



# Project Report Phase 2

**ICS 321 – sec 01**

**Abdullah Almajed ~ 202167470**

**Mohammed Hejazi ~ 202182750**

**Mohammed Busaleh ~ 202158210**

# CONSTRAINTS AS SCHEMA

Legend: D = domain constraint ,K = key constraint,  
EI = Entity Integrity, Refl = Referential Integrity Constraint,  
NN = Not Null constraint.

- Train

TrainID	Arabic name	English name
K, EI, D(Alphanumeric e.g. HHR100)	NN, D(Arabic letters)	NN, D(English letters)

- Schedule

TrainID	time	seqNumber
Refl	K, EI, D(time)	D(int)

- Station

StName	ArName
K, EI, D(English letters)	NN, D(Arabic letters)

- Reservation

ResID	Trip	UserId	....
K, EI, D(Numeric)	NN, Refl	Refl, NN	...

....	luggage	Class	Status
...	NN, D(int)	NN, D(custom)	NN, D(custom)

- waitingList

TrainID	ResID	Status	time	ExpDate	Waiting-Posotion
Refl, K, EI	Refl, K, EI	NN, D(varchar)	K, EI, D(time)	NN, D(numeric)	D(int)

- Payment

PaymentID	Status	Amount
K, EI, D(Numeric)	NN, D(varchar)	NN, D(numeric)

- User

Id	Email	Role	Name
K, EI, D(Varchar)	NN, D(Varchar)	NN, D(text)	NN, D(text)

- Staff

StaffId	UserName	StaffName
K, EI, D(Numeric)	NN, D(varchar)	NN, D(varchar)

- RegPassenger

UserId	PassengerName	IdDoc
K, EI, D(Numeric)	NN, D(text)	D(varchar)

- Dependent

DepId	Relationship	PassengerName	PassengerId
K, EI, D(Numeric)	NN, D(custom)	NN, D(varchar)	Refl, NN

- Miles

UserId	Class	miles
K, EI, Refl	D(custom)	NN, D(numeric)

- Trip

TripID	From	To	TrainId	Time	....
K, EI, D(Numeric)	NN, Refl	Refl, NN	Refl, NN	NN, D(time)	...

....	Price	Miles
...	D(numeric)	D(numeric)

- Seat

SeatID	isReserved	Class	TripID	....
K, EI, D(int)	NN, D(bool)	NN, D(custom)	Refl, NN	...

....	ResId	SeatLocation
...	Refl	D(text)

# CONSTRAINTS AS BULLET POINTS

---

- Trains must be identifiable by a unique ID
- Trains must have names in both Arabic and English.
- Train schedules should include arrival times at each station on the route
- A passenger can only book seats for available trains by specifying details like reservation number, train, date, origin, destination, coach, seat number, and passenger name.
- Luggage details must also be provided, and an ID document is required for verification.
- Dependent/Family Member details are maintained for each passenger.
- A 25% discount applies to each dependent when traveling together.
- A loyalty system is in place, offering discounts based on miles traveled. Loyalty tiers include:
  - Green: 5% discount after 10K miles
  - Silver: 10% discount after 50K miles
  - Gold: 25% discount after 100K miles

- A payment gateway must be available for ticket transactions.
- A VAT of 15% must be applied on top of payments.
- A waiting list must be managed for passengers when seats are unavailable, with temporary reservations that can be finalized before a set date.
- Reservations canceled/expired after payment incur a 15% charge.
- The system should support the retrieval of passenger summaries for any train, waiting lists for specific trains on a given day, and train lists for a particular day.

## Assumptions:

- Each Train runs only once.
- There is only one track between each two stations.
- There is no cap on the luggage weight.

```

    erDiagram
        Train ||--o{ Reservation : Via
        Reservation ||--o{ Staff : "Create/modify/delete"
        Reservation ||--o{ Payment : Bought
        Reservation ||--o{ Staff : Refund
        Reservation ||--o{ Trip : Reserves
        Reservation ||--o{ Station : IsOn
        Reservation ||--o{ Trip : waitingFor
        Reservation ||--o{ WaitingList : addedTo
        Station ||--o{ Trip : Visit
        Station ||--o{ Trip : StartFrom
        Station ||--o{ Trip : EndIn
        Trip ||--o{ Seat : goingOn
        Trip ||--o{ Seat : In
        Trip ||--o{ RspPassenger : Has
        RspPassenger ||--o{ Dependent : DependedOn
        User ||--o{ Staff : "d"
        User ||--o{ RspPassenger : "d"
        User ||--o{ Dependent : "d"
        User ||--o{ Trip : "d"

        Train {
            string TrainID PK
            string EnglishName
            string ArabicName
        }
        Reservation {
            string ResID PK
            string Status
            string Class
            string Luggage
        }
        Staff {
            string StaffID PK
            string UserName
            string StaffName
        }
        User {
            string Id PK
            string Name
            string Role
            string Email
        }
        Station {
            string SName PK
            string AName
        }
        Trip {
            string TripID PK
            string Time
            float Price
        }
        WaitingList {
            string time PK
            string Status
            string ExpDate
            string WaitingPosition
        }
        Seat {
            string SeatID PK
            boolean IsReserved
            string Class
            string SeatLocation
        }
        Payment {
            string PaymentID PK
            string Status
            float Amount
            string VAT
        }
        RspPassenger {
            string IdDoc PK
            string UserID
            string PassengerName
        }
        Dependent {
            string DesID PK
            string Relationship
            string PassengerName
        }
  
```

```

    erDiagram
        Train ||--o{ Schedule : "has"
        Train ||--o{ Visit : "has"
        Train ||--o{ Reservation : "has"
        Train ||--o{ Staff : "has"
        Train ||--o{ RegPassenger : "has"
        Train ||--o{ Dependent : "has"
        Train ||--o{ Miles : "has"
        Train ||--o{ User : "has"
        Schedule ||--o{ Station : "has"
        Visit ||--o{ Station : "has"
        Reservation ||--o{ Station : "has"
        Reservation ||--o{ Payment : "has"
        Reservation ||--o{ Seat : "has"
        Reservation ||--o{ Trip : "has"
        Reservation ||--o{ User : "has"
        Station ||--o{ Payment : "has"
        Seat ||--o{ Trip : "has"
        Trip ||--o{ User : "has"

        Train {
            string TrainID PK
            string EnglishName
            string ArabicName
        }
        Schedule {
            string TrainID FK
            string TrainID PK
            int SeqNumber
        }
        Visit {
            string TrainID FK
            string StName FK
        }
        Reservation {
            string ResID PK
            string Class
            string luggage
            string Status
            string Id FK
            string TrainID FK
            string PaymentID FK
        }
        Staff {
            string Id FK
            string StaffId U
            string UserName
            string StaffName
        }
        RegPassenger {
            string Id FK
            string IdDoc
            string UserId U
            string PassengerName
        }
        Dependent {
            string Relationship
            string DepId U
            string PassengerName
        }
        Miles {
            string Miles
            string Class
        }
        User {
            string Id PK
            string Name
            string Role
            string Email
        }
        Station {
            string StName PK
            string ArName
        }
        Payment {
            string PaymentID PK
            string Status
            float Amount
        }
        Seat {
            string SeatId PK
            bool isReserved
            string Class
            string SeatLocation
        }
        Trip {
            string TripId PK
            string Time
            float Price
            string StName FK
            string EndInStName FK
        }
  
```

## Tools

---

- **DB design**: ERDplus
- **Front end**: Flutter framework
- **Backend**: Supabase platform
- **Resources**: Flutter documentation, Supabase documentation, set of icons and images

## ISSUES

---

- Unclear requirements in phase 1 that became ambiguous in phase 2, resolved by repeatedly reviewing and analyzing the requirements.
- Challenges in understanding or implementing specific features, addressed through team brainstorming and collaboration.

## BOUNS FEATURES

---

- Localization: the application supports Arabic and English languages
- Compatibility: The application can run as a mobile application or as a website in the browser
- Average load factor
- Passenger's miles report
- List of dependents travelling on a given date on all trains



## Suggestions

---

The project was heavily front-end focused, with most of our effort dedicated to UI/UX rather than back-end logic and queries. This emphasis seems beyond the course's intended scope.

Thursday, November 7, 2024

<u>Name</u>	<b>Almajed</b>	<b>Hejazi</b>	<b>Busaleh</b>
<u>Task</u>	Frontend	Frontend	Backend, report

Files:

EER & Relational Schema:

<https://drive.google.com/drive/folders/14YP2VL3pH5t7wOMQTVKMgnbEopmk4iLw?usp=sharing>

Flutter project:

<https://github.com/boFawzi/database-project.git>