# Captura e Identificação de Veículos

* **História de Usuário**: Como usuário, quero que o sistema seja capaz de capturar o fluxo veículos que passam por um ponto específico para que eu possa monitorar o fluxo de tráfego em tempo real.
* **Tarefas**:
  + Configurar a câmera com o Arduino para capturar imagens de uma via pública.

o Desenvolver um script em Python usando OpenCV para processar as imagens e identificar os veículos.

* **Critérios de Aceitação**:
  + A identificação de veículos deve ser precisa, com uma margem de erro mínima.  
    o O sistema não deve armazenar dados pessoais dos veículos, apenas a contagem total.

# Registro de Fluxo em Diferentes Horários

* **História de Usuário**: Como usuário, preciso que o sistema registre o fluxo de veículos em horários variados do dia.
* **Tarefas**:
  + Implementar o armazenamento dos dados no MongoDB.  
    o Assegurar que o sistema registre os dados continuamente sem interrupções e repetição de dados.
* **Critérios de Aceitação**:
  + Os dados de contagem devem ser armazenados com o horário exato. o Os dados agregados devem ser facilmente recuperáveis para futuras análises.

# Exibição Intuitiva de Dados

* **História de Usuário**: Como usuário, gostaria que o sistema exiba gráficos de fácil interpretação, e que, eu consiga filtrar dados dentro desses gráficos
* **Tarefas**:
  + Criação da página e de gráficos dinâmicos usando python
  + Desenvolver filtros para seleção de dados dentro do gráfico
* **Critérios de Aceitação**:
  + A página deve conter filtros de datas e horários.
  + A página não deve conter dúvidas ou ambiguidade sobre valores mostrados em gráficos.

# Análise de Dados e Histórico

* **História de Usuário**: Como usuário, quero acessar históricos que mostram o fluxo de veículos em períodos específicos para entender os padrões de tráfego e tomar decisões de planejamento urbano.
* **Tarefas**:
  + Conseguir puxar dados históricos no sistema

o Implementar filtros para permitir a análise em diferentes datas

* **Critérios de Aceitação**:
  + Exibir gráfico com dados histórico especificados pelo usuário