**Objetivo do Projeto:** Continuar o desenvolvimento de uma aplicação de página única (SPA) para gestão de campanhas de marketing. A base de dados no Supabase já está populada. A tarefa atual é refinar o módulo de importação existente e depois conectar os dashboards restantes aos dados reais.

**Contexto:** O projeto já possui funcionalidades de CRUD, um módulo de importação funcional e o primeiro de três dashboards conectado aos dados. O código-fonte completo está abaixo. O próximo passo é adicionar validações de dados e depois construir os dashboards restantes.

**Requisitos Fundamentais:**

1. **Arquivo Único:** Todo o código (HTML, CSS e JavaScript) deve ser mantido em um único arquivo .html.
2. **Manter Layout:** A interface e o layout existentes não devem ser alterados.
3. **Sistema de Versão:** Cada nova entrega de código deve incrementar a versão do software em 0.01.

**Artefato Principal: Código Base Completo (Versão 62.12)** Abaixo está o código completo e funcional da aplicação no seu estado atual. Ele inclui todos os módulos de CRUD, importação funcional e o Dashboard de Performance conectado aos dados reais.

**Próximas Tarefas Detalhadas**

**Tarefa 1: Implementar Validações no Módulo de Importação**

O script de importação atual funciona, mas precisa de validações para melhorar a qualidade dos dados e o feedback ao usuário.

* **No Passo 1 (Inscrições):**
  1. Ao ler cada linha do CSV, verificar se a coluna email está presente.
  2. Verificar se o valor do e-mail tem um formato válido (contém @ e .).
  3. Se uma linha falhar em qualquer uma dessas validações, ela deve ser pulada, e uma mensagem de aviso específica deve ser registrada no log (ex: "AVISO: Linha 25 ignorada. Formato de e-mail inválido.").
* **No Passo 2 (Check-ins):**
  1. O script já verifica se o e-mail do check-in existe na base de inscritos. Manter esta lógica.
  2. Ao calcular o score, para perguntas que não são de "Texto Aberto", verificar se a resposta do CSV existe no array de opcoes da pergunta no banco de dados.
  3. Se a resposta não for uma das opções válidas, registrar um aviso no log (ex: "AVISO: Resposta 'Talvez' para o lead 'joao@email.com' é inválida para a pergunta 'Você já comprou?'. Score não atribuído para esta resposta."), salvar a resposta na tabela respostas\_leads, mas não somar seu peso ao score do lead.

**Tarefa 2: Conectar o Dashboard de Performance.**

Efetuar novos testes após os ajustes das importações.

**Conectar o Dashboard de Lead Scoring**

* **Filtros:** Adicionar um dropdown para selecionar um "Lançamento". Todas as queries devem ser filtradas pelo launch\_id selecionado.
* **KPIs:** Calcular "Total Inscrições", "Total Check-ins" e "Taxa de Check-in" para o lançamento selecionado.
* **Tabela de Resumo por Score (lead-scoring-table):**
  + Agrupar os leads por utm\_content.
  + Para cada utm\_content, contar quantos leads se encaixam em cada faixa de score: Frio (<35), Morno-Frio (35-49), Morno (50-64), Quente-Morno (65-79), Quente (80+).
* **Gráficos (Chart.js):**
  + **qualified-leads-chart (Pizza):** Mostrar a distribuição percentual de leads "Qualificados" (score >= 65) por utm\_content.
  + **daily-activity-chart (Linha):** Mostrar a evolução diária de Inscrições (created\_at) e Check-ins (check\_in\_at).

**Tarefa 3: Conectar o Dashboard de Acompanhamento de Vendas**

* **Filtros:** Adicionar um dropdown para selecionar um "Lançamento".
* **Tabela de Resumo (sales-summary-table):** Agrupar por utm\_content e utm\_medium e exibir as métricas.
* **Tabela Detalhada (sales-detail-table):** Exibir uma linha para cada combinação única de todas as 5 UTMs.
* **Gráfico (vendas-chart, Barras Empilhadas):** Eixo X para utm\_content, com barras representando o total de inscrições, e cada barra empilhada por utm\_medium.

**Como o Score é Calculado e Registrado?**

O cálculo e registro do score é um dos processos centrais do Passo 2. Ele acontece em 3 etapas:

conversacional

**Etapa 1: Preparação (Antes de ler o arquivo)**

* O script primeiro faz uma consulta na sua tabela perguntas no Supabase e carrega **todas as perguntas e suas opcoes (com os respectivos pesos)** para a memória, em um mapa de fácil acesso.

**Etapa 2: Cálculo (Durante a leitura de cada linha do CSV)**

* Para cada linha do seu arquivo de check-in, o script faz o seguinte: a. Inicia um contador de score para aquele lead: let currentScore = 0;. b. Ele percorre **cada coluna** daquela linha. c. Se o nome da coluna (ex: "qual a sua idade?") corresponde a uma pergunta que ele carregou do banco... d. ...ele pega a resposta daquela célula (ex: "35 a 44"). e. Procura essa resposta ("35 a 44") dentro das opcoes da pergunta e, se encontra, pega o peso associado (ex: 20 pontos). f. Soma esse peso ao contador: currentScore += 20;. g. Ele repete isso para todas as respostas daquele lead, acumulando o score.

**Etapa 3: Registro (Após processar um lote)**

* Após calcular o score final para um lead, o script armazena esse valor.
* Quando ele junta um lote de 200 leads processados, ele envia um único comando upsert para o Supabase. Esse comando encontra cada lead pelo email e **atualiza a coluna score** com o valor final que foi calculado.