RENEWABLE ENERGY TRACKING SYSTEM

DATABASE MANAGEMENT MINI-PROJECT

UCE2021422: Niharika Gurav
UCE2021423: Aaditee Iche
UCE2021424: Bhargavi Jadhav
UCE2021434: Hitali Khachane
SY COMP A (A2 Batch)

INTRODUCTION:

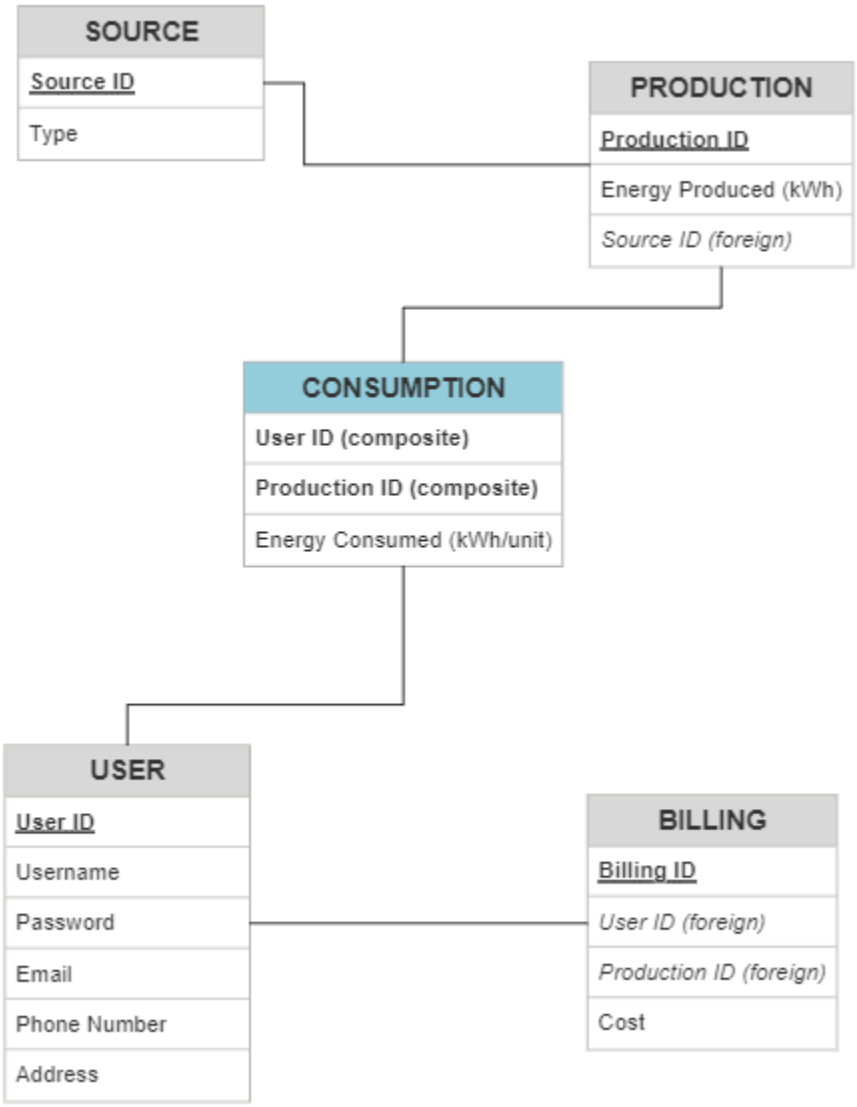
A renewable energy tracking system is a software platform that allows users to monitor and analyze renewable energy production and consumption data in real-time.The system typically aggregates data from various sources, including weather data, power production data, and energy consumption data, and presents it in an easy-to-understand dashboard.

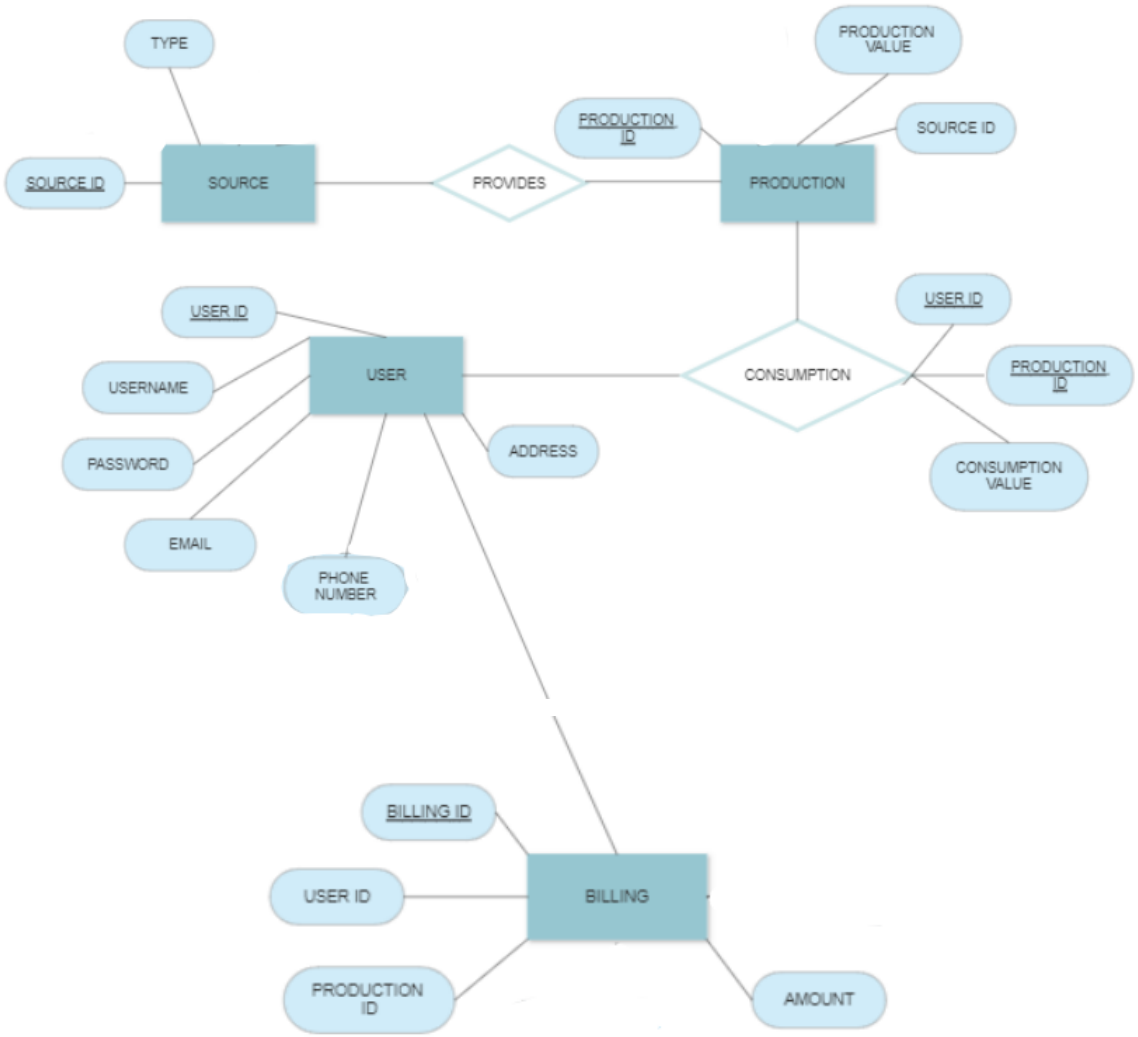
DATABASE REQUIREMENTS:

- Data Sources: Meteorological Organizations, Government and Energy Regulatory Websites
- Data Types: Energy Production and Consumption Data, Operational Data
- Data Access: Admin and User

1.IDE:Eclipse 4.23.0
2.Web Browser Chrome
3.Database Support:MYSQL Server 8.0.31
4.Operating System:Windows 11
5.C:\User\Hitali>java--version
java 18.0.1.2022-04-22
java(TM) SE Runtime Environment(build 18.0.1.1+2-6)
java HotSpot(TM) 64-Bit Server VM (build 18.0.1.1+2-6,mixed mode,sharing)
Swing (version 6)

ENTITY RELATIONSHIP DIAGRAM:





NORMALIZATION:

To justify the normalization of the provided tables to 1NF, 2NF, and 3NF, let's analyze each table and its columns.

1. Source table:

- Source ID (Primary Key): Unique identifier for each energy source.
- Type: Type or category of the energy source.

The Source table is already in 3NF because it has a primary key and atomic (indivisible) values in each column.

2. Production table:
- Production ID (Primary Key): Unique identifier for each production entry.
- Energy Produced: Amount of energy produced.
- Source ID (Foreign Key): Relates to the Source table.

To normalize this table to 2NF, we need to ensure that there are no partial dependencies. In this case, we have a single candidate key, which is the combination of Production ID and Source ID. The Energy Produced column depends on the Production ID, but not on the Source ID, so it is not a partial dependency.

To normalize this table to 3NF, we need to eliminate transitive dependencies. There are no transitive dependencies in this table since Energy Produced depends directly on the Production ID.

3. Consumption table:
- User ID: Identifier for each user.
- Production ID (Foreign Key): Relates to the Production table.
- Energy Consumed: Amount of energy consumed.

To normalize this table to 2NF, we need to ensure that there are no partial dependencies. In this case, we have a composite key, which is the combination of User ID and Production ID. The Energy Consumed column depends on both User ID and Production ID, so it is not a partial dependency.

To normalize this table to 3NF, we need to eliminate transitive dependencies. There are no transitive dependencies in this table since Energy Consumed depends directly on the User ID and Production ID.

4. User table:
- User ID (Primary Key): Unique identifier for each user.
- Username: User's username.
- Password: User's password.
- Email: User's email address.

- Phone Number: User's phone number.
- Address: User's address.

The User table is already in 1NF because it has a primary key and atomic values in each column.

5. Billing table:
- User ID (Foreign Key): Relates to the User table.
- Production ID (Foreign Key): Relates to the Production table.
- Cost: Cost associated with the energy consumption.

To normalize this table to 2NF, we need to ensure that there are no partial dependencies. In this case, we have a composite key, which is the combination of User ID and Production ID. The Cost column depends on both User ID and Production ID, so it is not a partial dependency.

To normalize this table to 3NF, we need to eliminate transitive dependencies. There are no transitive dependencies in this table since Cost depends directly on the User ID and Production ID.

In conclusion, based on the information provided, the tables have been normalized to at least the 3rd Normal Form (3NF) as they meet the criteria for 1NF, 2NF, and 3NF.
