SUPERMARKET QUEUE SIMULATION
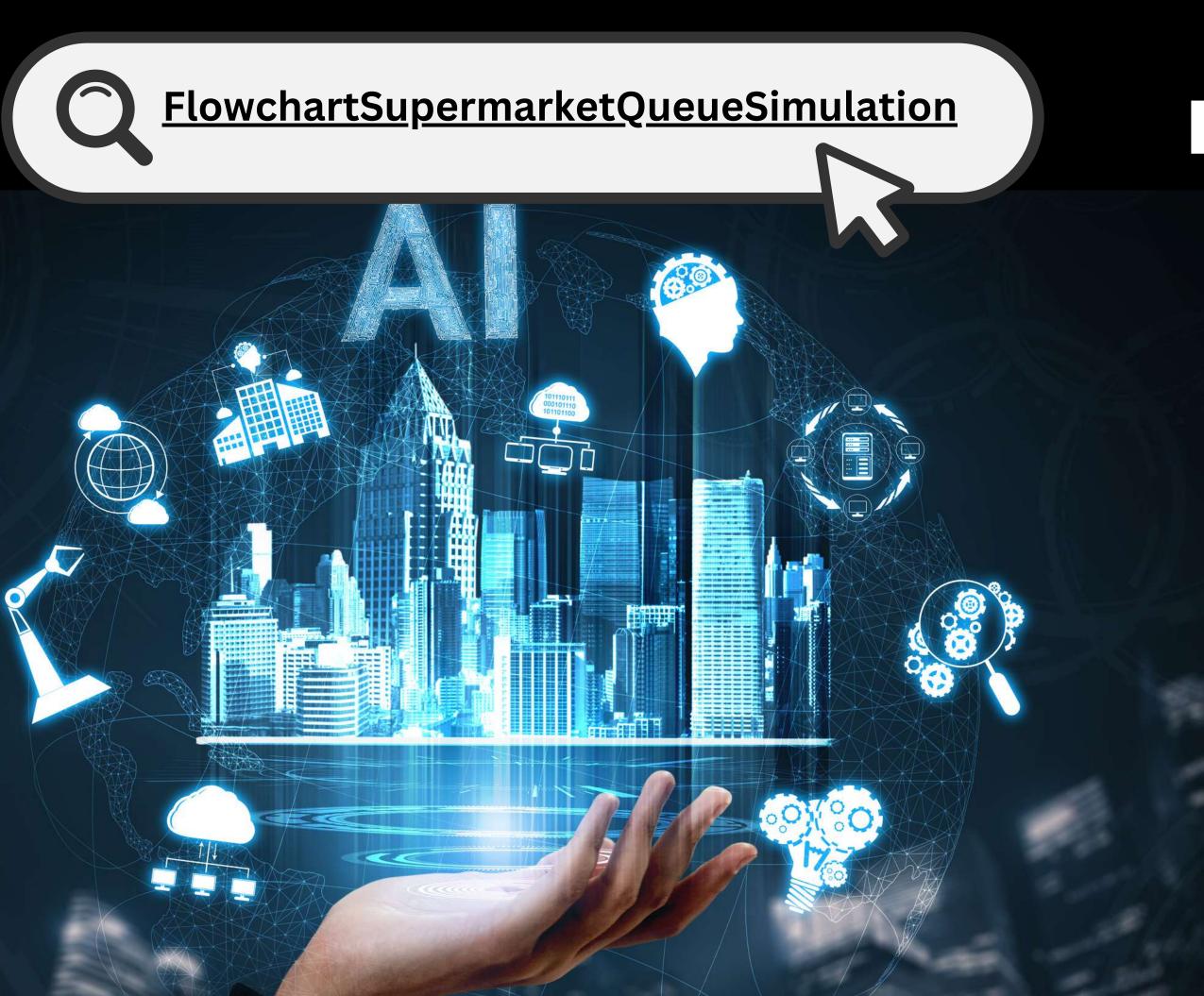
DATA STRUCTURE AND ALGORITHMS

INTRODUCTION

The Supermarket Queue Simulation project aims to optimize supermarket operations by managing customer queues at cash counters. It uses Java programming and interactive user inputs to analyze queue management strategies, providing real-time feedback on customer waiting times. The simulation also generates comprehensive outputs, enabling data-driven decision-making for supermarket management, reducing waiting times and enhancing customer satisfaction.

FlowchartSupermarketQueueSimulation

FRAMEWORK

We are eager to illustrate the comprehensive workflow of our project through a detailed flowchart. For a visual representation of the Supermarket Queue Simulation's operational processes.
Please refer to the link above.

ALTERNATIVE

An alternative data structure that could be used instead of a standard queue is a Linked List. Unlike a queue, which follows a First-In-First-Out (FIFO) order

Advantages of Using Linked List are Dynamic Size, Efficient Insertion and Removal, Easy Positional Access

For see the flowchart please refer to the link below

🔍 FlowchartAlternativeSupermarketQueue

ADDITIONAL FEATURES

The Supermarket Queue Simulation project not only meets the basic project requirements but also incorporates several key features to enrich the simulation's realism and adaptability. These enhancements, including Service Time Variability and Real-time Status Display, aim to emulate the complexities of real-world service environments, offering users a more comprehensive and engaging experience. Service Time Variability introduces variability in service times across different counters, reflecting the diverse pace of service in real-world scenarios.

ADDITIONAL FEATURES

This addition enhances realism and provides users with a more immersive understanding of service-oriented settings. Meanwhile, the Real-time Status Display dynamically updates the simulation interface to reflect the system's current status, allowing users to observe changes in real-time and facilitating a deeper understanding of system dynamics. Overall, these features elevate the simulation, offering users a more realistic and interactive experience.

TESTING

Test Case: Basic Simulation Setup

- Description: This test case verifies the correct initialization of the simulation with basic parameters.
- Input:
  Number of cash counters = 3,
  Number of customer = 10
  Service time = 5 minutes
  Arrival rate = 10 customers per minute
- Expected Result: The simulation initializes without errors, and basic statistics such as average wait time and queue length are within expected ranges.

Test Case: Service Time Variability

- Description: This test assesses the impact of service time variability among counters on overall system performance.
- Input:
  Number of cash counters = 3,
  Number of cash counters = 5
  Number of customer = 5
  Service time = 60 minutes
  Arrival rate = 10 customers per minute
- Expected Result: The simulation reflects varied service times among counters, with some serving faster than others. The average wait time should demonstrate the effects of this variability.

Conclusion

- **The Supermarket Queue Simulation project, driven by Java programming, addresses challenges in large supermarket queues effectively.**

- **User-friendly features like real-time feedback enhance the simulation's dynamic nature.**

- **Future work, including refining queueing strategies and enhancing visualization, highlights ongoing development opportunities.**

- **Its impact extends beyond theoretical insights, making it a valuable tool for supermarket operations.**

Thank You!