MOTOR INC.

PROJECT_0502_09

Rashi Madnani, Gunjan Sharma, Hridyansh Gupta, Minghao Fang

TABLE OF CONTENTS

BACKGROUND

- The users of the Motor Inc are the students, faculty and guests who commute to the university.
- The data is beneficial in making key business decisions such as where to divert traffic if a lot is at full capacity, to analyze which customer category occupies the maximum number of parking slots.
- Data Source:
  - Random Data Generator
  - UMD DOTS

MISSION STATEMENT

To create and deploy a parking lot management system for data storage, data processing, data analysis, and to derive insights by monitoring and maximizing parking space, controlling the influx of automobiles.

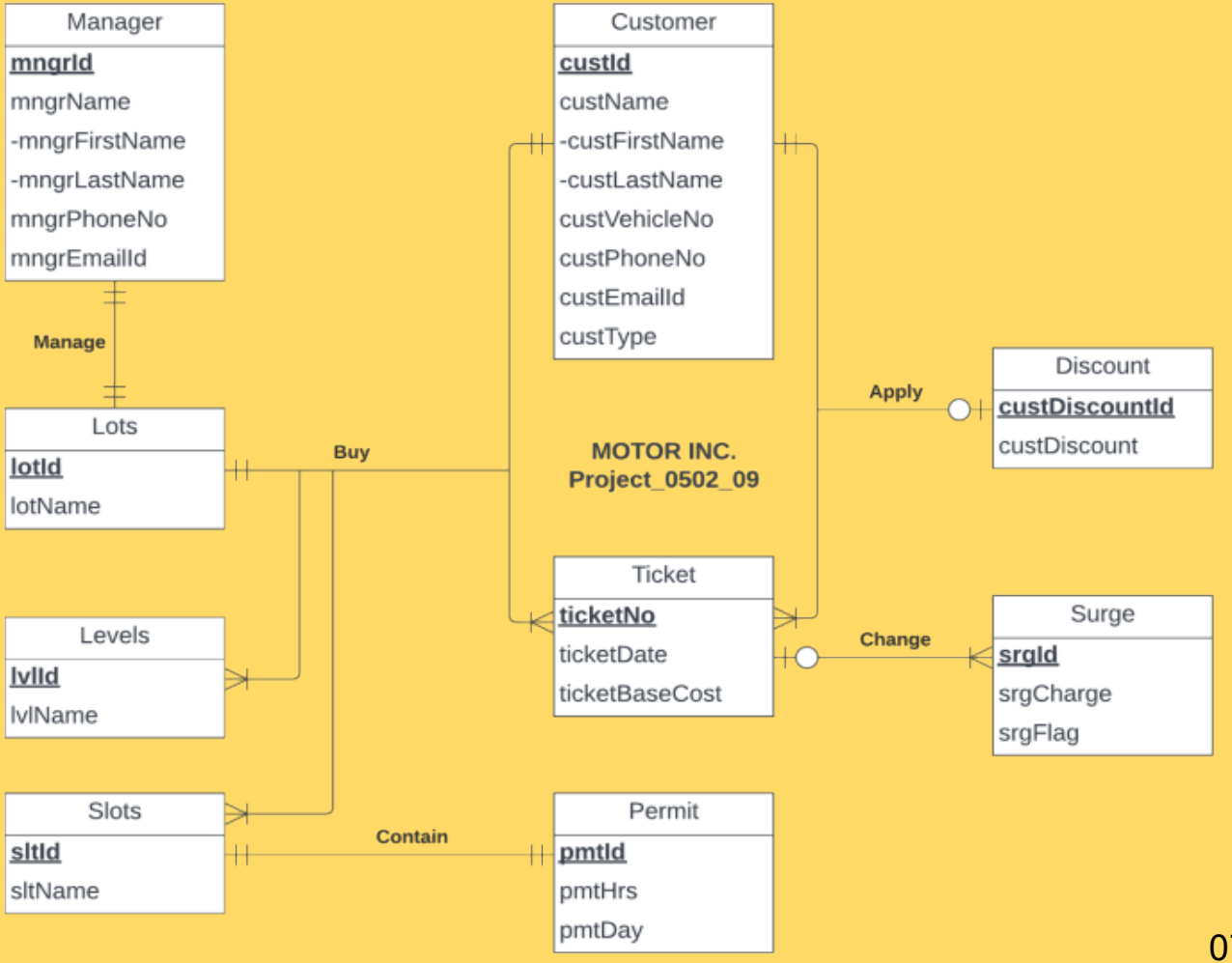
MISSION OBJECTIVES

- Determine which days have the highest volume of traffic.
- Determine the most heavily congested parking lot.
- Assess if parking lots are adequate for regular commuters comprising students, faculty, and staff.
- Analyze the volume of traffic on different game days.
- Examine if providing a discount to a customer helps the University increase revenue.

04

CONCEPTUAL DATABASE DESIGN

ER DIAGRAM


Manager
mngrId
mngrName
-mngrFirstName
-mngrLastName
mngrPhoneNo
mngrEmailId
Manage
Customer
custId
custName
-custFirstName
-custLastName
custVehicleNo
custPhoneNo
custEmailId
custType
Discount
custDiscountId
custDiscount
Apply
Lots
lotId
lotName
Buy
MOTOR INC.
Project_0502_09
Ticket
ticketNo
ticketDate
ticketBaseCost
Change
Surge
srgId
srgCharge
srgFlag
Levels
lvlId
lvlName
Slots
sltId
sltName
Contain
Permit
pmtId
pmtHrs
pmtDay

05

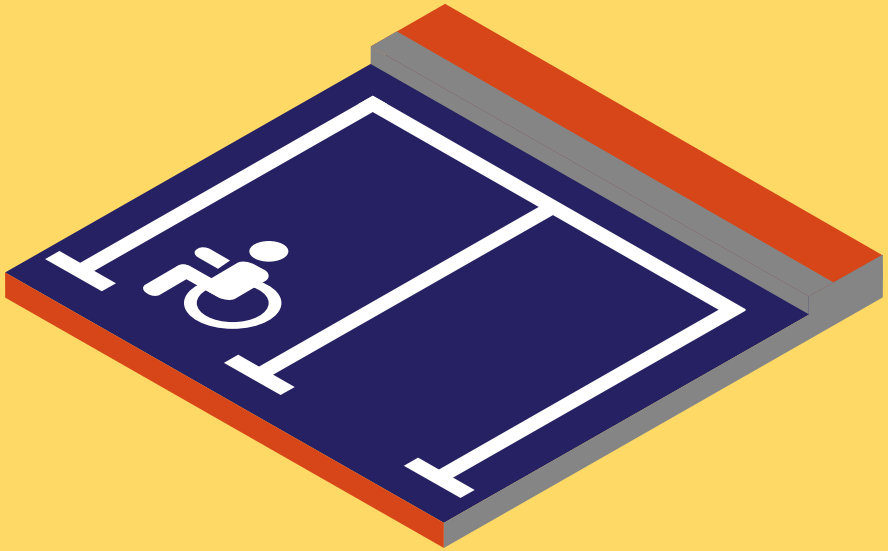
LOGICAL DATABASE DESIGN

RELATIONAL SCHEMA

- Ticket (**ticketNo**, ticketDate, ticketBaseCost)
- Customer (**custId**, custFirstName, custLastName, custVehicleNo, custPhoneNo, custEmailId, custType)
- Discount (**custDiscountId**, custDiscount)
- Lots (**lotId**, lotName)
- Levels (**lvlId**, lvlName)
- Slots (**sltId**, sltName)
- Manager (**mngrId**, mngrFirstName, mngrLastName, mngrPhoneNo, mngrEmailId, *lotId*)
- Surge (**srgId**, srgCharge, srgFlag, *ticketNo*)
- Permit (**pmtId**, pmtHrs, pmtDay, *sltId*)
- Buy (***ticketNo, lotId, custId, lvlId, sltId***)
- Apply (***custId, ticketNo, custDiscountId***)

06

PHYSICAL DATABASE DESIGN



CREATE TABLE

```
CREATE TABLE [UMDParking.Buy] (
        ticketNo CHAR(10) NOT NULL,
        lotId CHAR(10) NOT NULL,
        custId CHAR(10) NOT NULL,
        lvlId CHAR(10) NOT NULL,
        sltId CHAR(10) NOT NULL,
        CONSTRAINT pk_Buy_ticketNo_lotId_custId PRIMARY KEY (ticketNo, lotId, custId, lvlId, sltId),
        CONSTRAINT fk_Buy_ticketNo FOREIGN KEY (ticketNo)
                REFERENCES [UMDParking.Ticket] (ticketNo)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT fk_Buy_lotId FOREIGN KEY (lotId)
                REFERENCES [UMDParking.Lots] (lotId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT fk_Buy_custId FOREIGN KEY (custId)
                REFERENCES [UMDParking.Customer] (custId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT fk_Buy_lvlId FOREIGN KEY (lvlId)
                REFERENCES [UMDParking.Levels] (lvlId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT fk_Buy_sltId FOREIGN KEY (sltId)
                REFERENCES [UMDParking.Slots] (sltId)
                ON DELETE NO ACTION ON UPDATE NO ACTION )
```

What will be the cost of ticket on gameday weekday and gameday weekend for a certain amount of time?

QUERY

```
USE BUDT703_Project_0502_09
SELECT DISTINCT
    t.ticketNo AS 'Ticket Number',
    t.ticketDate AS 'Ticket Date',
    CONCAT(c.custFirstName,' ', c.custLastName) AS 'Customer Name',
    p.pmtDay AS 'Weekday/Weekend',
    l.lotName AS 'Lot Name',
    lv.lvlName AS 'Level Name',
    sl.sltName AS 'Slot Name',
    p.pmtHrs AS 'Parking Duration (in hrs)',
    t.ticketBaseCost AS 'Base Cost of Ticket($)',
    s.srgCharge AS 'Surge Charge($)',
  (ticketBaseCost+ 1.50*(p.pmtHrs-1)- d.custDiscount+s.srgCharge) AS 'Total Ticket Price($)'
FROM  [UMDParking.Permit] p,
  [UMDParking.Ticket] t,
  [UMDParking.Surge] s,
  [UMDParking.Customer] c,
  [UMDParking.Buy] b,
  [UMDParking.Lots] l,
  [UMDParking.Levels] lv,
  [UMDParking.Apply] a,
  [UMDParking.Discount] d,
[UMDParking.Slots] sl

WHERE c.custId = b.custId
AND t.ticketNo = b.ticketNo
AND l.lotId = b.lotId
AND lv.lvlId = b.lvlId
AND sl.sltId = b.sltId
AND sl.sltId = p.sltId
AND c.custId = a.custId
AND t.ticketNo = a.ticketNo
AND d.custDiscountId = a.custDiscountId
AND t.ticketNo = s.ticketNo
AND s.srgFlag = 'Y'
```

OUTPUT

	Ticket Number	Ticket Date	Customer Name	Weekday/Weekend	Lot Name	Level Name	Slot Name	Parking Duration (in hrs)	Base Cost of Ticket($)	Surge Charge($)	Total Ticket Price($)
1	0000000003	2022-08-11	Hiruzen Sarutobi	Weekend	Regents	RA	RAC	3	2	5.00	10
2	0000000004	2022-08-11	Sasuke Uchiha	Weekend	Regents	RA	RAD	1	3	5.00	6.5
3	0000000005	2022-07-11	Itachi Uchiha	Weekday	Regents	RB	RBK	2.5	3	5.00	10.25
4	0000000006	2022-08-26	Metztli Clarity	Weekday	Regents	RB	RBL	1.5	3	5.00	7.25
5	0000000009	2022-07-26	Tena Alevtina	Weekend	Regents	RD	RDX	1	3	5.00	6.5
6	0000000010	2022-08-24	Onora Annetta	Weekday	Regents	RD	RDY	2.5	2	5.00	9.25
7	0000000013	2022-06-27	Humbert Achan	Weekend	Smith	SA	SAC	1	2	5.00	7
8	0000000014	2022-08-23	Torben Enheduna	Weekday	Smith	SA	SAD	2.5	3	5.00	8.75
9	0000000015	2022-08-26	Ulrik Paftiy	Weekday	Smith	SB	SBJ	2.5	3	5.00	10.25
10	0000000016	2022-07-26	Gisleruis Gloria	Weekday	Smith	SB	SBK	1.5	3	5.00	7.25
11	0000000019	2022-06-28	Yadira Ambrozy	Weekday	Smith	SD	SDX	2.5	3	5.00	8.75
12	0000000023	2022-08-26	Neelam Ferdowsi	Weekday	Union	UA	UAE	1.5	3	5.00	8.75
13	0000000024	2022-07-26	Livia Borna	Weekend	Union	UA	UAF	3	3	5.00	9.5
14	0000000025	2022-08-25	Clement Mille	Weekend	Union	UB	UBL	1	2	5.00	7
15	0000000026	2022-09-26	Alina Kennet	Weekday	Union	UB	UBM	2.5	3	5.00	8.75
16	0000000029	2022-08-26	Carrol Vespasian	Weekday	Union	UD	UDW	2.5	2	5.00	7.75
17	0000000030	2022-08-26	Margarita Bohus...	Weekday	Union	UD	UDX	2.5	3	5.00	10.25
18	0000000033	2022-08-26	Vihtori Cai	Weekday	Mowatt	MA	MAC	2.5	3	5.00	10.25
19	0000000034	2022-06-25	Szabolcs Honoria	Weekday	Mowatt	MA	MAD	1.5	3	5.00	7.25
20	0000000035	2022-08-26	Mesud Virgilius	Weekday	Mowatt	MB	MBN	1.5	2	5.00	7.75
21	0000000036	2022-07-27	Valdemar Kori	Weekend	Mowatt	MB	MBO	3	3	5.00	9.5
22	0000000039	2022-09-24	Helmut Raisa	Weekday	Mowatt	MD	MDX	1.5	3	5.00	7.25
23	0000000040	2022-09-26	Hristina Gnaeus	Weekend	Mowatt	MD	MDY	3	2	5.00	10
24	0000000043	2022-07-26	Severin Ragna	Weekday	Downto...	DA	DAC	1.5	3	5.00	8.75
25	0000000044	2022-08-26	Barclay Suad	Weekend	Downto...	DA	DAD	3	2	5.00	8.5
26	0000000045	2022-08-25	Anton Connla	Weekday	Downto...	DB	DBK	2.5	2	5.00	9.25
27	0000000046	2022-06-26	Tshofelo Mikhail	Weekday	Downto...	DB	DBL	2.5	2	5.00	7.75
28	0000000049	2022-06-26	Hildbrand Nikole	Weekend	Downto...	DD	DDY	1	2	5.00	5.5

What is the total revenue generated by the Motor_Inc for each customer type?

QUERY

```
USE BUDT703_Project_0502_09
SELECT  SUM(student.Ticket_Price) AS 'Student Ticket Total ($)' ,
        SUM(guest.Ticket_Price ) AS 'Guest Ticket Total ($)',
        SUM(faculty.Ticket_Price) AS 'Faculty Ticket Total ($)'
FROM
    (SELECT  c.custId,c.custFirstName,c.custLastName,c.custType,p.pmtHrs,p.pmtDay,t.ticketBaseCost,d.custDiscount,s.srgCharge,sl.sltName,
            (t.ticketBaseCost+ 1.50*(p.pmtHrs-1)-d.custDiscount+s.srgCharge) AS 'Ticket_Price'
     FROM   [UMDParking.Customer] c,[UMDParking.Ticket] t,[UMDParking.Apply] a,[UMDParking.Buy] b,
            [UMDParking.Discount] d,[UMDParking.Surge] s,[UMDParking.Permit] p,[UMDParking.Slots] sl
    WHERE  c.custType = 'Student'
          AND c.custId = a.custId
          AND c.custId = b.custId
          AND t.ticketNo = a.ticketNo
          AND t.ticketNo = b.ticketNo
          AND sl.sltId = b.sltId
          AND sl.sltId = p.sltId
          AND d.custDiscountId = a.custDiscountId
          AND s.ticketNo = t.ticketNo
    ) AS student,
```

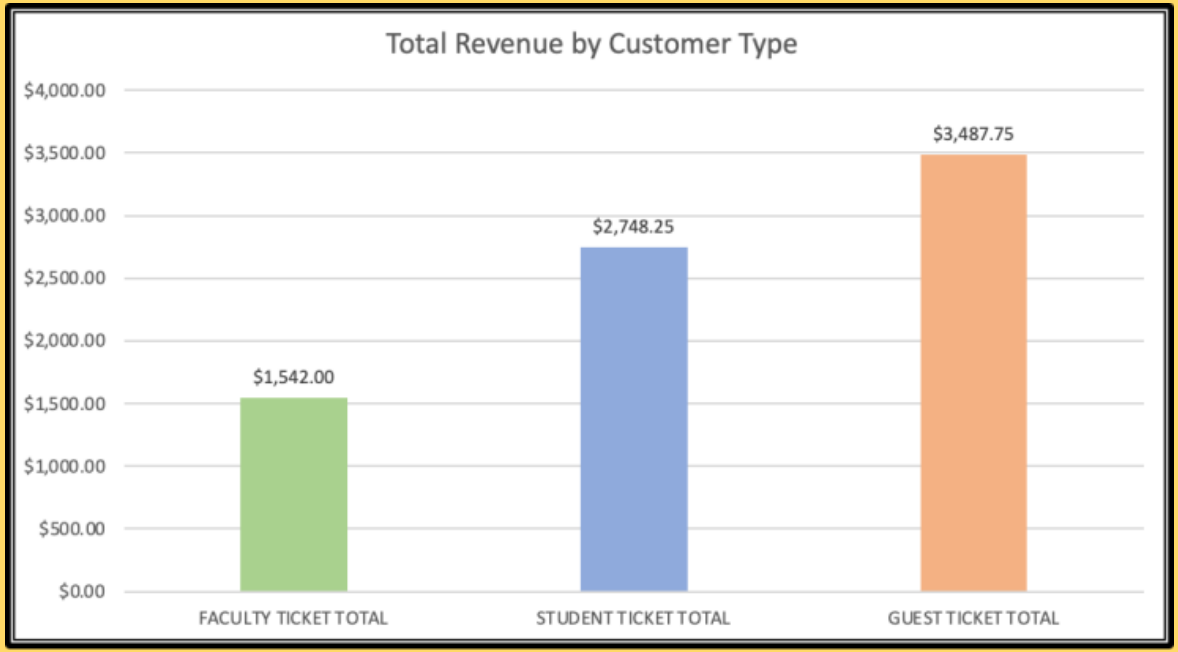
```
    (SELECT  c.custId,c.custFirstName,c.custLastName,c.custType,p.pmtHrs,p.pmtDay,t.ticketBaseCost,d.custDiscount,s.srgCharge,sl.sltName,
            (t.ticketBaseCost+ 1.50*(p.pmtHrs-1)-d.custDiscount+s.srgCharge) AS 'Ticket_Price'
     FROM   [UMDParking.Customer] c,[UMDParking.Ticket] t,[UMDParking.Apply] a,[UMDParking.Buy] b,
            [UMDParking.Discount] d,[UMDParking.Surge] s,[UMDParking.Permit] p,[UMDParking.Slots] sl
    WHERE  c.custType = 'Guest'
          AND c.custId = a.custId
          AND c.custId = b.custId
          AND t.ticketNo = a.ticketNo
          AND t.ticketNo = b.ticketNo
          AND sl.sltId = b.sltId
          AND sl.sltId = p.sltId
          AND d.custDiscountId = a.custDiscountId
          AND s.ticketNo = t.ticketNo
    ) AS guest,
```

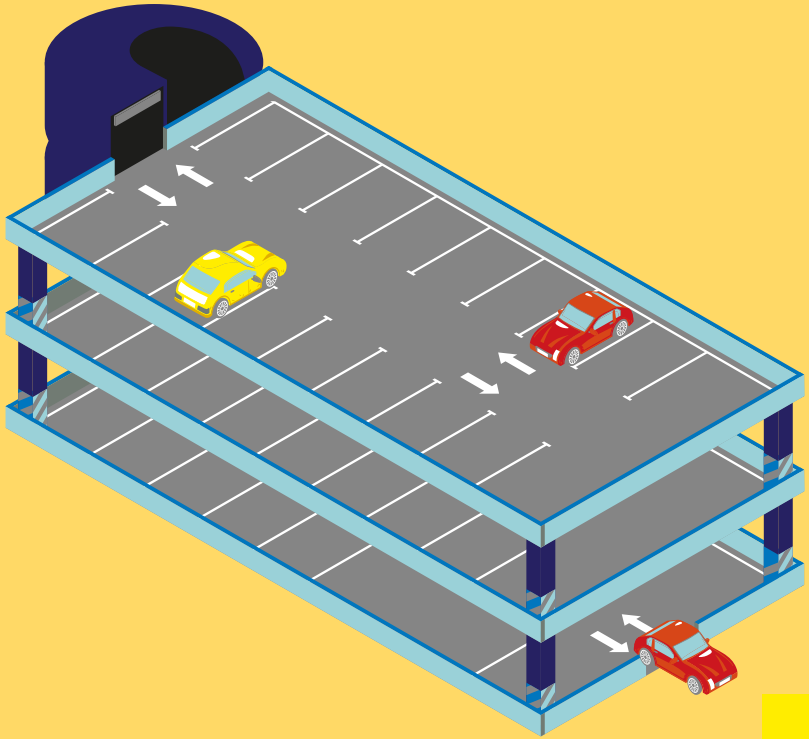
```
    (SELECT  c.custId,c.custFirstName,c.custLastName,c.custType,p.pmtHrs,p.pmtDay,t.ticketBaseCost,d.custDiscount,s.srgCharge,sl.sltName,
            (t.ticketBaseCost+ 1.50*(p.pmtHrs-1)-d.custDiscount+s.srgCharge) AS 'Ticket_Price'
     FROM   [UMDParking.Customer] c,[UMDParking.Ticket] t,[UMDParking.Apply] a,[UMDParking.Buy] b,
            [UMDParking.Discount] d,[UMDParking.Surge] s,[UMDParking.Permit] p,[UMDParking.Slots] sl
    WHERE  c.custType = 'Faculty'
          AND c.custId = a.custId
          AND c.custId = b.custId
          AND t.ticketNo = a.ticketNo
          AND t.ticketNo = b.ticketNo
          AND sl.sltId = b.sltId
          AND sl.sltId = p.sltId
          AND d.custDiscountId = a.custDiscountId
          AND s.ticketNo = t.ticketNo
          ) AS faculty

WHERE   student.pmtHrs = guest.pmtHrs
        AND faculty.pmtHrs = student.pmtHrs
```

OUTPUT

	Student Ticket Total ($)	Guest Ticket Total ($)	Faculty Ticket Total ($)
1	2748.25	3487.75	1542





THANK YOU

DO YOU HAVE ANY QUESTIONS?