

# Projeto CRM para Igreja

---

## Introdução

Este documento descreve o projeto de desenvolvimento de uma solução de CRM para a igreja. O objetivo é organizar o cadastro dos visitantes através das fichas coletadas e automatizar o envio de mensagens via WhatsApp, guiando os visitantes para formulários específicos de ministérios.

## Objetivos

1. Criar um sistema de cadastro para visitantes baseado nas fichas coletadas pela igreja.
2. Disparar automaticamente uma mensagem via WhatsApp para os visitantes cadastrados.
3. Direcionar os visitantes para formulários específicos de ministérios com base em suas respostas.
4. Armazenar e organizar as informações dos visitantes para acompanhamento futuro.
5. Fornecer um dashboard para líderes acompanharem o status dos visitantes e as respostas coletadas.

## Tecnologias Sugeridas

Frontend:

- React.js ou Next.js: Para uma interface moderna e responsiva.
- HTML/CSS/JavaScript: Alternativa mais simples para o protótipo.

Backend:

- Node.js com Express: Para criar a API que gerencia o envio de mensagens e conexão ao banco de dados.
- Python (Flask ou Django): Alternativa para quem prefere Python.

Integração com WhatsApp:

- Twilio API ou WhatsApp Business API: Para disparar mensagens pelo WhatsApp.

Banco de Dados:

- MongoDB: Para protótipo rápido e flexível.
- MySQL ou PostgreSQL: Para quem prefere um banco de dados relacional.

## Fluxo da Solução

1. Cadastro de Visitante: Um formulário simples (nome, telefone, e-mail, interesses) que salva os dados no banco de dados.
2. Disparo de Mensagem via WhatsApp: Após o cadastro, o sistema dispara uma mensagem

automática, guiando o visitante para um formulário específico.

3. Formulários de Ministério: O visitante recebe links para preencher formulários de interesse, como Google Forms ou formulários desenvolvidos na plataforma.

4. Dashboard de Acompanhamento: Um painel para os líderes visualizarem e acompanharem as respostas e o progresso de cada visitante.

## **Protótipo Básico**

1. Frontend: Crie um formulário simples em HTML para cadastro de visitantes.

2. Backend: Desenvolva uma API com Node.js para salvar os dados no banco de dados.

3. Integração com WhatsApp: Use a Twilio API para disparar mensagens automáticas após o cadastro.

4. Formulários Dinâmicos: Utilize Google Forms ou crie formulários dinâmicos para capturar os interesses dos visitantes.

## **Próximos Passos**

1. Desenvolver o protótipo com cadastro, banco de dados e integração com WhatsApp.

2. Testar a solução com visitantes e obter feedback.

3. Ajustar a solução com base no feedback e adicionar funcionalidades como dashboards e automações avançadas.