1 Introduction

In this project, we will need to design a web-based Smart Home Energy Management System. The project consists of two parts: database design and system implementation. In this part, we will describe a comprehensive database design. The organization of this part is as follows: In Section 2, we will conduct a needs analysis for the project, followed by table design, as well as assumptions and justifications for the design. Section 3 introduces the detailed creation of the database schema, specifying each table's columns, data types, and constraints. Finally, we will include some testing samples of the schema in Section 4.

2 Schema Design

2.1 Need Analysis

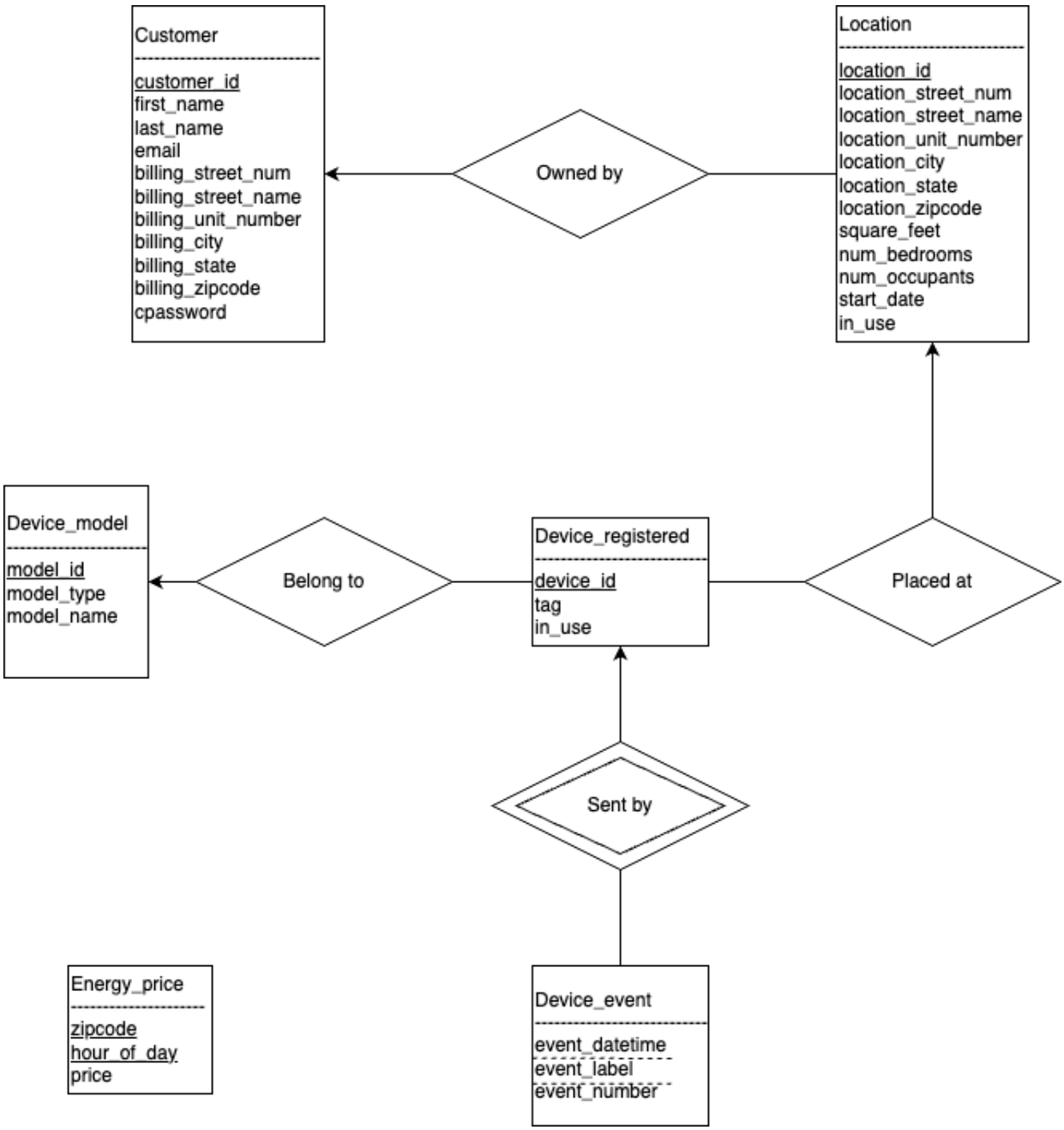
Based on the problem description, the information and functionalities needed can be concluded as follows:

- User System:
  - Sign up: Name, billing address, email
  - One user <- many locations
- Location:
  - Able to modify locations
  - Attributes:
    - Full address, including unit number
    - Start date
    - Square feet
    - Num bedrooms
    - Number of occupants of the unit
    - List of registered devices
- Devices
  - Type: AC, fridge, ...
  - Model: Samsung AC R100, ...
  - Ownership: User, Locations. (Can have more than one same model in a location by a user)
  - Can have an ID
- Devices Events(Signal / Information)
  - Switching On / Off (Time)
  - Setting Changed (Time, setting[possibly]). e.g. AC temperature, light bulb brightness, fridge door opened / closed, ...
  - Energy consumption
    - info / 5min: energy use
    - switched off: energy use since last update
- Event format
  - Device ID

    - time stamp
    - label (e.g. energy_consumption, temperature_lowered, door_opened, ...)
    - number (corresponding to label)
- Energy Price
    - Vary according to:
        - Hour
        - Zip Code

2.2 Entity-Relation Design

Based on the need analysis, the Entity-Relation Graph can be drawn:



2.3 Assumptions and Justifications

The ER-Gram is calculated based on some assumptions, some of which will actually affect the schema. Below is an description of all the assumptions made:

2.3.1 User & Location

Assumptions:

1. Each user can only have one billing address. So billing_address could be an attribute in user table.
2. Each home location can only belong to one account, so customer_id & start_date should be an attribute in address table.
3. We assume the location format as the following: (Line 1)street_number street_name (Line 2)unit_number (zip code info)city, state zipcode. Based on the atomic principals, addresses are separated into different columns (e.g. street_num, street_name, ...).
4. We also asuume the input addresses are all legal(they really exist, and in the right format).

2.3.2 Device & Event

Assumptions:

1. All the device models are in the list of device_model table, whenever a new device is promoted, we can modify the database to put it into the table.
2. User can only register devices of which the models are in the device_model table. So there won't be any records in the *device_registered* table with ambiguous model_id.
3. There are only limited number of event_label, and every the event_label revceived by the system should be legal.
4. As mentioned in the problem desciption, we don't have to model how the system prestored all the event_labels. So we assume that the events are automatically stored into the database(such process is not modeled). In the project, this process might be simulated by manaully insert data into the model_event table.
5. We assume that devices could generate two (or more) events at the same timeframe. However, such events will not have a same event_label. For example, when an AC is turened off, it generates two events: 'Off' and 'EnergyReport' with a same timeframe. Consequently, (device_id, event_label, event_datetime) is a candidate key of the *device_event* table.

2.3.3 Energy Price & Calculation

Assumptions:

1. The unit of energy price is *dollars / kwh*. The unit of the energy consumption reported by devices is *kwh*.
2. Because the energy price might vary between two consecutive hours, there might be situations where one EnergyReport has a time span across tow hours(e.g. first 2 minites in 7 and the other 3 minutes in 8). In this case, we choose the unit price of the second hour for calculation, resulting in an approximate energy price, which would not have significant difference with the actual price, given that the numbers of kwh used by home devices per 5 minutes are very small, and that the unit price of energy would not vary a lot.

2.4 Tables Design

2.4.1 User & Location Table

customer: (**customer_id**, first_name, last_name, email, billing_street_num, billing_street_name, billing_unit_number, billing_city, billing_state, billing_zipcode, cpassword, in_use)
location: (**location_id**, customer_id, location_street_num, location_street_name, location_unit_number, location_city, location_state, location_zipcode, square_feet, num_bedrooms, start_date, num_occupants, in_use)

2.4.2 Device & Event

device_model(**model_id**, model_type, model_name), *This is for prestoring devices for user to register*
device_registered(**device_id**, model_id, location_id, tag, in_use), *This is for devices registered by user*
device_event(**device_id**, **event_label**, **event_datetime**, event_number), *event_number corresponds to event_label*

2.4.3 Energy Price

energy_price(**zipcode**, **hour_of_day**, price), *Energy prices vary on hourly and locational basis*

2.4.4 Modifications from the design of part1*

The last part of the project designed some foreign key constraints for table *device_event* and *device_registered*, but did not specify the on delete action for them. Taking into account of the large amount of data in these table that is associated with *device_register* and *location*, and the logic that if any devices and locations are deleted, the records of the corresponding energy consumption should be reserved (in order to show the correct energy statistic charts), 'in_use' attributes are added into these table. When some locations or devices are deleted, 'in_use' column are set to false, thus preserving the integrity constraints.

3 Database Creation

In this part, we choose MySQL to implement the schema.

3.1 customer

```
CREATE TABLE customer(
    customer_id INT AUTO_INCREMENT,
    first_name VARCHAR(63) NOT NULL,
    last_name VARCHAR(63) NOT NULL,
    email VARCHAR(255) UNIQUE NOT NULL,
    billing_street_num INT NOT NULL,
```

```
    billing_street_name VARCHAR(127) NOT NULL,
    billing_unit_number VARCHAR(127) NOT NULL,
    billing_city VARCHAR(127) NOT NULL,
    billing_state VARCHAR(16) NOT NULL,
    billing_zipcode VARCHAR(5) NOT NULL,
    cpassword VARCHAR(127) NOT NULL,
    in_use BOOLEAN NOT NULL DEFAULT TRUE,
    PRIMARY KEY (customer_id)
);
```

3.2 location

```
CREATE TABLE location(
    location_id INT AUTO_INCREMENT,
    customer_id INT NOT NULL,
    location_street_num INT NOT NULL,
    location_street_name VARCHAR(127) NOT NULL,
    location_unit_number VARCHAR(127) NOT NULL,
    location_city VARCHAR(127) NOT NULL,
    location_state VARCHAR(127) NOT NULL,
    location_zipcode VARCHAR(5) NOT NULL,
    square_feet FLOAT NOT NULL,
    num_bedrooms INT NOT NULL,
    num_occupants INT NOT NULL,
    start_date DATETIME NOT NULL,
    in_use BOOLEAN NOT NULL DEFAULT TRUE,
    PRIMARY KEY (location_id),
    FOREIGN KEY (customer_id) REFERENCES customer(customer_id) ON
DELETE CASCADE
);
```

3.3 device_model

```
CREATE TABLE device_model(
    model_id INT AUTO_INCREMENT,
    model_type VARCHAR(127) NOT NULL,
    model_name VARCHAR(127) NOT NULL,
    PRIMARY KEY (model_id)
);
```

3.4 device_registered

```
CREATE TABLE device_registered(
    device_id INT AUTO_INCREMENT,
    model_id INT NOT NULL,
    location_id INT NOT NULL,
    tag VARCHAR(255),
```

```
    in_use BOOLEAN DEFAULT TRUE,
    PRIMARY KEY (device_id),
    FOREIGN KEY (model_id) REFERENCES device_model(model_id) ON
DELETE CASCADE,
    FOREIGN KEY (location_id) REFERENCES location(location_id) ON
DELETE CASCADE
);
```

3.5 device_event

```
CREATE TABLE device_event(
    device_id INT NOT NULL,
    event_datetime DATETIME DEFAULT CURRENT_TIMESTAMP,
    event_label VARCHAR(63) NOT NULL,
    event_number FLOAT,
    PRIMARY KEY (device_id, event_datetime, event_label),
    FOREIGN KEY (device_id) REFERENCES device_registered(device_id)
ON DELETE CASCADE
);
```

3.6 energy_price(zipcode, hour_of_day, price)

```
CREATE TABLE energy_price(
    zipcode VARCHAR(5),
    hour_of_day INT NOT NULL,
    price FLOAT NOT NULL,
    PRIMARY KEY (zipcode, hour_of_day)
);
```

4 SQL

Before testing the queries, meaningful data has been inserted into the database to ensure the testing results:

customer_id	first_name	last_name	email	billing_street_num	billing_street_name	billing_unit_number	billing_city	billing_state	billing_zipcode	cpassword
1	John	Doe	john.doe@example.com	123	5th Avenue	Apt 1005	New York	NY	10003	HelloPassWord1
2	Jane	Smith	jane.smith@example.com	456	Madison Avenue	Apt 2406	New York	NY	10022	HelloPassWord1
3	Alice	Johnson	alice.johnson@example.com	789	Park Avenue	Apt 0701	New York	NY	11206	HelloPassWord1
4	Bob	Williams	bob.williams@example.com	101	Broadway	Apt 2803	New York	NY	11249	HelloPassWord1
5	Emily	Brown	emily.brown@example.com	202	Wall Street	Apt 1502s	New York	NY	10005	HelloPassWord1

5 rows in set (0.03 sec)

location_id	customer_id	location_street_num	location_street_name	location_unit_number	location_city	location_state	location_zipcode	square_feet	num_bedrooms	num_occupants	start_date
1	1	123	5th Avenue	Apt 1005	New York	NY	10003	865	2	2	2022-07-31 23:59:59
2	1	465	Washington Boulevard	Apt 1303	Jersey City	NJ	07310	789	2	2	2022-07-31 23:59:59
3	2	456	Madison Avenue	Apt 2406	New York	NY	10022	713	1	2	2022-07-31 23:59:59
4	3	789	Park Avenue	Apt 0701	New York	NY	11206	803	2	3	2022-07-31 23:59:59
5	3	25	Christopher Columbus Drive	Apt 3403	Jersey City	NJ	07302	875	2	3	2022-07-31 23:59:59
6	4	101	Broadway	Apt 2803	New York	NY	11249	698	2	1	2022-07-31 23:59:59
7	5	202	Wall Street	Apt 1502s	New York	NY	10005	715	1	1	2022-07-31 23:59:59

7 rows in set (0.04 sec)

model_id	model_type	model_name
1	Refrigerator	Samsung Family Hub Refrigerator
2	Refrigerator	LG Smart Wi-Fi Enabled Refrigerator
3	Air Conditioner	Daikin Smart Inverter AC
4	Air Conditioner	Samsung Wind-Free Air Conditioner
5	Light Bulb	Philips Hue White and Color Ambiance
6	Light Bulb	LIFX A60 Smart Bulb
7	Smart Speaker	Amazon Echo Dot
8	Smart Speaker	Google Nest Audio

8 rows in set (0.03 sec)

device_id	model_id	location_id	tag
1	5	1	Light Bulb in the Bedroom
2	6	1	Light Bulb in the Kitchen
3	5	1	Light Bulb in the Living Room
4	1	1	Refrigerator in the Kitchen
5	3	1	AC in the Living Room
6	7	1	Smart Speaker in the Bedroom
7	6	2	Light Bulb in the Bedroom
8	5	2	Light Bulb in the Kitchen
9	6	2	Light Bulb in the Living Room
10	2	2	Refrigerator in the Kitchen
11	4	2	AC in the Living Room
12	5	3	Light Bulb in the Bedroom
13	5	3	Light Bulb in the Kitchen
14	6	3	Light Bulb in the Living Room
15	1	3	Refrigerator in the Kitchen
16	3	3	AC in the Living Room
17	8	3	Smart Speaker in the Bedroom
18	6	4	Light Bulb in the Bedroom
19	6	4	Light Bulb in the Kitchen
20	5	4	Light Bulb in the Living Room
21	2	4	Refrigerator in the Kitchen
22	4	4	AC in the Living Room
23	5	5	Light Bulb in the Bedroom
24	5	5	Light Bulb in the Kitchen
25	6	5	Light Bulb in the Living Room
26	1	5	Refrigerator in the Kitchen
27	3	5	AC in the Living Room
28	7	5	Smart Speaker in the Bedroom
29	6	6	Light Bulb in the Bedroom
30	6	6	Light Bulb in the Kitchen
31	5	6	Light Bulb in the Living Room
32	2	6	Refrigerator in the Kitchen
33	4	6	AC in the Living Room
34	5	7	Light Bulb in the Bedroom
35	6	7	Light Bulb in the Kitchen
36	5	7	Light Bulb in the Living Room
37	1	7	Refrigerator in the Kitchen
38	3	7	AC in the Living Room
39	8	7	Smart Speaker in the Bedroom

39 rows in set (0.03 sec)

```
mysql> select * from device_event limit 30
    -> ;
+-----------+---------------------+-------------+--------------+
| device_id | event_datetime      | event_label | event_number |
+-----------+---------------------+-------------+--------------+
|         1 | 2022-08-01 07:06:00 | On          |         NULL |
|         1 | 2022-08-01 07:11:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:16:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:21:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:26:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:31:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:36:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:41:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:46:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:51:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 07:56:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:01:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:06:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:11:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:16:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:21:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:26:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:31:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:36:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:41:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:46:00 | EnergyReport |       0.0075 |
|         1 | 2022-08-01 08:48:00 | DarkMode    |         NULL |
|         1 | 2022-08-01 08:51:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 08:56:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:01:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:06:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:11:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:16:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:21:00 | EnergyReport |        0.005 |
|         1 | 2022-08-01 09:26:00 | EnergyReport |        0.005 |
+-----------+---------------------+-------------+--------------+
30 rows in set (0.03 sec)
```

```
mysql> select * from energy_price limit 30;
+---------+-------------+-------+
| zipcode | hour_of_day | price |
+---------+-------------+-------+
| 07097   |           0 |  0.08 |
| 07097   |           1 |  0.08 |
| 07097   |           2 |  0.08 |
| 07097   |           3 |  0.08 |
| 07097   |           4 |  0.08 |
| 07097   |           5 |  0.08 |
| 07097   |           6 |  0.08 |
| 07097   |           7 |  0.08 |
| 07097   |           8 |  0.13 |
| 07097   |           9 |  0.13 |
| 07097   |          10 |  0.13 |
| 07097   |          11 |  0.13 |
| 07097   |          12 |  0.13 |
| 07097   |          13 |  0.13 |
| 07097   |          14 |  0.13 |
| 07097   |          15 |  0.13 |
| 07097   |          16 | 0.184 |
| 07097   |          17 | 0.184 |
| 07097   |          18 | 0.184 |
| 07097   |          19 | 0.184 |
| 07097   |          20 | 0.184 |
| 07097   |          21 | 0.184 |
| 07097   |          22 | 0.184 |
| 07097   |          23 | 0.184 |
| 07302   |           0 |  0.08 |
| 07302   |           1 |  0.08 |
| 07302   |           2 |  0.08 |
| 07302   |           3 |  0.08 |
| 07302   |           4 |  0.08 |
| 07302   |           5 |  0.08 |
+---------+-------------+-------+
30 rows in set (0.03 sec)
```

1. List all enrolled devices with their total energy consumption in the last 24 hours, for a specific customer identified by customer ID.
   Assume the current time is 2022-08-17 14:00:00(replaced with NOW() in real application scenarios)

```
SELECT
    dr.device_id,
    SUM(de.event_number) AS total_energy_consumption
FROM
    device_event de
    JOIN device_registered dr ON de.device_id = dr.device_id
    JOIN location l ON l.location_id = dr.location_id
WHERE
    de.event_label = 'EnergyReport'
    AND customer_id = 1
GROUP BY
    de.device_id
HAVING
    MAX(de.event_datetime) >= '2022-08-17 14:00:00' - INTERVAL 24
HOUR;
```

```
+-----------+----------------------------+
| device_id | total_energy_consumption   |
+-----------+----------------------------+
|         1 |       1.3749999713618308   |
|         2 |       1.5439999671652913   |
|         3 |       1.0474999791476876   |
|         4 |        107.0999982431531   |
|         7 |       1.4579999695997685   |
|         8 |       0.8994999811984599   |
|         9 |       1.8564999600639567   |
|        10 |        88.76600099727511   |
+-----------+----------------------------+
8 rows in set (0.08 sec)
```

2. Calculate the average monthly energy consumption per device type, for the month of August 2022, considering only devices that have been on (i.e., reported data) at least once during that month.

```
SELECT
    dm.model_type,
    AVG(total_energy_consumption) AS average_energy_consumption
FROM
    (
        # first compute all the device monthly energy consumption
        SELECT
            device_id,
            SUM(event_number) AS total_energy_consumption
        FROM
            device_event
```

```
        WHERE
            event_label = 'EnergyReport'
            AND event_datetime >= '2021-08-01'
            AND event_datetime < '2023-09-01'
        GROUP BY
            device_id
        HAVING
            SUM(event_number) IS NOT NULL # only consider devices
that have on at least once
    ) AS subquery
    JOIN device_registered dr ON subquery.device_id = dr.device_id
    JOIN device_model dm ON dr.model_id = dm.model_id
    JOIN location l ON dr.location_id = l.location_id
    WHERE l.customer_id = 2
GROUP BY
    dm.model_type;
```

```
+--------------+-----------------------------+
| model_type   | average_energy_consumption  |
+--------------+-----------------------------+
| Refrigerator |          49.482142542355824 |
| Light Bulb   |          0.7478571271390787 |
+--------------+-----------------------------+
2 rows in set (0.05 sec)
```

3. Identify cases where a refrigerator door was left open for more than 30 minutes. Output the date and time, the service location, the device ID, and the refrigerator model.

```
SELECT
    CONCAT(
        l.location_street_num,
        ' ',
        l.location_street_name,
        ', Unit ',
        l.location_unit_number,
        ', ',
        l.location_city,
        ', ',
        l.location_state,
        ' ',
        l.location_zipcode
    ) AS location,
    dr.device_id,
    dm.model_type,
    de.event_datetime AS OpenTime
FROM
    device_event de
    JOIN device_registered dr ON de.device_id = dr.device_id
    JOIN device_model dm ON dr.model_id = dm.model_id
    JOIN location l ON l.location_id = dr.location_id
```

```
WHERE
    de.event_label = 'DoorOpen'
    AND dm.model_type = 'Refrigerator'
    AND NOT EXISTS(
        SELECT
            *
        FROM
            device_event de2
        WHERE
            de2.device_id = de.device_id
            AND de2.event_label = 'DoorClose'
            AND de2.event_datetime > de.event_datetime
            AND TIMEDIFF(de2.event_datetime, de.event_datetime) <=
'00:30:00'
    );
```

location	device_id	model_type	OpenTime
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-08-05 07:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-08-10 03:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-08-15 04:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-08-20 03:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-08-25 05:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-05 06:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-10 04:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-15 04:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-20 07:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-25 07:00:00
123 5th Avenue, Unit Apt 1005, New York, NY 10003	4	Refrigerator	2022-09-30 03:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-08-05 07:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-08-10 00:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-08-25 07:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-08-30 04:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-09-05 02:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-09-15 05:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-09-20 04:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-09-25 04:00:00
465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310	10	Refrigerator	2022-09-30 00:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-08-05 02:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-08-10 03:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-08-15 03:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-08-20 01:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-09-05 00:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-09-15 05:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-09-20 03:00:00
456 Madison Avenue, Unit Apt 2406, New York, NY 10022	15	Refrigerator	2022-09-30 07:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-08-05 03:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-08-10 03:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-08-15 07:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-08-20 01:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-09-10 03:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-09-15 05:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-09-20 02:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-09-25 04:00:00
789 Park Avenue, Unit Apt 0701, New York, NY 11206	21	Refrigerator	2022-09-30 06:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-08-05 01:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-08-10 04:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-08-15 00:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-08-20 03:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-09-05 04:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-09-10 06:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-09-15 01:00:00
25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302	26	Refrigerator	2022-09-20 06:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-08-05 02:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-08-15 03:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-08-20 07:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-08-25 01:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-08-30 06:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-09-15 06:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-09-20 06:00:00
101 Broadway, Unit Apt 2803, New York, NY 11249	32	Refrigerator	2022-09-30 02:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-08-10 03:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-08-15 01:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-08-20 07:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-08-25 05:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-08-30 07:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-05 01:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-10 02:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-15 06:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-20 01:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-25 05:00:00
202 Wall Street, Unit Apt 1502s, New York, NY 10005	37	Refrigerator	2022-09-30 02:00:00

64 rows in set, 88 warnings (0.16 sec)

4. Calculate the total energy cost for each service location during August 2022, considering the hourly changing energy prices based on zip code.

```
SELECT
    CONCAT(
        l.location_street_num,
        ' ',
        l.location_street_name,
        ', Unit ',
        l.location_unit_number,
        ', ',
        l.location_city,
        ', ',
```

```
        l.location_state,
        ' ',
        l.location_zipcode
    ) AS location,
    dr.location_id,
    SUM(ep.price * de.event_number) as monthlyCostSum
FROM
    device_event de
    JOIN device_registered dr ON de.device_id = dr.device_id
    JOIN location l ON dr.location_id = l.location_id
    JOIN energy_price ep ON ep.zipcode = l.location_zipcode
    AND ep.hour_of_day = HOUR(de.event_datetime)
WHERE
    de.event_label = 'EnergyReport'
    AND de.event_datetime BETWEEN "2022-08-01"
    AND "2022-08-31"
GROUP BY
    dr.location_id;
```

```
+-----------------------------------------------------------------+-------------+--------------------+
| location                                                        | location_id | monthlyCostSum     |
+-----------------------------------------------------------------+-------------+--------------------+
| 123 5th Avenue, Unit Apt 1005, New York, NY 10003               |           1 |  8.779507985096869 |
| 465 Washington Boulevard, Unit Apt 1303, Jersey City, NJ 07310  |           2 |  6.077616970085041 |
| 456 Madison Avenue, Unit Apt 2406, New York, NY 10022           |           3 | 10.043581828314661 |
| 789 Park Avenue, Unit Apt 0701, New York, NY 11206              |           4 |  7.505406670015111 |
| 25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302 |       5 |  7.145635752277388 |
| 101 Broadway, Unit Apt 2803, New York, NY 11249                 |           6 |  6.854043528100691 |
| 202 Wall Street, Unit Apt 1502s, New York, NY 10005             |           7 |  8.881512977744334 |
+-----------------------------------------------------------------+-------------+--------------------+
7 rows in set (0.07 sec)
```

5. For each service location, compute its total energy consumption during August 2022, as a percentage of the average total energy consumption during the same time of other service locations that have a similar square footage (meaning, at most 5% higher or lower square footage). Thus, you would output 150% if a service location with 1000 sqft had 50% higher energy consumption than the average of other service locations that have between 950 and 1050 sqft.

```
# obtain each cost of the location
WITH EachCosts AS(
    SELECT
        dr.location_id,
        SUM(ep.price * de.event_number) as monthlyCostSum,
        l.square_feet
    FROM
        device_event de
        JOIN device_registered dr ON de.device_id = dr.device_id
        JOIN location l ON dr.location_id = l.location_id
        JOIN energy_price ep ON ep.zipcode = l.location_zipcode
        AND ep.hour_of_day = HOUR(de.event_datetime)
    WHERE
        de.event_label = 'EnergyReport'
        AND de.event_datetime BETWEEN "2022-08-01"
        AND "2022-08-31"
    GROUP BY
        dr.location_id
```

```
),
AvgCosts AS (
    SELECT
        E1.location_id,
        AVG(E2.monthlyCostSum) AS avgSimilarSquareFeetCost
    FROM
        EachCosts E1
        JOIN EachCosts E2 ON E1.location_id != E2.location_id
        AND E2.square_feet BETWEEN E1.square_feet * 0.95
        AND E1.square_feet * 1.05
    GROUP BY
        E1.location_id
)
SELECT
    e.location_id,
    e.monthlyCostSum,
    a.avgSimilarSquareFeetCost,
    (e.monthlyCostSum / a.avgSimilarSquareFeetCost) * 100 AS
percentageCost
FROM
    EachCosts e
    JOIN AvgCosts a ON e.location_id = a.location_id;
```

```
+-------------+--------------------+--------------------------+--------------------+
| location_id | monthlyCostSum     | avgSimilarSquareFeetCost | percentageCost     |
+-------------+--------------------+--------------------------+--------------------+
|           1 |  8.779507985096869 |        7.145635752277388 | 122.86531652972583 |
|           2 |  6.077616970085041 |        7.505406670015111 |  80.97651782635256 |
|           3 | 10.043581828314661 |        7.867778252922513 |  127.6546123371479 |
|           4 |  7.505406670015111 |        6.077616970085041 | 123.49259104280294 |
|           5 |  7.145635752277388 |        8.779507985096869 |  81.38993397360123 |
|           6 |  6.854043528100691 |        9.462547403029497 |  72.43338644630012 |
|           7 |  8.881512977744334 |        8.448812678207677 | 105.12143322401664 |
+-------------+--------------------+--------------------------+--------------------+
7 rows in set (0.08 sec)
```

6. Identify service location(s) that had the highest percentage increase in energy consumption between August and September of 2022.

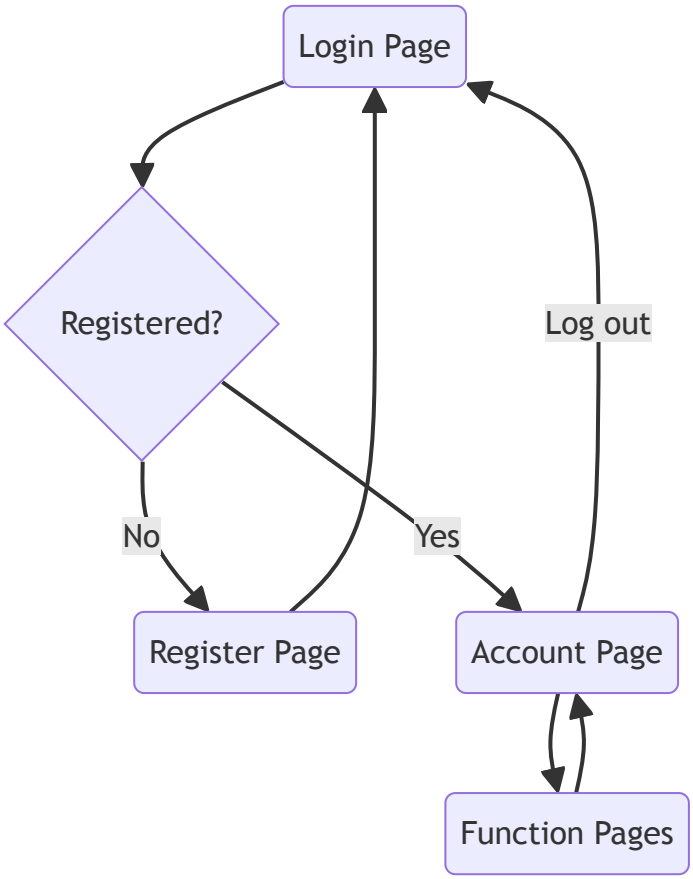
```
WITH AugCosts AS (
    SELECT
        CONCAT(
            l.location_street_num,
            ' ',
            l.location_street_name,
            ', Unit ',
            l.location_unit_number,
            ', ',
            l.location_city,
            ', ',
            l.location_state,
            ' ',
            l.location_zipcode
        ) AS location,
        dr.location_id,
        SUM(ep.price * de.event_number) as monthlyCostSum
```

```
    FROM
        device_event de
        JOIN device_registered dr ON de.device_id = dr.device_id
        JOIN location l ON dr.location_id = l.location_id
        JOIN energy_price ep ON ep.zipcode = l.location_zipcode
        AND ep.hour_of_day = HOUR(de.event_datetime)
    WHERE
        de.event_label = 'EnergyReport'
        AND de.event_datetime BETWEEN "2022-08-01"
        AND "2022-08-31"
    GROUP BY
        dr.location_id
),
SepCosts AS (
    SELECT
        CONCAT(
            l.location_street_num,
            ' ',
            l.location_street_name,
            ', Unit ',
            l.location_unit_number,
            ', ',
            l.location_city,
            ', ',
            l.location_state,
            ' ',
            l.location_zipcode
        ) AS location,
        dr.location_id,
        SUM(ep.price * de.event_number) as monthlyCostSum
    FROM
        device_event de
        JOIN device_registered dr ON de.device_id = dr.device_id
        JOIN location l ON dr.location_id = l.location_id
        JOIN energy_price ep ON ep.zipcode = l.location_zipcode
        AND ep.hour_of_day = HOUR(de.event_datetime)
    WHERE
        de.event_label = 'EnergyReport'
        AND de.event_datetime BETWEEN "2022-09-01"
        AND "2022-09-30"
    GROUP BY
        dr.location_id
)
SELECT
    a.location,
    a.location_id,
    (
        (s.monthlyCostSum - a.monthlyCostSum) / a.monthlyCostSum
    ) * 100 AS percentage_increase
FROM
    AugCosts a
    JOIN SepCosts s ON a.location_id = s.location_id
ORDER BY
    percentage_increase DESC
```

```
  LIMIT
      1;
```

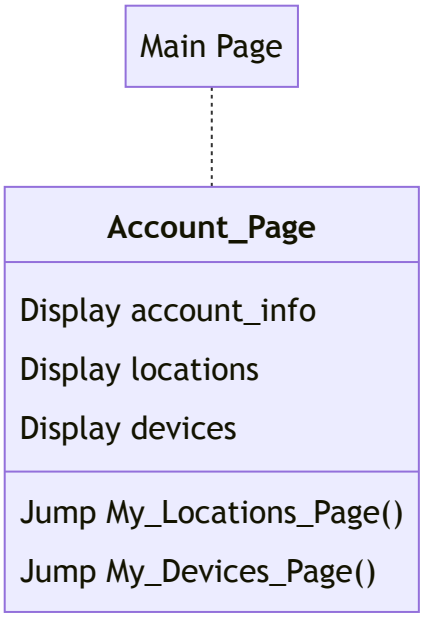
```
+------------------------------------------------------------------------+-------------+---------------------+
| location                                                               | location_id | percentage_increase |
+------------------------------------------------------------------------+-------------+---------------------+
| 25 Christopher Columbus Drive, Unit Apt 3403, Jersey City, NJ 07302 |           5 | -15.689982043561837 |
+------------------------------------------------------------------------+-------------+---------------------+
1 row in set (0.13 sec)
```

Basic Pages and Functions Design



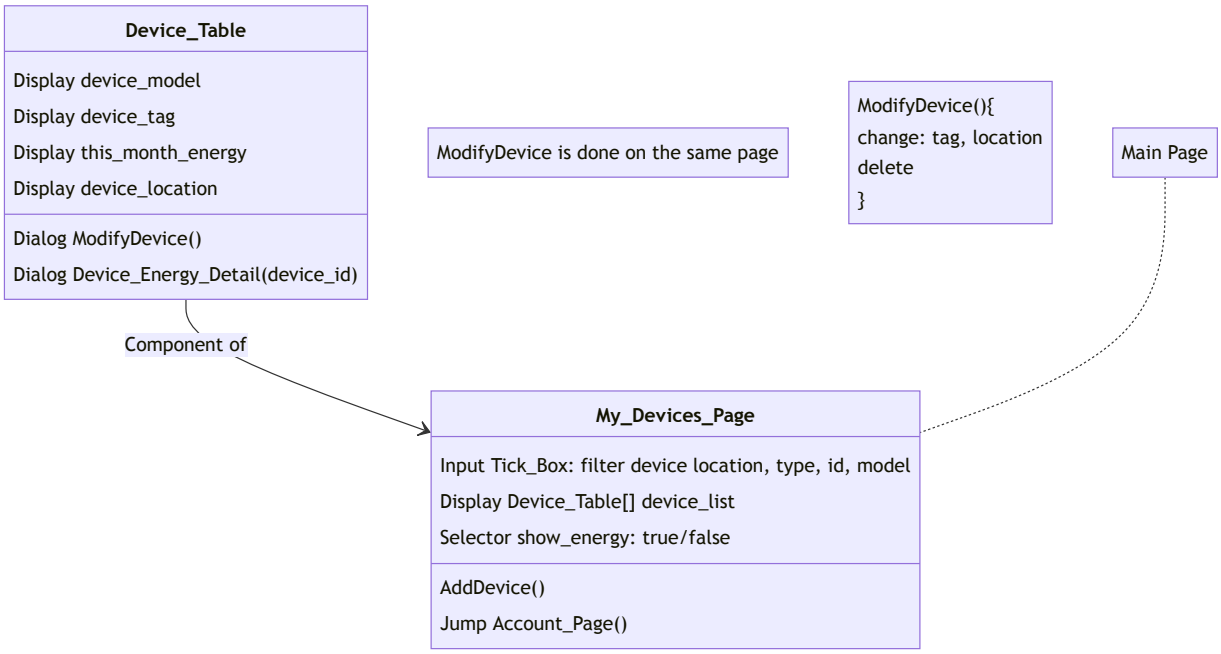
All the pages, excluding the login and register page, will require login. The server should detect if the user have logged in to the system, if not, they should be redirected to login page.

Account



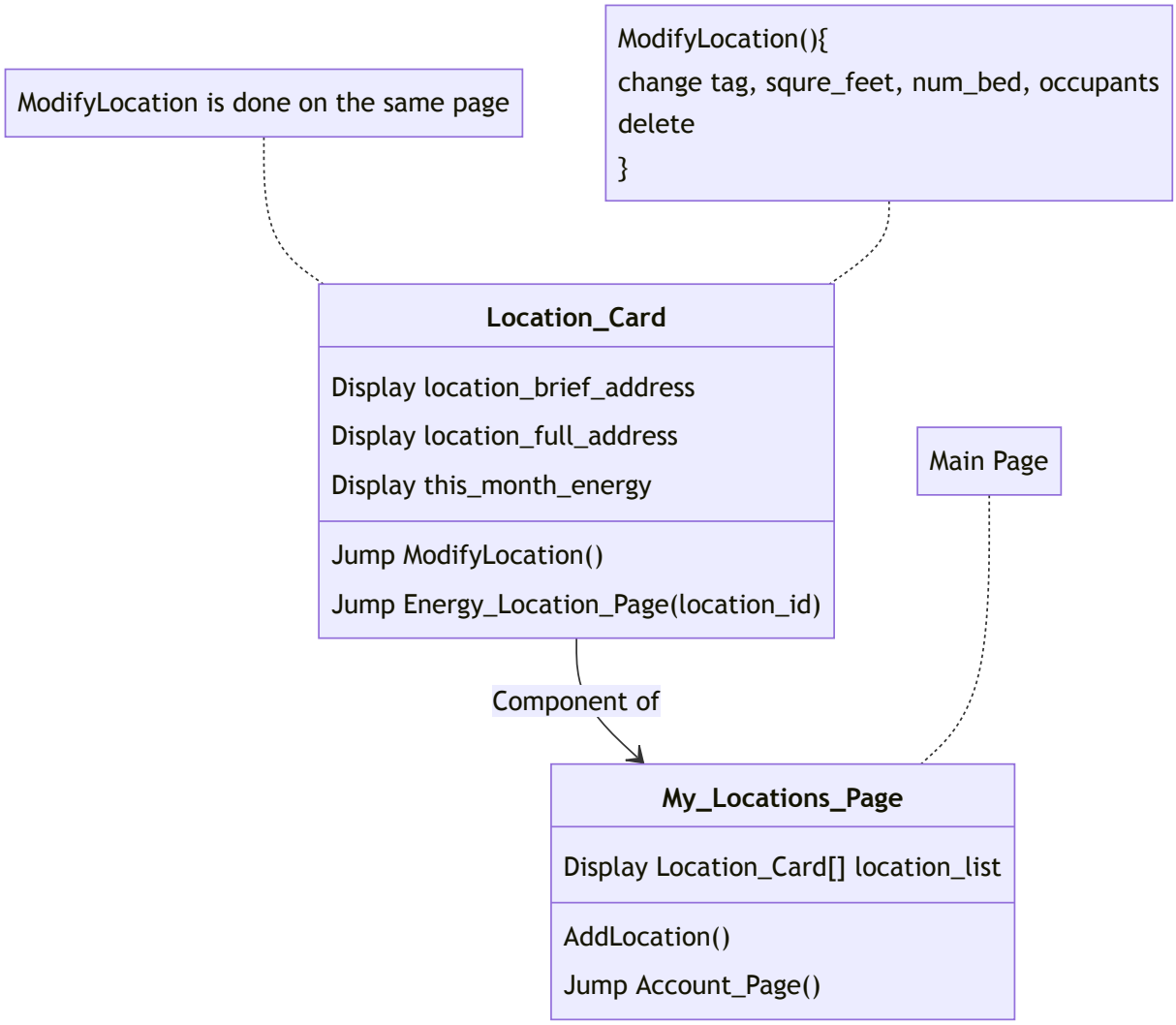
Account page should display the basic information of the customer: account information(name, email, billing address, etc.), list of locations(short address for each location), list of devices(device model and type for each device). This page will also provide the buttons navigating to other pages as well.

Devices



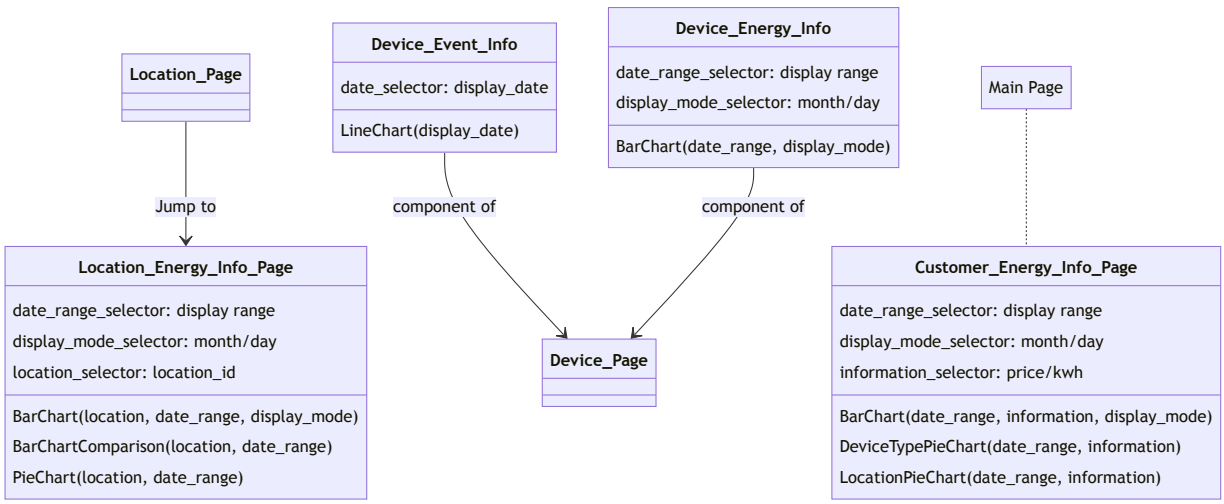
Device page should show a table listing all the registered devices belonging to the account. For each device, the table entry is designed to show all the information of the deivce(model, tag). The energy overview of each device should also be shown in the entry. To provide more intuitive interactions, a selector defining what the table is showing(device info / energy) should be provided in this page.

Locations



Location page is designed to display a list of location cards, each showing the information of one location, including location address, detailed information, and buttons linking to the modification dialog and location energy page, which will be shown later.

EnergyInfo



There would be three components/pages of energy information, corresponding to the field of account, device, location. Account information page would show the overview of all the energy usage of all locations registered by the customer. It should also provide date range selector, display mode, etc. to facilitate various choices of view for the energy/fee consumption.

Location energy page is designed similarly, adding an extra barchart comparing the energy information between current location and other locations with a close number of square feet. It also should provide vairous selection of displayed view. Device Event and Energy are set to be components of the device information page. Triggered by the buttons of the entry in device table, they would show the energy consumption on daily/monthly basis, and the event(Turned on/off, door opened/closed, etc.) genreated by the device.