April 2020

WALMART SPARKPLUG

TEAM: PandemicSparks

THE TEAM

L SRIHARI	BITS Hyderabad
DAKSH YASHLAHA	BITS Hyderabad
SHREYAM KUMAR	BITS Hyderabad

Problem Statement 2

Spark innovation with a robust Wal-Ket basket.

Problem Statement and Interpretation

- The associate navigates through the store and picks up items from the shelves/aisles belonging to a given order
- Each order may contain multiple items
- A set of associates are always available to fulfil orders
- If no associate is available, the waiting time till next assignment will not be considered towards the inefficiency of the Wal-Ket system
- Each associate can be given a maximum of 3 orders which may each contain any number of items
- Given a set of 3 or less orders, an associate will pick all the items without sharing the work with other associates
- In addition to specifying the fastest path (most efficient store navigation) to associates, the problem statement's task also extends to the design of how the Wal-Ket system assigns orders to the associates (How the customer orders are batched)
- We also suggest other ideas, in addition to the solution for the two main problems mentioned above, for more efficient fulfilment

CONCEPTS INVOLVED

Supply Chain Concepts

- LEAN
- DMAIC SIX SIGMA

Programming and Presenting

- MERN Stack framework
- System Architecture
- Data Mining and Clustering
- Graph Algorithms

SPARK 1

CLUSTERING AND LOCALITY SENSITIVE HASHING

(to Batch and assign orders)

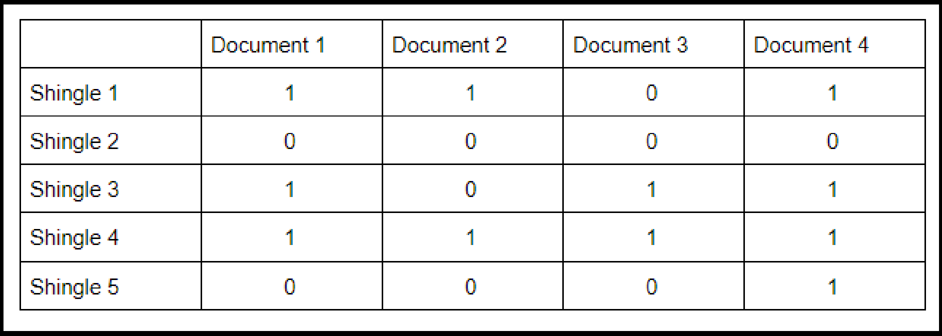
Clustering

- The Wal-Ket system has a stored mapping of items to its location within the facility
- Initially, we take a number of orders (maximum of 3 x number of waiting associates) and map each item onto a layout of the facility
- Please note that currently, no distinction has been made about which item belongs to which order. All items currently involved in any order is plotted.
- The facility is divided into a number of rectangular regions based on department / section
- All items falling within a rectangle are clustered and given a cluster ID.
- We call this a primary clustering
- Then a secondary clustering is performed on the primary clusters containing at-least one point. (The need and further information about secondary clustering is discussed after LSH)

Locality Sensitive Hashing with minhash

The algorithm

- Locality sensitive hashing or LSH is a concept of Information retrieval or Data Mining
- Documents are represented as a collection of words called shingles
- Each document is represented as a boolean vector over the shingles where the document is given 1 if the shingle is present and 0 if it is not.
- Let us call such a vector, a document vector.
- The algorithm takes a set of these document vectors as input and performs the minhashing technique to decrease the document size
- It then maps the minhash signatures into buckets so that similar items are hashed to the same bucket. (Please refer section 3.3 and 3.4 of Mining of Massive Datasets)



	Document 1	Document 2	Document 3	Document 4
Shingle 1	1	1	0	1
Shingle 2	0	0	0	0
Shingle 3	1	0	1	1
Shingle 4	1	1	1	1
Shingle 5	0	0	0	1

Conversion of our problem to an instance of LSH

- After secondary hashing the plotted points are assigned new cluster IDs
- We can now consider the cluster IDs as shingles and represent each order (a list of items) as a boolean vector over cluster IDs
- Minhashing can be performed and the LSH algorithm can be run to group similar orders based on the number of clusters they have in common
- Orders grouped within a bucket are chosen 3 at a time and given to the associates
- The parameters 'b' and 'r' (in LSH) are adjusted so that too many orders do not pile up within the same bucket

Correctness

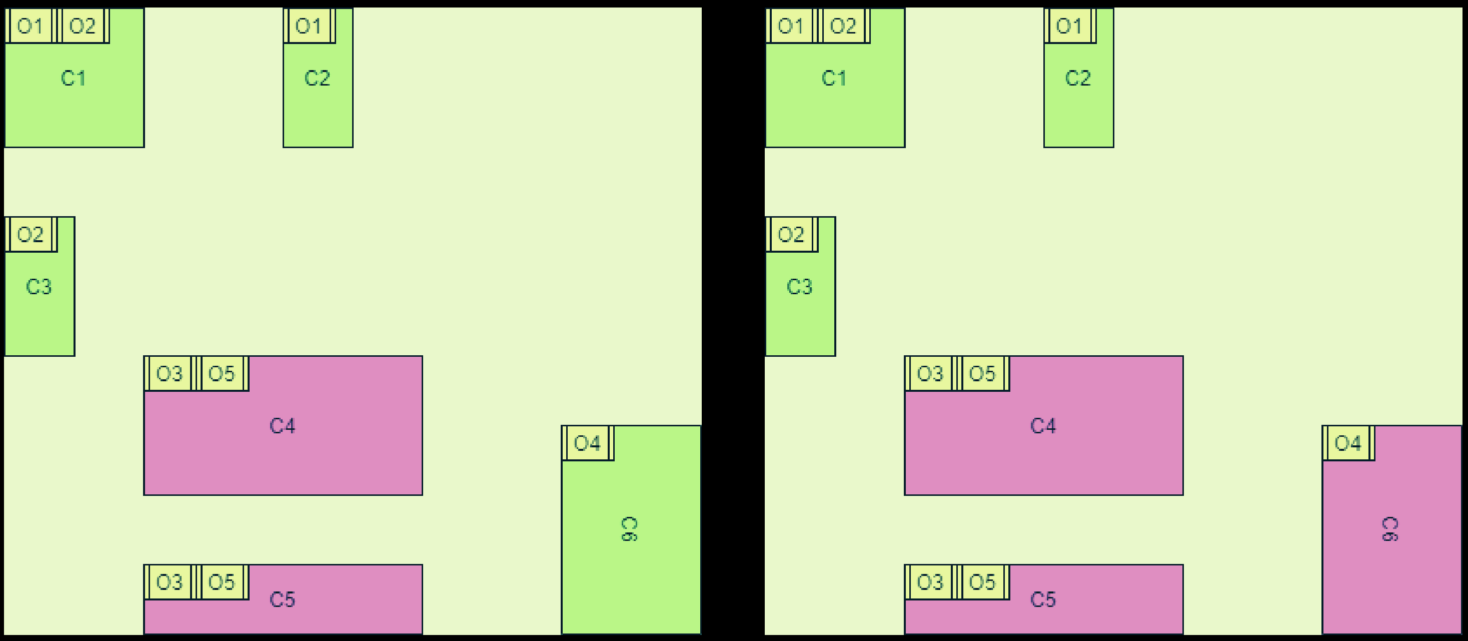
- This algorithm ensures that multiple orders having items very close to each other go to the same associate
- This is because, the clustering enables all items close by to be grouped together and the LSH algorithm makes certain that orders having maximum close by items are bucketed together

Why Primary clustering is insufficient?

- Consider the following simple case of five orders and their respective plots in the diagram
- Each order has two items

Order 1: { item 1: C1, item 2: C2 }
Order 2: { item 1: C3, item 2: C1 }
Order 3: { item 1: C4, item 2: C5 }
Order 4: { item 1: C6, item 2: C6 }
Order 5: { item 1: C4, item 2: C5 }

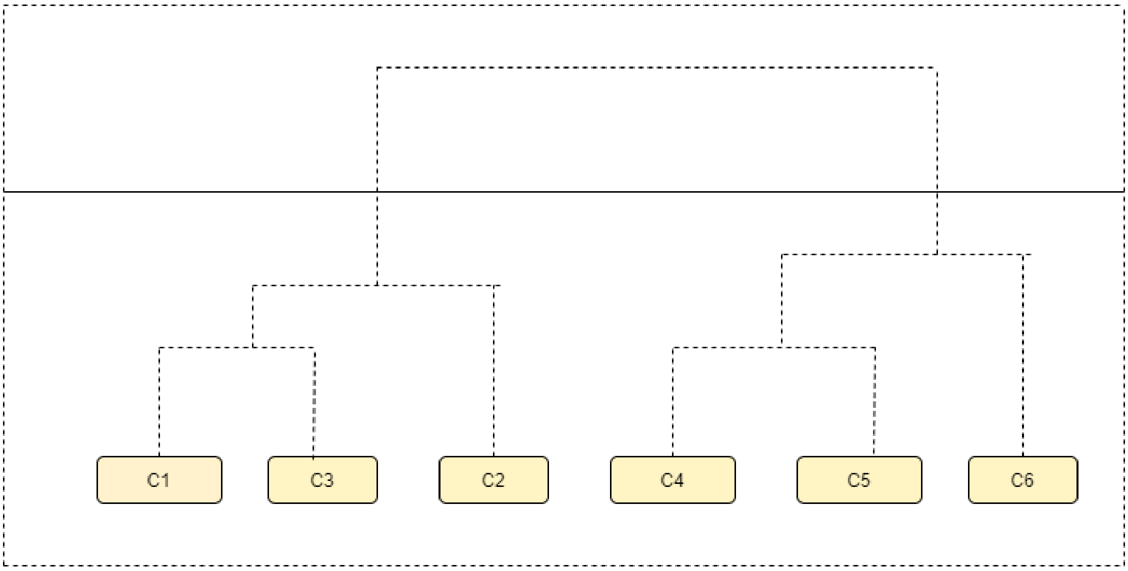
- Green clusters are given to one associate and pink to another



- It is evident that O1 and O2 will be grouped together since they have maximum common clusters, so will O3 and O5
- They real problem is with the assignment of O4
- Both the above cases are possible with primary clustering since relative distance is not taken into account
- But it is clear that assignment 2 is better since C6 is much closer to the pinks than the greens

Secondary Clustering

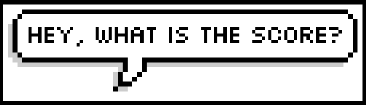
- For distance based clustering, we propose to use Agglomerative Clustering
  on those primary clusters containing at least one point
- All clusters may not contain points and there are limited primary clusters to consider, hence, this can be approximated to be a constant time calculation
- Even though all pairwise distances are considered, it is still efficient
- Dendograms can be later constructed to break the clusters at required distances



SPARK 2

SHORTEST MANHATTAN PATH FINDING

- For the most effective utilisation of time and resources, once a order is assigned to an employee, we need to find an optimal (shortest) path to all the sections where the items are available
- This is achieved by making use of A-star pathfinding algorithm
- The heuristic function used is Manhattan Distance
- Other algorithms like Djikstra's are not considered since we need the shortest route to a single destination rather than all destinations
- The path keeps being updated dynamically based on the position of other employees (for clash free pathing)

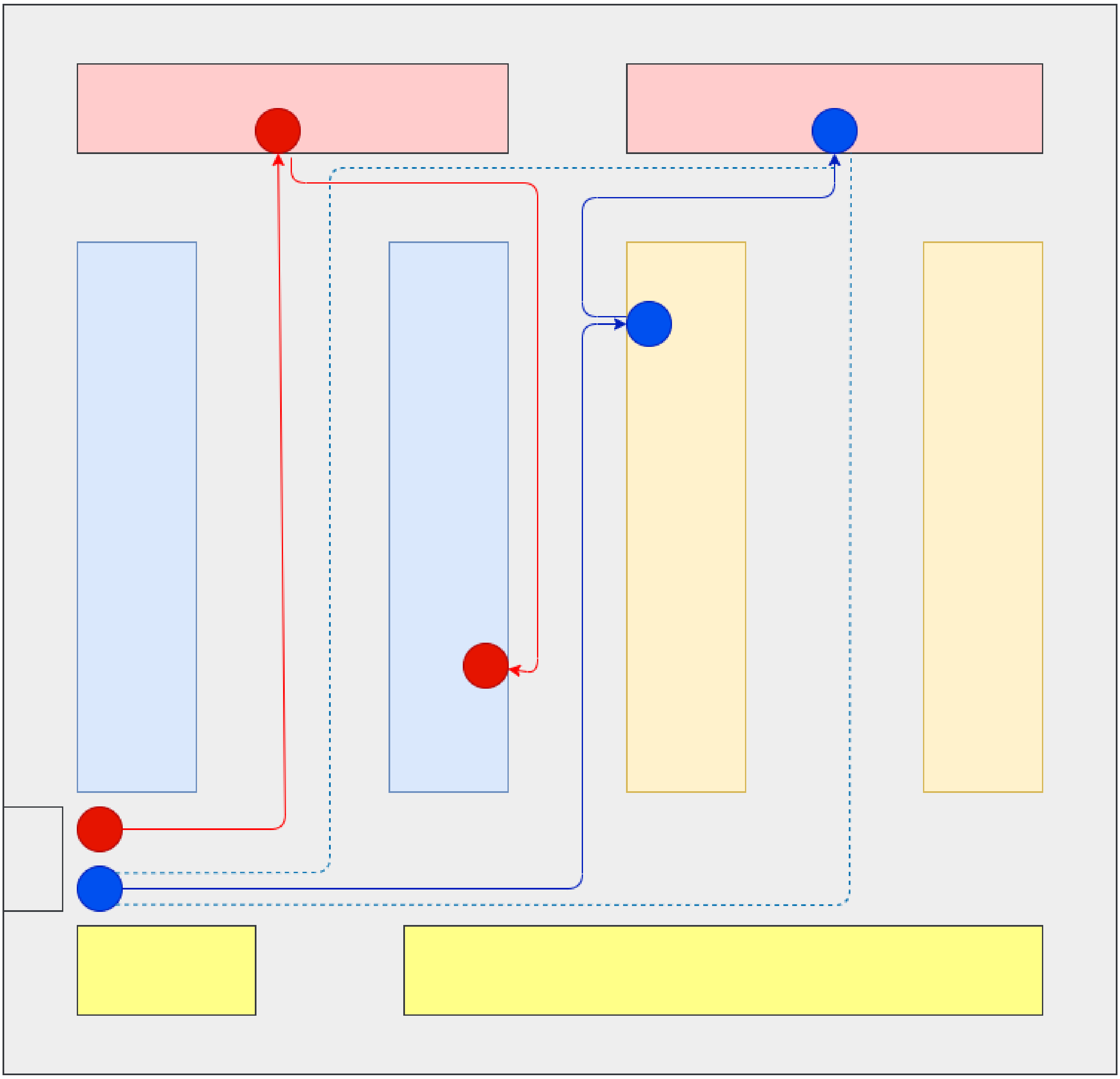


SPARK 3

COLLISION FREE PATHFINDING

- Collision free pathfinding is necessary as the aisles of supermarkets are usually not conducive for multiple people crossing at the same time
- To resolve such clashes, we propose an algorithm that takes a time based cost into account when searching for least cost path
- A modified A-star algorithm is what we will be using where the heuristic is Manhattan distance
- The cost for the shelves is set extremely high as we cannot cross shelves when finding the least cost path
- The cost for all other paths is initially set to one
- When a path is found using A star, the traversed cells are given an updated cost for the time till which current expansion is completed
- Since path till prior positions are optimal, any simultaneous intersection will find a different path
- This allows for avoiding clashes as two paths will not readily share the same cells at the same instant of time

- Using this time based cost updation in A star, we can avoid collisions as much as possible



SPARK 4

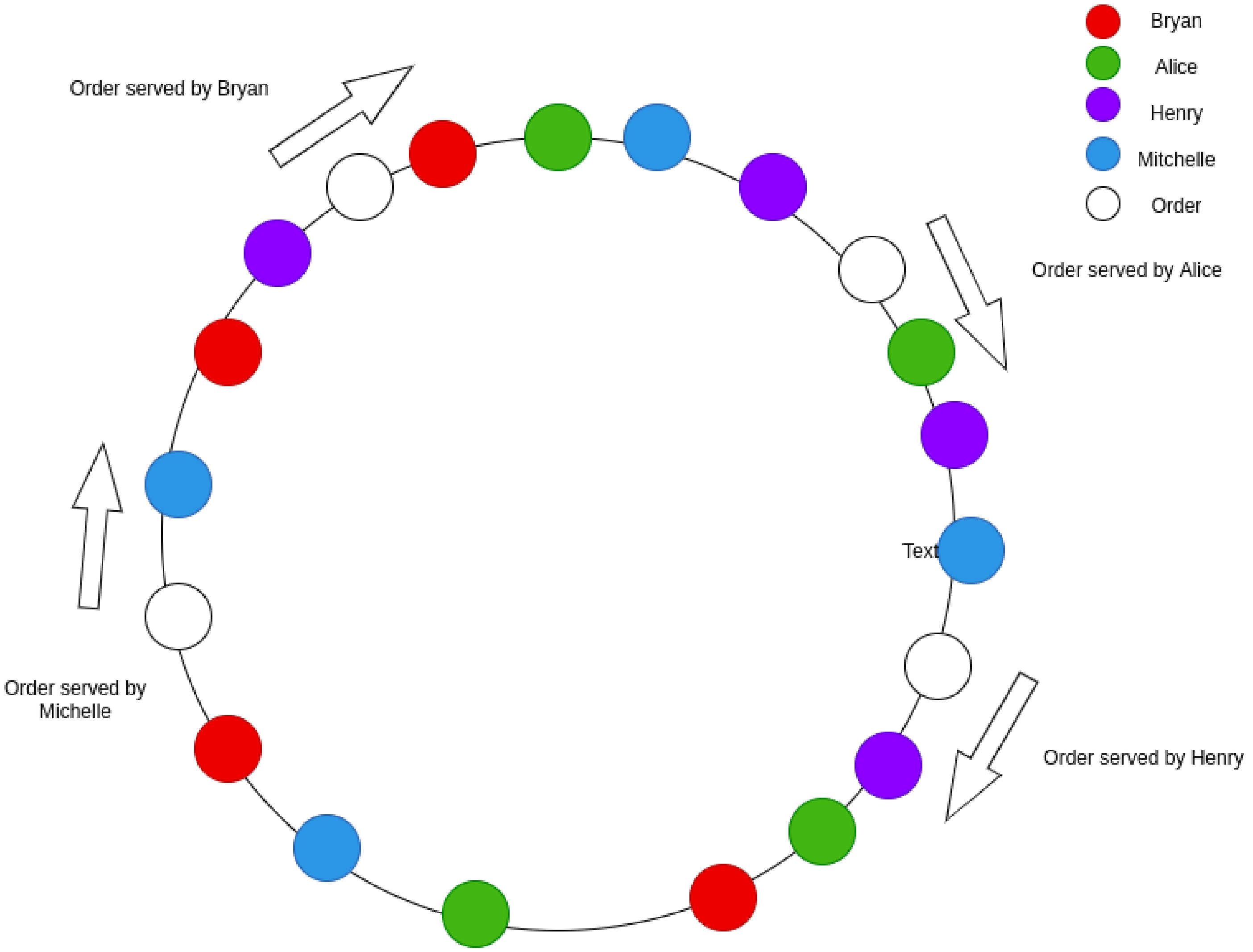
SYSTEM ARCHITECTURE

- Walmart stores receive a large number of orders at a given time and handling these orders simultaneously becomes a very difficult task
- In order to solve this problem in an effective manner, we employ the use of Consistent Hashing.
- This technique is used exclusively in load balancers to distribute a large amount of incoming requests equally amonst the available servers.
- In a similar fashion, this can be used to distribute the orders amongst the employees in an uniform way to prevent under-utilization of resources.

ALGORITHM

- Consider a total of M orders which are to be collected by N employees. These orders have already been preprocessed by clustering techniques mentioned above
- Each order and each employee has a unique ID in the form of an integer
- Consider multiple hash functions which hash to the same range of values (let it be from a to b)

- Consider a circular data structure that ranges from a to b. The hash functions maps the employees (by using their IDs) into this structure.
- Using multiple hashes results in multiple mapping of each employee
- As each order is received, it is hashed into this structure. The order is served by the first employee reached by moving in clockwise direction from the order's hash value
- It can be proved that on an average each employee receives 1/N of the total orders. Thus no employee is over-worked and the load is distributed effectively



SPARK 5

AUDIO BASED ITEM READOUT

- Since the associates are always busy with their hands, reading out of items and directions to aisles will be given through audio (text-to-speech)
- Checklists will be included in the API (Touch based device would be convenient)

SPARK 6

INCENTIVES TO ASSOCIATES BASED ON DISTANCE WALKED

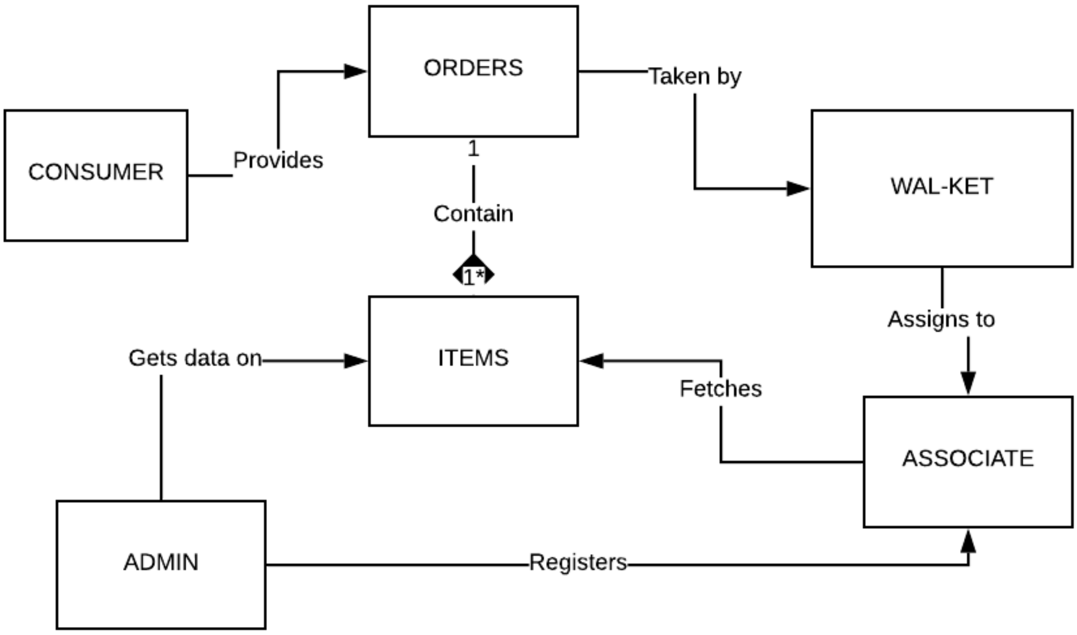
- The Wal-Ket system divides work efficiently among the associates
- Fast and consistent workers are promoted by giving incentives (as points to be used later)
- A nominal upper bound is fixed so that asscociates do not overwork or work in haste

SPARK 7

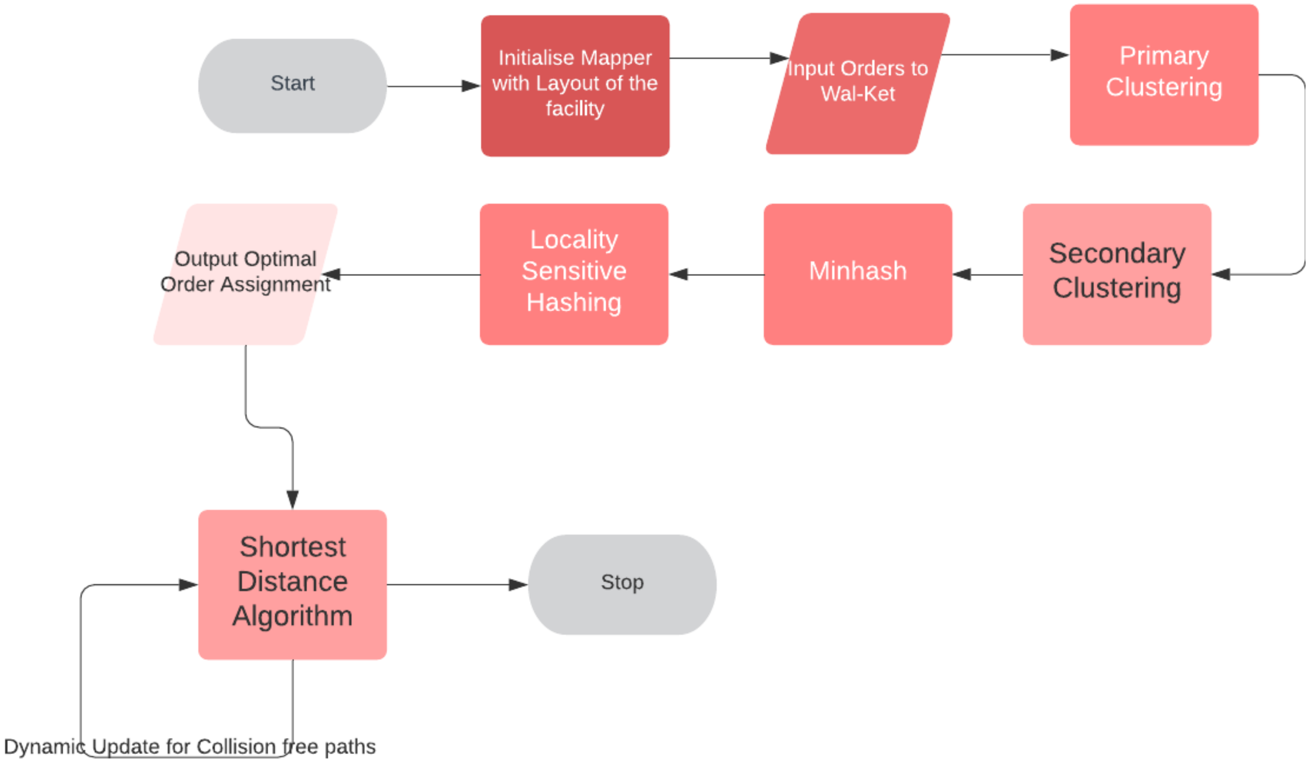
COLLECT DATA ON ITEMS FREQUENTLY BOUGHT TOGETHER

- Use the Frequent Itemset algorithm with recent research improvements and provide the data to Admin
- This also helps in management of different sections within a store: frequent itemsets are stored close to one another

OUR CLASSES



FLOW CHART

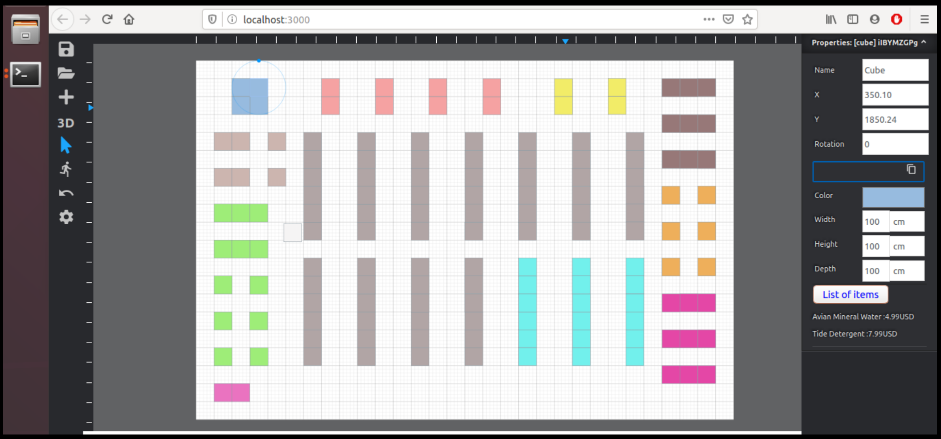

Start
Initialise Mapper with Layout of the facility
Input Orders to Wal-Ket
Primary Clustering
Output Optimal Order Assignment
Locality Sensitive Hashing
Minhash
Secondary Clustering
Shortest Distance Algorithm
Stop
Dynamic Update for Collision free paths

- These stages were considered after define, measure and analyze stages of six sigma with Lean support

OUTPUT OF CODE COMPLETED TILL NOW

Admin's view

- Admin can modify the layout
- Load and save existing layouts
- Check items belonging to each grid component



localhost:3000

Properties: [cube] iIBYMZGPg

Name	Cube
X	350.10
Y	1850.24
Rotation	0
Color	
Width	100 cm
Height	100 cm
Depth	100 cm

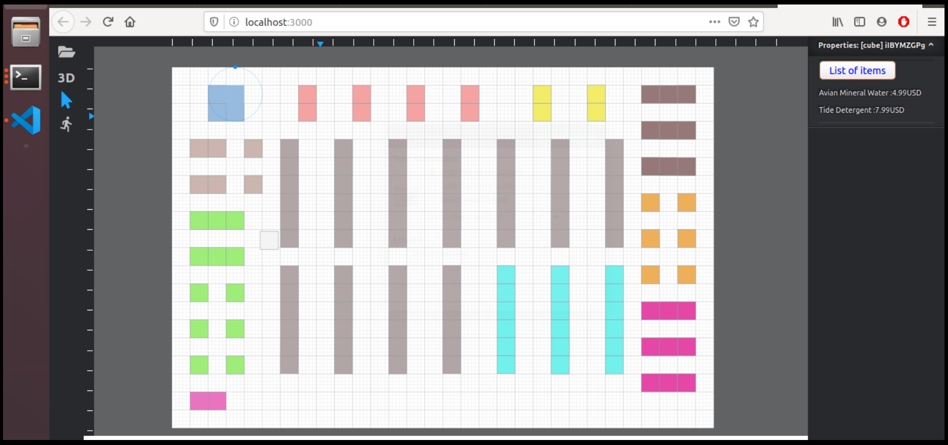
List of items

Avian Mineral Water :4.99USD

Tide Detergent :7.99USD

Associate's view

- Load existing layout
- Check items belonging to each grid component



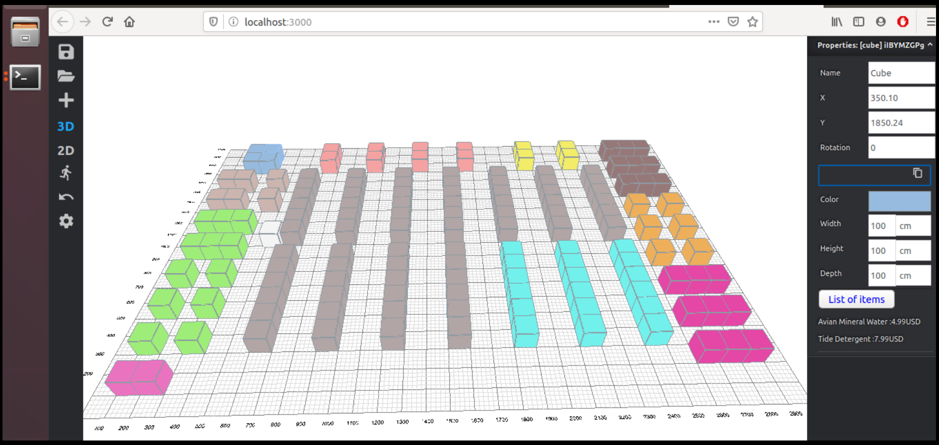
localhost:3000

Properties: [cube] iIBYMZGPg

List of items

Avian Mineral Water :4.99USD

Tide Detergent :7.99USD



localhost:3000

Properties: [cube] iIBYMZGPg

Name	Cube
X	350.10
Y	1850.24
Rotation	0
Color	
Width	100 cm
Height	100 cm
Depth	100 cm

List of items

Avian Mineral Water :4.99USD

Tide Detergent :7.99USD

3D VIEW

In addition to this

- Path finding algorithm was completed
- LSH was implemented
- Login and registration pages were created

THANK YOU