Date:22.10.25

TASK:11

Implementation of MAP NAVIGATION.

CO1, CO2, CO3 S8/L7

PROBLEM STATEMENT

Finding the shortest and fastest route between locations can be difficult using manual maps or without real-time updates. The problem is to create a map navigation system that accurately detects the user's location, calculates the best route, provides turn-by-turn directions, and adapts to traffic or road changes for easier and faster travel.

AIM

The aim of map navigation is to help users identify their current position, plan routes, and navigate efficiently from one point to another by using maps, GPS technology, and intelligent pathfinding algorithms, ensuring accuracy, safety, and time optimization.

OBJECTIVE

1. To develop an interactive map interface that allows users to view and explore different locations.
2. To determine the user's current location accurately using GPS technology.
3. To calculate the shortest and most efficient route between two or more points.
4. To provide step-by-step or turn-by-turn navigation guidance to users.
5. To analyze real-time data such as traffic, distance, and travel time for route optimization.
6. To enhance user experience through easy, reliable, and accurate navigation support.

DESCRIPTION

The Map Navigation System is a digital tool designed to help users locate places and find the most efficient routes between two or more locations. It uses GPS technology and digital maps to track the user's current position in real time. The

system provides step-by-step directions, displays distance and estimated travel time, and suggests the shortest or fastest route available.

In addition, it can update routes automatically based on traffic conditions, road blocks, or user preferences, ensuring smooth and accurate navigation. This system is widely used in smartphones, vehicles, and transport applications, making travel and delivery operations more convenient and efficient.

ALGORITHM

Step 1: Start the process**.**
Step 2: Get the user's current location using GPS**.**
Step 3: Ask the user to enter the destination location**.**
Step 4: Fetch map data (roads, distances, landmarks, etc.) from the database or map API.
Step 5: Use a pathfinding algorithm (like Dijkstra's or A* algorithm) to find the shortest or fastest route**.**
Step 6: Display the route on the map with distance and estimated travel time.
Step 7: Provide turn-by-turn directions for navigation**.**
Step 8: Continuously update the user's position and adjust the route if necessary (e.g., traffic or deviation).
Step 9: Stop when the user reaches the destination**.**

PROGRAM

```
import heapq


# Function to find shortest path using Dijkstra's Algorithm

def dijkstra(graph, start, destination):

    # Priority queue for managing routes

    queue = [(0, start)]

    distances = {node: float('inf') for node in graph}

    distances[start] = 0

    path = {start: None}
```

```
    while queue:
        (current_distance, current_node) = heapq.heappop(queue)

        if current_node == destination:
            break

        for neighbor, weight in graph[current_node].items():
            distance = current_distance + weight
            if distance < distances[neighbor]:
                distances[neighbor] = distance
                path[neighbor] = current_node
                heapq.heappush(queue, (distance, neighbor))

    # Reconstruct shortest path
    route = []
    node = destination
    while node is not None:
        route.insert(0, node)
        node = path[node]

    return route, distances[destination]

# Example map (graph representation)
```

```
graph = {
    'A': {'B': 4, 'C': 2},
    'B': {'A': 4, 'C': 1, 'D': 5},
    'C': {'A': 2, 'B': 1, 'D': 8, 'E': 10},
    'D': {'B': 5, 'C': 8, 'E': 2, 'Z': 6},
    'E': {'C': 10, 'D': 2, 'Z': 3},
    'Z': {'D': 6, 'E': 3}
}

# Input locations
start = input("Enter starting point: ")
destination = input("Enter destination point: ")

# Find and display shortest route
route, distance = dijkstra(graph, start, destination)
print("\nShortest Route:", " -> ".join(route))
print("Total Distance:", distance)
```

OUTPUT

```
Enter starting point: A
Enter destination point: Z

Shortest Route: A -> C -> B -> D -> E -> Z
Total Distance: 13
```

CONCLUSION

The **Map Navigation System** provides an effective way to **find and follow the best route** between two locations using **GPS and digital mapping technologies**. It helps users reach their destinations easily by providing **accurate directions, shortest paths, and real-time updates**. By using efficient algorithms like **Dijkstra's or A***, the system ensures **time-saving and reliable navigation**. Overall, it enhances travel convenience, reduces effort, and improves route planning for everyday users, drivers, and delivery services.