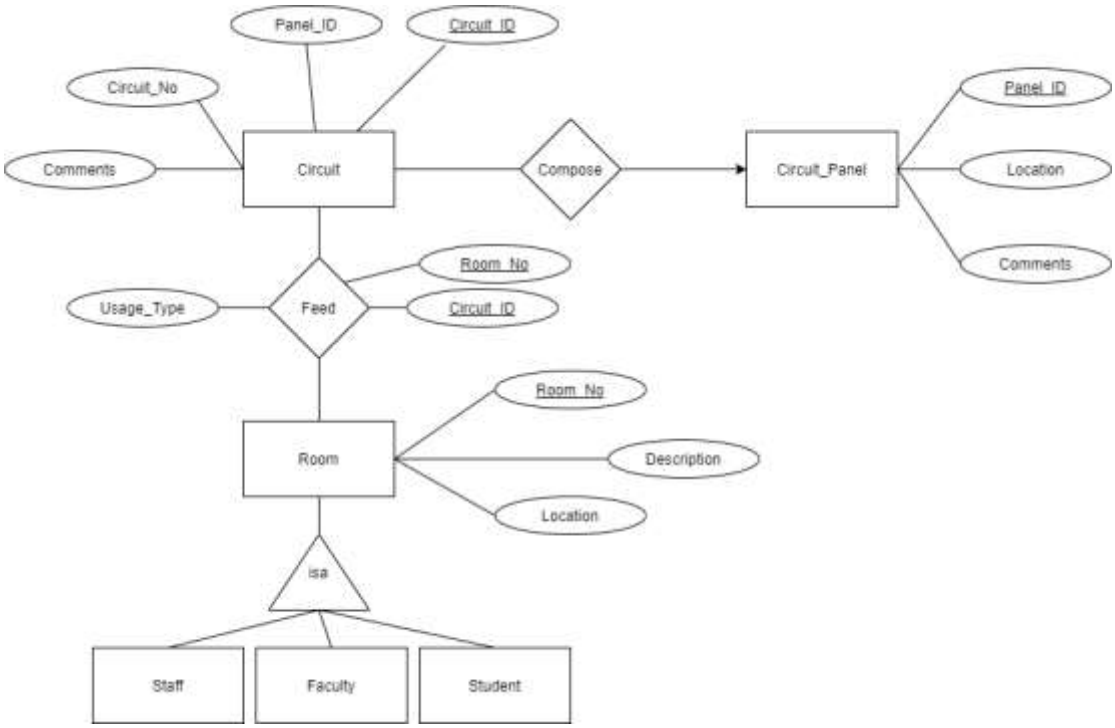
1



2.1

Circuit (Circuit_ID,Circuit_No, Panel_ID, Comments)

Circuit_ID=> Circuit_No, Panel_ID, Comments

Room (Room_No, Description, Location)

Room_No => Description, Location

Circuit_Panel (Panel_ID,Location,Comments)

Panel_ID => Location,Comments

Feed (Circuits_ID,Room_No, Usage_Type)

Circuits_ID,Room_No => Usage_Type

Staff (Room_No, Description, Location)

Room_No => Description, Location

Faculty (Room_No, Description, Location)

Room_No => Description, Location

Student (Room_No, Description, Location)

Room_No => Description, Location

Compose (Panel_ID,Circuit_ID)

2.2

No, it does not violate the BCNF because for all functional dependencies, the left-hand side is a key. Where a key is an attribute that determines all other attributes.

3.1

```
SELECT Description
FROM Room
WHERE Room_No = “118E”;
```

3.2

```
CREATE TABLE Circuit(
        Circuit_ID INT NOT NULL PRIMARY KEY,
        Circuit_No INT,
        Comments VARCHAR(50),
        Panel_ID INT
);
```

3.3

```
DELETE FROM Circuit_Panel
WHERE Comments = “TO BE DELETED”;
```

4.1

(e)

First, create tables that do not contain foreign keys (Circuit_Panel, Room), then create tables that contains foreign keys but are strong entities (Circuit), lastly create tables that are relations (with composite primary keys) that depend on other strong entities (Feeds).

4.2

```
SELECT *
FROM Circuit
WHERE circuitComments LIKE “%old%”
ORDER BY Panel_ID, Circuit_No;
```

4.3

```
SELECT COUNT( DISTINCT Circuit_ID)
FROM Feeds
WHERE Room_No = 121;
```