Transportation Company

User class:

Constructor:

User(String username, String password):-

Constructor method that initializes a User object with the provided username and password.

register(): Registers a new user by writing their username and password to a file :-

Method to register a new user by writing their username and password to a file. Returns true if the registration is successful, false otherwise.

login(String username, String password): Static method that checks if the provided username and password match any existing user in the system:-

Static method that checks if the provided username and password match any existing user in the system by reading from a file. Returns true if the login is successful, false otherwise.

readIDFromFile(): Static method that reads the last used user ID from a file:-

Static method that reads the last used user ID from a file (id.txt) and returns it.

updateID(int id): Static method that updates the last used user ID in the file:-

Static method that updates the last used user ID in the file (id.txt) with the provided value.

usernameExists(String username): Private method that checks if a given username already exists in the system:-

Private method that checks if a given username already exists in the system by reading from a file (users.txt). Returns true if the username exists, false otherwise.

Passenger class (extends User):

Constructors:

Passenger(String username, String password):-

Constructor method that initializes a Passenger object with the provided username and password.

Passenger(String username, String password, int id):-

Constructor method that initializes a Passenger object with the provided username, password, and ID.

saveToFile(): Method to save the passenger's information (ID, username, password) to a file (passenger.txt).

bookTicket(String tripname): Books a trip for the passenger.

cancelBookedTicket(String username): Cancels the last booked trip for the passenger.

displayPassenger(): Static method that displays the information of all passengers stored in a file.

reviewBookedTrips(String username): Displays the booked trips for the passenger with the given username.

Employee class (extends User):

Constructors: Various constructors for initializing Employee objects with different parameters.

Employee(String username)

Employee(String username, String password, long idCounter)

Employee(String username, String password)

Employee(String username, String password, String type)

Employee(String username, String password, String type, ArrayList<Employee> employees)

viewBasicInfo(): Method to return a string representation of the employee's basic information, including username, ID, and type.

Driver class (extends Employee):

Constructors:

Various constructors for initializing Driver objects with different parameters.

Driver(String username, String password, int driverID, String phoneNumber, String carType)

Driver(String username, String password, long id, vehicle vehicle, ArrayList<Trip> assignedTrips, ArrayList<Driver> drivers)

Driver(int driverID, vehicle vehicle, ArrayList<Trip> assignedTrips, String carType, String filename, String phoneNumber)

Driver(String username, String password)

Driver(String username, String password, long idCounter)

saveToFile(): Method to save the driver's information (ID, username, password, car type) to a file (driver.txt).

driverDetails(): Method to return a string representation of the driver's details, including username, ID, vehicle, and assigned trips.

displayDriver(): Static method that displays the information of all drivers stored in the driver.txt file

Manager class (extends Employee):

Methods:

Various methods for managing trips, vehicles, drivers, and employees, including saving to files, adding and removing entities, generating reports, and canceling trips.

1. saveToFile(): Saves the manager's information to a file.
2. removeTrips(Trip trip): Removes a trip from the list of trips.
3. addTrips(Trip trip): Adds a trip to the list of trips.
4. reviewTrips(ArrayList<Trip> trips): Reviews the list of trips.
5. addVehicles(vehicle vehicle): Adds a vehicle to the list of vehicles.
6. removeVehicles(vehicle vehicle): Removes a vehicle from the list of vehicles.
7. addDriver(Driver driver): Adds a driver to the list of employees.
8. removeDriver(Driver driver): Removes a driver from the list of employees.
9. generateEmployeeReport(): Generates a report of employees.
10. displayManager(): Static method that displays the information of all managers stored in a file.
11. AssignTripToDriver(int id, String Name): Assigns a trip to a driver based on ID.
12. AssignDriver(String driverName, String trip): Assigns a driver to a trip.
13. addTrip(Trip trip): Adds a trip to the list of trips.
14. addEmployee(String username, String password, String Phone Number, String carType): Adds an employee to the list of employees.
15. addVehicle(vehicle vehicle): Adds a vehicle to the list of vehicles.
16. cancelTrip(String tripName): Cancels a trip from the list of trips.

Vehicle class:

Constructors:

Various constructors for initializing Vehicle objects with different parameters.

vehicle(String type, int capacity, String licensePlate)

vehicle(String type, int capacity, String licensePlate, ArrayList<vehicle> vehicles)

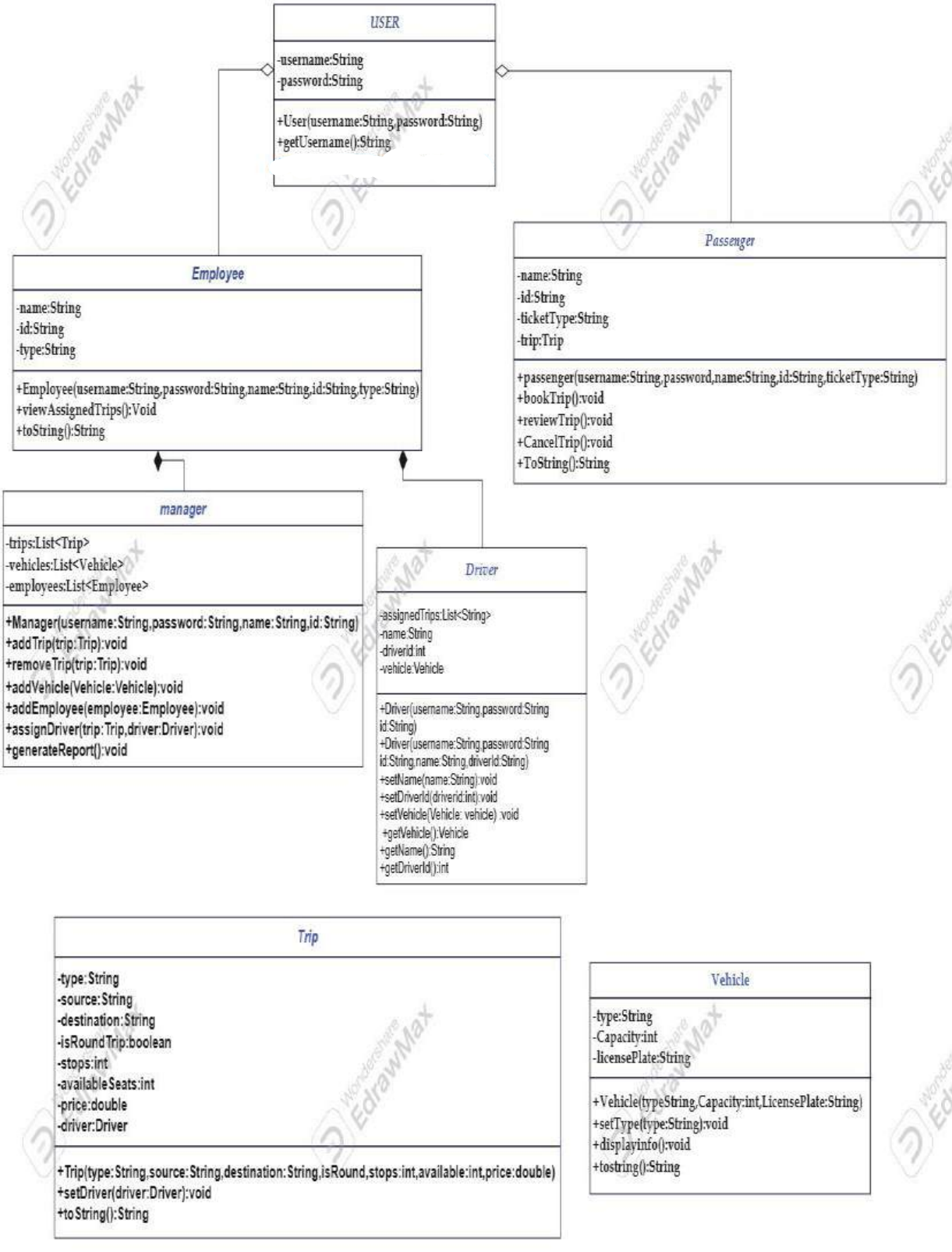
addVehicle(vehicle vehicle): Method to add a vehicle to the list of vehicles.

removeVehicle(vehicle vehicle): Method to remove a vehicle from the list of vehicles.

vehicleInfo(): Method to return a string representation of the vehicle's information, including type, capacity, and license plate.

saveToFile(): Method to save the vehicle's information (type, capacity, license plate) to a file (vehicle.txt).

displayVehicle(): Static method that displays the information of all vehicles stored in the vehicle.txt file.


USER
-username:String
-password:String
+User(username:String,password:String)
+getUsername():String
Employee
-name:String
-id:String
-type:String
+Employee(username:String,password:String,name:String,id:String,type:String)
+viewAssignedTrips():Void
+toString():String
Passenger
-name:String
-id:String
-ticketType:String
-trip:Trip
+passenger(username:String,password,name:String,id:String,ticketType:String)
+bookTrip():void
+reviewTrip():void
+CancelTrip():void
+ToString():String
manager
-trips:List<Trip>
-vehicles:List<Vehicle>
-employees:List<Employee>
+Manager(username:String,password:String,name:String,id:String)
+addTrip(trip:Trip):void
+removeTrip(trip:Trip):void
+addVehicle(Vehicle:Vehicle):void
+addEmployee(employee:Employee):void
+assignDriver(trip:Trip,driver:Driver):void
+generateReport():void
Driver
-assignedTrips:List<String>
-name:String
-driverId:int
-vehicle:Vehicle
+Driver(username:String,password:String
id:String)
+Driver(username:String,password:String
id:String,name:String,driverId:String)
+setName(name:String):void
+setDriverId(driverid:int):void
+setVehicle(Vehicle: vehicle) :void
+getVehicle():Vehicle
+getName():String
+getDriverId():int
Trip
-type:String
-source:String
-destination:String
-isRoundTrip:boolean
-stops:int
-availableSeats:int
-price:double
-driver:Driver
+Trip(type:String,source:String,destination:String,isRound,stops:int,available:int,price:double)
+setDriver(driver:Driver):void
+toString():String
Vehicle
-type:String
-Capacity:int
-licensePlate:String
+Vehicle(typeString,Capacity:int,LicensePlate:String)
+setType(type:String):void
+displayinfo():void
+tostring():String

User Management:

- Users can register with a username and password.
- Registered users can log in to the system.
- Passenger Management:
- Passengers are a type of user.
- Passengers can book and cancel trips.
- Information about passengers is stored in a file (passenger.txt).

Employee Management:

- Employees are also users.
- Different types of employees include drivers and managers.
- Employees have basic information such as username, ID, and type.
- Managers have additional functionalities like managing trips, vehicles, and employees.

Driver Management:

- Drivers are a type of employee.
- They have details such as username, ID, vehicle, and assigned trips.
- Information about drivers is stored in a file (driver.txt)

Manager Management:

- Managers are a type of employee with advanced privileges.
- They can manage trips, vehicles, drivers, and employees.
- Managers can add or remove trips, vehicles, drivers, and employees.
- They can generate reports on employees.

Vehicle Management:

- Vehicles have attributes like type, capacity, and license plate.
- Vehicles can be added to or removed from the system.
- Information about vehicles is stored in a file (vehicle.txt).

Trip Management:

- Trips can be booked by passengers.
- Trips can be assigned to drivers by managers.
- Managers can add, remove, or cancel trips.

Overall, the system provides functionalities for managing users, passengers, employees, vehicles, and trips efficiently. It utilizes file storage to maintain data persistence across sessions. The system allows for user authentication, registration, booking trips, managing employees and vehicles, and generating reports for managerial purposes.

NAME:- Adel Sobhy Abdelaziz

ID:-23101870

NAME:- Marwan Tamer Mohamed

ID:-23101477

NAME:- Youssef Mohamed Moharm

ID:-23101517