INTELLIGENT PEOPLE AND VEHICLE COUNTING SYSTEM FOR SECRETERIAT

A Project Report

Submitted by

D. AKALYASRI	711620106001
G. SANDRA GRACE	711620106323
M. DHIVYADHARSHINI	711620104007
N. ANUSURYA	711620104001

TEAM ID : NM2023TMID15643

BATCHELOR OF DEGREE

IN

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

&

COMPUTER SCIENCE AND ENGINEERING

KATHIR COLLEGE OF ENGINEERING

INDEX

1. INTRODUCTION

1.1 PROJECT OVERVIEW:

The Intelligent People and Vehicle Counting System for Secretariat in IoT is a project aimed at developing an advanced surveillance and monitoring system using Internet of Things (IoT) technologies. The system utilizes IoT devices, computer vision algorithms, and data analytics to count and track the number of people and vehicles entering and exiting a secretariat facility. The project aims to enhance security, optimize resource allocation, and provide real-time insights for efficient management within the secretariat.

OBJECTIVES:

Develop an IoT-based system capable of accurately detecting, counting, and tracking people and vehicles in real-time. Integrate IoT devices, such as cameras and sensors, to capture data from various locations within the secretariat. Implement computer vision algorithms to analyse the captured data and provide accurate counting information. Utilize data analytics to generate valuable insights, trends, and statistical data related to people and vehicle movement within the secretariat. Enhance security measures by identifying unauthorized access, monitoring suspicious activities, and generating alerts in real-time. Provide a user-friendly interface for monitoring, reporting, and managing the counting system using IoT-enabled devices.

KEY FEATURES:

a. People Counting:

Utilize computer vision techniques to detect and track individuals entering and exiting the secretariat.

Accurately count the number of people in real-time, considering multiple entry and exit points.

Handle complex scenarios, such as crowded areas, overlapping individuals, and occlusions.

b. Vehicle Counting:

Deploy computer vision algorithms to identify and track vehicles entering and leaving the secretariat premises.

Provide accurate vehicle counting statistics, including vehicle type classification (e.g., cars, bikes, trucks).

Handle varying traffic conditions and adapt to different lighting and weather conditions.

c. Real-time Monitoring and Alerts:

Set up a live monitoring interface displaying real-time counts and statistical data.Generate alerts and notifications for anomalies, such as sudden increases in vehicle or people counts.Integrate with existing security systems for immediate response to potential threats.

d. Data Analytics and Reporting:

Perform data analysis to generate insights and trends related to people and vehicle movements.

Generate comprehensive reports and visualizations to aid decision-making and resource allocation.

Provide historical data for future planning, optimization, and resource management.

IMPLEMENTATION PLAN:

a**. SYSTEM DESIGN AND ARCHITECTURE:**

Define the system requirements and design the architecture for the counting system.

Identify the appropriate hardware infrastructure and camera placements.

Select suitable computer vision algorithms and techniques for accurate counting.

B. DATA COLLECTION AND ANNOTATION:

Collect a diverse dataset of video footage capturing various scenarios within the secretariat.

Annotate the dataset to label people and vehicle instances for training and evaluation purposes.

C. MODEL DEVELOPMENT AND TRAINING:

Develop and train computer vision models for people and vehicle detection and tracking.

Optimize the models for real-time performance and accuracy.

Perform rigorous testing and validation to ensure reliability and robustness.

D. SYSTEM INTEGRATION AND DEPLOYMENT:

Integrate the developed models into a unified system with live camera feeds.

Deploy the system on the selected hardware infrastructure within the secretariat.

Configure network connectivity, storage, and backup mechanisms for seamless operation.

E. USER INTERFACE AND REPORTING:

Develop an intuitive user interface for real-time monitoring and system management.

Implement reporting features to generate comprehensive statistical reports and visualizations.

PROJECT DELIVERABLES:

Fully functional Intelligent People and Vehicle Counting System for Secretariat.

Detailed documentation covering system architecture, installation, configuration, and usage instructions.

Trained computer vision models and dataset for further research or system enhancement.

Comprehensive reports on people and vehicle counts, analytics, and insights.

1.2 PROJECT PURPOSE:

The project aims to improve the security of the secretariat facility by accurately detecting and tracking people and vehicles entering and exiting the premises.

By integrating IoT devices and computer vision algorithms, the system can identify unauthorized access and monitor suspicious activities in real-time.

This helps in preventing security breaches and ensuring the safety of the secretariat.

The intelligent counting system provides valuable insights and statistical data regarding people and vehicle movements within the secretariat.

By analysing this data, decision-makers can optimize resource allocation based on the traffic patterns, peak hours, and occupancy levels.

This helps in streamlining operations, managing staffing requirements, and improving overall efficiency within the secretariat.

The project aims to provide a user-friendly interface accessible through IoT-enabled devices, allowing real-time monitoring, reporting, and management of the counting system.

Authorized personnel can access the system remotely, view live counts, generate reports, and receive alerts for any anomalies or security breaches.

This enables efficient monitoring and swift decision-making to ensure the smooth functioning of the secretariat.

By leveraging data analytics, the project generates valuable insights and trends related to people and vehicle movement.

These insights can be used for strategic planning, future resource allocation, and optimizing the secretariat's operations.

The system provides comprehensive reports and visualizations that aid decision-makers in understanding the patterns, identifying areas for improvement, and making data-driven decisions.

2. IDEATION AND PROPOSED SOLUTION

2.1 Problem Statement:

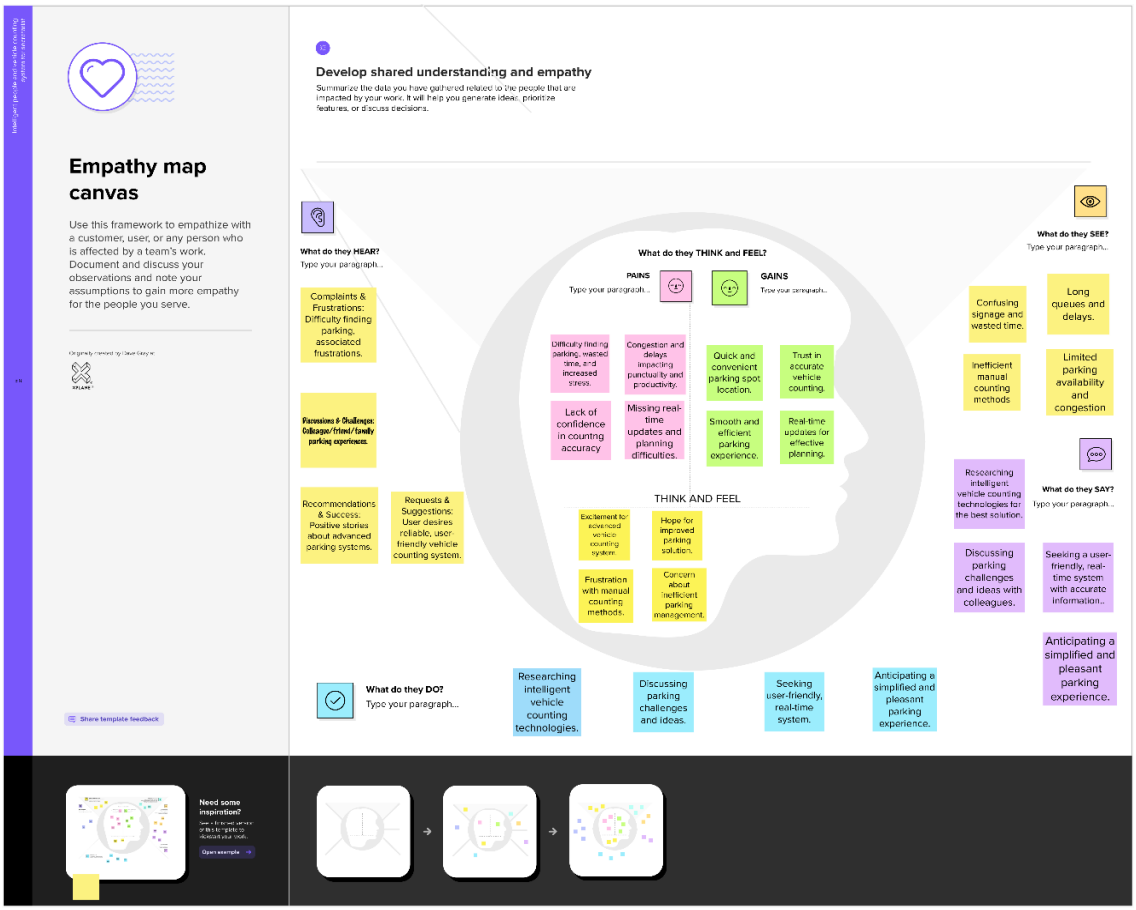
Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love. A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

Problem Statement (PS	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Secretariat security personnel	enhance the security and surveillance of the premises	the current manual counting system is not efficient and reliable in tracking the number of people and vehicles entering and exiting the Secretariat	It's challenging to monitor and analyze the data accurately, which makes it difficult to ensure the safety and security of the premises.	Anxious
PS-2	a Secretariat traffic management personnel	improve the traffic flow and reduce congestion on the premises.	the current manual counting system is not efficient and reliable in tracking the number of people and vehicles entering and exiting the Secretariat	It's challenging to manage the traffic flow and reduce congestion, which makes it difficult to ensure the smooth and efficient movement of vehicles.	frustrated

2.2 EMPATHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

The motive behind implementing an intelligent people and vehicle counting system for the secretariat is to enhance efficiency, accuracy, and user experience in parking management. By leveraging innovative technologies and data analytics the system aims to streamline the parking process, optimize space utilization, and provide real-time information to users. The goal is to alleviate parking challenges, reduce congestion, create a more sustainable and user- centric



parking environment. Ultimately, the motive is to improve productivity, user satisfaction, and overall operational effectiveness at the secretariat.

2.3 IDEATION & BRAINSTORMING

Brainstorm & Idea Prioritization:

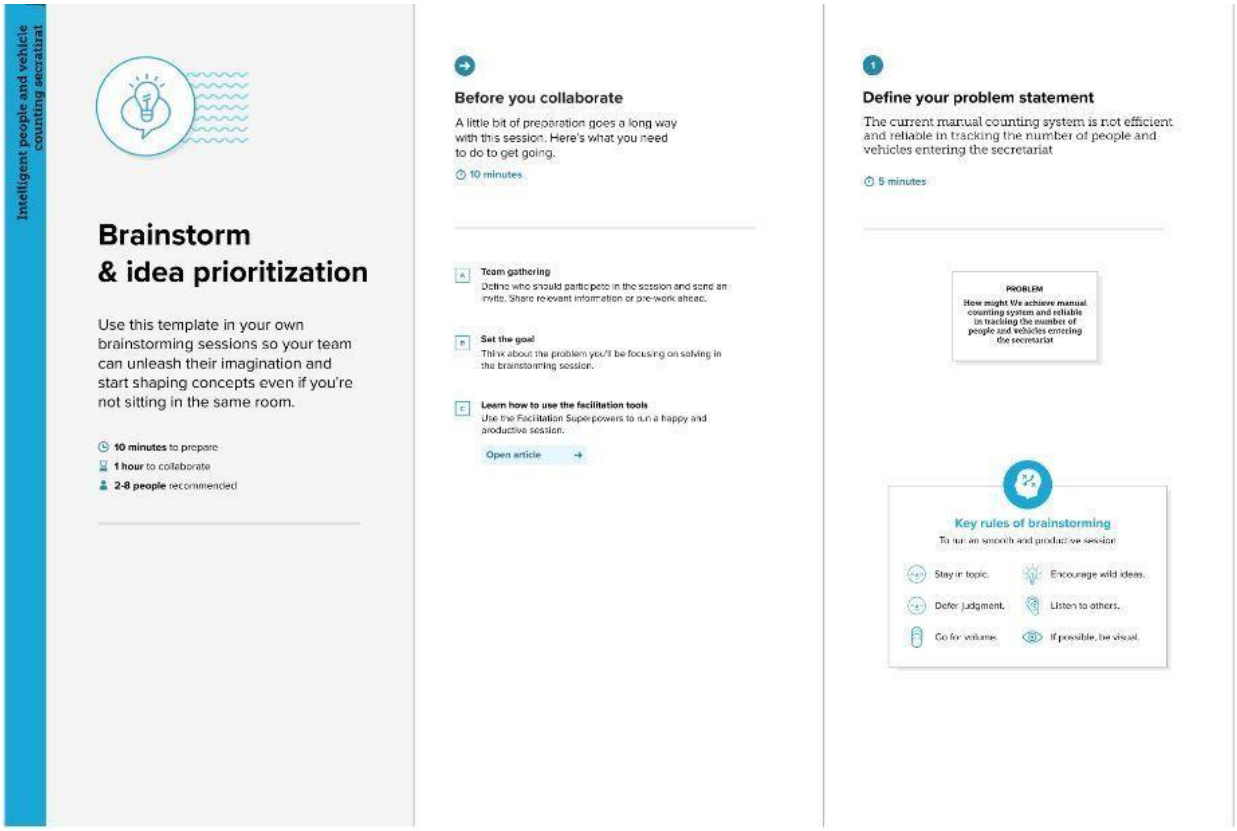
Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-box ideas are welcome and built upon, and all participants are encouraged to collaborate , helping each other develop a rich amount of creative solutions.

Step-1: Team Gathering, Collaboration and Select the problem Statement.

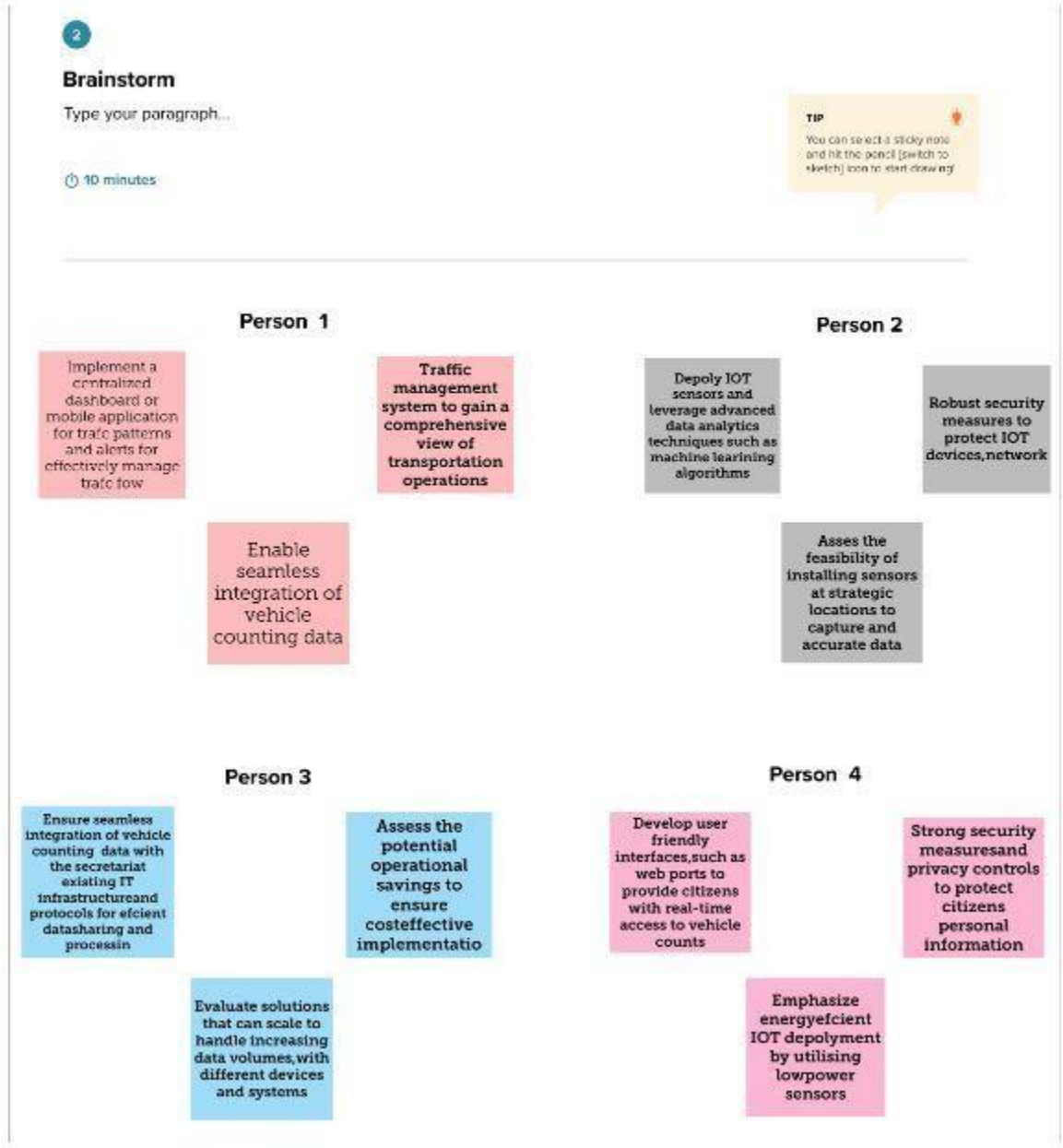
Step-2: Brainstorm

Step-3: Idea Prioritization

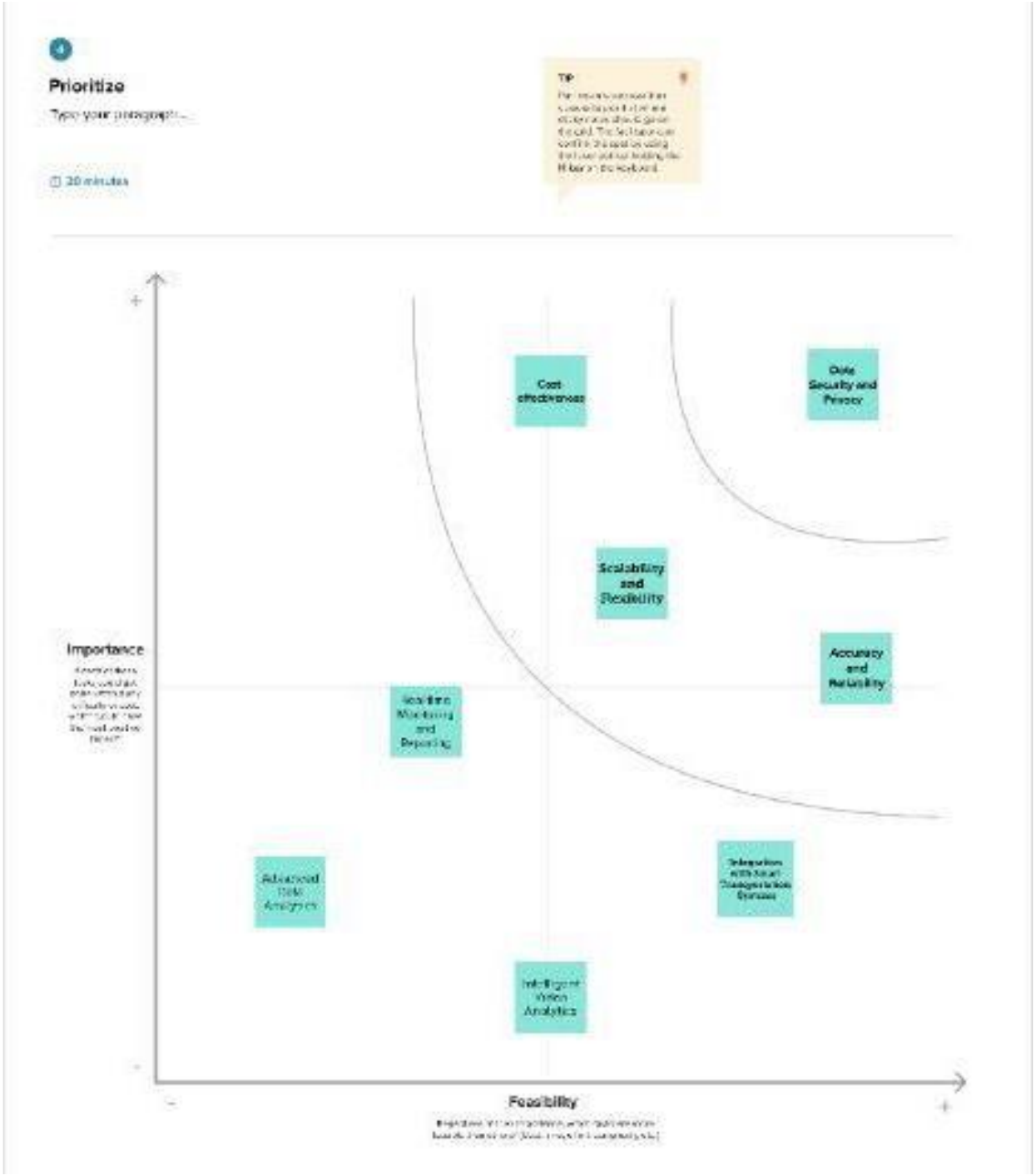
Step-1:



STEP 2:


2
Brainstorm
Type your paragraph...
10 minutes
TIP
Person 1
Implement a centralized dashboard or mobile application for traffic patterns and alerts for effectively manage traffic flow
Traffic management system to gain a comprehensive view of transportation operations
Enable seamless integration of vehicle counting data
Person 2
Depoly IOT sensors and leverage advanced data analytics techniques such as machine learining algorithms
Robust security measures to protect IOT devices,network
Asses the feasibility of installing sensors at strategic locations to capture and accurate data
Person 3
Ensure seamless integration of vehicle counting data with the secretariat existing IT infrastructureand protocols for efcient datasharing and processin
Assess the potential operational savings to ensure costeffective implementatio
Evaluate solutions that can scale to handle increasing data volumes,with different devices and systems
Person 4
Develop user friendly interfaces,such as web ports to provide citizens with real-time access to vehicle counts
Strong security measuresand privacy controls to protect citizens personal information
Emphasize energyefcient IOT depolyment by utilising lowpower sensors

STEP 3:


Prioritize
Cost effectiveness
Data Security and Privacy
Scalability and Flexibility
Accuracy and Reliability
Importance
Feasibility

2.4PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The current manual counting system is not efficient and reliable in tracking the number of people and vehicles entering and exiting the Secretariat. It's challenging to monitor and analyze the data accurately, which makes it difficult to ensure the safety and security of the premises.
2.	Idea / Solution description	An intelligent people and vehicle counting system should be implemented that can accurately count and track the number of people and vehicles entering and exiting the Secretariat. The systemcan use a combination of sensors, cameras, and machine learning algorithms to detect and differentiate between people and vehicles.The system can also generate real-time data that can be analyzed tomonitor the movement of people and vehicles, identify any suspicious behavior or activity and take appropriate securitymeasures.
3.	Novelty / Uniqueness	This system can efficiently and accurately monitor and analyze the movement of people and vehicles within the Secretariat. This systemcan provide valuable data that can be used to optimize traffic flow, improve security measures, and enhance the overall visitor experience.
4.	Social Impact / Customer Satisfaction	Security personnel can identify and respond to potential security threats quickly and efficiently. This can help ensure the safety and security of visitors and staff, which is essential in public spaces. Thedata generated by the intelligent counting system can be used to optimize traffic flow within the Secretariat. This system can create employment opportunities for people with technical skills in the installation, operation, and maintenance of the system.
5.	Business Model (Revenue Model)	The business model can be customized based on the needs of the Secretariat and the goals of the company. The company can offer

3. REQUIREMENT ANALYSIS

CUSTOMER JOUNEY MAP:

3.1 FUNCTIONAL REQUIREMENTS:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**Counting Accuracy**	1) Accurately count the number of people entering andexiting the secretariat premises. 2)Accurately count the number of vehicles entering andexiting the secretariat premises. 3)Maintain counting accuracy within an acceptable margin oferror.
FR-2	**Real-time Counting**	1) Provide real-time updates of the current count of people and vehicles within the secretariat premises. 2) Make real-time updates accessible to authorizedpersonnel through a user-friendly interface.

FR-3	**Multiple Entry/Exit Points**	1) Support counting people and vehicles at multiple entry and exit points of the secretariat premises. 2) Handle simultaneous counting at different points withoutcompromising accuracy.
FR-4	**Intelligent Detection**	1) Employ intelligent detection techniques, such as computer vision or sensor-based technologies, to identify and track individuals and vehicles. 2) Differentiate accurately between people and vehicles.
FR-5	**Integration with Access ControlSystems**	1) Integrate with existing access control systems, such as IDcard readers or vehicle identification systems 2) Associate counted individuals and vehicles with theirrespective credentials. 3) Synchronize entry and exit data with the counting system.
FR-6	**Data Logging and Reporting**	1) Maintain a log of all entry and exit events, includingtimestamps, for audit and reporting purposes. 2) Generate periodic reports summarizing the total count ofpeople and vehicles over specific time intervals.
FR-7	**Scalability and Flexibility**	1) Scale the system to accommodate future expansion or changes in the secretariat premises, including additionalentry/exit points. 2) Adapt to different counting requirements based on specificevents or time periods.
FR-8	**Security and Privacy**	1) Ensure the security and privacy of collected data, adheringto relevant data protection regulations. 2) Restrict access to the counting system and its data to authorized personnel using authentication and authorizationmechanisms.
FR-9	**Maintenance and Support**	1) Provide ease of maintenance, including regularmaintenance, updates, and bug fixes. 2) Offer timely technical support and assistance to addressany issues or concerns.
FR-10	**Cost-effectiveness**	1) Be cost-effective, considering both initial implementationcosts and long-term operational expenses. 2) Provide a reasonable return on investment by improvingsecurity, efficiency, and resource allocation within the secretariat premises.

3.2 NON-FUNCTIONAL REQUIREMENTS:

Following are the non-functional requirements of the proposed solution.

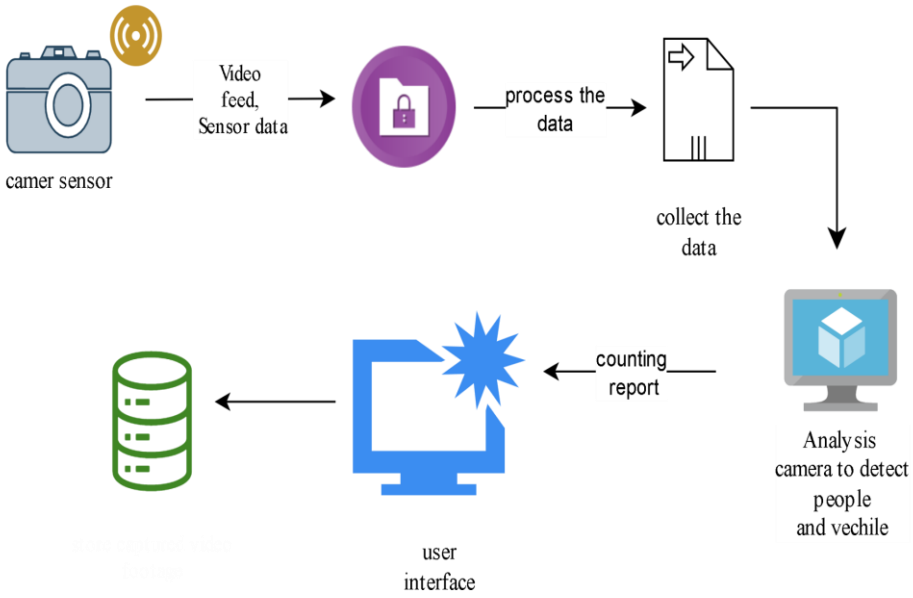
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	1) Provide a user-friendly interface that is easy to understand and navigate for authorized personnelaccessing real-time updates and reports. 2)Ensure system controls and interactions are intuitive andwell-designed.
NFR-2	**Security**	1)Implement robust security measures to prevent unauthorized access to the counting system and protectthe integrity and confidentiality of collected data.
NFR-3	**Reliability**	1)Ensure the system operates reliably without frequent failures or disruptions in counting functionality.
NFR-4	**Performance**	1)Ensure the system can handle high traffic and accuratelycount people and vehicles in real-time without significant delays or performance degradation.
NFR-5	**Availability**	1)Ensure the system is always available and accessible to authorized personnel, with minimal downtimeor scheduled maintenance windows.
NFR-6	**Scalability**	1)Design the system to handle increasing numbers of people and vehicles as the secretariat premises and trafficgrow over time.

4 PROJECT DESIGN

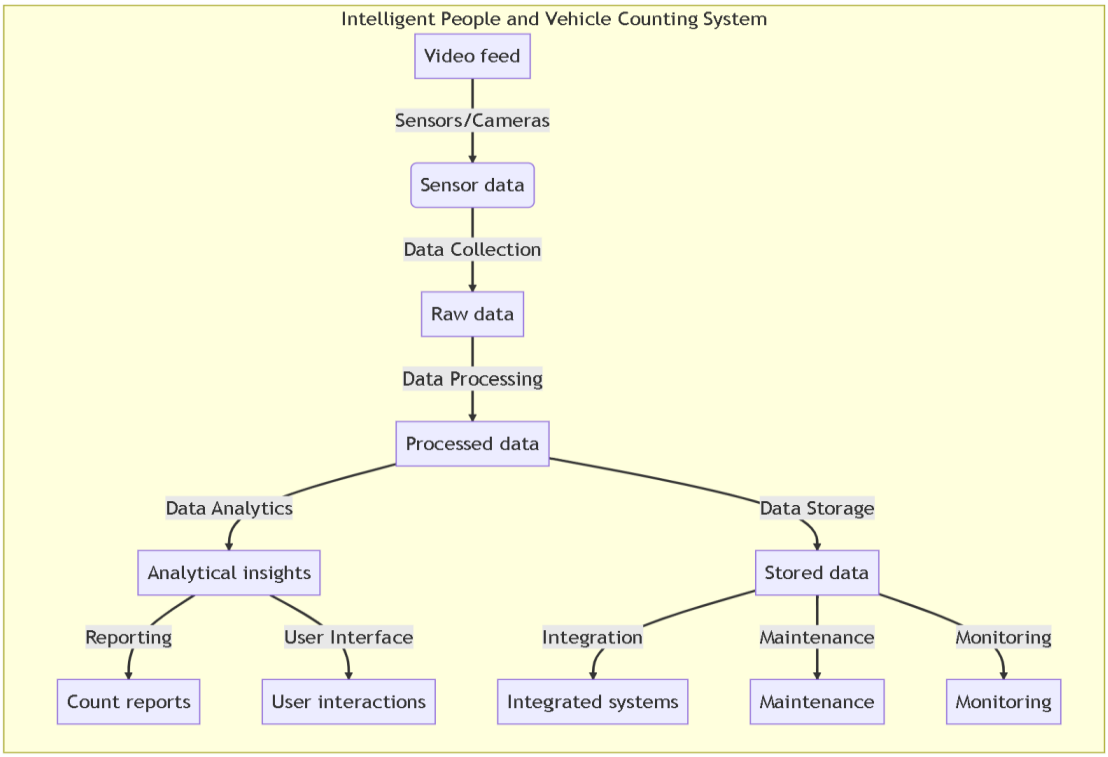
4.1 Data Flow Diagrams:

Data Flow Diagrams: A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Example:



Flowchart:

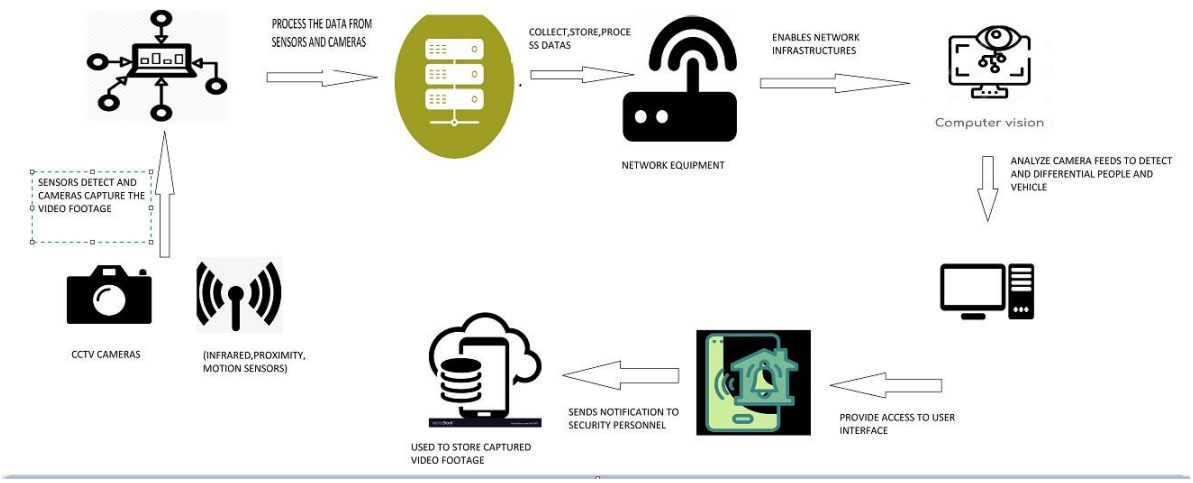


4.2 SOLUTION ARCHITECTURE:

Solution architecture is a complex process – with many sub-processes – that bridgesthe gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behaviour, and other aspects of thesoftware to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, anddelivered.

Solution Architecture Diagram:



TECHNICAL ARCHITECTURE:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2.

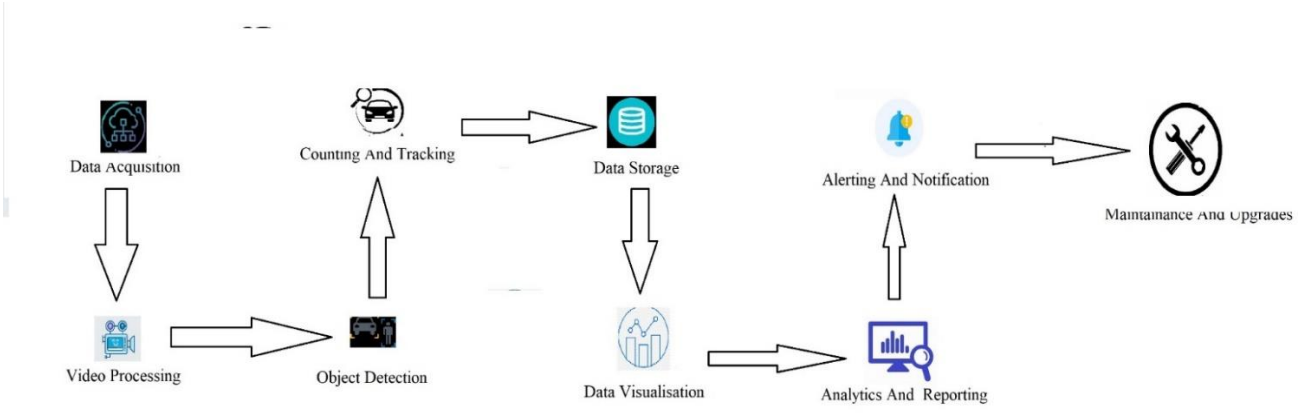


Table-1 : Components & Technologies

s.no	Component	Description	Technology
1.	User Interface	How user interacts with application e.g.,Web UI, Mobile App, Chatbot etc.	Web UI using React.js
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant API
5.	Database	Data Type, Configurations etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM Cloudant
7.	File Storage	File storage requirements	Local Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API
9.	External API-2	Purpose of External API used in the application	Aadhar API
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / CloudLocal Server Configuration: Cloud Server Configuration:	Local Server Configuration: Apache Tomcat Cloud Server Configuration: AWS (Amazon Web Services)

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	React.js
2.	Security Implementations	List all the security / access controls implemented,use of firewalls etc.	SHA-256 encryption, user authentication, role-based access control (RBAC), OWASP security practices.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Microservices architecture using Docker and Kubernetes for containerization and horizontal scaling.
4.	Availability	Justify the availability of application (e.g., use ofload balancers, distributed servers etc.)	Load balancers, distributed servers, and redundant infrastructure.
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Caching mechanisms, Content Delivery Network (CDN) for static assets, load testing, and query optimization.

4.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team
Security Personnel	Real-time People Counting	USN-1	As a security personnel, I want real-time people counting at entry/exit points so that I can monitor and control access.	People count is updated in real-time at entry/exit points.	High	Akalyasri D

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team
Facility Manager	Parking Occupancy Tracking	USN-2	As a facility manager, I want to track parking occupancy levels to efficiently manage parking space availability.	Real-time parking occupancy information is displayed.	High	Dhivyadharshini M
Management Staff	Comprehensive Reports	USN-3	As a management staff member, I want comprehensive reports on people and vehicle counts for different areas	Reports include historical and real-time data.	Medium	Anusurya N
Maintenance Team	Sensor/Camera Notifications	USN-4	As a maintenance team member, I want to receive notifications of sensor or camera malfunctions for timely troubleshooting.	Automated notifications are sent for sensor/camera malfunctions.	Medium	Sandra Grace G
User	Integration with Security Infrastructure	USN-5	As a user, I want the counting system to integrate with the existing security infrastructure for seamless access control.	Integration enables authentication and authorization processes.	Medium	Akalyasri D

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team
Administrator	User-Friendly Interface	USN-6	As an administrator, I want a user-friendly interface to configure counting zones, adjust sensitivity settings, and customize reporting parameters.	Interface allows easy configuration and customization.	High	Anusurya N
Data Analyst	Clean and Structured Data	USN-7	As a data analyst, I want clean and structured data on people and vehicle counts for in-depth analysis and insights.	Data is clean, well-structured, and suitable for analysis.	High	Sandra Grace G
Visitor	Signage and Wayfinding Assistance	USN-8	As a visitor, I want clear signage and wayfinding assistance based on the counting system data.	Signage and assistance are based on real-time counting data.	Medium	Dhivyadharshini M
Security Manager	Data Privacy Compliance	USN-9	As a security manager, I want the counting system to ensure compliance with data privacy regulations.	System follows data privacy regulations for captured data.	High	Akalyasri D

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team
System Administrator	Scalability	USN-10	As a system administrator, I want the counting system to be scalable for future expansion and growth.	System can easily scale to accommodate additional areas.	Medium	Dhivyadharshini M

5. CODING AND SOLUTIONING:

5.1 FEATURE 1:

Display a live count of people on the screen:

Add a text overlay on the frame to display the current count of people.

Update the count whenever a person crosses the designated lines.

```
# before the cv2.imshow('Frame', frame) line
count_text = "Count: {}".format(cnt_up + cnt_down)
cv2.putText(frame, count_text, (10, 140), font, 0.5, (255, 255, 255), 2, cv2.LINE_AA)
cv2.putText(frame, count_text, (10, 140), font, 0.5, (0, 0, 255), 1, cv2.LINE_AA)
```

5.2 FEATURE 2:

Save a video with annotated results:

Create a VideoWriter object to save the processed frames with annotations to a video file.

Write each annotated frame to the video file.

```
# before the main loop
output_video = cv2.VideoWriter('output.mp4', cv2.VideoWriter_fourcc(*'mp4v'), 25, (int(w), int(h)))
# inside the main loop, after annotating the frame
output_video.write(frame)
# Add this line after the main loop to release the video writer
output_video.release()
```

6. RESULT:

6.1 PERFORMANCE METRICS:

METRICS:

For the Intelligent People and Vehicle Counting System for the secretariat, several application performance metrics can be measured and monitored to ensure optimal system functionality. Here are some key performance metrics to consider:

Counting Accuracy:

- Measure the accuracy of people and vehicle counting.
- Calculate the percentage of correctly identified individuals and vehicles.
- Monitor any discrepancies or errors in the counting process.

Real-time Updates:

- Measure the time it takes to update and display real-time counting data.
- Monitor the latency between data capture and availability in the user interface.
- Ensure timely updates to provide up-to-date information to authorized personnel.

Response Time:

- Measure the time taken by the system to respond to user requests or queries.
- Monitor the overall system responsiveness during peak usage periods.
- Optimize response time to provide a smooth and efficient user experience.

Scalability:

- Monitor system performance under increasing traffic or load.
- Measure the system's ability to handle a growing number of people and vehicles.
- Evaluate the scalability of the infrastructure and adjust resources as needed.

Availability:

- Measure the system's uptime and availability.
- Monitor any downtime or service interruptions.
- Aim for a high availability percentage to ensure continuous operation.

Resource Utilization:

- Monitor the utilization of system resources such as CPU, memory, and disk space.
- Identify any bottlenecks or areas where resource consumption can be optimized.
- Ensure efficient resource allocation to maintain system performance.

Error and Exception Handling:

- Monitor the occurrence of errors or exceptions within the system.
- Measure the frequency and severity of errors and exceptions.
- Implement proper error handling and logging mechanisms to track and resolve issues promptly.

Data Processing Speed:

- Measure the time taken to process and analyse captured data.
- Monitor the efficiency of data pre-processing, object detection, and tracking algorithms.
- Optimize data processing to minimize delays and enable real-time analysis.

Security and Compliance:

- Monitor the effectiveness of security measures and access control mechanisms.
- Track any unauthorized access attempts or security breaches.
- Ensure compliance with data protection regulations and industry standards.

User Satisfaction:

- Gather user feedback and satisfaction ratings.
- Conduct surveys or interviews to understand user experience and identify areas for improvement.
- Monitor user adoption and engagement with the system.

7 .ADVANTAGES:

Enhanced Efficiency:

An intelligent vehicle and people counting system can improve the overall efficiency of operations within a secretariat. It can automate tasks such as vehicle access control, parking management, and people counting, reducing the need for manual intervention and streamlining processes.

Accurate Data Collection:

Such a system can provide accurate and real-time data on vehicle and people movements within the secretariat. This data can be valuable for analyzing traffic patterns, optimizing resource allocation, and enhancing security measures.

Improved Security:

By implementing an intelligent vehicle and people counting system, security measures can be strengthened. The system can detect unauthorized access attempts, monitor suspicious activities, and generate alerts or notifications in case of security breaches.

Traffic Management:

The system can assist in managing traffic flow within the secretariat premises. By analyzing the data collected, it can identify congestion areas, optimize traffic routes, and facilitate smooth movement of vehicles, reducing traffic jams and delays.

DISADVANTAGES:

Cost:

Implementing an intelligent vehicle and people counting system requires an initial investment in infrastructure, hardware, and software. The cost may be a significant factor for organizations with limited budgets.

Technical Challenges:

Setting up and maintaining such a system can pose technical challenges. Integration with existing infrastructure, ensuring data accuracy, and addressing issues such as false readings or system failures may require technical expertise and ongoing support.

Privacy Concerns:

Collecting data on vehicle and people movements within a secretariat raises privacy concerns. Measures must be in place to ensure the collected data is stored securely and used only for authorized purposes. Transparency and clear privacy policies are essential to address these concerns.

Dependency on Technology:

Relying on an intelligent vehicle and people counting system means the operations of the secretariat become dependent on the technology. Any system failures or technical glitches could disrupt the normal functioning of the secretariat, requiring backup plans and contingencies.

8. CONCLUSION:

In conclusion, the implementation of an intelligent people and vehicle counting system for a secretariat offers several advantages and disadvantages. On the positive side, such a system can enhance operational efficiency, provide accurate data collection, improve security measures, and facilitate traffic management within the secretariat premises. However, there are also challenges to consider, including the initial cost of implementation, technical complexities, privacy concerns, and the dependency on technology.

Looking ahead, the future scope for intelligent people and vehicle counting systems in secretariats is promising. Continuous advancements in technology, data analysis, and integration with smart city infrastructure can further improve efficiency, security, and the overall user experience. Advanced sensor technologies, AI and ML algorithms, real-time data processing, and seamless integration with smart city initiatives are among the key areas of development.

By embracing these future possibilities, secretariats can benefit from enhanced accuracy in counting, proactive decision-making based on predictive analytics, improved resource allocation, and a more streamlined and intelligent management system. It is crucial to address privacy concerns, ensure transparency, and maintain robust security measures to build trust and confidence in the system. Ultimately, the successful implementation and evolution of an intelligent people and vehicle counting system can contribute to the overall effectiveness and modernization of secretariat operations.

Enhanced Visitor Experience: By implementing an intelligent counting system, secretariats can provide a seamless and efficient experience for visitors. Real-time data on parking availability, optimized traffic flow, and streamlined access control can reduce waiting times and improve overall satisfaction.

Resource Optimization: The data collected by the intelligent counting system can be utilized to optimize resource allocation within the secretariat. By analyzing patterns and trends, organizations can make informed decisions about staffing, space utilization, and facility management, leading to cost savings and increased productivity.

Integration with IoT and Smart Devices: As the Internet of Things (IoT) continues to expand, intelligent counting systems can integrate with various smart devices and sensors. This integration can provide a comprehensive view of the secretariat's operations, enabling automated processes, such as adjusting lighting and HVAC systems based on occupancy levels, further enhancing energy efficiency.

Customization and Scalability: Future systems can offer customization options to adapt to the specific needs of different secretariats. Whether it's tailoring the system to handle peak hours or accommodating specific security protocols, the flexibility of customization allows organizations to optimize the system according to their unique requirements. Moreover, the system should be scalable to accommodate potential growth or changes in the secretariat's infrastructure.

Data-Driven Decision Making: The intelligent counting system generates valuable data that can be leveraged for data-driven decision making. By analyzing historical and real-time data, secretariats can identify trends, anticipate future needs, and implement evidence-based strategies to improve operations, security measures, and resource planning.

Collaboration and Integration with Stakeholders: Secretariats can collaborate with relevant stakeholders, such as law enforcement agencies, transportation departments, and neighboring organizations, to integrate their systems and share data. This collaboration can lead to a more connected and efficient ecosystem, where data is shared, and responses to incidents or emergencies are coordinated effectively.

Continuous Improvement: The future scope of intelligent counting systems includes a focus on continuous improvement. Organizations can regularly update and upgrade the system to incorporate the latest technological advancements, address emerging security threats, and improve the user interface and experience.

9.FUTURE SCOPE:

Integration with Smart City Infrastructure:

Intelligent systems can be integrated with broader smart city initiatives, allowing for seamless coordination between different sectors, such as transportation, emergency services, and public utilities.

Advanced Analytics and Predictive Capabilities:

Future systems could incorporate advanced analytics and machine learning techniques to provide predictive insights. This could include predicting traffic patterns, estimating parking availability, and optimizing resource allocation based on historical data.

Multi-modal Data Integration:

Integrating data from various sources, such as CCTV cameras, sensors, and mobile devices, could provide a more comprehensive understanding of vehicle and people movements. This could lead to more accurate counting and better analysis of patterns and trends.

Enhanced User Experience:

Future systems could focus on improving the user experience by offering features such as mobile applications for real-time updates, navigation assistance, and personalized services based on user preferences.

10.APPENDIX:

```
##Contador de personas
##Federico Mejia
import numpy as np
import cv2
import time
import pyttsx3
import requests
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
organization = "4roq77"
deviceType = "abcd"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"
engine = pyttsx3.init()
engine.say('Hello')
engine.runAndWait()
#Contadores de entrada y salida
cnt_up   = 0
```

```
cnt_down = 0


#Fuente de video
#cap = cv2.VideoCapture(0)
#cap = cv2.VideoCapture('people.mp4')

#Propiedades del video
##cap.set(3,160) #Width
##cap.set(4,120) #Height

#Imprime las propiedades de captura a consola
cap = cv2.VideoCapture('people.mp4')
#cap = cv2.VideoCapture(0)
for i in range(19):
    print (i, cap.get(i))


w = cap.get(3)
h = cap.get(4)
frameArea = h*w
areaTH = frameArea/250
print ('Area Threshold', areaTH)

#Lineas de entrada/salida
```

```
line_up = int(2*(h/5))

line_down   = int(3*(h/5))


up_limit =   int(1*(h/5))

down_limit = int(4*(h/5))

print ("Red line y:",str(line_down))

print ("Blue line y:", str(line_up))

line_down_color = (255,0,0)

line_up_color = (0,0,255)

pt1 =  [0, line_down];

pt2 =  [w, line_down];

pts_L1 = np.array([pt1,pt2], np.int32)

pts_L1 = pts_L1.reshape((-1,1,2))

pt3 =  [0, line_up];

pt4 =  [w, line_up];

pts_L2 = np.array([pt3,pt4], np.int32)

pts_L2 = pts_L2.reshape((-1,1,2))


pt5 =  [0, up_limit];

pt6 =  [w, up_limit];

pts_L3 = np.array([pt5,pt6], np.int32)

pts_L3 = pts_L3.reshape((-1,1,2))

pt7 =  [0, down_limit];
```

```
pt8 =  [w, down_limit];

pts_L4 = np.array([pt7,pt8], np.int32)

pts_L4 = pts_L4.reshape((-1,1,2))


#Substractor de fondo

fgbg = cv2.createBackgroundSubtractorMOG2(detectShadows = True)


#Elementos estructurantes para filtros morfoogicos

kernelOp = np.ones((3,3),np.uint8)

kernelOp2 = np.ones((5,5),np.uint8)

kernelCl = np.ones((11,11),np.uint8)


#Variables

font = cv2.FONT_HERSHEY_SIMPLEX

persons = []

max_p_age = 5

pid = 1

def ibmwork(cnt_up,cnt_down,deviceCli):

    data = { 'UP' : cnt_up, 'down': cnt_down}

        #print data

    def myOnPublishCallback():

        print ("Published Up People Count = %s" % str(cnt_up), "Down People Count =
%s " % str(cnt_down), "to IBM Watson")
```

```
    success = deviceCli.publishEvent("PeopleCounter", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
    deviceCli.disconnect()
def ibmstart(cnt_up,cnt_down):

    try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-
method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        print(type(deviceCli))
        #.....
    except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()
    deviceCli.connect()
    ibmwork(cnt_up,cnt_down,deviceCli)

while(cap.isOpened()):
##for image in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
    #Lee una imagen de la fuente de video
```

```
    ret, frame = cap.read()
##    frame = image.array

    for i in persons:
        i.age_one() #age every person one frame
    ############################
    #   PRE-PROCESAMIENTO   #
    ############################

    #Aplica substraccion de fondo
    fgmask = fgbg.apply(frame)
    fgmask2 = fgbg.apply(frame)

    #Binariazcion para eliminar sombras (color gris)
    try:
        ret,imBin= cv2.threshold(fgmask,200,255,cv2.THRESH_BINARY)
        ret,imBin2 = cv2.threshold(fgmask2,200,255,cv2.THRESH_BINARY)
        #Opening (erode->dilate) para quitar ruido.
        mask = cv2.morphologyEx(imBin, cv2.MORPH_OPEN, kernelOp)
        mask2 = cv2.morphologyEx(imBin2, cv2.MORPH_OPEN, kernelOp)
        #Closing (dilate -> erode) para juntar regiones blancas.
        mask =  cv2.morphologyEx(mask , cv2.MORPH_CLOSE, kernelCl)
        mask2 = cv2.morphologyEx(mask2, cv2.MORPH_CLOSE, kernelCl)
```

```
    except:
        print('EOF')
        print ('UP:',cnt_up)
        print ('DOWN:',cnt_down)
        break
    ##################
    #   CONTORNOS    #
    ##################

    # RETR_EXTERNAL returns only extreme outer flags. All child contours are left behind.
    contours0, hierarchy = cv2.findContours(mask2,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
    for cnt in contours0:
        area = cv2.contourArea(cnt)
        if area > areaTH:
            ##################
            #   TRACKING    #
            ##################
            #Falta agregar condiciones para multipersonas, salidas y entradas de pantalla.

            M = cv2.moments(cnt)
            cx = int(M['m10']/M['m00'])
```

```
            cy = int(M['m01']/M['m00'])
            x,y,w,h = cv2.boundingRect(cnt)


            new = True
            if cy in range(up_limit,down_limit):
                for i in persons:
                    if abs(cx-i.getX()) <= w and abs(cy-i.getY()) <= h:
                        # el objeto esta cerca de uno que ya se detecto antes
                        new = False
                        i.updateCoords(cx,cy)   #actualiza coordenadas en el objeto and resets
age
                        if i.going_UP(line_down,line_up) == True:
                            cnt_up += 1;
                            print ("ID:",i.getId(),'crossed going up at',time.strftime("%c"))
                            engine.say('A Person is Going UP ')
                            engine.runAndWait()
                        elif i.going_DOWN(line_down,line_up) == True:
                            cnt_down += 1;
                            print ("ID:",i.getId(),'crossed going down at',time.strftime("%c"))
                            engine.say('A Person is Going Down')
                            engine.runAndWait()
                        break
                    if i.getState() == '1':
```

```
                if i.getDir() == 'down' and i.getY() > down_limit:
                    i.setDone()
                elif i.getDir() == 'up' and i.getY() < up_limit:
                    i.setDone()
            if i.timedOut():
                #sacar i de la lista persons
                index = persons.index(i)
                persons.pop(index)
                del i     #liberar la memoria de i
        if new == True:
            p = Person.MyPerson(pid,cx,cy, max_p_age)
            persons.append(p)
            pid += 1
        ##################
        #   DIBUJOS     #
        ##################
        cv2.circle(frame,(cx,cy), 5, (0,0,255), -1)
        img = cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2)
        #cv2.drawContours(frame, cnt, -1, (0,255,0), 3)

#END for cnt in contours0

############################
```

```
    # DIBUJAR TRAYECTORIAS  #
    ##########################
    for i in persons:
##        if len(i.getTracks()) >= 2:
##            pts = np.array(i.getTracks(), np.int32)
##            pts = pts.reshape((-1,1,2))
##            frame = cv2.polylines(frame,[pts],False,i.getRGB())
##        if i.getId() == 9:
##            print str(i.getX()), ',', str(i.getY())
        cv2.putText(frame,
str(i.getId()),(i.getX(),i.getY()),font,0.3,i.getRGB(),1,cv2.LINE_AA)


    ##################
    #   IMAGANES    #
    ##################
    str_up = 'UP: '+ str(cnt_up)
    str_down = 'DOWN: '+ str(cnt_down)
    print('----------------------------------------')
    print ('UP:',cnt_up)
    print ('DOWN:',cnt_down)


    #r1 =
requests.get('https://api.thingspeak.com/update?api_key=4BGMGGBRLQM3VRHO&fi
eld1='+str(cnt_up))
```

```
    # r2 =
requests.get('https://api.thingspeak.com/update?api_key=4BGMGGBRLQM3VRHO&fi
eld2='+str(cnt_down))
    # print(r1.status_code)
    # print(r2.status_code)
     frame = cv2.polylines(frame,[pts_L1],False,line_down_color,thickness=2)
     frame = cv2.polylines(frame,[pts_L2],False,line_up_color,thickness=2)
     frame = cv2.polylines(frame,[pts_L3],False,(255,255,255),thickness=1)
     frame = cv2.polylines(frame,[pts_L4],False,(255,255,255),thickness=1)
     cv2.putText(frame, str_up ,(10,40),font,0.5,(255,255,255),2,cv2.LINE_AA)
     cv2.putText(frame, str_up ,(10,40),font,0.5,(0,0,255),1,cv2.LINE_AA)
     cv2.putText(frame, str_down ,(10,90),font,0.5,(255,255,255),2,cv2.LINE_AA)
     cv2.putText(frame, str_down ,(10,90),font,0.5,(255,0,0),1,cv2.LINE_AA)


     cv2.imshow('Frame',frame)
     #cv2.imshow('Mask',mask)


     #preisonar ESC para salir
     ibmstart(cnt_up,cnt_down)
# Disconnect the device and application from the cloud


     k = cv2.waitKey(30) & 0xff
     if k == 27:
```

```
        break
    #END while(cap.isOpened())


    ##################
    #   LIMPIEZA    #
    ##################


    cap.release()
    cv2.destroyAllWindows()
```

GITHUB LINK:

https://github.com/naanmudhalvan-SI/PBL-NT-GP--6369-1680926857/tree/main

VIDEO DEMO LINK:

https://drive.google.com/drive/folders/1mELjD4HRIQExR3J1yQ6c_EoBm6qPxcOp