plano



A partir da “fig. acima exemplo de página”, há várias oportunidades para tornar o anúncio mais convincente e orientado à conversão imediata. Segue análises detalhadas e sugestões de melhorias, dividindo-as por áreas estratégicas:

**1. Design e Layout**

* **Profissionalismo e Modernidade:** Apesar do caráter lúdico (com o personagem animado e o visual “divertido”), é fundamental transmitir confiança e seriedade para o negócio. Sugiro uma atualização do layout com um design mais minimalista, moderno e responsivo. Considere uma paleta de cores que transmita credibilidade—tons de azul ou verde, por exemplo—e tipografias limpas que facilitem a leitura, principalmente para clientes que buscam serviços de autopeças.
* **Hierarquia da Informação:** Organize as seções de maneira a guiar o olhar do visitante rumo à ação principal: a contratação do anúncio. Destaque o plano Premium (R$ 30,00/ano) com elementos visuais marcantes, como bordas diferenciadas, cores de destaque ou efeitos de sombra. Isso pode ser feito através de um “badge” ou selo que remeta ao “destaque” e “exclusividade”.
* **Imagens e Ícones:** Use imagens de alta qualidade das autopeças e da oficina, gráficos ou mesmo pequenos ícones para ilustrar os benefícios dos planos. Uma representação visual comparativa entre o Plano Cortesia e o Premium pode ser muito eficaz para reforçar a vantagem adicional do investimento.

**2. Conteúdo e Mensagem**

* **Clareza na Proposta de Valor:** É essencial que, logo de início, o visitante compare rapidamente as vantagens entre a opção gratuita e o Premium. Utilize uma linguagem direta e impactante para explicar que, investindo apenas R$ 30,00 por ano, o anúncio ganhará maior visibilidade, destaque e, consequentemente, potencial de retorno maior. Frases como “Destaque seu anúncio e venda mais” ou “Seu negócio no topo da lista” podem ajudar.
* **Benefícios Tangíveis:** Liste os benefícios de forma clara e objetiva. Por exemplo, se o plano Premium inclui recursos exclusivos (como busca por localização, alternância entre anúncios Premium ou uma vitrine de destaque), crie uma tabela comparativa ou uma lista com ícones que deixe evidente como cada recurso pode aumentar a visibilidade e a conversão. Se houver provas de aumento de vendas ou casos de sucesso, insira testemunhos rápidos e números (ex.: “Nosso cliente X aumentou suas vendas em 25% após migrar para o Premium”).
* **Call to Action (CTA) Forte:** Os botões “Atualizar Dados” e “Ver Mapa” são úteis, mas o foco principal deve ser em “Assine Agora” ou “Publique Seu Anúncio”. Garanta que o CTA esteja bem destacado, com cores contrastantes e uma mensagem irresistível. Considere testar phrases como “Garanta seu destaque agora” ou “Impulsione suas vendas por apenas R$ 30,00/ano.”

**3. Usabilidade e Jornada do Usuário**

* **Navegação Simplificada:** A estrutura da página deve levar o usuário a entender rapidamente qual é a próxima ação a ser tomada. Reduza a quantidade de informações secundárias na página inicial e crie um fluxo de navegação intuitivo que leve direto à ação de publicação do anúncio.
* **Processo de Conversão Rápido:** O formulário de “atualizar informações” pode ser simplificado para que seja preenchido em poucos cliques. Considere uma abordagem em etapas (passo a passo) para não sobrecarregar o potencial anunciante logo de cara. Quanto mais simples e ágil o processo, maior a probabilidade de conversão.
* **Mobile First:** Certifique-se de que a experiência seja perfeita em dispositivos móveis. Muitos anunciantes acessam a internet pelo celular; um layout responsivo e adaptado para telas menores é imprescindível para não perder nenhuma oportunidade.

**4. Estratégias de Persuasão e Marketing**

* **Gatilhos Mentais:** Prova social (testemunhos reais de clientes satisfeitos) para incentivar a ação imediata. Essa abordagem ajuda a reduzir as dúvidas e a acelerar o processo decisório.
* **Demonstração de ROI:** Mostre, de forma concreta, como o investimento de R$ 30,00 é rentável. Se você conseguir 1 cliente no ano o investimento já estará mais do que pago.
* **Integração com Redes Sociais e SEO:** Otimize a página para mecanismos de busca utilizando palavras-chave relevantes, o que pode atrair ainda mais negócios para o site e, consequentemente, mais anunciantes.

**5. Extras para Elevar a Conversão**

* **Live Chat ou Suporte Imediato:** Uma ferramenta de atendimento via chat pode ajudar a tirar dúvidas dos possíveis anunciantes em tempo real, resolvendo objeções e facilitando a tomada de decisão. (isso poderá ser implementado mais adiante)
* **Teste A/B:** Considere implementar testes A/B para diferentes layouts, cores e CTAs. Assim, você poderá identificar quais elementos funcionam melhor para converter os visitantes em clientes pagantes.
* **Landing Page Dedicada:** Uma alternativa é desenvolver uma landing page exclusiva para explicar os benefícios do plano Premium. Nessa página, todo o conteúdo seria focado em convencer o anunciante, com depoimentos, chamadas para a ação fortes e uma apresentação visual que reforce a exclusividade e o baixo custo do investimento.

**Conclusão**

Para atingir o objetivo de convencer o cliente e vender imediatamente o anúncio, é fundamental alinhar um design profissional e moderno com uma mensagem clara, objetiva e persuasiva. A experiência do usuário deve ser ágil, permitindo que o anunciante compreenda rapidamente os benefícios do investimento de R$ 30,00 (será mais ou menos esse valor) anual e tome a decisão de publicar o anúncio sem hesitações.

Esse conjunto de melhorias – desde a atualização do design até a implementação de gatilhos mentais e ferramentas de conversão – não apenas aumentará a taxa de conversão, mas também posicionará a marca de forma mais forte no mercado automotivo.

**Fale Conosco**

Da página gfauto.com.br leva para o arquivo: <https://www.gfauto.com.br/aa_anuncio/form_contato.html> e envia o e-mail e imagem para [rluizweber@yahoo.com.br](mailto:rluizweber@yahoo.com.br) (testado e funcionando em jun/2025)

# ## Documentação da Integração do Fluxo do Visitante - GFauto

## Visão Geral

Este documento descreve a integração completa do fluxo\_app ao projeto GFauto, incluindo:

1.Modelagem de dados (schema.prisma)

2.APIs para busca dinâmica e rotação de anúncios

3.Componentes frontend

4.Configuração de rotas Next.js

## 1. Alterações no Schema Prisma

O schema.prisma foi estendido para incluir os seguintes modelos:

### Novos Modelos Adicionados:

•Estado: Armazena estados brasileiros (id, nome, sigla)

•Cidade: Armazena cidades vinculadas a estados (id, nome, estadoId)

•Especialidade: Armazena especialidades automotivas (id, nome, descricao, slug, icone)

•RotacaoPremium: Controla a rotação circular dos anúncios premium (id, especialidadeId, cidadeId, ultimaPosicao)

•EspecialidadeDisponivel: Rastreia especialidades disponíveis em cada cidade (id, cidadeId, especialidadeId)

•ImagemAnuncio: Armazena múltiplas imagens para cada anúncio (id, anuncioId, url, ordem)

### Modelo Anuncio Estendido:

O modelo Anuncio existente foi estendido com os seguintes campos:

•titulo, descricao, endereco, telefone, whatsapp, email, site

•plano (premium/cortesia)

•especialidadeId e cidadeId (com relacionamentos)

•imagemPrincipal, latitude, longitude

•dataExpiracao

## 2. APIs Implementadas

Foram implementadas as seguintes APIs:

### /api/estados

•Método: GET

•Descrição: Retorna a lista completa de estados brasileiros ordenados por nome

•Arquivo: /app/api/estados/route.ts

### /api/cidades

•Método: GET

•Parâmetros: estado\_id (obrigatório)

•Descrição: Retorna a lista de cidades do estado especificado

•Arquivo: /app/api/cidades/route.ts

### /api/especialidades

•Método: GET

•Parâmetros: cidade\_id (obrigatório)

•Descrição: Retorna a lista de especialidades disponíveis na cidade especificada

•Arquivo: /app/api/especialidades/route.ts

### /api/anuncios

•Método: GET

•Parâmetros: cidade\_id, especialidade\_id (obrigatórios), page, pageSize (opcionais)

•Descrição: Retorna anúncios filtrados por cidade e especialidade, com rotação circular para anúncios premium

•Arquivo: /app/api/anuncios/route.ts

### /api/anuncios/[id]

•Método: GET

•Parâmetros: id (na rota)

•Descrição: Retorna detalhes completos de um anúncio específico

•Arquivo: /app/api/anuncios/[id]/route.ts

## 3. Componentes Frontend

Foram implementados os seguintes componentes:

### Componentes Principais:

•BuscaForm: Formulário de busca em três etapas (Estado > Cidade > Especialidade)

•HeroSection: Banner principal da página inicial

•ResultadosList: Lista de resultados da busca com paginação

•AnuncioCard: Card de anúncio com layout em três colunas

•Pagination: Componente de paginação para resultados

•LoadingResults: Estado de carregamento para resultados

### Páginas:

•Página Inicial (/app/page.tsx): Página principal com formulário de busca

•Página de Resultados (/app/resultados/page.tsx): Exibe resultados da busca

•Página de Detalhes do Anúncio (/app/anuncio/[id]/page.tsx): Exibe detalhes completos de um anúncio

## 4. Sistema de Rotação Circular

O sistema de rotação circular para anúncios premium foi implementado na API /api/anuncios e funciona da seguinte forma:

1.Busca anúncios premium filtrados por cidade\_id e especialidade\_id

2.Busca ou cria um registro na tabela rotacao\_premium para a combinação cidade+especialidade

3.Aplica a rotação com base na última posição registrada

4.Atualiza a posição para a próxima consulta

5.Adiciona os anúncios cortesia (sem rotação) após os premium

## 5. Fluxo de Navegação

O fluxo de navegação do visitante funciona da seguinte forma:

1.Usuário acessa a página inicial (/)

2.Seleciona estado, cidade e especialidade no formulário de busca

3.É redirecionado para a página de resultados (/resultados?cidade\_id=X&especialidade\_id=Y)

4.Visualiza os anúncios com rotação circular para premium

5.Pode clicar em um anúncio para ver detalhes (/anuncio/[id])

6.Pode navegar entre páginas de resultados usando a paginação

## 6. Próximos Passos

1.Realizar testes completos do fluxo integrado

2.Implementar integração com biblioteca de mapas para exibir localização dos anúncios

3.Importar dados reais de estados e cidades brasileiros

4.Adicionar testes automatizados para garantir a qualidade do código

**Dizer para ele acrescentar isso abaixo no Resumo 01 e que eu vou pedir para ele consultar o Resumo 01 cada vez que retornarmos de uma pausa (um dia para o outro)**

**=============================================================**

**Convenções:**

**Convenções de Nomenclatura e Localização**

**4.1. Ambientes de Referência**

**Repositório GitHub**: Repositório público do Projeto GFauto

**Estrutura Local**: Arquivos e pastas na máquina do desenvolvedor (deve estar igual o Repositório GitHub)

**Estrutura Vercel**: Ambiente de produção (deve estar igual também)

**SandboxGFauto**: Ambiente do Desenvolvedodr (deve estar igual também)

Meu Local: cd /w/A\_Weber/Pai/Hostmachine/gfauto/githubVercel/Gfauto Para **(main)** e **(admin-panel)**

28/06