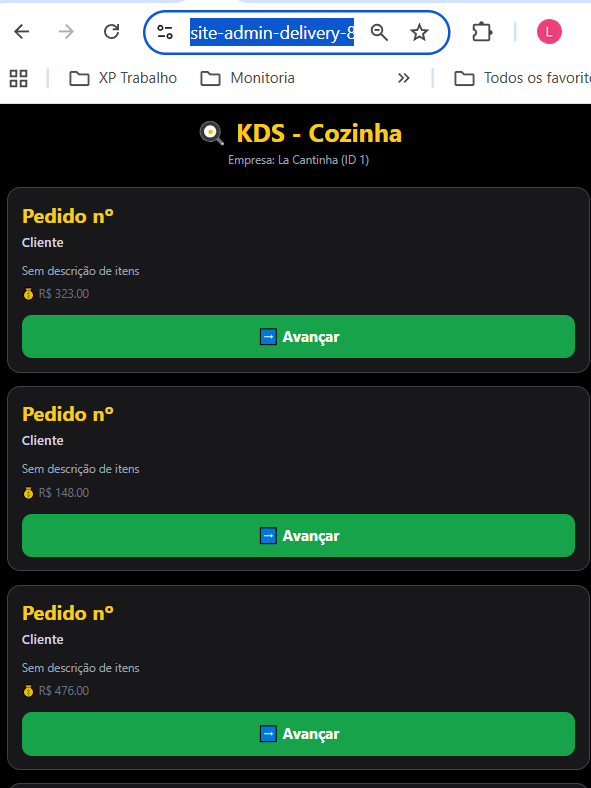
Requisitos KDS

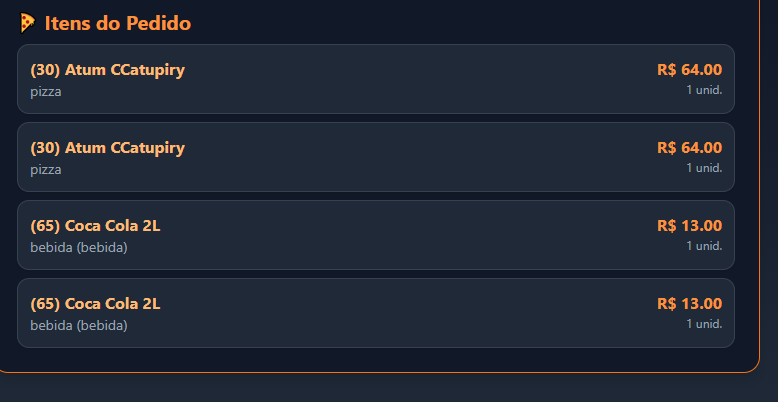
**💡 O que mudou**

* Mostra **número do pedido como link clicável** → abre o detalhe no navegador  
  (/webhook/detalhes?numero=...&id\_empresa=...)
* Mostra **nome do cliente**, **itens** e **valor**.
* Botão verde mudou para **➡️ “Avançar”**, que chama  
  POST /webhook/pedido\_avancar (você pode apontar pro mesmo endpoint que usa no dashboard).
* Remove o card da tela após avançar (igual comportamento do painel admin).

Se quiser, posso deixar o **modal interno** (em vez de abrir link externo), tipo o detalhe que aparece no painel — só em overlay.  
Quer que eu faça esse próximo passo?

**Você disse:**





precisamos volar pra prancheta ok nada de codigo.. vamos discutir requisitos.. o que é mais fácil para funcionario que recisam entender de aplicativo de celular na cozinhao que fica mais faciL vamos la sem codigo so em resquisitos ok ? nome do cliente fora.. numero do pedido tem que aparecer.. talves dois botoes visualiar avancar pode ser alaranjado... quando clicar em visualiar vai abrir o pedido.. (vou criar um webhook visualizarcozinha .. somente com o que ele precisa ver... ) bem bacana visual jovem .. vou mostrar do pedido o valor e numero e todos os item que eu vou fazer uma proc o que vc acha ? uma barra de rolagem ? bem proximo do celular e ipad... não deixar duvida dos itens vou fazer uma proc que qualquer analfaneto entenda .. sem codigo ok vamos falar de requisistos.. faltou algo ?

**O ChatGPT disse:**

Perfeito 👌 — agora sim estamos falando de **requisitos funcionais e de usabilidade**, não de código.  
Vamos listar juntos o **escopo ideal da tela KDS – Modo Cozinha** para o seu caso (tablet/celular, funcionário de cozinha, sem confusão visual).

**🍕 Objetivo Geral**

Permitir que o **funcionário da cozinha veja, entenda e avance pedidos** rapidamente — sem precisar interpretar muito, ler textos pequenos ou navegar em telas complexas.

**🧱 Estrutura da Tela KDS (visão cozinha)**

**1. Cabeçalho fixo (topo)**

* Título: **“🍳 KDS – Cozinha”**
* Nome da empresa (ex: “La Cantinha”)
* Mostrador de tempo (ex: “Atualizado há 5s”)
* Botão discreto “🔄 Atualizar” (caso a conexão caia)
* Fundo preto ou cinza escuro, fonte grande e legível

**2. Cards de Pedido (grid flexível)**

Cada pedido é um **card** grande e limpo:

* Fundo escuro (cinza carvão)
* Bordas arredondadas
* Boa separação entre pedidos (espaço respirando)

**Conteúdo do card:**

1. **Número do pedido** em destaque → “Pedido nº 45”
2. **Valor total** → “💰 R$ 157,00”
3. **Resumo rápido dos itens** (ex: “2 pizzas, 2 bebidas”)
   * Pode ser texto sintético (“🍕 x2 🧃 x2”)
4. **Duas ações principais:**
   * 🟧 **Visualizar** (abre detalhes do pedido)
   * 🟩 **Avançar** (muda status para “pronto”)

**3. Ação “Visualizar”**

* Abre **modal sobreposto** (ou página simples) com:
  + **Número do pedido**
  + **Lista dos itens** (com nome completo, borda, observações, etc.)
  + **Quantidade e tipo de item**
  + **Valor total**
* Visual limpo: letras grandes, ícones de pizza, bebida, borda.
* Scroll vertical suave (caso o pedido tenha muitos itens)
* Botão “Fechar” bem visível no topo.

**4. Ação “Avançar”**

* Confirmação visual imediata: card some da tela (feedback instantâneo).
* Som opcional (bipe curto ou cor piscando) pra indicar sucesso.
* O botão “Avançar” deve:
  + Ter **cor laranja/amarela** enquanto o pedido está “Recebido”
  + Virar **verde** quando está “Em preparo”
  + Desaparecer após marcar “Pronto”

**5. Design e Experiência de Uso**

* **Responsivo** (1 coluna no celular, 2–3 no tablet)
* **Texto grande** (mínimo 18 px)
* **Ícones de apoio** (pizza, bebida, etc.)
* **Sem scroll lateral** (somente vertical)
* **Autoatualização a cada 5–10 s**
* **Indicação de status** por cor:
  + 🟧 Recebido
  + 🟦 Em preparo
  + 🟩 Pronto
* **Sem menus ou distrações** (modo kiosk)
* Ideal para tela sensível ao toque

**6. Apoio visual e feedback**

* Quando há poucos pedidos → mostrar “Nenhum pedido em preparo 🍕”
* Quando entra um novo pedido → animação curta de destaque (ex: brilha por 2 s)
* Feedback ao clicar em “Avançar”:  
  “✅ Pedido nº 45 avançado para entrega!”

**7. Fluxo de uso**

1. Novo pedido entra → aparece na tela
2. Cozinheiro toca em “Visualizar” → vê detalhes simples
3. Prepara → toca “Avançar”
4. Pedido some da tela KDS e aparece na aba seguinte (ex: Entrega)

**💡 Ideias extras (opcionais)**

* Filtro no topo: “📦 Todos | 🔪 Em preparo | 🚀 Pronto”
* Indicador de tempo de preparo (contador desde o “Recebido”)
* Modo **horizontal automático** no tablet (rotação fixa)
* Cores mais quentes para a cozinha (laranja, amarelo, preto)

**✅ Resumo final (sem código)**

**Requisitos obrigatórios:**

* Mostrar número e valor do pedido
* Exibir itens de forma visual e legível
* Dois botões: **Visualizar** e **Avançar**
* Visualizar → abre modal simples com todos os itens
* Avançar → muda status e remove da lista
* Interface 100% touchscreen
* Layout adaptado pra celular/iPad (sem menus)

Quer que eu monte o **protótipo visual (mockup)** dessa tela com esse layout pra você validar antes de codar?  
Posso gerar a imagem realista de como ficaria no iPad/cozinha.

**Você disse:**

muito legal.. o avancar tem pronto o webhook no dashboad podemos aprovietar.. o usuario nçao pode avancar sem querer ou na verdade pode e perder o pedido e ficar perdido ,. lembre-se não tem imprensao logo talvez ele tenha que visualizar um pedido avançado sem querer ok .. segue o webhoo do avancar com pedido e empresa é so aproveitar const avancarPedido = async (numero) => { const id\_empresa = getIdEmpresaSafe(); if (!id\_empresa) { alert("Empresa não identificada. Abra o cardápio/logue novamente."); return; } try { const response = await fetch( "https://webhook.lglducci.com.br/webhook/avancar", { method: "POST", headers: { "Content-Type": "application/json" }, body: JSON.stringify({ numero, id\_empresa }), } ); let data = null; try { data = await response.json(); } catch {} if (!response.ok) { console.error("Falha ao avançar:", response.status, data); alert("Falha ao avançar o pedido."); return; } window.location.reload(); } catch (err) { console.error("Erro ao avançar pedido:", err); alert("Erro ao avançar pedido!"); } };

**O ChatGPT disse:**

Perfeito 👏 — e ótimo ponto: **sem impressora, o risco de avançar “sem querer” é real**.  
Então, antes de pensar em código, o requisito é proteger o fluxo do cozinheiro pra ele **nunca perder um pedido**.

**🍕 Requisitos funcionais revisados para o botão “Avançar”**

**1️⃣ Confirmação antes de avançar**

* Sempre que o usuário tocar em **“Avançar”**, o sistema deve perguntar:

“Tem certeza que o pedido nº 45 foi concluído?”

* + Botões: **[Sim, avançar]** | **[Cancelar]**
* Somente após confirmar, o webhook avancar é disparado.

**2️⃣ Histórico ou reabertura rápida**

* Criar uma visão **“Pedidos Avançados (últimos 10 min)”**, visível com um toque, mostrando:
  + Nº do pedido
  + Status atual
  + Botão **“Reabrir”** (regrava status anterior no banco)
* Isso resolve o caso de “avancei sem querer”.
* Essa tela pode ser opcional, acessada por ícone discreto “🕓 Histórico”.

**3️⃣ Bloqueio de toque acidental**

* Botão “Avançar” grande, mas exige **toque longo (1 s)** ou **duplo toque** para acionar.
  + Exemplo: “Toque e segure para avançar”.
* Visual: muda de cor durante o toque (“Confirmando…”).

**4️⃣ Feedback visual forte**

* Ao avançar:
  + Card pisca em verde por 1 s (“✅ Pedido avançado”).
  + Some da lista.
* Evita o medo de “será que foi?”.

**5️⃣ Fluxo de exceção (sem imprimir)**

* O cozinheiro deve conseguir **rever um pedido avançado**, pelo número:
  + Campo de busca simples: “🔍 Buscar pedido nº \_\_\_”.
  + Usa o mesmo webhook visualizarcozinha que você mencionou.
* Assim, se o cara avançar errado, ele consegue achar e ver o conteúdo.

**🧩 Resumo de comportamento ideal**

| **Etapa** | **Ação** | **Reação** |
| --- | --- | --- |
| Toque no card | Abre detalhes (modal) | Mostra itens e botão “Avançar” |
| Toque em “Avançar” | Pergunta confirmação | “Deseja avançar o pedido nº X?” |
| Confirmar | Chama webhook /avancar | Card pisca verde e some |
| Erro | Mostra alerta vermelho | “Falha ao avançar pedido” |
| Avançado por engano | Vai em 🕓 Histórico | Pode visualizar e reabrir |

**💡 Experiência mobile**

* Tudo “full touch” com fonte grande.
* Evita popups complexos — usa modais grandes e botões largos.
* No histórico, destacar visualmente o tempo (“⏱️ avançado há 2min”).

👉 Esse conjunto de requisitos garante que o **cozinheiro não se perca**, não avance por engano e ainda consiga **corrigir um erro sem ajuda do admin**.

Quer que eu monte agora o **wireframe (mockup visual)** dessa tela, com esses dois botões e o fluxo de confirmação / histórico, pra validar o layout antes de codar?

**Você disse:**

em portugues::: falta alguma coisa que deixamos pra tras ?

**O ChatGPT disse:**

Excelente pergunta 👏 — vamos revisar **tudo o que já definimos** e ver se há lacunas antes de passar para o design e implementação.

**🍳 Revisão geral da tela KDS – Cozinha**

**✅ Objetivo**

Exibir pedidos de forma simples e visual para que o funcionário da cozinha:

* entenda **rapidamente o que deve preparar**,
* **avance o status** quando terminar,
* e consiga **corrigir erros** (ex: avançar sem querer),  
  sem precisar de impressão ou treinamento técnico.

**🧱 Elementos principais da tela**

1. **Cabeçalho fixo**
   * Título: “🍳 KDS – Cozinha”
   * Nome da empresa (ex: “La Cantinha (ID 1)”)
   * Indicador de atualização (“Atualizado há 5 segundos”)
   * Botão de atualização manual 🔄 (caso a conexão caia)
2. **Lista de pedidos (cards grandes)**
   * Fundo escuro e letras grandes (para leitura fácil)
   * Cada card mostra:
     + **Número do pedido** (grande e destacado)
     + **Valor total**
     + **Resumo rápido dos itens** (ex: 2 pizzas, 1 bebida)
     + Botões:
       - 🟧 **Visualizar** → abre os detalhes
       - 🟩 **Avançar** → muda status (com confirmação)
   * Cores por status:
     + 🟧 Recebido
     + 🟦 Em preparo
     + 🟩 Pronto / Finalizado
3. **Modal “Visualizar pedido”**
   * Abre sobre a tela (sem sair da página)
   * Mostra:
     + Nº do pedido
     + Itens (nome, tipo, quantidade, observações)
     + Valor total
   * Scroll vertical suave (para pedidos longos)
   * Botões:
     + “Fechar”
     + “Avançar” (repete função do card)
   * Fonte grande, layout limpo, ícones de apoio (🍕, 🧃, 🧀 etc.)
4. **Ação “Avançar”**
   * Usa webhook /webhook/avancar
   * Regras:
     + **Confirmação antes de enviar** (“Tem certeza que o pedido nº 45 foi concluído?”)
     + Pode exigir toque longo ou duplo toque
     + Mostra feedback visual (“✅ Pedido avançado!”)
     + Card desaparece da tela
   * Erro → alerta vermelho (“Falha ao avançar pedido”)
5. **Histórico / Reabertura**
   * Ícone discreto “🕓 Histórico”
   * Mostra últimos pedidos avançados (ex: últimos 10 minutos)
   * Permite visualizar ou reabrir (se avançou por engano)
   * Campos:
     + Nº do pedido
     + Status
     + Tempo desde o avanço (“há 3 min”)
     + Botão “Reabrir”
6. **Apoio visual**
   * Ícones: 🍕 pizza, 🧃 bebida, 🧀 borda
   * Mensagens curtas e diretas (“Nenhum pedido em preparo 🍕”)
   * Feedback imediato (sem depender de reload)
7. **Usabilidade mobile**
   * 1 coluna (celular), 2 ou 3 colunas (tablet)
   * Texto mínimo 18 px
   * Botões grandes e bem separados
   * Sem menus, sem rolagem lateral
   * Autoatualização suave (a cada 5 s)

**🔍 Verificação final – O que falta**

| **Área** | **Situação** | **Observação** |
| --- | --- | --- |
| Cabeçalho e autoatualização | ✅ | Já previsto |
| Exibição de pedidos | ✅ | Layout definido |
| Visualizar detalhes | ✅ | Falta apenas definir o **webhook visualizarcozinha** |
| Avançar pedido | ✅ | Usa webhook /avancar, com confirmação |
| Proteção contra erro | ✅ | Toque longo ou confirmação + histórico |
| Reabrir pedido avançado | 🟡 | Planejado, ainda sem endpoint (precisa /reabrir) |
| Feedback visual/sonoro | 🟢 | Planejado (piscar verde, opcional som curto) |
| Filtros de status (Recebido, Em preparo, Pronto) | 🟡 | Opcional, pode entrar depois |
| Busca por número de pedido | 🟡 | Recurso extra para emergências |
| Responsividade | ✅ | Já definido (1–3 colunas) |
| Impressão | ❌ | Intencionalmente fora |

**💡 Conclusão**

👉 **O sistema KDS está conceitualmente completo.**  
O único ponto pendente é **decidir o comportamento do “Histórico / Reabrir”**, que pode ser implementado depois da primeira versão.

Se quiser, posso montar o **fluxo visual (mockup de tela)** com esses elementos para você aprovar o layout antes da implementação.  
Quer que eu gere essa prancheta visual (em estilo iPad, com os dois botões e modal de pedido)?

**Você disse:**

pensei em algo .. se der uma dor de barriga no frente de loja a cozinha pode ter um botão requisitar que avanca todos os pedios recebidos para produção ....

**O ChatGPT disse:**

Perfeito 👏 — isso é **muito inteligente operacionalmente** e totalmente coerente com o dia a dia da pizzaria.  
Vamos documentar esse novo requisito “**Botão Requisitar Tudo**” (ou “Avançar Todos para Produção”) dentro do contexto do KDS.

**⚙️ Nova funcionalidade: “Requisitar / Avançar Todos”**

**🎯 Objetivo**

Garantir que, em situações emergenciais (ex: atendente ausente, pico de pedidos, ou problema no frente de loja),  
a **cozinha consiga puxar todos os pedidos “Recebidos”** e colocá-los em **“Produção”** de forma centralizada — sem precisar do painel principal.

**🧩 Requisitos funcionais**

**1️⃣ Local do botão**

* Fica **no topo da tela KDS**, próximo ao título, ou em uma **barra fixa inferior**.
  + Ícone sugerido: 🔄 ou ⚡
  + Texto: **“Requisitar pedidos recebidos”** ou **“Puxar pedidos para produção”**
* Cor **laranja intensa** (diferente do “Avançar” verde).

**2️⃣ Ação do botão**

* Ao tocar, o sistema exibe **uma confirmação clara**:

“Deseja enviar todos os pedidos *Recebidos* para *Produção*?”

* + Botões: **[Confirmar]** | **[Cancelar]**
* Após confirmar:
  + Chama o webhook:
  + POST /webhook/avancar\_todos
  + { id\_empresa, destino: "producao" }
  + Backend atualiza o status de todos os pedidos “Recebidos” → “Produção”.

**3️⃣ Feedback visual**

* Enquanto processa → mostra animação “⏳ Atualizando pedidos...”
* Após sucesso → banner verde:

“✅ Todos os pedidos foram enviados para produção!”

* Tela se atualiza automaticamente e os pedidos aparecem nos cards do KDS.

**4️⃣ Restrições de segurança**

* Só disponível para **usuário tipo\_admin = 'cozinha'**.
* Pode ter **limite de frequência** (ex: só pode acionar 1x a cada 5 minutos).
* (Opcional) Registrar no log:
  + usuario\_cozinha, data\_hora, qtd\_pedidos\_afetados.

**5️⃣ Casos de uso**

| **Situação** | **Resultado esperado** |
| --- | --- |
| Atendente ausente | Cozinha clica “Requisitar” → todos os pedidos entram em produção |
| Pico de pedidos | Permite ao cozinheiro “adiantar” o fluxo |
| Pedido travado no “Recebido” | Força atualização e evita fila parada |
| Falha temporária de comunicação com o painel | Cozinha pode continuar produção sem depender do frente |

**6️⃣ Design e experiência**

* Botão **grande**, **bem separado** dos botões normais (para evitar toques acidentais).
* Cor **alaranjada / âmbar**, com ícone de relâmpago ou engrenagem ⚡.
* Após o clique, botão desabilita por alguns segundos (“Executando…”).
* Pode exibir **contador de pedidos movidos** (ex: “5 pedidos enviados”).

**💡 Extras (opcionais para versão 2)**

* Configuração no painel admin para **habilitar ou não** essa função no KDS.
* Enviar **alerta interno** (ex: mensagem no WhatsApp do gerente: “Cozinha requisitou todos os pedidos às 22:35”).
* Integrar ao log de auditoria (“Quem puxou todos os pedidos e quando”).

**✅ Resumo Final**

| **Item** | **Status** |
| --- | --- |
| Nome da função | “Requisitar pedidos recebidos” |
| Local | Barra superior ou inferior no KDS |
| Cor | Laranja ⚠️ (diferente do verde “Avançar”) |
| Confirmação | Sim, obrigatória |
| Webhook | /webhook/avancar\_todos (usa id\_empresa) |
| Feedback | Mensagem “✅ Todos enviados para produção” |
| Segurança | Apenas tipo\_admin = 'cozinha' |
| Logs | Recomendado (quem acionou e quando) |

Quer que eu te ajude a desenhar **como essa função vai aparecer visualmente na tela** (posição, tamanho, cor e comportamento da confirmação)?  
Posso gerar um layout visual da versão “KDS com botão Requisitar Tudo”.