International Islamic University Chittagong

Department of Computer Science and Engineering

EazyTrip

Instructor

Prof. Mohammed Shamsul Alam

Dept. of CSE, IIUC

Team Members

1.Mohammad Hasan_C231005

2.Shanjid Mahammad_C231007

3.Prasenjit Chowdhury_C231028

Easy Trip: A Transport System

Introduction:-

The Easy Trip project is a modern transport system similar to Uber and Pathao, designed to provide efficient and user-friendly transportation services. The system includes various user roles such as Admin, Rider, and User, each with distinct functionalities. The core objective of Easy Trip is to offer a seamless travel experience by leveraging custom maps, shortest path algorithms, and a robust database to manage rides and user interactions.

Data Structure Implementation:-

1. **Graph Representation:-**

Purpose: Represents the transport network consisting of nodes (locations) and edges (routes) between them.

Implementation: Typically implemented using an adjacency list or adjacency matrix.

Adjacency List: Efficient for sparse graphs where each node stores a list of adjacent nodes and possibly edge weights.

Adjacency Matrix: Suitable for dense graphs where each cell represents the presence of an edge between two nodes and may store edge weights.

2. **Custom Map Representation:-**

Purpose: Visual representation of the transport network for user interface (UI) and route planning.

Implementation: Utilizes graphical elements (like maps from third-party providers or custom graphics libraries) to display nodes (locations) and edges (routes) with geographical context.

Mapping APIs: Integration with mapping APIs (e.g., Google Maps API) for displaying real-world locations and routes.

Custom Graphics: Creation of custom maps using graphical libraries (e.g., Leaflet.js, Mapbox) to render nodes and edges according to project-specific requirements.

Algorithms Used:

1. Dijkstra's Algorithm:-

Purpose: Finding the shortest path between two nodes (locations) on the transport network.

Implementation:

Algorithm: Utilizes a priority queue (min-heap) to efficiently retrieve the node with the smallest distance from the source node and updates the shortest path distances to neighboring nodes.

Application: Enables route planning for ride bookings by calculating the optimal path based on distance or travel time.

2. User Authentication and Data Management:

Purpose: Secure management of user and administrative data within the application.

Implementation:

Database: Uses a relational database (e.g., MySQL, PostgreSQL) for persistent storage of user profiles, ride history, administrative logs, and network data.

Encryption: Ensures passwords are securely stored using hashing algorithms (e.g., bcrypt) to protect user credentials.

3. **Dynamic Pricing Algorithms:-**

Purpose: Calculates fares based on various factors such as distance traveled, time of day, and demand-supply dynamics.

Implementation:

Pricing Rules: Configures pricing rules and algorithms within the application logic to dynamically adjust fares in response to real-time conditions.

Fairness: Ensures transparent and fair pricing for users and riders based on the service provided and market conditions.

Features:

Admin Features:-

- **Fixed Admin Login**: Admins have a fixed login for security and management purposes.
- **Node Management**: Admins can manage a predefined set of 9 nodes on a custom map.
- **Edge Management**: Admins can add edges between nodes to define possible routes.

- **Monitoring**: Admins have the ability to oversee all activities, including user and rider interactions.

User Features:-

- **Sign Up and Login**: New users can sign up, and existing users can log in to access their profiles.
- **Profile Management**: Users can view and update their profiles.
- **Ride Booking**: Users can book rides by providing a destination, after which the system calculates the shortest path and the fare based on distance.
- **Fare Calculation**: The system provides the fare for the selected route, allowing users to confirm the booking.
- **Booking Confirmation**: Once a ride is confirmed, the details are stored in the database.

Rider Features:-

- **Sign Up and Login:** Riders can sign up and log in to access their accounts.
- **Ride Acceptance:** Riders can view ride requests and choose to accept or decline them.

Implementation:

System Architecture:

The Easy Trip system is designed with a multi-layer architecture to ensure scalability, maintainability, and security. The key components include:

1. **User Interface (UI)**: Interfaces for Admin, User, and Rider to interact with the system.

2. **Backend Services**: Handles business logic, shortest path calculations, and database interactions.

3. **Database**: Stores user profiles, ride details, and node/edge configurations.

Custom Map and Shortest Path Algorithm:-

- **Custom Map**: The system uses a custom map with 9 fixed nodes. Admins add edges to define routes between these nodes.

Graph-Based Pathfinding: The shortest path between nodes is calculated using graph algorithms, providing users with the most efficient route and corresponding fare.

Database Management:-

The database is designed to store all necessary information securely and efficiently:

User Information: Stores details of all users and riders.

Ride Details: Logs all ride requests, confirmations, and completions.

Node and Edge Data: Maintains the structure of the custom map for route calculations.

Future Plan/Possible Extensions:-

Dynamic Mapping: Integrate dynamic mapping capabilities to expand beyond the fixed 9-node system, allowing for a more extensive and flexible network.

Real-Time Traffic Updates: Implement real-time traffic data integration to optimize route calculations and provide more accurate fare estimates.

Advanced Analytics: Develop analytics tools for admins to better understand user behavior, ride patterns, and system performance.

Enhanced Security Features: Introduce advanced security measures, including multi-factor authentication and encryption of sensitive data.

Additional Payment Options: Expand the range of payment methods to include digital wallets and other modern payment systems.

User Feedback System: Implement a feedback system for users and riders to provide reviews and ratings, improving service quality.

Functions Used- Short Description:-

1. **User Authentication Functions:-**

register_user(username, password, email): Registers a new user with username, password, and email.

login_user(username, password): Authenticates user credentials and starts a session.

update_profile(user_id, new_data): Updates user profile information like name, phone number, etc.

view_booking_history(user_id): Retrieves past ride bookings for a specific user.

2. Ride Booking Functions:-

calculate_shortest_path(source, destination): Utilizes graph-based algorithms (like Dijkstra's) to find the shortest path between source and destination nodes on the map.

estimate_fare(distance): Calculates the fare estimate based on distance using predefined pricing rules.

book_ride(user_id, source, destination): Initiates a ride booking for a user from source to destination, including updating the database with ride details.

3. Rider Functions:-

register_rider(username, password, email): Registers a new rider with username, password, and email.

login_rider(username, password): Authenticates rider credentials and starts a session.

view_ride_requests(rider_id): Retrieves pending ride requests for a specific rider.

accept_ride_request(rider_id, ride_id): Accepts a ride request identified by ride_id for a specific rider.

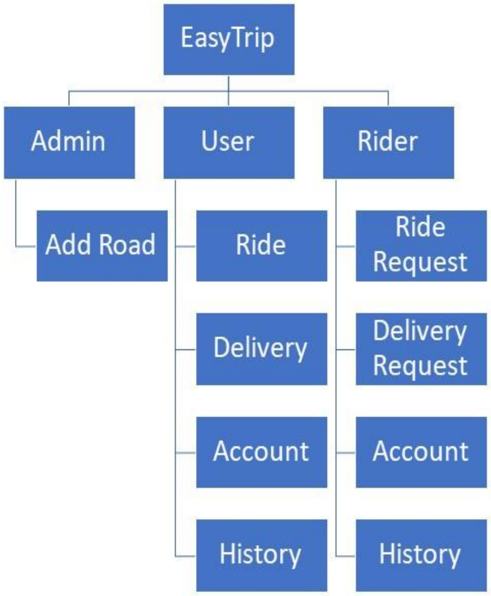
4. Admin Functions:-

admin_login(username, password): Authenticates admin credentials and starts an administrative session.

add_edge(node1, node2): Adds an edge between two nodes on the transport network map, allowing new routes.

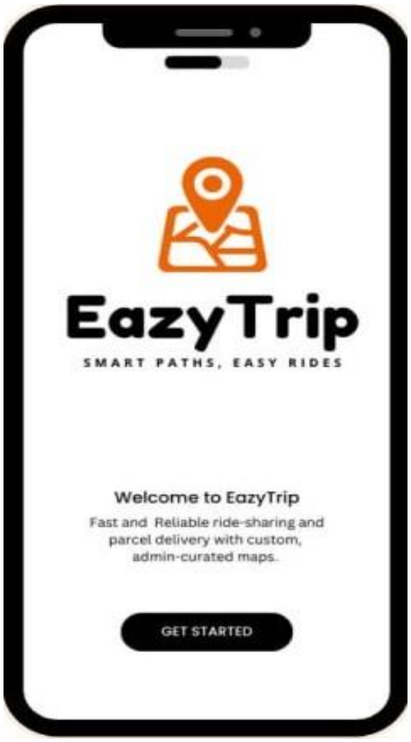
view_system_logs(): Retrieves logs of system activities for administrative review and monitoring.

FLOWCHART



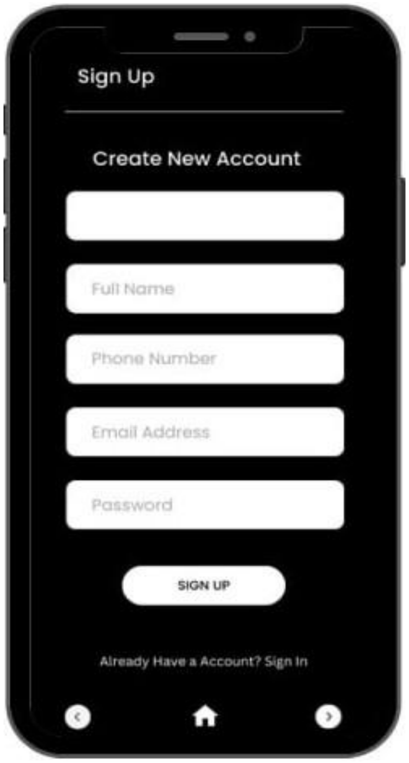
Screenshots of the Project

Admin:-



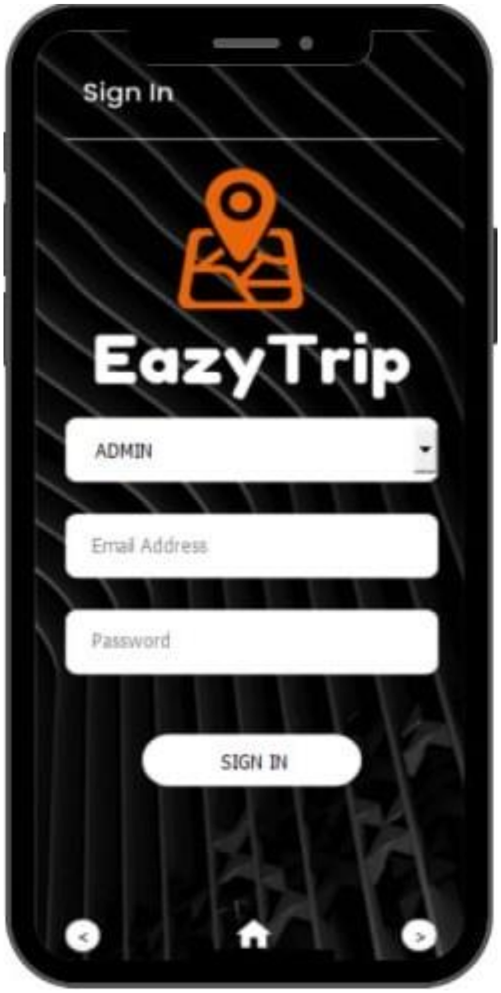
Welcome to EazyTrip

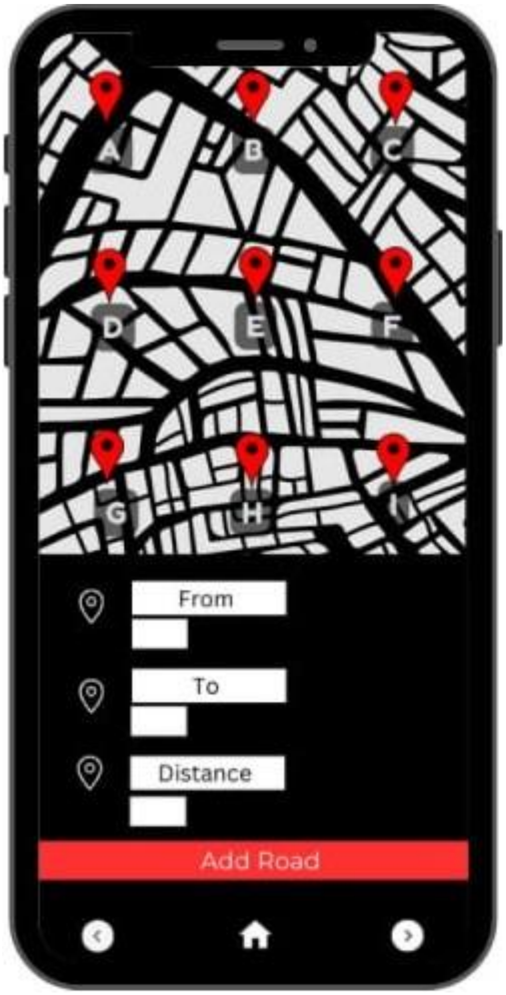
Fast and Reliable ride-sharing and parcel delivery with custom, admin-curated maps.



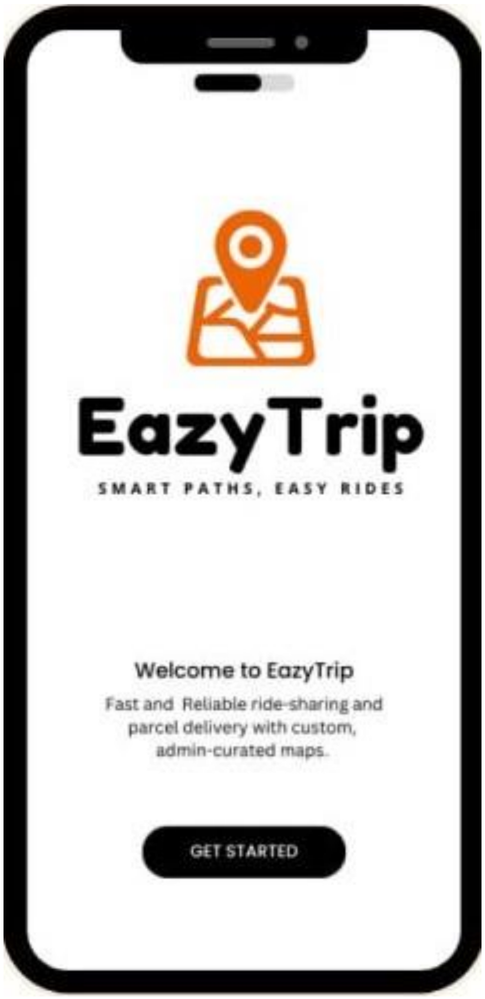
Create New Account

Already Have a Account? Sign In


Sign In
EazyTrip
ADMIN
Email Address
Password
SIGN IN

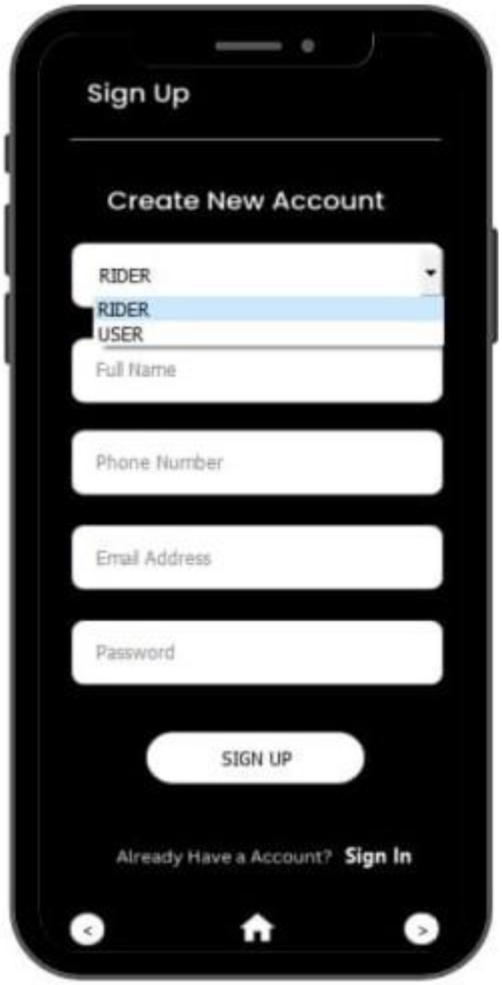

A
B
C
D
E
F
G
H
I
From
To
Distance
Add Road

USER:-

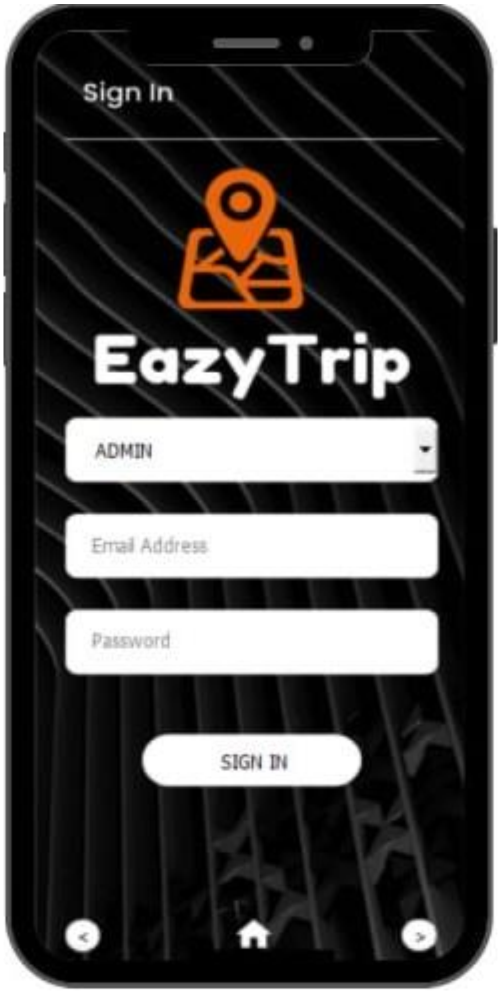


Welcome to EazyTrip

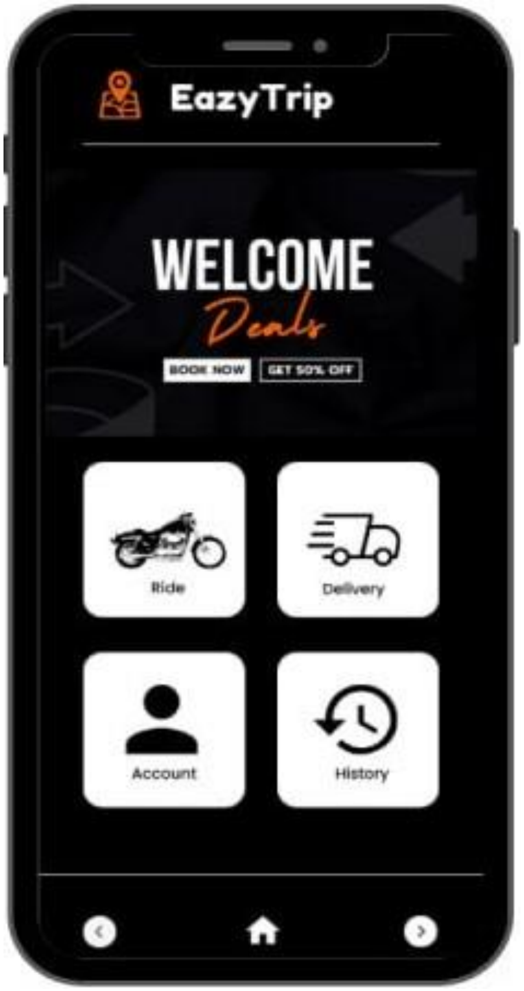
Fast and Reliable ride-sharing and parcel delivery with custom, admin-curated maps.



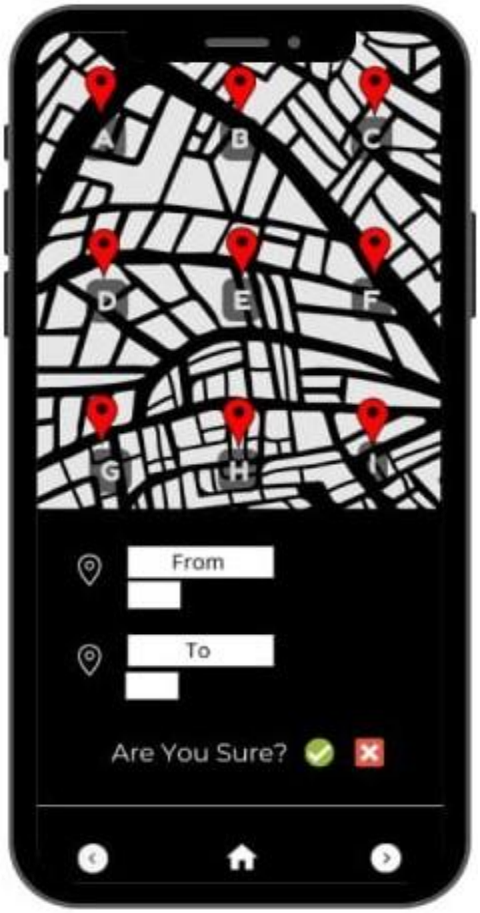
RIDER

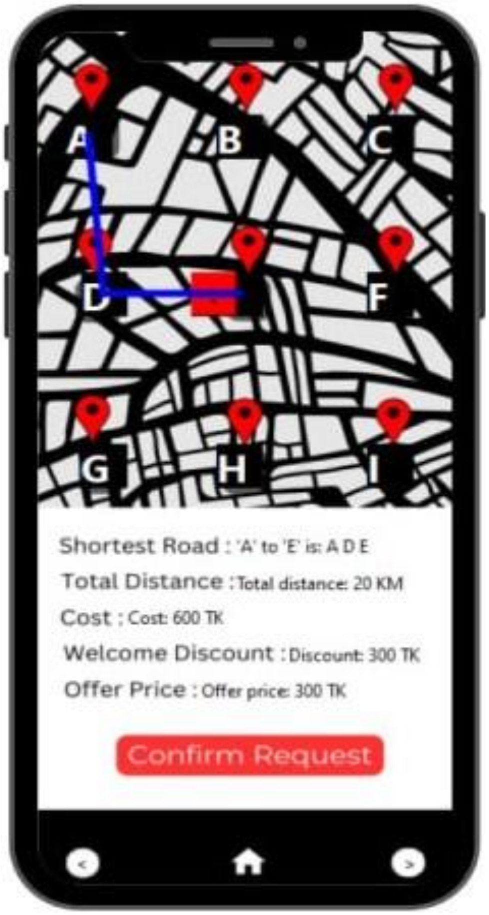


Sign In
EazyTrip
ADMIN
Email Address
Password
SIGN IN



EazyTrip
WELCOME
Deals
BOOK NOW
GET 50% OFF
Ride
Delivery
Account
History





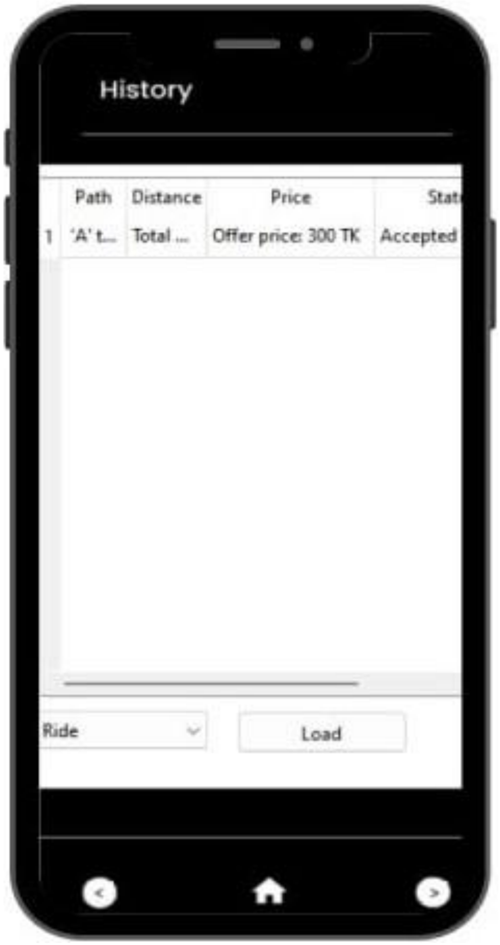
Shortest Road : 'A' to 'E' is: A D E

Total Distance : Total distance: 20 KM

Cost : Cost: 600 TK

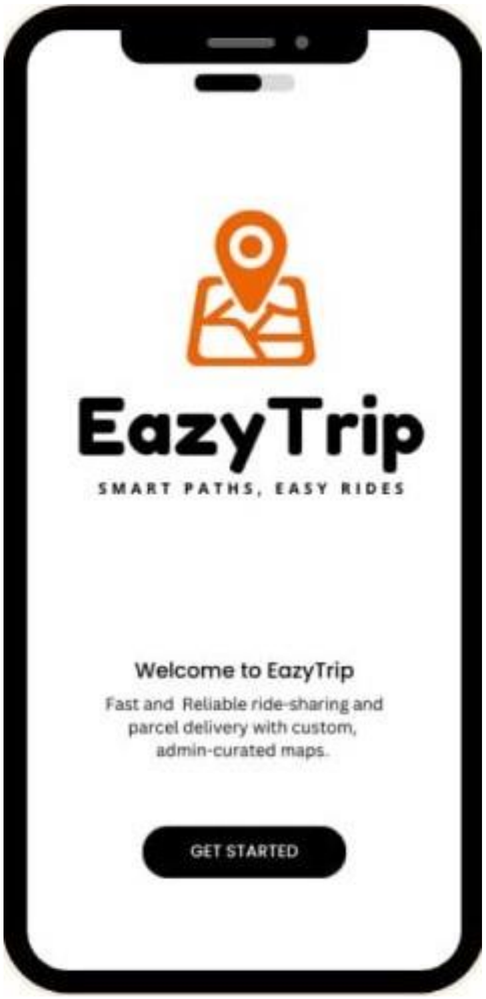
Welcome Discount : Discount: 300 TK

Offer Price : Offer price: 300 TK



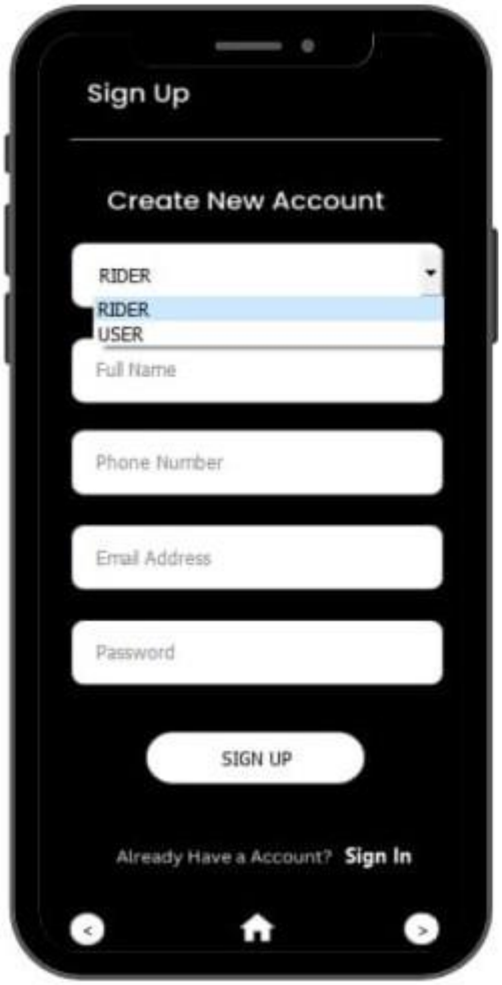
	Path	Distance	Price	Stat
1	'A' t...	Total ...	Offer price: 300 TK	Accepted

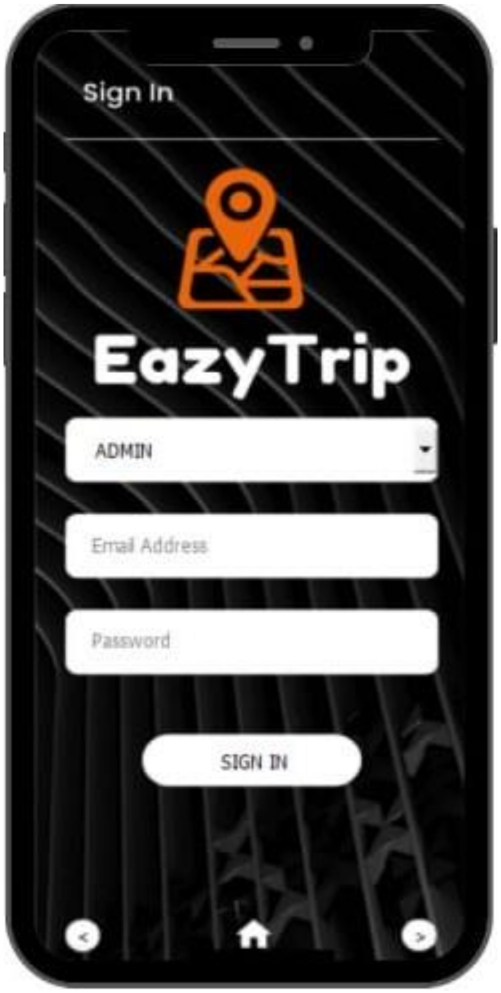
Rider:-

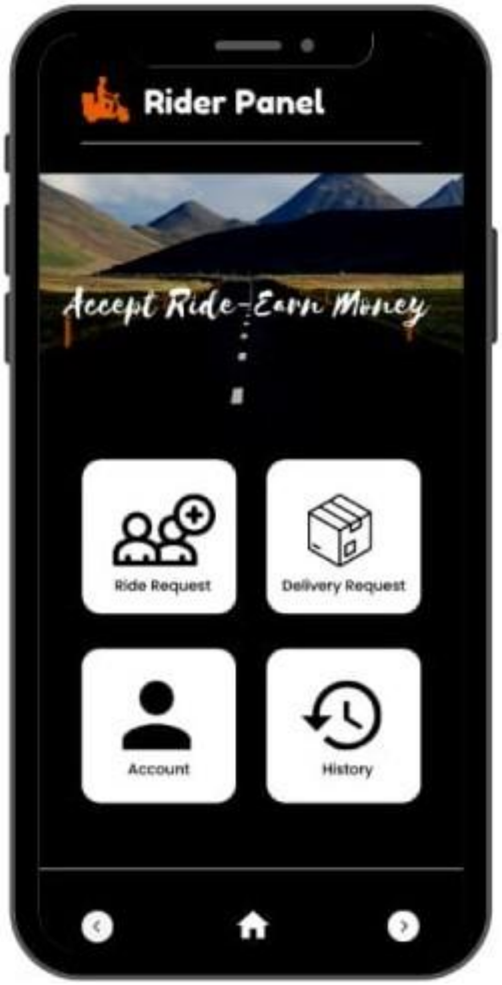


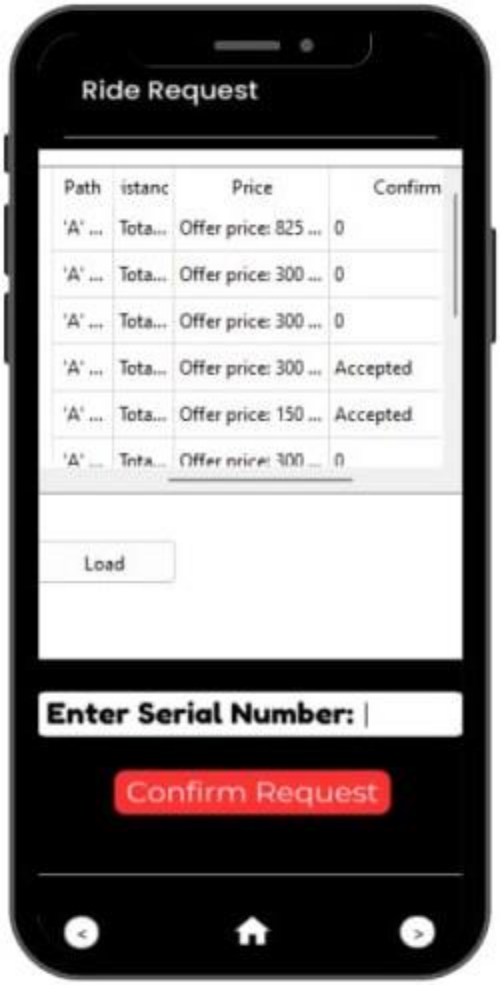
Welcome to EazyTrip

Fast and Reliable ride-sharing and parcel delivery with custom, admin-curated maps.




Sign In
EazyTrip
ADMIN
Email Address
Password
SIGN IN

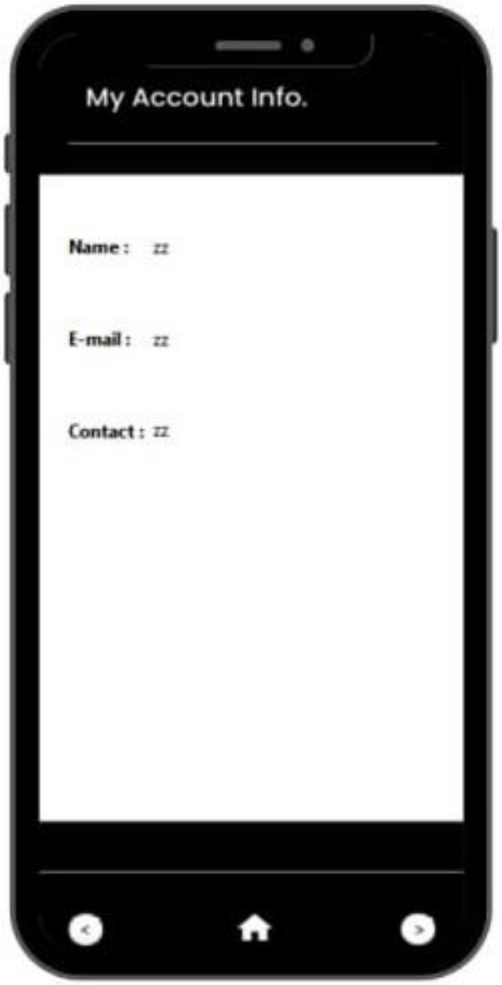

Rider Panel
Accept Ride-Earn Money
Ride Request
Delivery Request
Account
History



Ride Request

Path	istanc	Price	Confirm
'A' ...	Tota...	Offer price: 825 ...	0
'A' ...	Tota...	Offer price: 300 ...	0
'A' ...	Tota...	Offer price: 300 ...	0
'A' ...	Tota...	Offer price: 300 ...	Accepted
'A' ...	Tota...	Offer price: 150 ...	Accepted
'A' ...	Tota...	Offer price: 300 ...	0

Enter Serial Number: |



My Account Info.

Name : zz

E-mail : zz

Contact : zz

Conclustion:-

The Easy Trip project is a comprehensive transport system that simplifies travel planning and execution for users while providing robust management tools for admins and flexible options for riders. By leveraging custom maps and efficient algorithms, Easy Trip offers a reliable and user-friendly transportation service. Future enhancements will further improve system capabilities and user experience, making Easy Trip a competitive choice in the modern transport market.