University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: 1 March 2024

Group Number: 75

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Dwayne Dmello	13046024	F7t7m	dwayne.dmello6@gmail.com
Theo Obadiah Teguh	67719955	k2c7t	theo.obadiah@gmail.com
Adeeb Khan	49330046	f4j3s	adeeb1037@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia.

University of British Columbia, Vancouver
Department of Computer Science

Car Parking Management System (CPMS)

Deliverables

2. Project Description

Our project involves the development of a database system tailored for multi-level car parks. This application is designed to streamline the management of parking facilities, providing a solution for overseeing operations, assessing profitability, and analyzing car distribution within a network of parking lots. The primary goal is to enhance efficiency and user experience in parking lots while addressing common challenges such as finding parking spaces in crowded areas, locating parked cars, and planning ahead for reservations.

3. ER Diagram

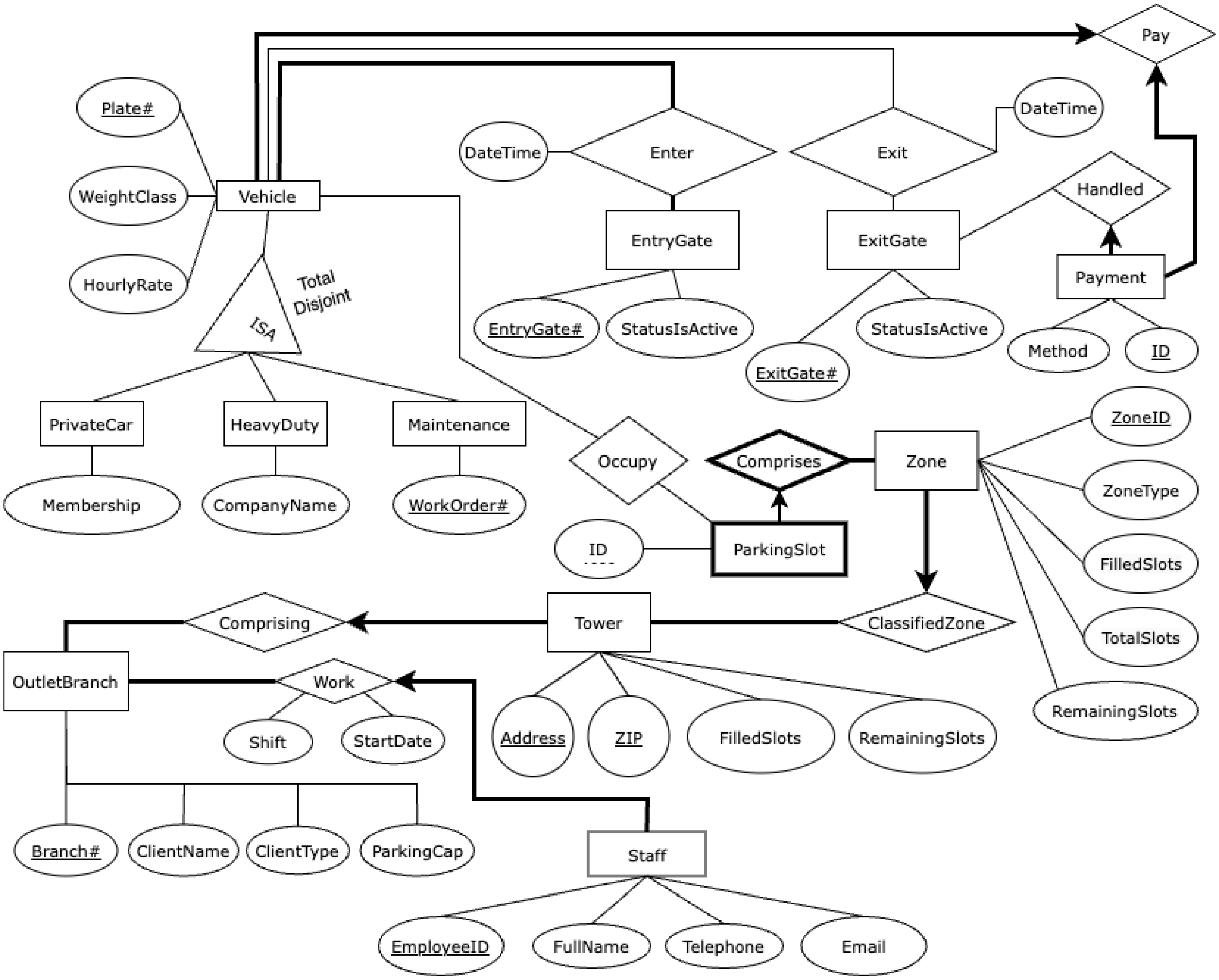
We made the following adjustments to the ER Diagram from the previous milestone.

- The Gate# key in EntryGate and ExitGate has been updated to be EntryGate# and ExitGate# respectively. This is done in order to clearly distinguish between the two attributes.
- We have updated our ISA constraint to *Total Disjoint*.

We also added the following attributes to facilitate the creation of functional dependencies.

- A TotalSlots attribute has been added to Zone.
- A WeightClass attribute has been added to Vehicle.
- A ClientType attribute has been added to OutletBranch.

The updated ER Diagram is as follows.



4. Schema

Keys
Bold: **Foreign Key**
Underlined: Primary Key

- Vehicle(Plate#: Varchar(10), HourlyRate: INT, WeightClass: Char(1))
- PrivateCar(**Plate#:** Varchar(10), Membership: Varchar(12))
- HeavyDuty(**Plate#:** Varchar(10), CompanyName: Varchar(255))
- Maintenance(**Plate#:** Varchar(10), WorkOrder#: Varchar(12)), Candidate key: WorkOrder#
- Payment(ID: Varchar(10), Method: Varchar(255), **Plate#:** Varchar(10), **ExitGate#:** INT)
- Enter(DateTime: DATETIME, **Plate#:** Varchar(10), **EntryGate#:** INT)
- Exit(DateTime, **Plate#:** Varchar(10), **ExitGate#:** INT)
- EntryGate(EntryGate#: INT, StatusIsActive: INT)
- ExitGate(ExitGate#: INT, StatusIsActive: INT)

- Zone(ZoneID: INT, ZoneType: INT, FilledSlots: INT, RemainingSlots: INT, TotalSlots: INT, **Address:** Varchar(255), **Zip:** Char(6))
- ParkingSlot(ID: Varchar(4), **ZoneID**: INT)
- Occupy(**ID** Varchar(4), **ZoneID** INT, **Plate#** Varchar(10))
- OutletBranch(Branch#: Varchar(12), ClientName: Varchar(255), ParkingCap: INT, ClientType: Char(1))
- Tower(Address: Varchar(255), Zip: Char(6), FilledSlots: INT, RemainingSlots: INT, **Branch#:** Varchar(12))
- Staff(EmployeeID: INT, FullName: Varchar(255), Telephone: INT, email: Varchar(255), Shift: Varchar(255), StartDate: DATE, **Branch#:** Varchar(12)) Candidate key: email (must be unique)

5. Functional Dependencies

- Plate# -> HourlyRate
- Plate# -> Membership
- Plate# -> CompanyName
- Plate# -> WorkOrder#
- Plate# -> WeightClass
- WeightClass -> HourlyRate

- ID -> Method
- EntryGate# -> StatusIsActive
- ExitGate# -> StatusIsActive

- ZoneID -> ZoneType, FilledSlots, RemainingSlots, TotalSlots
- ZoneType -> TotalSlots /

- TotalSlots, RemainingSlots -> FilledSlots
- TotalSlots, FilledSlots -> RemainingSlots
- RemainingSlots, FilledSlots -> TotalSlots

- Branch# -> ClientName, ParkingCap, ClientType
- Address, Zip -> FilledSlots, RemainingSlots
- EmployeeID -> FullName, Telephone, email, Shift, StartDate, Branch#
- email -> FullName, Telephone, EmployeeID, Shift, StartDate, Branch#
- ClientType -> ParkingCap

6. **Decomposing to BCNF**

- We see that WeightClass -> HourlyRate violates BCNF in the Vehicle table so we decompose it. On decomposing, we get 2 relations:

- VehichleClass(Plate#: Varchar(10), WeightClass: Char(1))
- WeightRate(**WeightClass**: Char(1), HourlyRate: INT)

These satisfy BCNF.

- We also see that ZoneType -> TotalSlots violates BCNF in Zone so we decompose on that relation to get these 2 relations:

- ZoneType(ZoneID: INT, ZoneType: INT, FilledSlots: INT, RemainingSlots: INT, **Address:** Varchar(255), **Zip:** Char(6))

- TypeSlots(**ZoneType**: INT, TotalSlots: INT)

These are both in BCNF as well.

Finally, we see ClientType -> ParkingCap violates BCNF in BranchOutlet. We decompose this function to get:

- BranchClient(Branch#: Varchar(12), ClientName: Varchar(255), ClientType: Char(1))
- BranchCap(**ClientType:** Char(1), ParkingCap: INT)

These 2 are in BCNF.

All of our remaining FDs and relations are In BCNF and therefore the decomposition is complete.

FINAL TABLES AFTER BCNF DECOMPOSITION:

-VehichleClass(Plate#: Varchar(10), WeightClass: Char(1))
-WeightRate(**WeightClass**: Char(1), HourlyRate: INT)

- PrivateCar(**Plate#:** Varchar(10)**,** Membership: Varchar(12))
- HeavyDuty(**Plate#:** Varchar(10)**,** CompanyName: Varchar(255))
- Maintenance(**Plate#:** Varchar(10)**,** WorkOrder#: Varchar(12)), Candidate key: WorkOrder#
- Payment(ID: Varchar(10), Method: Varchar(255), **Plate#:** Varchar(10)**, ExitGate#:** INT)
- EntryGate(EntryGate#: INT, StatusIsActive: INT)
- ExitGate(ExitGate#: INT, StatusIsActive: INT)
- Enter(DateTime: DATETIME, **Plate#:** Varchar(10), **EntryGate#:** INT)
- Exit(DateTime, **Plate#:** Varchar(10), **ExitGate#:** INT)

- ZoneType(ZoneID: INT, ZoneType: INT, FilledSlots: INT, RemainingSlots: INT, **Address:** Varchar(255), **Zip:** Char(6))

-TypeSlots(**ZoneType**: INT, TotalSlots: INT)

- ParkingSlot(ID: Varchar(4), **ZoneID**: INT)
- Occupy(**ID** Varchar(4), **ZoneID** INT**,** **Plate#** Varchar(10))

- BranchClient(Branch#: Varchar(12), ClientName: Varchar(255), ClientType: Char(1))
- BranchCap(**ClientType:** Char(1), ParkingCap: INT)

- Tower(Address: Varchar(255), Zip: Char(6), FilledSlots: INT, RemainingSlots: INT, **Branch#:** Varchar(12))
- Staff(EmployeeID: INT, FullName: Varchar(255), Telephone: INT, email: Varchar(255), Shift: Varchar(255), StartDate: DATE, **Branch#:** Varchar(12)) Candidate key: email (must be unique)

7. **SQL DDL Statements**

- VehicleClass Table

```
CREATE TABLE VehicleClass (
   Plate# VARCHAR(10) PRIMARY KEY,
   WeightClass CHAR(1)
);
```

- WeightRate Table

```
CREATE TABLE WeightRate (
   WeightClass CHAR(1) PRIMARY KEY,
   HourlyRate INT,
    FOREIGN KEY (WeightClass) REFERENCES VehicleClass
);

- PrivateCar Table
CREATE TABLE PrivateCar (
    Plate# VARCHAR(10) PRIMARY KEY,
    Membership VARCHAR(12),
    FOREIGN KEY (Plate#) REFERENCES VehicleClass
);

- HeavyDuty Table
CREATE TABLE HeavyDuty (
    Plate# VARCHAR(10) PRIMARY KEY,
    CompanyName VARCHAR(255),
    FOREIGN KEY (Plate#) REFERENCES VehicleClass
);

- Maintenance Table
CREATE TABLE Maintenance (
    Plate# VARCHAR(10) PRIMARY KEY,
    WorkOrder# VARCHAR(12),
    FOREIGN KEY (Plate#) REFERENCES VehicleClass
);

- Payment Table
CREATE TABLE Payment (
    ID VARCHAR(10) PRIMARY KEY,
    Method VARCHAR(255),
    Plate# VARCHAR(10),
    ExitGate# INT,
    FOREIGN KEY (Plate#) REFERENCES VehicleClass,
    FOREIGN KEY (ExitGate#) REFERENCES ExitGate
);

- Enter Table
CREATE TABLE Enter (
    DateTime DATETIME,
    Plate# VARCHAR(10),
    EntryGate# INT,
    PRIMARY KEY (EntryGate#, Plate#),
    FOREIGN KEY (Plate#) REFERENCES VehicleClass,
    FOREIGN KEY (EntryGate#) REFERENCES EntryGate
);
```

```
- Exit Table
CREATE TABLE Exit (
   DateTime DATETIME,
   Plate# VARCHAR(10),
   ExitGate# INT,
   PRIMARY KEY (ExitGate#, Plate#),
   FOREIGN KEY (Plate#) REFERENCES VehicleClass,
   FOREIGN KEY (ExitGate#) REFERENCES ExitGate
);

- EntryGate Table
CREATE TABLE EntryGate (
   EntryGate# INT PRIMARY KEY,
   StatusIsActive INT
);

- ExitGate Table
CREATE TABLE ExitGate (
   ExitGate# INT PRIMARY KEY,
   StatusIsActive INT
);

- ZoneType Table
CREATE TABLE ZoneType (
   ZoneID INT PRIMARY KEY,
   ZoneType INT,
   FilledSlots INT,
   RemainingSlots INT,
   Address VARCHAR(255),
   Zip CHAR(6),
   FOREIGN KEY (Address, Zip) REFERENCES Tower
);

- TypeSlots Table
CREATE TABLE TypeSlots (
   ZoneType INT PRIMARY KEY,
   TotalSlots INT,
   FOREIGN KEY (ZoneType) REFERENCES ZoneType

);


- ParkingSlot Table
```

```
CREATE TABLE ParkingSlot (
    ID VARCHAR(4),
    ZoneID INT,
    PRIMARY KEY (ID, ZoneID),
    FOREIGN KEY (ZoneID) REFERENCES ZoneType(ZoneID)
);

- Occupy Table
CREATE TABLE Occupy (
    ID VARCHAR(4),
    ZoneID INT,
    Plate# VARCHAR(10),
    PRIMARY KEY (ID, ZoneID, Plate#),
    FOREIGN KEY (ID) REFERENCES ParkingSlot(ID),
    FOREIGN KEY (ZoneID) REFERENCES ZoneType(ZoneID),
    FOREIGN KEY (Plate#) REFERENCES VehicleClass(Plate#)
);

- BranchClient Table
CREATE TABLE BranchClient (
    Branch# VARCHAR(12) PRIMARY KEY,
    ClientName VARCHAR(255),
    ClientType CHAR(1)
);

- BranchCap Table
CREATE TABLE BranchCap (
    ClientType CHAR(1) PRIMARY KEY,
    ParkingCap INT,
FOREIGN KEY (ClientType) REFERENCES BranchClient
);

- Tower Table
CREATE TABLE Tower (
    Address VARCHAR(255),
    Zip CHAR(6),
    FilledSlots INT,
    RemainingSlots INT,
    Branch# VARCHAR(12),
    PRIMARY KEY (Address, Zip),
    FOREIGN KEY (Branch#) REFERENCES BranchClient(Branch#)
);
```

```
- Staff Table
CREATE TABLE Staff (
   EmployeeID INT PRIMARY KEY,
   FullName VARCHAR(255),
   Telephone INT,
   email VARCHAR(255) UNIQUE,
   Shift VARCHAR(255),
   StartDate DATE,
   Branch# VARCHAR(12),
   FOREIGN KEY (Branch#) REFERENCES BranchClient(Branch#)
);
```

8. **Insert Statements**

```
- VehicleClass Table
INSERT INTO VehicleClass VALUES ('PLATE1234', 'A');
INSERT INTO VehicleClass VALUES ('PLATE2345', 'B');
INSERT INTO VehicleClass VALUES ('PLATE3456', 'C');
INSERT INTO VehicleClass VALUES ('PLATE4567', 'D');
INSERT INTO VehicleClass VALUES ('PLATE5678', 'E');

- WeightRate Table
INSERT INTO WeightRate VALUES ('A', 10);
INSERT INTO WeightRate VALUES ('B', 15);
INSERT INTO WeightRate VALUES ('C', 20);
INSERT INTO WeightRate VALUES ('D', 25);
INSERT INTO WeightRate VALUES ('E', 30);

- PrivateCar Table
INSERT INTO PrivateCar VALUES ('PLATE1234', 'MEM123');
INSERT INTO PrivateCar VALUES ('PLATE2345', 'MEM234');
INSERT INTO PrivateCar VALUES ('PLATE3456', 'MEM345');
INSERT INTO PrivateCar VALUES ('PLATE4567', 'MEM456');
INSERT INTO PrivateCar VALUES ('PLATE5678', 'MEM567');

- HeavyDuty Table
INSERT INTO HeavyDuty VALUES ('PLATE6789', 'CompA');
INSERT INTO HeavyDuty VALUES ('PLATE7890', 'CompB');
INSERT INTO HeavyDuty VALUES ('PLATE8901', 'CompC');
INSERT INTO HeavyDuty VALUES ('PLATE9012', 'CompD');
INSERT INTO HeavyDuty VALUES ('PLATE0123', 'CompE');
```

```
- Maintenance Table
INSERT INTO Maintenance VALUES ('PLATE1234', 'WO1234');
INSERT INTO Maintenance VALUES ('PLATE2345', 'WO2345');
INSERT INTO Maintenance VALUES ('PLATE3456', 'WO3456');
INSERT INTO Maintenance VALUES ('PLATE4567', 'WO4567');
INSERT INTO Maintenance VALUES ('PLATE5678', 'WO5678');

- Payment Table
INSERT INTO Payment VALUES ('ID123', 'Cash', 'PLATE1234', 1);
INSERT INTO Payment VALUES ('ID234', 'Card', 'PLATE2345', 2);
INSERT INTO Payment VALUES ('ID345', 'Online', 'PLATE3456', 3);
INSERT INTO Payment VALUES ('ID456', 'Check', 'PLATE4567', 4);
INSERT INTO Payment VALUES ('ID567', 'Mobile', 'PLATE5678', 5);

- Enter Table
INSERT INTO Enter VALUES ('2023-01-01 08:00:00', 'PLATE1234', 1);
INSERT INTO Enter VALUES ('2023-01-02 09:00:00', 'PLATE2345', 2);
INSERT INTO Enter VALUES ('2023-01-03 10:00:00', 'PLATE3456', 3);
INSERT INTO Enter VALUES ('2023-01-04 11:00:00', 'PLATE4567', 4);
INSERT INTO Enter VALUES ('2023-01-05 12:00:00', 'PLATE5678', 5);

- Exit Table
INSERT INTO Exit VALUES ('2023-01-01 18:00:00', 'PLATE1234', 1);
INSERT INTO Exit VALUES ('2023-01-02 19:00:00', 'PLATE2345', 2);
INSERT INTO Exit VALUES ('2023-01-03 20:00:00', 'PLATE3456', 3);
INSERT INTO Exit VALUES ('2023-01-04 21:00:00', 'PLATE4567', 4);
INSERT INTO Exit VALUES ('2023-01-05 22:00:00', 'PLATE5678', 5);
- EntryGate Table
INSERT INTO EntryGate VALUES (1, 1);
INSERT INTO EntryGate VALUES (2, 1);
INSERT INTO EntryGate VALUES (3, 1);
INSERT INTO EntryGate VALUES (4, 1);
INSERT INTO EntryGate VALUES (5, 1);

- ExitGate Table
INSERT INTO ExitGate VALUES (1, 1);
INSERT INTO ExitGate VALUES (2, 1);
INSERT INTO ExitGate VALUES (3, 1);
INSERT INTO ExitGate VALUES (4, 1);
INSERT INTO ExitGate VALUES (5, 1);

- ZoneType Table
```

```
INSERT INTO ZoneType VALUES (1, 1, 20, 30, 'Address1', 'Zip1');
INSERT INTO ZoneType VALUES (2, 2, 25, 25, 'Address2', 'Zip2');
INSERT INTO ZoneType VALUES (3, 3, 30, 20, 'Address3', 'Zip3');
INSERT INTO ZoneType VALUES (4, 4, 4, 15, 35, 'Address4', 'Zip4');
INSERT INTO ZoneType VALUES (5, 5, 10, 40, 'Address5', 'Zip5');

- TypeSlots Table
INSERT INTO TypeSlots VALUES (1, 50);
INSERT INTO TypeSlots VALUES (2, 50);
INSERT INTO TypeSlots VALUES (3, 50);
INSERT INTO TypeSlots VALUES (4, 50);
INSERT INTO TypeSlots VALUES (5, 50);

- ParkingSlot Table
INSERT INTO ParkingSlot VALUES ('SLOT1', 1);
INSERT INTO ParkingSlot VALUES ('SLOT2', 2);
INSERT INTO ParkingSlot VALUES ('SLOT3', 3);
INSERT INTO ParkingSlot VALUES ('SLOT4', 4);
INSERT INTO ParkingSlot VALUES ('SLOT5', 5);

- Occupy Table
INSERT INTO Occupy VALUES ('SLOT1', 1, 'PLATE1234');
INSERT INTO Occupy VALUES ('SLOT2', 2, 'PLATE2345');
INSERT INTO Occupy VALUES ('SLOT3', 3, 'PLATE3456');
INSERT INTO Occupy VALUES ('SLOT4', 4, 'PLATE4567');
INSERT INTO Occupy VALUES ('SLOT5', 5, 'PLATE5678');

- BranchClient Table
INSERT INTO BranchClient VALUES ('BR001', 'Client1', 'A');
INSERT INTO BranchClient VALUES ('BR002', 'Client2', 'B');
INSERT INTO BranchClient VALUES ('BR003', 'Client3', 'C');
INSERT INTO BranchClient VALUES ('BR004', 'Client4', 'D');
INSERT INTO BranchClient VALUES ('BR005', 'Client5', 'E');

- BranchCap
INSERT INTO BranchCap (ClientType, ParkingCap) VALUES ('A', 100);
INSERT INTO BranchCap (ClientType, ParkingCap) VALUES ('B', 200);
INSERT INTO BranchCap (ClientType, ParkingCap) VALUES ('C', 300);
INSERT INTO BranchCap (ClientType, ParkingCap) VALUES ('D', 400);
INSERT INTO BranchCap (ClientType, ParkingCap) VALUES ('E', 500);
```

```
-Tower Table
INSERT INTO Tower VALUES ('123 Main St', 'A1B2C3', 20, 30, 'BR001');
INSERT INTO Tower VALUES ('456 Oak Rd', 'D4E5F6', 25, 25, 'BR002');
INSERT INTO Tower VALUES ('789 Pine Ave', 'G7H8I9', 30, 20, 'BR003');
INSERT INTO Tower VALUES ('101 River Ln', 'J1K2L3', 15, 35, 'BR004');
INSERT INTO Tower VALUES ('202 Lake St', 'M4N5O6', 10, 40, 'BR005');

- Staff Table
INSERT INTO Staff VALUES (1, 'John Doe', 1234567890, 'jdoe@example.com', 'Morning',
'2023-01-01', 'BR001');
INSERT INTO Staff VALUES (2, 'Jane Smith', 2345678901, 'jsmith@example.com', 'Evening',
'2023-02-01', 'BR002');
INSERT INTO Staff VALUES (3, 'Mike Brown', 3456789012, 'mbrown@example.com', 'Night',
'2023-03-01', 'BR003');
INSERT INTO Staff VALUES (4, 'Sara White', 4567890123, 'swhite@example.com', 'Morning',
'2023-04-01', 'BR004');
INSERT INTO Staff VALUES (5, 'Alex Green', 5678901234, 'agreen@example.com', 'Evening',
'2023-05-01', 'BR005');
```