# Documento de Requisitos: Projeto Cruzamento 4.0

### 1. Visão Geral

O objetivo é desenvolver um MVP (Produto Mínimo Viável) de um semáforo inteligente que corrija falhas atuais, adapte-se a condições climáticas e de fluxo, e garanta segurança na comunicação de dados.

### 2. Requisitos Funcionais (RF)

*Descrevem o que o sistema deve fazer, suas funções e comportamentos.*

* **[RF001] Controle de Tempo por Fluxo:** O sistema deve monitorar o fluxo de veículos e aumentar o tempo do sinal verde automaticamente quando o fluxo for alto (horários de pico).
* **[RF002] Monitoramento Climático (Chuva):** O sistema deve ler dados de um sensor de umidade para identificar se está chovendo e a intensidade da chuva.
* **[RF003] Adaptação à Chuva:** Em caso de chuva forte, o algoritmo deve aumentar o tempo do sinal amarelo para permitir a redução de velocidade segura dos veículos.
* **[RF004] Modo de Segurança (Falha de Sensor):** Caso o sistema detecte falha na leitura dos sensores, o semáforo deve entrar imediatamente em modo de segurança, acionando o "pisca amarelo".
* **[RF005] Operação Local (Fallback):** O algoritmo deve ser capaz de rodar localmente e manter o funcionamento do semáforo mesmo se a conexão com o servidor ou broker cair.
* **[RF006] Armazenamento de Dados de Tráfego:** O sistema deve armazenar o histórico de fluxo de veículos utilizando vetores ou matrizes para processamento lógico.
* **[RF007] Simulação de Atuadores:** O sistema deve fornecer saídas visuais (simulação textual ou gráfica) para representar o estado dos LEDs (Verde, Amarelo, Vermelho).
* **[RF008] Comunicação de Dados:** O controlador deve enviar os dados coletados pelos sensores para o servidor local ou broker.

### 2.1. Requisitos Não Funcionais (RNF)

*Descrevem como o sistema deve ser, suas restrições de qualidade, técnicas e de desempenho.*

* **[RNF001] Protocolo de Comunicação:** A comunicação entre os sensores/controlador e o servidor deve utilizar o protocolo MQTT ou TCP/IP.
* **[RNF002] Hardware do Controlador:** O sistema embarcado deve ser projetado para rodar em arquitetura compatível com ESP32 ou Arduino.
* **[RNF003] Qualidade de Código:** O código-fonte do algoritmo deve seguir estritamente as boas práticas de *Clean Code* (código limpo).
* **[RNF004] Segurança de Rede (Firewall):** O servidor deve possuir firewall configurado permitindo tráfego apenas nas portas estritamente necessárias (ex: porta 1883 para MQTT).
* **[RNF005] Controle de Acesso:** O Sistema Operacional do servidor deve ter usuários criados com permissões limitadas e protegidos por senha, conforme política de segurança.
* **[RNF006] Versionamento:** Todo o código e documentação do projeto devem estar versionados em um repositório Git/GitHub com mensagens de commit claras.

### 3. Regras de Negócio e Condições Especiais

Lógica específica derivada do cenário proposto:

1. **Prioridade de Segurança:** A segurança física (evitar acidentes) tem prioridade sobre o fluxo. Se chover, a segurança aumenta (mais tempo de amarelo) em detrimento da velocidade.
2. **Resiliência:** O sistema não pode parar completamente. Se um componente crítico (servidor ou sensor) falhar, ele degrada para um estado seguro (modo local ou pisca amarelo) em vez de travar no vermelho ou verde.
3. **Horários de Pico:** O sistema deve considerar horários críticos (ex: 07:00, 12:00, 18:00) como gatilhos para verificação de fluxo intenso.