

CPSC 304 Project Cover Page

Milestone #: 2

Date: Oct 20, 2023

Group Number: 10

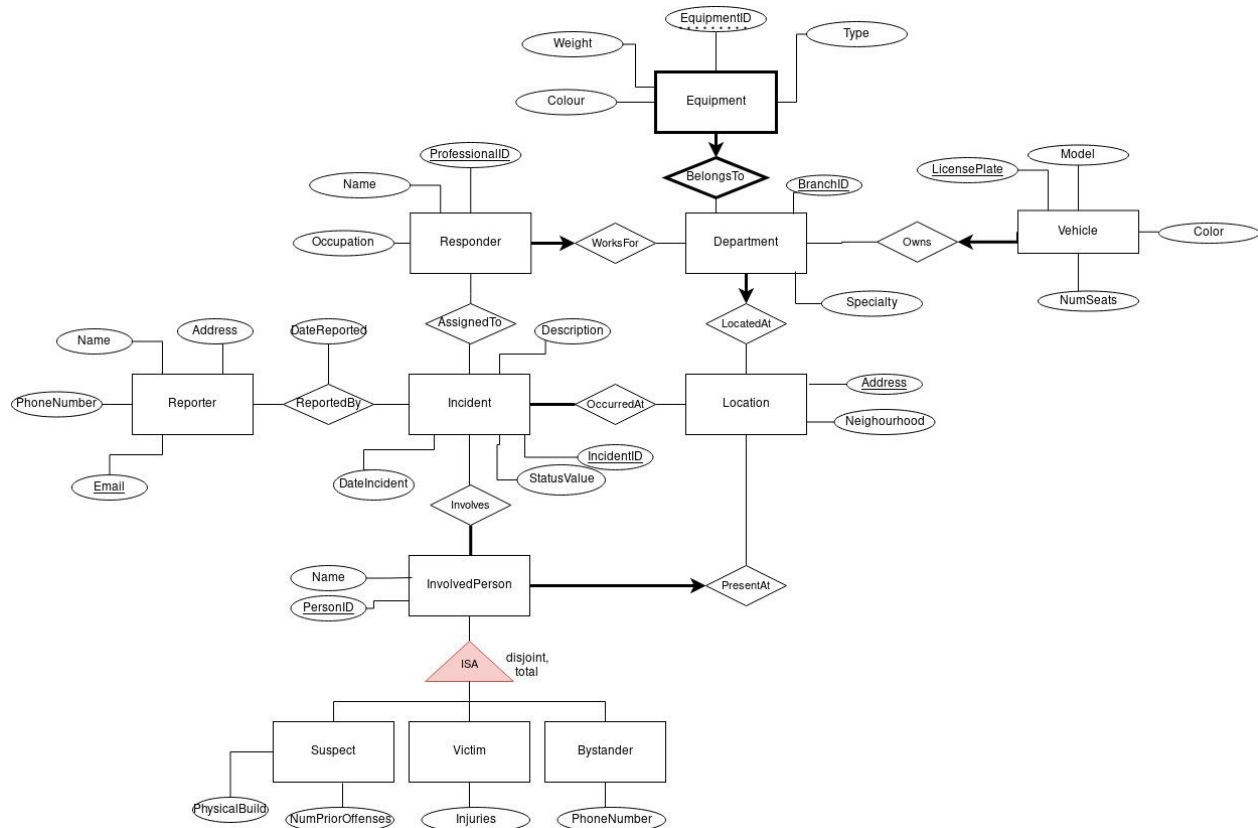
Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Bryan Chang	56321920	i9j6q	bc2003@student.ubc.ca
Maziyar Dowlatabadibazaz	94539046	b1q8n	maziyardowlat@gmail.com
William Xiao	92878362	v3m4r	munce@student.ubc.ca

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

Summary: This is the database for a municipal system tracking down all reported physical incidents in a city. Our database aims to keep track of many aspects of an incident, including the location it was located at, all people involved (the reporter of the incident, first responders, potential suspects, victims, etc.), and what equipment/vehicles were used in aiding the case.

ER Diagram:



changes to the ER Diagram:

- changed the PK for Reporter entity from Email, Name to just Email
- renamed date in Incident to dateIncident (due to name collision with the DATE type)
- renamed status to statusValue (collision with Oracle names)

Relational Schema:

Incident(
 statusValue: varchar,
 dateIncident: Date,
 incidentID: int,

description: varchar
)

Location(
 address: varchar,
 neighbourhood: varchar
)

OccurredAt(
 incidentID: int,
 address: varchar
)

Department(
 branchID: int,
 specialty: varchar,
 address: varchar NOT NULL
 other CKs: (specialty, address)
)

Responder(
 professionalID: int,
 name: varchar,
 occupation: varchar,
 branchID: int NOT NULL
 other CKs: (name)
)

AssignedTo(
 incidentID: int,
 professionalID: int
)

InvolvedPerson(
 name: varchar,

personID: int,
presentAtAddress: varchar NOT NULL
other CKs: (name)
)

Involves(
 incidentID: int,
 personID: int
)

Suspect(
 personID: int,
 physicalBuild: varchar,
 numPriorOffenses: int
)

Victim(
 personID: int,
 injuries: varchar
)

Bystander(
 personID: int,
 phoneNumber: char(10)
 other CKs: (phoneNumber)
)

Reporter(
 name: varchar,
 address: varchar,
 phoneNumber: char(10),
 email: varchar
 other CKs: (name), (address, phoneNumber)
)

```
ReportedBy(  
    incidentID: int,  
    email: varchar,  
    dateReported: date  
)
```

```
Equipment(  
    type: varchar,  
    weight: int,  
    color: varchar,  
    equipmentID: int,  
    belongsToBranchID: int  
)
```

```
Vehicle(  
    licensePlate: char,  
    model: varchar,  
    color: varchar,  
    numSeats: int,  
    ownedByBranchID: varchar NOT NULL  
)
```

Functional Dependencies

1. **Incident:**
incidentID -> statusValue, date, description
description -> statusValue
2. **Location:**
address -> neighborhood
3. **OccurredAt:**
incidentID -> address
4. **Department:**
branchID -> specialty, address
specialty, address -> branchID

5. Responder:

professionalID -> name, occupation, branchID

name -> occupation, professionalID, branchID

6. AssignedTo:

no non-trivial

7. InvolvedPerson:

personID -> name, presentAtAddress

name -> personID, presentAtAddress

8. Involves:

no non-trivial

9. Suspect:

personID -> physicalBuild, numPriorOffenses

10. Victim:

personID -> injuries

11. Bystander:

personID -> phoneNumber

phoneNumber -> personID

12. Reporter:

email -> phoneNumber, address, name

name -> email, phoneNumber, address

phoneNumber -> name

address, phoneNumber -> name, email

13. ReportedBy:

incidentID, email -> dateReported

14. Equipment:

equipmentID, belongsToBranchID -> type, weight, color

type -> weight, color

15. Vehicle:

licensePlate -> model, numSeats, color, ownedByBranchID

model -> numSeats

Normalization

We know that entities with only the primary/candidate key FDs are already in BCNF, so we will only consider entities that include more than one FD.

Incident:

description -> statusValue NOT in BCNF, therefore decompose into:

IncidentInfo(incidentID, date, **description**), IncidentStatus(description, statusValue)

IncidentInfo and IncidentStatus are in BCNF

Department:

Department is already in BCNF, no changes needed.

Responder:

Responder is already in BCNF, no changes needed.

InvolvedPerson:

InvolvedPerson is already in BCNF, no changes needed.

Bystander:

Bystander is already in BCNF, no changes needed.

Reporter:

Reporter is already in BCNF, no changes needed.

Equipment:

type -> weight, color NOT in BCNF, therefore decompose into:

EquipmentItem(equipmentID, belongsToBranchID, **type**), EquipmentInfo(type, weight, color)

EquipmentItem and EquipmentInfo are in BCNF.

Vehicle:

model -> numSeats NOT in BCNF, therefore decompose into:

VehicleInfo(licensePlate, **model**, color, **ownedByBranchID**), VehicleSpecs(model, numSeats)

VehicleInfo and VehicleSpecs are in BCNF.

SQL DDL:

We are using Oracle so there is no ON UPDATE CASCADE; but we indicated the situations where they would have been used below in *italics*.

```
CREATE TABLE IncidentStatus(  
    description VARCHAR(200) PRIMARY KEY,  
    statusValue VARCHAR(20)  
);
```

```
CREATE TABLE IncidentInfo(  
    incidentID INT PRIMARY KEY,  
    dateIncident DATE,  
    description VARCHAR(200),  
    FOREIGN KEY (description) REFERENCES IncidentStatus(description)  
        ON DELETE SET NULL  
        ON UPDATE CASCADE  
);
```

```
CREATE TABLE Location(  
    address VARCHAR(100) PRIMARY KEY,  
    neighbourhood VARCHAR(30)  
);
```

```
CREATE TABLE OccurredAt(  
    incidentID INT,  
    address VARCHAR(100),  
    PRIMARY KEY(incidentID, address),  
    FOREIGN KEY (incidentID) REFERENCES IncidentInfo(incidentID)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE,  
    FOREIGN KEY (address) REFERENCES Location(address)  
        ON DELETE SET NULL  
);
```

```
CREATE TABLE Department(  
    branchID INT PRIMARY KEY,  
    speciality VARCHAR(20),  
    locatedAtAddress VARCHAR(100) NOT NULL,
```


FOREIGN KEY (locatedAtAddress) REFERENCES Location(address) *ON UPDATE CASCADE*
);

CREATE TABLE Responder(
 professionalID INT PRIMARY KEY,
 name VARCHAR(20),
 occupation VARCHAR(20),
 worksForBranchID INT NOT NULL,
 FOREIGN KEY (worksForBranchID) REFERENCES Department(branchID) *ON UPDATE CASCADE*
);

CREATE TABLE AssignedTo(
 incidentID INT,
 professionalID INT,
 PRIMARY KEY (incidentID, professionalID),
 FOREIGN KEY (incidentID) REFERENCES IncidentInfo(incidentID)
 ON DELETE CASCADE
 ON UPDATE CASCADE,
 FOREIGN KEY (professionalID) REFERENCES Responder(professionalID)
 ON DELETE CASCADE
 ON UPDATE CASCADE
);

CREATE TABLE InvolvedPerson(
 name VARCHAR(20),
 personID INT PRIMARY KEY,
 presentAtAddress VARCHAR(100) NOT NULL,
 FOREIGN KEY (presentAtAddress) REFERENCES Location(address)
);

CREATE TABLE Involves(
 incidentID INT,
 personID INT,
 PRIMARY KEY (incidentID, personID),

```
FOREIGN KEY (incidentID) REFERENCES IncidentInfo(incidentID)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
FOREIGN KEY (personID) REFERENCES InvolvedPerson(personID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

```
CREATE TABLE Suspect(
    personID INT PRIMARY KEY,
    physicalBuild VARCHAR(20),
    numPriorOffenses INT,
    FOREIGN KEY (personID) REFERENCES InvolvedPerson(personID)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);
```

```
CREATE TABLE Victim(
    personID INT PRIMARY KEY,
    injuries VARCHAR(100),
    FOREIGN KEY (personID) REFERENCES InvolvedPerson(personID)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);
```

```
CREATE TABLE Bystander(
    personID INT PRIMARY KEY,
    phoneNumber CHAR(10),
    FOREIGN KEY (personID) REFERENCES InvolvedPerson(personID)
        ON DELETE CASCADE
        ON UPDATE CASCADE
);
```

```
CREATE TABLE Reporter(
    name VARCHAR(50),
```

```
address VARCHAR(100),  
phoneNumber CHAR(10),  
email VARCHAR(50) PRIMARY KEY  
);
```

```
CREATE TABLE ReportedBy(  
    incidentID INT,  
    email VARCHAR(50),  
    dateReported DATE,  
    PRIMARY KEY (incidentID, email),  
    FOREIGN KEY (incidentID) REFERENCES IncidentInfo(incidentID)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE,  
    FOREIGN KEY (email) REFERENCES Reporter(email)  
        ON DELETE CASCADE  
);
```

```
CREATE TABLE EquipmentInfo(  
    type VARCHAR(20) PRIMARY KEY,  
    weight INT,  
    color VARCHAR(10)  
);
```

```
CREATE TABLE EquipmentItem(  
    equipmentID INT,  
    belongsToBranchID INT,  
    type VARCHAR(20),  
    PRIMARY KEY (equipmentID, belongsToBranchID),  
    FOREIGN KEY (belongsToBranchID) REFERENCES Department(branchID)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE,  
    FOREIGN KEY (type) REFERENCES EquipmentInfo(type)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE  
);
```

```
CREATE TABLE VehicleSpecs(  
    model VARCHAR(20) PRIMARY KEY,  
    numSeats INT  
);
```

```
CREATE TABLE VehicleInfo(  
    licensePlate CHAR(7) PRIMARY KEY,  
    model VARCHAR(20),  
    color VARCHAR(10),  
    ownedByBranchID INT NOT NULL,  
    FOREIGN KEY (model) REFERENCES VehicleSpecs(model)  
        ON DELETE CASCADE  
        ON UPDATE CASCADE,  
    FOREIGN KEY (ownedByBranchID) REFERENCES Department(branchID)  
);
```

Populate Tables:

```
INSERT  
INTO IncidentStatus(description, statusValue)  
VALUES  
(‘stabbing’, ‘in-progress’),  
(‘murder’, ‘in-progress’),  
(‘shooting’, ‘closed’),  
(‘gum-robbery’, ‘closed’),  
(‘shoe-robbery’, ‘in-progress’);
```

```
INSERT  
INTO IncidentInfo(incidentID, dateIncident, description)  
VALUES  
(1, 20-OCT-2023, ‘stabbing’),  
(2, 17-NOV-1981, ‘arson’),  
(3, 29-FEB-2020, ‘shooting’),
```

(4, 09-MAY-2017, 'bomb threat'),
(5, 05-JAN-2010, 'gun-robbery');

```
INSERT
INTO Location(address, neighbourhood)
VALUES
('2000 Granville St.', 'Downtown'),
('1000 Granville St.', 'Downtown'),
('3000 Granville St.', 'Downtown'),
('4000 Granville St.', 'Downtown'),
('6000 Granville St.', 'Downtown');
```

```
INSERT
INTO OccurredAt(incidentID, address)
VALUES
(1, '2000 Granville St.'),
(2, '1000 Granville St.'),
(3, '3000 Granville St.'),
(4, '4000 Granville St.'),
(5, '6000 Granville St.');
```

```
INSERT
INTO Department(branchID, specialty, address)
VALUES
(1, fire, 2000 Granville St.),
(2, police, 1000 Granville St.),
(3, police, 3000 Granville St.),
(4, hospital, 4000 Granville St.),
(5, court, 6000 Granville St.);
```

```
INSERT
INTO Responder(professionalID, name, occupation, worksForBranchID)
VALUES
(21, 'Jim Jones', 'firefighter', 1),
(31, 'Jason Somerville', 'police officer', 2),
```

```
(36, 'Charity Johnson', 'police officer', 3),  
(40, 'Melina del Cortez', 'nurse', 4),  
(53, 'Tyler Chu', 'lawyer', 5);
```

```
INSERT  
INTO AssignedTo(incidentID, professionalID)  
VALUES  
(1, 21),  
(2, 31),  
(3, 36),  
(4, 40),  
(5, 53);
```

```
INSERT  
INTO InvolvedPerson(name, personID, address)  
VALUES  
( 'John Smith', 10, '2000 Granville St.'),  
( 'Jon Smith', 12, '1000 Granville St.'),  
( 'Gianni Smith', 14, '3000 Granville St.'),  
( 'Johnny Smith', 16, '4000 Granville St.'),  
( 'Joanna Smith', 18, '6000 Granville St.');
```

```
INSERT  
INTO Involves(incidentID, personID)  
VALUES  
(1, 10),  
(2, 12),  
(3, 14),  
(4, 16),  
(5, 18);
```

```
INSERT  
INTO Suspect(personID, physicalBuild, numPriorOffenses)  
VALUES  
(10, 'stocky', 0),
```

```
(12, 'stocky', 1),  
(14, 'stocky', 3),  
(16, 'stocky', 20),  
(18, 'stocky', 0);
```

```
INSERT  
INTO Victim(personID, injuries)  
VALUES  
(10, 'broken arm'),  
(12, 'broken leg'),  
(14, 'heart failure'),  
(16, 'stroke'),  
(18, 'third degree burns');
```

```
INSERT  
INTO Bystander(personID, phoneNumber)  
VALUES  
(10, '6043223280'),  
(12, '7786043914'),  
(14, '6043210987'),  
(16, '6046046046'),  
(18, '7787787788');
```

```
INSERT  
INTO Reporter(name, address, phoneNumber, email)  
VALUES  
( 'Captain Marvel', '123 Hero Lane, Super City', '555-1111-987',  
'captain.marvel@gmail.com'),  
( 'Iron Man', '456 Stark Tower, New York City', '555-2222-987', 'iron.man@gmail.com'),  
( 'Wonder Woman', '789 Amazon Island, Paradise', '555-3333-987',  
'wonder.woman@gmail.com'),  
( 'Spider-Man', '101 Web Avenue, Marvel City', '555-4444-987',  
'spider.man@gmail.com'),  
( 'Black Widow', '202 Spy Street, Shadowland', '555-5555-987',  
'black.widow@gmail.com');
```

```
INSERT
INTO ReportedBy(incidentID, email, dateReported)
VALUES
(1, 'captain.marvel@example.com', '2023-09-15'),
(2, 'iron.man@gmail.com', '2023-08-28'),
(3, 'wonder.woman@gmail.com', '2023-07-12'),
(4, 'spider.man@gmail.com', '2023-06-05'),
(5, 'black.widow@gmail.com', '2023-05-20');
```

```
INSERT
INTO EquipmentInfo(type, weight, color)
VALUES
('fire-extinguisher', 10, 'red'),
('grenade', 5, 'black'),
('fish-extinguisher', 3, 'blue'),
('cat-catcher', 2, 'grey'),
('medical', 1, 'green');
```

```
INSERT
INTO EquipmentItem(equipmentID, belongsToBranchID, type)
VALUES
(1234, 1, 'fire-extinguisher'),
(1235, 2, 'grenade'),
(1236, 3, 'fish-extinguisher'),
(1237, 4, 'cat-catcher'),
(1238, 5, 'medical');
```

```
INSERT
INTO VehicleSpecs(model, numSeats)
VALUES
('corolla', 5),
('camry', 5),
('Rav4', 5),
```



```
('m3', 5),  
( 'GWagon', 5);
```

```
INSERT  
INTO VehicleInfo(licensePlate, model, color, ownedByBranchID)  
VALUES  
( 'DONDA4', 'corolla', 'black', 1),  
( 'YANDHI', 'camry', 'purple', 2),  
( 'DRIZZY', 'Rav4', 'beige', 3),  
( 'CHANCE3', 'm3', 'red', 4),  
( 'SHREK6', 'GWagon', 'orange', 5);
```