



# REAL-TIME OBJECT DETECTION SYSTEM

**PRESENTED TO**  
Mrs Nitila Adnaik

**PRESENTED BY**  
Sentinle Squad

# ❖ Team Members



Bhakti Ayarekar

01

Ankita Yadav

10

Rohan Chopade

21

Soham Mangore

29

[Back to Agenda](#)



# ◆ Agenda

[Back to Agenda](#)



1

Introduction

2

Problem Definition

3

Objective

4

Methodology.

5

Software Requirement  
Specification

6

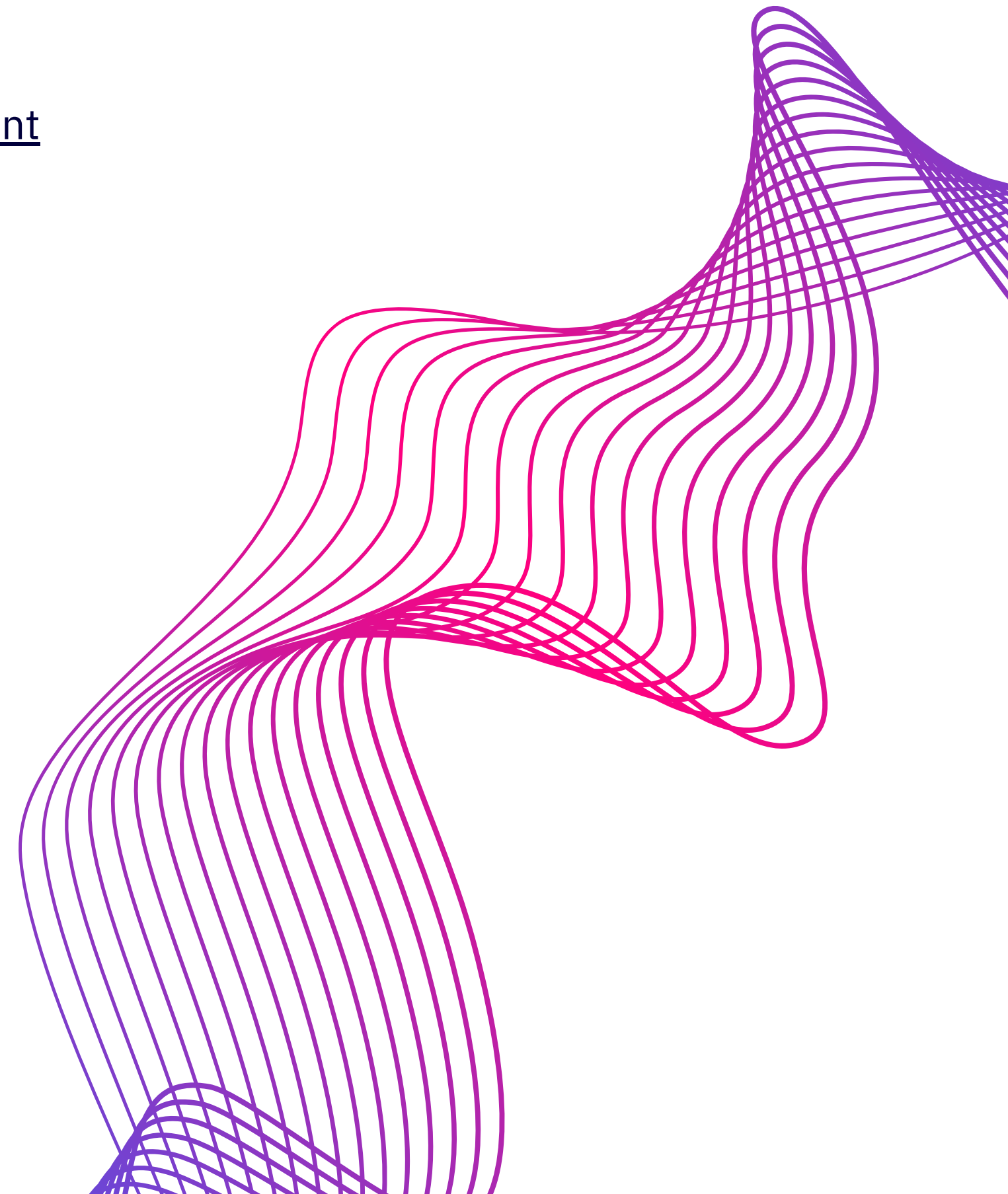
Flowchart

7

Advantages

8

Applications

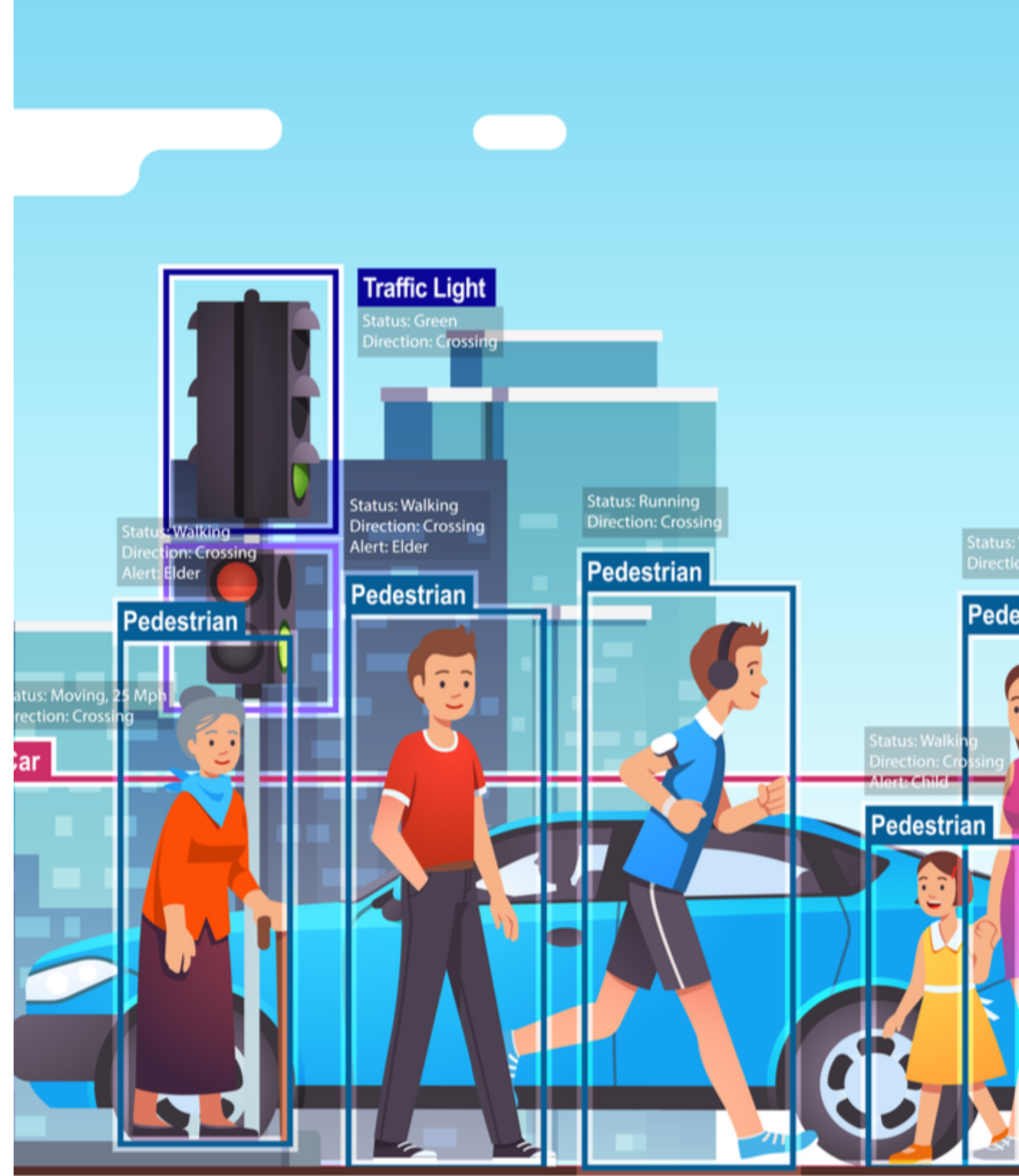




# ❖ Introduction

The security of our homes is of utmost importance in today's world. Traditional home security systems often rely on static cameras and manual monitoring, which can be time-consuming and prone to human error. To address these limitations, we have developed a real-time video-based home security system that leverages the power of object detection and socket programming.

[Back to Agenda](#)



# ❖ Problem Statement

To create an intelligent home security system that can actively monitor and analyze video feeds from multiple cameras in real-time. By integrating object detection algorithms into the system, we aim to detect and classify various objects of interest, such as humans, vehicles, and potential intruders.



[Back to Agenda](#)



# ❖ Objectives

[Back to Agenda](#)



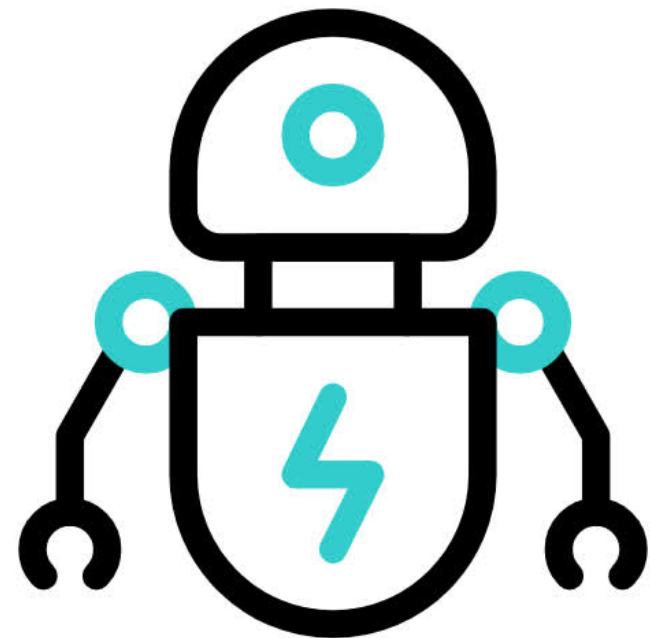
- Implementing real-time video streaming using socket programming to enable clients (home security cameras) to transmit video feeds to the server.
- Developing an object detection model capable of accurately detecting and classifying objects of interest in the video frames received from clients.
- Enabling intrusion detection by analyzing the video frames for potential unauthorized access or suspicious activities.





# ❖ Methodology

[Back to Agenda](#)

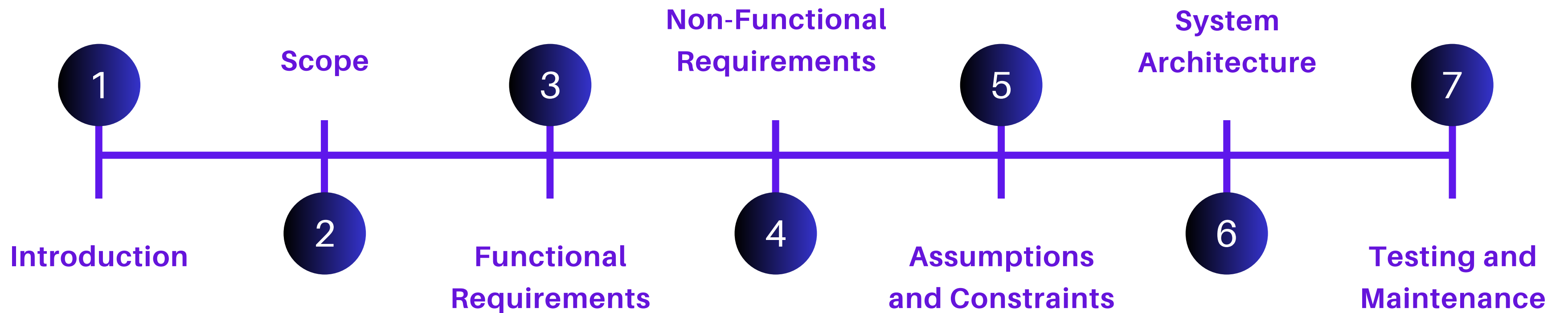


Socket programming in Python enables the communication between clients and the server, while OpenCV is utilized for video frame fetching and processing. Object detection is done by using a pre-trained model, but we can still use ML libraries like TensorFlow or Pytorch for better performance.



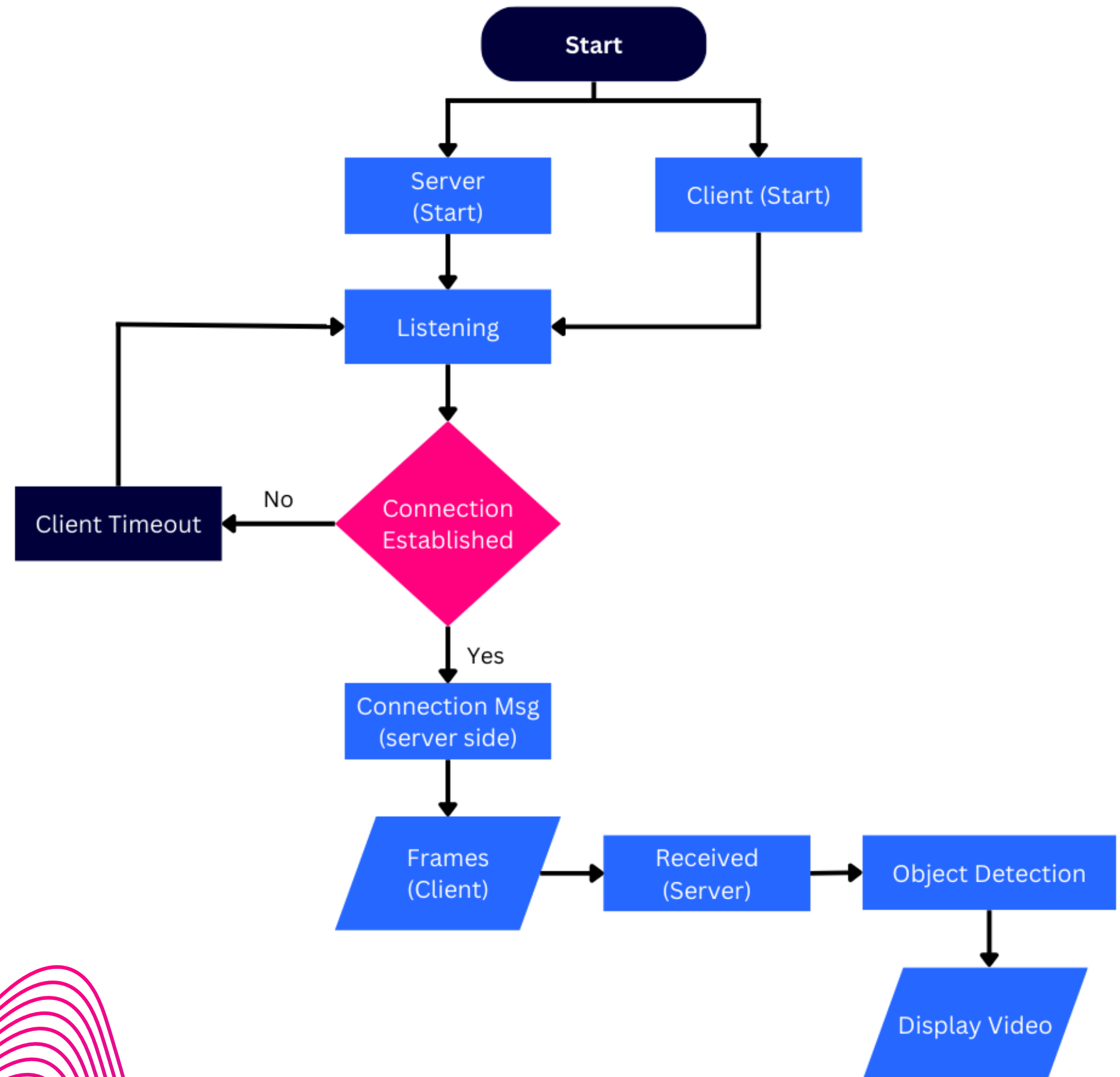
# ❖ Software Requirements Specification

[Back to Agenda](#)





# Flowchart



[Back to Agenda](#)

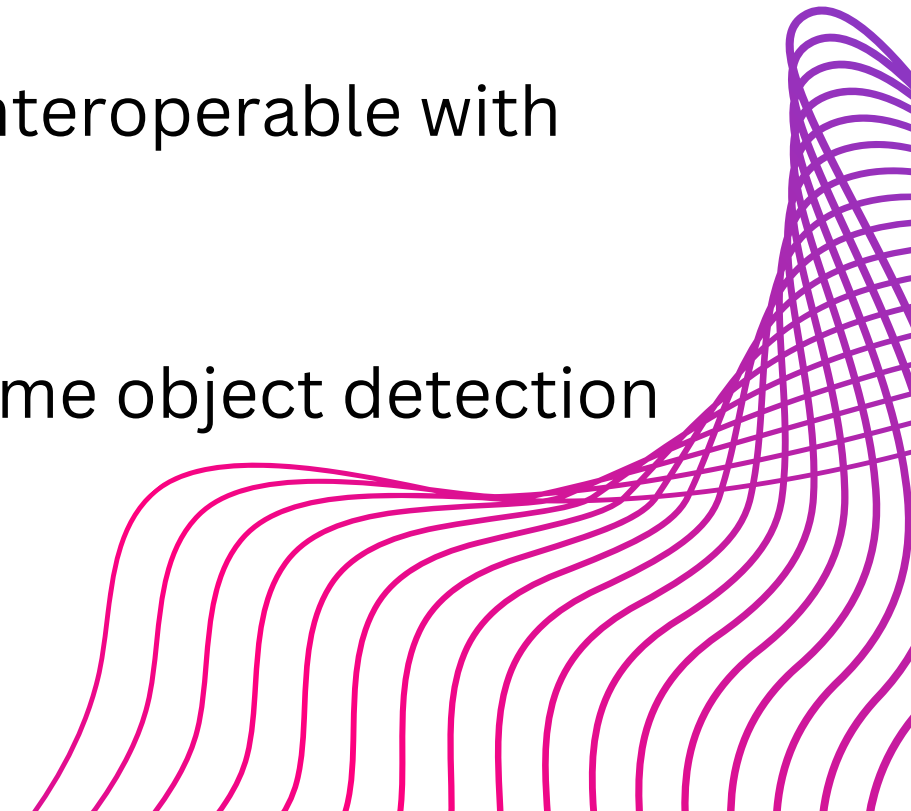


# ❖ Advantages

[Back to Agenda](#)



- **Real-time Updates:** Socket programming allows real-time updates and notifications about detected objects, ensuring prompt responses to changing situations.
- **Scalability and Flexibility:** Socket programming facilitates communication between multiple devices or systems, making it scalable and flexible.
- **Remote Monitoring and Control:** Socket programming enables remote monitoring and control of object detection systems. Users can access the system from different locations or devices, allowing for remote surveillance, management, and decision-making.
- **Compatibility and Interoperability:** Socket programming it is compatible and interoperable with different platforms, programming languages, and operating systems.
- **Enhanced User Experience:** Socket programming enables the delivery of real-time object detection updates and notifications to end-users, enhancing their overall experience.



# ❖ Applications

[Back to Agenda](#)



- **Video Surveillance** : The system can analyze live video feeds from cameras and detect objects of interest, such as people, vehicles, or specific objects.
- **Industrial Safety**: The system can monitor the environment for potential hazards or anomalies, such as unauthorized personnel, misplaced objects, or equipment malfunctions.
- **Traffic Management**: Real-time object detection can be applied to traffic management systems to monitor and optimize traffic flow.
- **Smart Home Applications**: Detect objects like people, pets, or specific items within the home, the system can trigger actions such as sending alerts to homeowners about potential security breaches or unusual activities.
- **Fire and Smoke Detection**: By analyzing the video feed, the system can detect signs of fire or smoke within the monitored area.



*Thank You*

[Back to Agenda](#)

