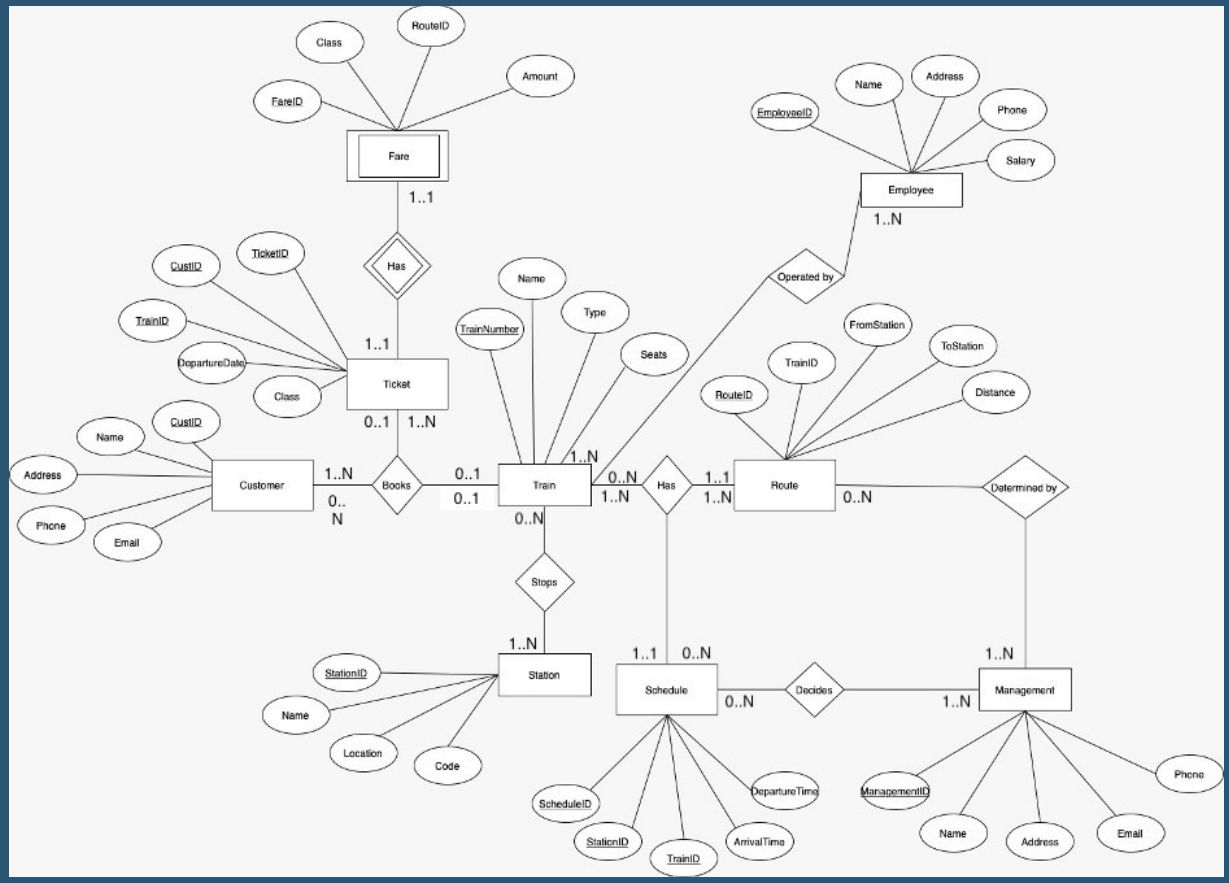
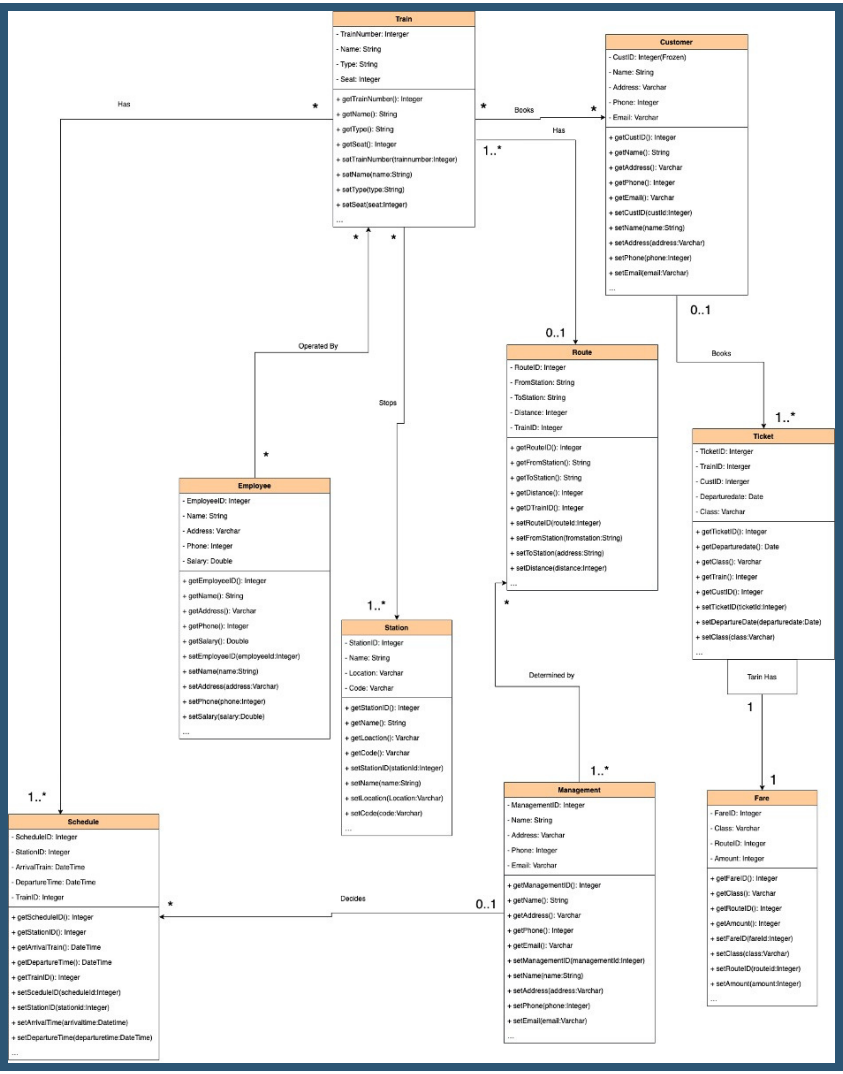
# Problem Statement

**The railway reservation system is important for resource efficiency and passenger flow control. However, the current manual system has issues like long wait times, ticket shortages, and lack of transparency. We propose a project to automate the reservation process and provide a smooth experience for passengers. We'll use the latest technology and best practices for database management in a centralized database system. The aim is to develop a reliable and effective reservation system that enhances customer satisfaction, streamlines business processes, and boosts the railway company's reputation.**

# EER Diagram


Class
RouteID
Amount
FareID
Fare
1..1
Has
CustID
TicketID
TrainID
DepartureDate
Class
Ticket
1..1
0..1
1..N
Name
CustID
Address
Phone
Email
Customer
1..N
0..
N
Books
0..1
0..1
Train
1..N
0..N
1..N
Has
1..1
1..N
Route
0..N
Determined by
Name
Type
TrainNumber
Seats
0..N
Stops
1..N
StationID
Name
Location
Code
Station
1..1
0..N
Schedule
0..N
Decides
1..N
Management
1..N
EmployeeID
Name
Address
Phone
Salary
Employee
1..N
Operated by
FromStation
ToStation
TrainID
RouteID
Distance
ScheduleID
StationID
TrainID
ArrivalTime
DepartureTime
ManagementID
Name
Address
Email
Phone

# UML Diagram


Train
- TrainNumber: Interger
- Name: String
- Type: String
- Seat: Integer
+ getTrainNumber(): Integer
+ getName(): String
+ getType(): String
+ getSeat(): Integer
+ setTrainNumber(trainnumber:Integer)
+ setName(name:String)
+ setType(type:String)
+ setSeat(seat:Integer)
...
Customer
- CustID: Integer(Frozen)
- Name: String
- Address: Varchar
- Phone: Integer
- Email: Varchar
+ getCustID(): Integer
+ getName(): String
+ getAddress(): Varchar
+ getPhone(): Integer
+ getEmail(): Varchar
+ setCustID(custId:Integer)
+ setName(name:String)
+ setAddress(address:Varchar)
+ setPhone(phone:Integer)
+ setEmail(email:Varchar)
...
Route
- RouteID: Integer
- FromStation: String
- ToStation: String
- Distance: Integer
- TrainID: Integer
+ getRouteID(): Integer
+ getFromStation(): String
+ getToStation(): String
+ getDistance(): Integer
+ getDTrainID(): Integer
+ setRouteID(routeId:Integer)
+ setFromStation(fromstation:String)
+ setToStation(address:String)
+ setDistance(distance:Integer)
...
Ticket
- TicketID: Interger
- TrainID: Interger
- CustID: Interger
- Departuredate: Date
- Class: Varchar
+ getTicketID(): Integer
+ getDepturedate(): Date
+ getClass(): Varchar
+ getTrain(): Integer
+ getCustID(): Integer
+ setTicketID(ticketId:Integer)
+ setDepartureDate(departuredate:Date)
+ setClass(class:Varchar)
...
Employee
- EmployeeID: Integer
- Name: String
- Address: Varchar
- Phone: Integer
- Salary: Double
+ getEmployeeID(): Integer
+ getName(): String
+ getAddress(): Varchar
+ getPhone(): Integer
+ getSalary(): Double
+ setEmployeeID(employeeId:Integer)
+ setName(name:String)
+ setAddress(address:Varchar)
+ setPhone(phone:Integer)
+ setSalary(salary:Double)
...
Station
- StationID: Integer
- Name: String
- Location: Varchar
- Code: Varchar
+ getStationID(): Integer
+ getName(): String
+ getLocation(): Varchar
+ getCode(): Varchar
+ setStationID(stationId:Integer)
+ setName(name:String)
+ setLocation(Location:Varchar)
+ setCode(code:Varchar)
...
Schedule
- ScheduleID: Integer
- StationID: Integer
- ArrivalTrain: DateTime
- DepartureTime: DateTime
- TrainID: Integer
+ getScheduleID(): Integer
+ getStationID(): Integer
+ getArrivalTrain(): DateTime
+ getDepartureTime(): DateTime
+ getTrainID(): Integer
+ setScheduleID(scheduleId:Integer)
+ setStationID(stationid:Integer)
+ setArrivalTime(arrivaltime:Datetime)
+ setDepartureTime(departuretime:DateTime)
...
Management
- ManagementID: Integer
- Name: String
- Address: Varchar
- Phone: Integer
- Email: Varchar
+ getManagementID(): Integer
+ getName(): String
+ getAddress(): Varchar
+ getPhone(): Integer
+ getEmail(): Varchar
+ setManagementID(managementId:Integer)
+ setName(name:String)
+ setAddress(address:Varchar)
+ setPhone(phone:Integer)
+ setEmail(email:Varchar)
...
Fare
- FareID: Integer
- Class: Varchar
- RouteID: Integer
- Amount: Integer
+ getFareID(): Integer
+ getClass(): Varchar
+ getRouteID(): Integer
+ getAmount(): Integer
+ setFareID(fareId:Integer)
+ setClass(class:Varchar)
+ setRouteID(routeId:Integer)
+ setAmount(amount:Integer)
...
Has
Books
Has
Operated By
Stops
Books
Determined by
Tarin Has
Decides
1..*
0..1
1

# Scope of analytics

- Railway reservation system database can be leveraged to answer a variety of business problems, such as optimizing train schedules, forecasting demand for specific routes, analyzing passenger behavior patterns, and identifying areas for improvement in the reservation process.
- It can also enable railway companies to make data-driven decisions, such as improving customer experience, enhancing operational efficiency, and maximizing revenue.
- By leveraging analytics with a railway reservation system database, railway companies can gain valuable insights and make informed decisions that can positively impact their bottom line and overall performance.


Thank
You!