Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

# CAPSTONE
# REVIEW II

# E-Vehicle Routing with Parking System


Name : Aryaman Mishra

Register No:19BCE1027

Programme & Specialization : B.Tech Computer Science and Engineering

Guide Name:

**Dr. Suguna M**

# Outline

# Introduction

- The existing network of transportation can no longer keep up with the growing demand in metropolitan cities. Short distance travel has become an unresolved issue for daily commuters. The case presents how MMVs have emerged as an alternative mode of transport for resolving issues of daily commuters regarding the first-mile connectivity, last-mile connectivity and short distance travel to reach their final destination.
- MMVs are basically light-weight vehicles which occupy less space on road. These vehicles include bicycles, e-bikes, skateboards, hoverboards and other battery-operated vehicles. An electric vehicle venture promotes the concept of green consumerism among the daily commuters at affordable rates.

# Introduction

- Recently, E-vehicles systems are being found to have been damaged by users due to inconsistent routes, mechanical problems and driver errors. Thus, this project intends to create a tool for damage limitation and optimize routing of such E-vehicles.
- In cities having the availabilities of E-vehicles, we are going to create an AI function to map the optimal route in a provided city landscape structure to avoid damage, engine stagnation, battery leakage and discharge by providing users the route to their destination on a point-to-point basis which allows them to reach their destinations on time with the integrity of the structure of the E-Vehicle intact.

# Problem Statement

- Taking into accounts the background of the project, our problem statement would be: “ In cities having the availabilities of E-vehicles, we are going to create an AI function to map the optimal route in a provided city landscape structure to avoid damage, engine stagnation, battery leakage and discharge by providing users the route to their destination on a point-to-point basis which allows them to reach their destinations on time with the integrity of the structure of the E-Vehicle intact, along with providing time and distance values in case of recharging vehicles and charging stations,taking into account the aspect of slot fulfillment,or go for alternative routes. ”

# Research Challenges

- Optimize algorithm to treat large city landscape as a huge graph to implement graph operations and routing techniques on.
- Minimize time of output.
- Obtain machine power to train epochs and minimize loss.
- Create User Interface for any person using e-vehicles to commute and limit damage to public e-vehicles which are used in transit.
- Consider hosting options for Bot or User Interface to deploy on cloud and provide results as soon as possible and observe limits of varying cloud services for application of research.

# Research objectives

◉ Get proper routes to account if in case user reaches destination or needs to head to a recharging station.

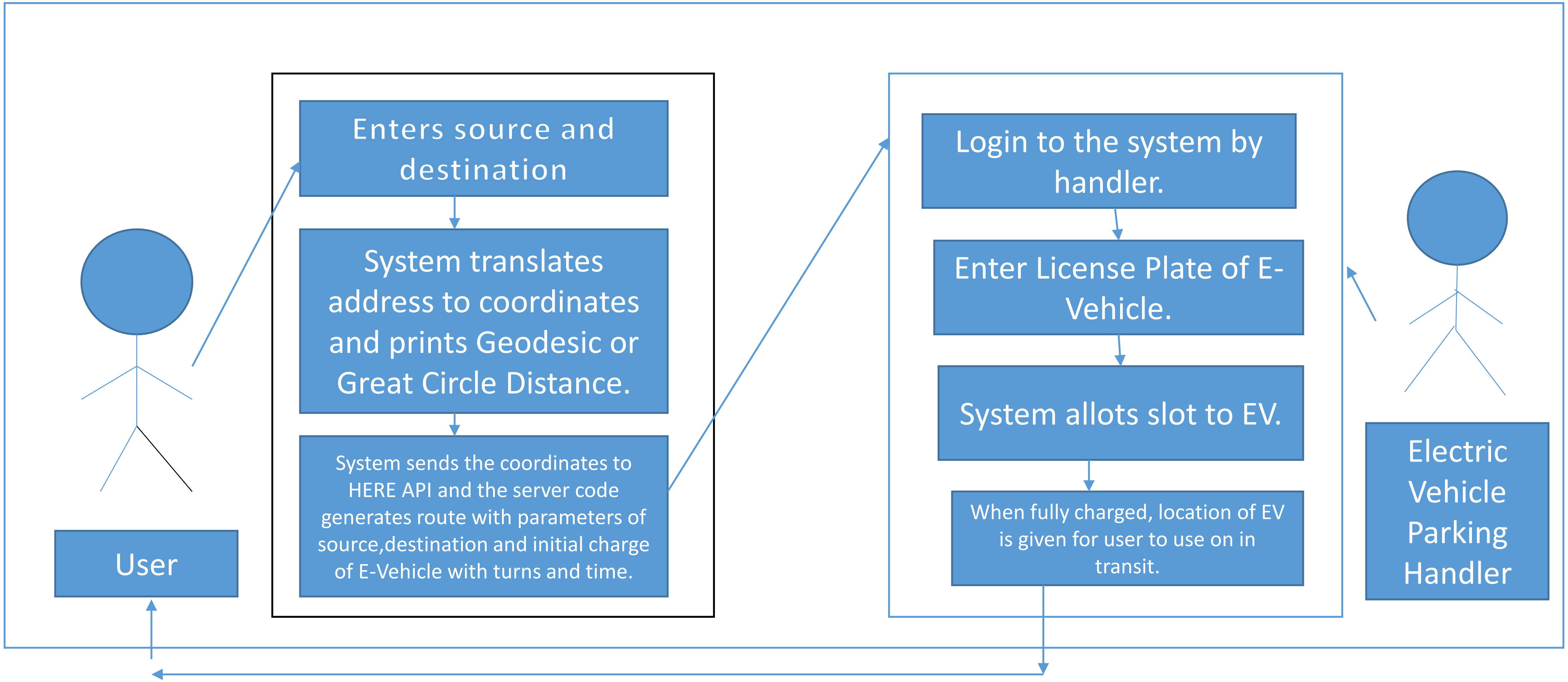
◉ At a recharging station, look for available slots or reroute to the nearest charging station with a free slot.

◉Implement a distance routing technique which adapts to rerouting functions incase slots are not available at a station for a vehicle to recharge without draining the battery.

# Proposed System Introduction

- The expanding demand in major cities is outpacing the capacity of the current transportation system. For daily commuters, short-distance transit has emerged as an unresolved problem. The instance illustrates how EVs have become a viable alternative mode of transportation for addressing daily commuters' concerns about first-mile connectivity, last-mile connectivity, and quick trips to their destination.
- EVs are essentially cars and bikes that on normal capacity cause less carbon emission and can be used for intra and intercity commute. Cars, Bicycles, e-bikes, skateboards, hoverboards, and other battery-powered vehicles are some of these. An electric vehicle venture promotes the concept of green consumerism among the daily commuters at affordable rates.
- Recently, E-vehicles systems are being found to have been damaged by users due to inconsistent routes, mechanical problems and driver errors. Thus, this project intends to create a tool for damage limitation and optimize routing of such E-vehicles.
- Electric vehicles have been gaining traction in India as a sustainable and eco-friendly alternative to traditional gasoline-powered vehicles. The Indian government has set ambitious targets for the adoption of EVs, with a goal of 30% of all vehicles on the road being electric by 2030.

# Proposed System Diagram



# List of Modules

- Parking System Login
- Parking System Vehicle Allotment
- Parking System Vehicle Finder
- Latitude/Longitude Coordinate to Address Generator
- Address to Latitude/Longitude Coordinate Generator
- Geodesic Distance Finder
- Great Circle Distance Finder
- Server Configuration

# Explanation of Modules (1)

- Our project makes use of 3 APIs: Google Geocoding API to generate coordinates for source and origin, OpenStreetMap to integrate map tile for our server-side program and HERE API to make use of Electric Vehicle Routing from Point A to Point B.
- We started by creating our command line program which would run the main script. An argument parser was integrated to  carry out 5 functions:
- Locate Coordinates from Address: We used the ‘geopy’ package in python n environment to accept an address as parameter which would call on the geocoding functions to print the Latitude and Longitude coordinates of the address as long as it is a valid address with no spelling errors.
- Locate Address from Coordinates: This function did the opposite from the first function as it would give an address by giving latitude-longitude coordinates.
- Great Circle Distance Calculator: Great Circle is the shortest distance between any two places on a sphere's surface when measured along the sphere's surface rather than in a direct line through the centre of the sphere. The length of a straight line connecting any two points in Euclidean space represents their distance, however there are no straight lines on the sphere. Geodesics are used in place of straight lines in curved spaces. Geodesics on the sphere are referred to as "great circles" and are circles on the sphere whose centres match the sphere's center.This function allows user to take 2 inputs as source and destination and calculate the Great Circle distance in kilometers.

# Explanation of Modules (2)

- Geodesic Distance Calculator:The number of edges in the shortest path (also known as a graph geodesic) between any two points in a graph is the geodesic distance between them. This function allows user to take 2 inputs as source and destination and calculate the Geodesic distance in kilometers.
- Server Configuration: This function accepts 2 locations as origin and destination ,calculates it's coordinates in latitude-longitude forms and configures data in our backend code and finally opens a new tab showing the best possible route,along with total time (depending on realtime traffic conditions) to reach destination,the program also shows time which would take at charging stations so that the EV completes its journey.
- Our python script interacts with our server-side javascript code by changing location details and opening a browser tab. The server-side code structure is as follows:
- We create a route function which accepts origin-destination data along with the amount of charge in the EV (in percentage).Inside the function we define a embedded json-array which would contain pre-defined data on destinations based in India which contains ID (primary-key),value(short form of destination),full name of destination and latitude-longitude coordinates.If the API detects a proper route based on information gathered,it will show the timings,route directions and steps to take on a separate panel on the browser tab else it would return result as 'No Route Available.'

# Explanation of Modules (3)

- To implement the functions we have designed, we will need 4 elements: a map, an initial route planner, a database of charging stations and finally a component that takes care of combining the route provided with the position of the columns to generate the route
- This allows to monitor EV status and control it remotely with any stops.
- The user on interacting with the command line based python script will provide information and invoke a path request which will involve the functions responsible for finding and generating the paths. The user will enter the origin point and the destination point and may choose charge capacity from the dropdown menu.
- The user should ensure that the locations are valid and see that the processed information is being sent correctly through OpenStreetMap and Google Geocoding API to convert addresses into coordinates and are successfully being sent into the server code.The received coordinates are then provided to OpenStreetMap to obtain the route between the 2 points.
- The route obtained from the data generates the definitive route and the result is presented to the user in satellite view,map view and realtime traffic view.The code won’t work if the 2 points are separated from each other by any water body or absence of road routes.

# Explanation of Modules (4)

- We need to route a group of electric vehicles that may need to be charged while travelling from one place to another. Let's say there are n cities (v1, v2,..., vn) and eij is the distance between vj and vi (if two cities are not directly connected, eij = and eij = eji). Assume that there is only one charging station per city, and that it can only charge one EV at a time. Consider a collection of k EVs, such as P1, P2,..., Pk. The following details are provided for each EV:
- Sr - source node
- Dr - destination node
- Br - battery charge status initially
- cr - battery charging rate at charging station (percent charged per unit time)
- dr - battery discharge rate while moving (distance travel per unit charge)Maximum battery capacity
- ats - average trip speed (distance per unit time).
- Assume that all vehicles leave at time t = 0 and arrive at their destinations at time t = Tr. In order to decrease maxTr, all of the vehicles must be routed from their respective sources to destinations.
- Path Taken: Each element I represents the route that a particular vehicle took to get where it was going.
- Battery Charge Matrix: The percentage of the car i's battery that is charged at node j's.
- Timing charges Starts...: Each element [i][j] denotes the beginning of the charging process for vehicle I at node "j."Each element [i][j] denotes the moment at which the vehicle I stops being charged at node [j].Time when Car I The individual time stamps for each element list represent the moment that vehicle I exits node "j." Since there is no charging or waiting at the destination, the departure time indicates the arrival time.

# Explanation of Modules (5)

**Parking System Algorithm**

Step 1:Create a max heap to represent the parking lot, where the key of each node is the departure time of the vehicle and the value is the vehicle's license plate number.

Step 2:When a vehicle arrives, assign them a departure time and insert their license plate number and departure time into the heap.

Step 3:When a vehicle leaves, remove the node with the maximum departure time from the heap.

Step 4:If the heap becomes empty, assign the next arriving vehicle the earliest departure time available.

Step 5:If a vehicle wants to leave before their assigned departure time, remove the node with the license plate number of that vehicle from the heap and assign the next vehicle with the earliest departure time available.

Step 6:Repeat steps 2-5 until all vehicles have left the parking lot.

**Routing Algorithm**

Step 1:Receive the starting location, ending location, and current state of charge (SOC) of the EV from the user.

Step 2:Use the EV's battery capacity and energy consumption data to calculate the maximum distance the EV can travel on its current SOC.

Step 3:Query a database of charging stations to find all stations within the EV's maximum range and their availability status.

Step 4:Generate multiple routes from the starting location to the destination, taking into account the locations and availability of charging stations.

Step 5:Evaluate each route based on factors such as distance, duration, and charging time, and select the most efficient route.

Step 6:Return the selected route to the user, along with information about the charging stations on the route, such as their location, availability, and charging times.

Step 7:Allow the user to specify additional preferences such as avoiding tolls or ferries, minimizing the amount of time spent charging, or maximizing the use of fast-charging stations.

Step 8:Continuously updating the algorithm by using machine learning techniques, traffic data, and weather forecast to improve the routing.

# What is to be completed?

- Provide test cases for route generation.
- Rectify and add content in research paper.
- Attempt to print paper in reputed journal or conference paper.

# Guide Approval Snapshot

# *Any other additional information to be added by Guide*

# References


- [1] IEA (2019), Global EV Outlook 2019, IEA, Paris https://www.iea.org/reports/global-ev-outlook-2019, License: CC BY 4.0
- [2] Electric Vehicle Industry in India might continue to draw investments https://www2.deloitte.com/in/en/pages/finance/articles/ev-covid.html
- [3] E-MOBILITY: NATIONAL MISSION ON TRANSFORMATIVE MOBILITY AND BATTERY STORAGE https://www.niti.gov.in/e-mobility-national-mission-transformative-mobility-and-battery-storage
- [4] Faster adoption of electric vehicles https://www.teriin.org/sites/default/files/2019-11/Faster%20adoption%20of%20electric%20vehicles%20in%20India
- [5] Corona-Gutiérrez, K., Nucamendi-Guillén, S., & Lalla-Ruiz, E. (2022, March 5). Vehicle routing with cumulative objectives: A State of the art and analysis. Computers & Industrial Engineering.

# References [Contd.]


- [6] Electric-Vehicle Routing Planning Based on the Law of Electric Energy Consumption Nan Ding 1 , Jingshuai Yang 1,*, Zhibin Han 1 and Jianming Hao

- [7] Foa, S. (2022, July 30). Solving the vehicle routing problem with deep reinforcement learning. arXiv.org. https://arxiv.org/abs/2208.00202

- [8] Rajesh, K. (2022, August 26). A Multi-Objective approach to the Electric Vehicle Routing Problem. arXiv.org. https://arxiv.org/abs/2208.12440

- [9] Qin, Hu & Su, Xinxin & Ren, Teng & Luo, Zhixing. (2021). A review on the electric vehicle routing problems: Variants and algorithms. Frontiers of Engineering Management. 8. 10.1007/s42524-021-0157-1.

# Thank You