CHAPTER 3

Research Methodology

To develop the software which is composed of decentralized power sources and utility grid, the data of power consumption and power production from Photovoltaic system was taken from 2 institutions. In the Case I, the energy/power flow behavior was analyzed from four buildings at School of Renewable Energy Technology (SERT), Naresuan University. Then, the energy/power flow was used to create the algorithm for identifying the real-time power flow and develop the software in Case II by using the algorithm for eleven buildings at University of Phayao, Thailand. The research method has been separated into 2 phase and 5 steps as shown in Figure 3.1 below:

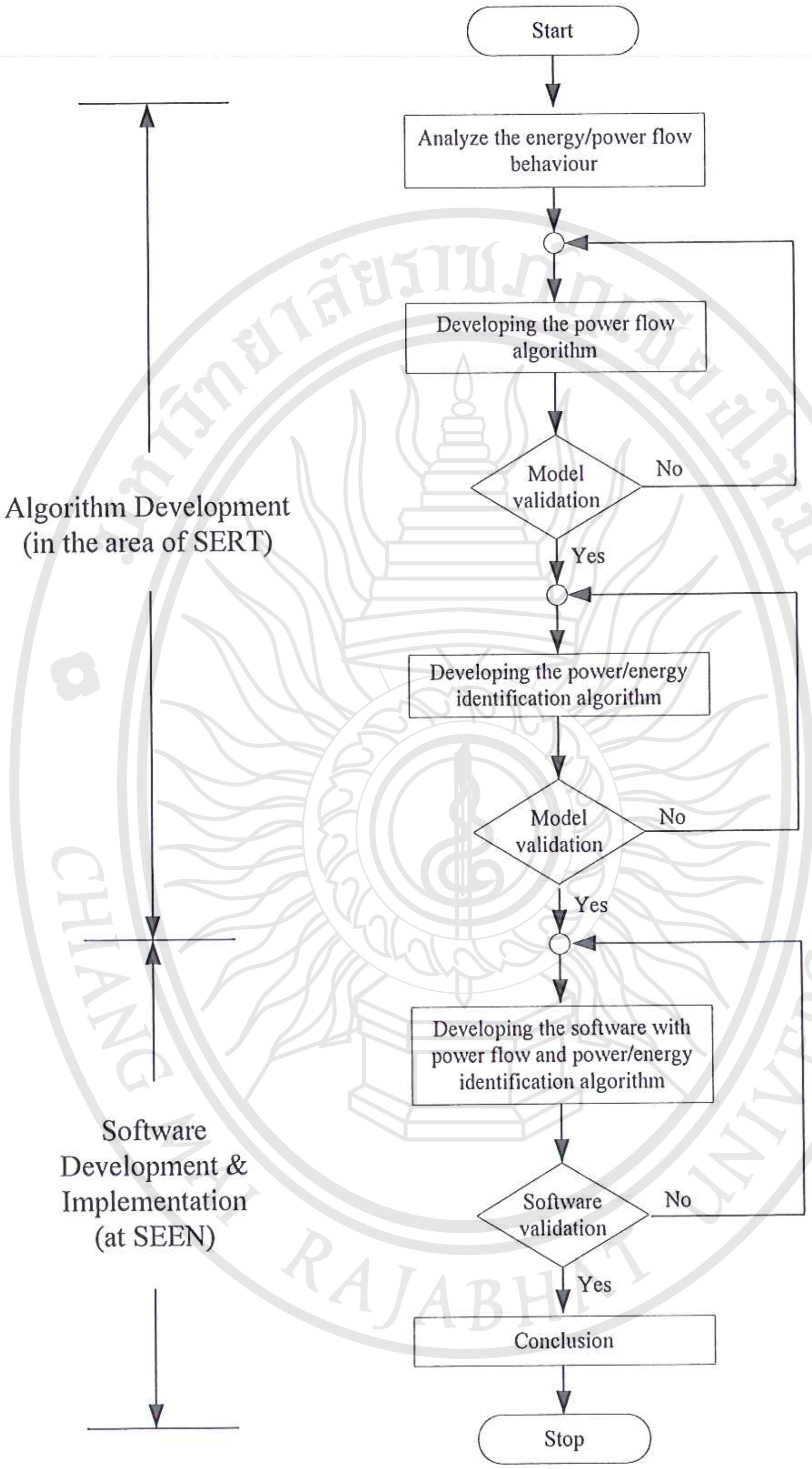

Figure 3.1 Flow chart of research methodology

Case I: Determining Energy / Power Flow Algorithm

In this case, four buildings at SERT were the case-study buildings for the modeling of the power flow exchange between renewable energy and electric utility network. Three buildings were installed with Photovoltaic systems, while one building did not have Photovoltaic systems. All buildings were connected and also used power from the local utility network. Figure 3.2 displays the line diagram of the 4 building connection to the main utility grid network. The building name, power demand and power producing capacity are indicated in Table 3.1. All buildings were currently used as office and seminar buildings.

Table 3.1 Building power demand and production capacity

Building name	Peak Demand (kW)	Photovoltaic installed	Photovoltaic Capacity (kW)
Service & Seminar	17	Yes	6
High temp	10	No	-
10 kW PV	3	Yes	10
Testing	19	Yes	9

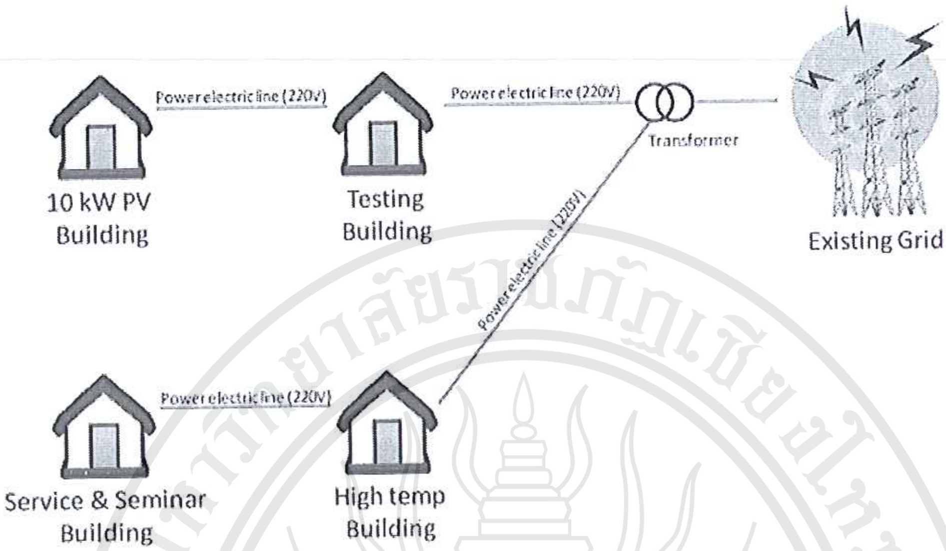

Figure 3.2 Building schematic electric line diagram at SERT

The power demand data of each building were collected and converted to the electrical load profile. The solar radiation data were collected and used to calculate the Photovoltaic performance. The algorithm for the balancing model depended on load profiles and Photovoltaic performance to calculate the net metering of each building; power exchange between building and the overall system; and the source of power.

1. Power flow and identification algorithm design

Algorithm involved in this research was separated into 2 algorithms; power flow (kW) and power unit identification algorithm. The first was created for reporting the direction of electricity in the system and second was for identifying the source of electricity.

1.1 Building power flow algorithm

The main goal of the power flow algorithm was for reporting the real time power flow routing of the building. Each building would be simultaneously indicating different power status and net metering values that were calculated from the building power flow algorithm. The net metering value depended on the summation of power consumption and power production. If the power consumption was more than the power production, the net metering value was equal or minus value and the status of the building was in demand mode. The demand mode was reported as the building that needs the power from outside of the building. On the other hand, if the power consumption was less than the power production, the status of the building was in supply mode and net metering value was equal or plus value. The supply mode was reported as the building that supplies the power to the outside building. In case of the power consumption was equal to power production, the status of the building was an island mode that meant the building did not need the power from the outside and dit not supply power to the outside. From different values, the overall power flow would be calculated from the status and net metering from each building and generate report as the power status and net metering value of all buildings. At the same time, it would use the electric price calculation function in reporting real-time energy cost. After finishing the calculation, the status of this algorithm was divided into 3 modes, which were demand, supply and island mode as show in Figure 3.3.

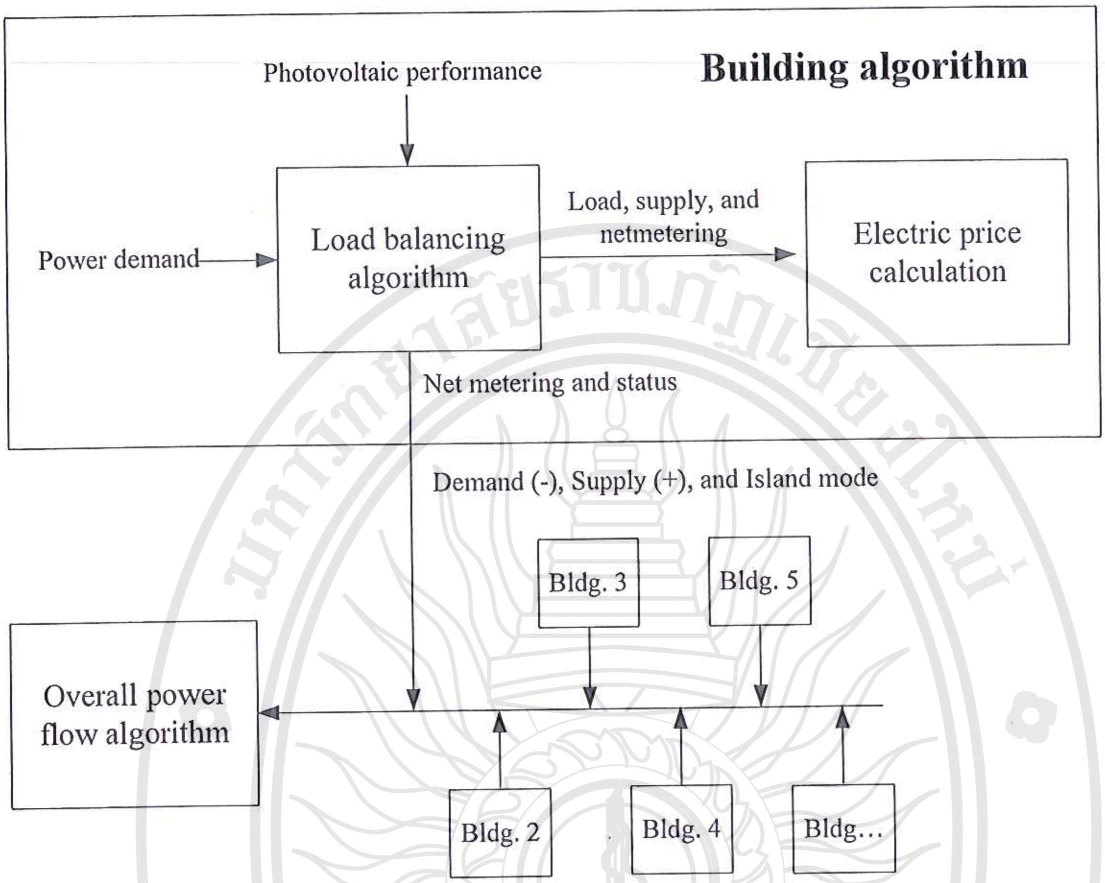

Figure 3.3 Block diagram buildings and overall power flow algorithm

1.2 Overall power flow algorithm

The overall power flow algorithm was reported as the power flow routing of the community that was calculated based on net metering value for all buildings. The overall net metering value was a summation of net metering in each building. If the overall net metering was minus value, then the status is demand. It meant that the community needed power from outside of the community. On the other hand, if the overall net metering was plus values, then the status was supply. The community was supplying power to the outside of the community. In case of island mode, the summation of the overall net metering value was zero, which meant that the community did not need power from the outside and no supply to the outside.

The overall power flow balancing algorithm flow chart is shown in Figure 3.4. The power flow algorithm from each building would be analyzed and processed. In the case where the building was shown as demand mode, the supply power may come from another building or existing utility grid. The algorithm would check the status and net metering from the nearby building first if the status from nearby building was the supply mode and the net metering value was enough for the building demand. The algorithms would report that the electricity was from another building. In another case, if the status from nearby building was shown as supply mode and net metering value was not enough to supply to the building demand, the algorithm would show that the source of power was from another building and the electrical utility network.

In the overall algorithm, after finishing the building power flow algorithm, each building should simultaneously indicated different power status and net metering values. From the different values, the overall power flow would be calculated every 5 minutes and generated report of the power status and net metering value of eleven buildings in university of Phayao (Figure 3.4). Finally, all status and net metering value in the system were recorded in the database and display the value from the software.

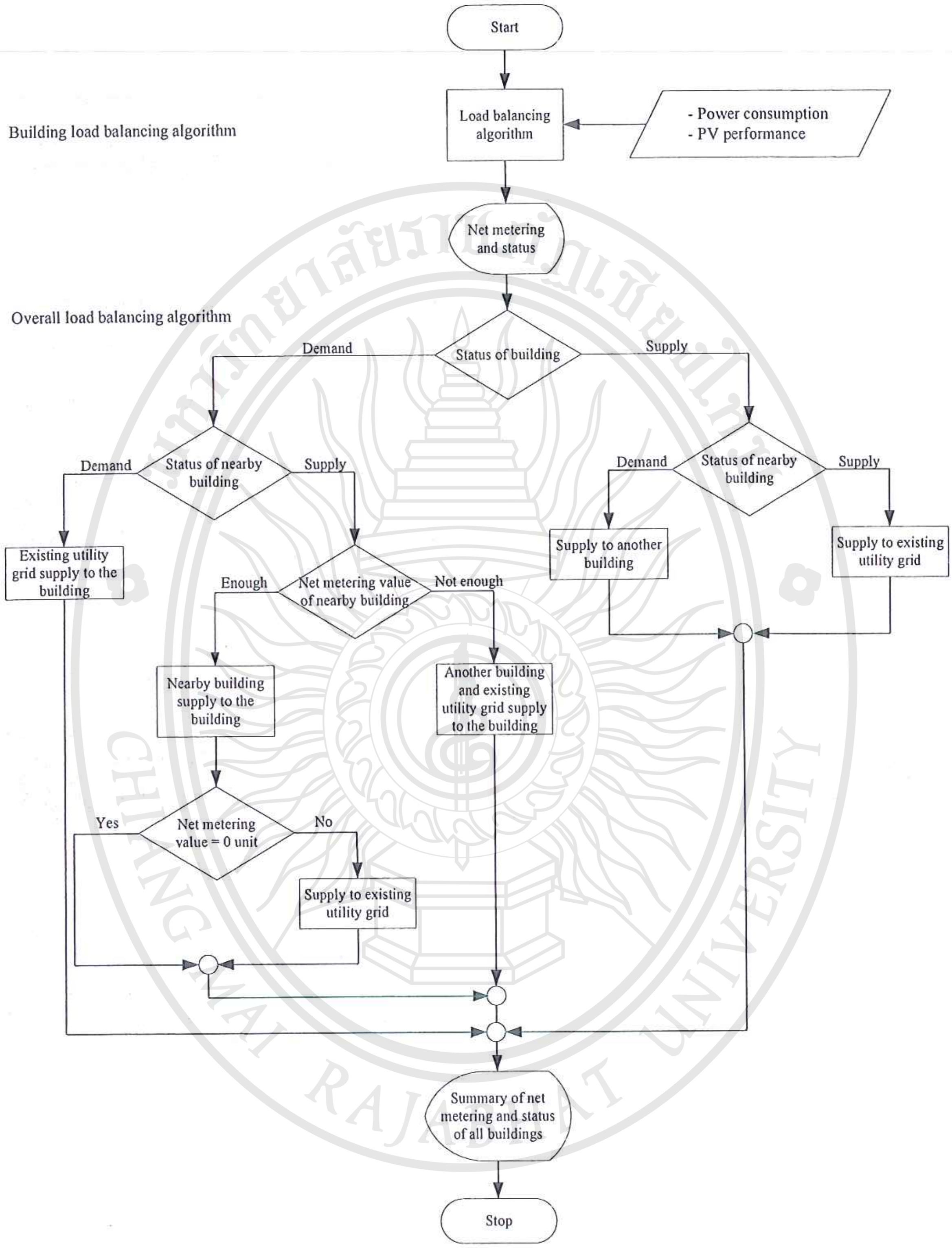

Figure 3.4 Flowchart diagram of the overall load balancing algorithm

1.3 Power unit identification algorithm

The algorithm started from building that was the demand status. The power of balancing should come from another building or utility grid. This sorting algorithm would find the building with the supply status and match the power unit. If the power units of the supply status for all buildings were not enough for matching, it would use the utility grid for balancing the system.

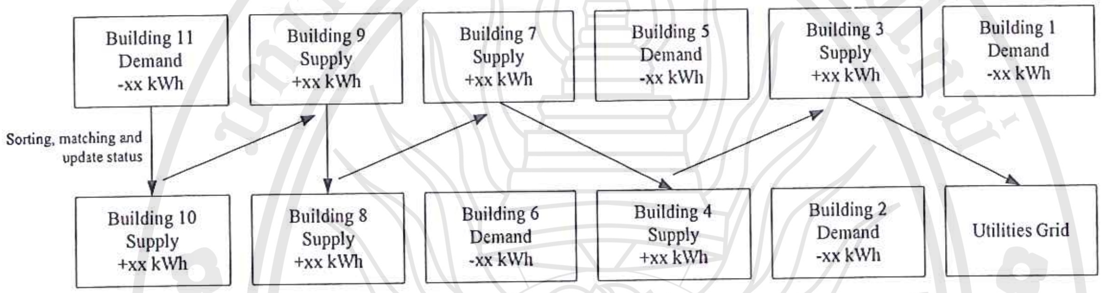

Figure 3.5 Sorting and matching power unit identification algorithm

Figure 3.5 displays the example of sorting and matching of power unit identification algorithm. First, building 11 would sort and match the net metering with building 10 because that building's status was "supply". After matching, it would update the net metering and status of the building 11 and 10. If net metering of the building 10 was enough for matching, the algorithm of building 11 was finished and will record the data into the database. The information in the database showed that building 11 was using the power from building 10 amounting xx kWh. In the case that net metering of the building10 was not enough for matching with the building 11, the algorithm would sort and select the building 9 and used the same method for matching and updating of the database until the net metering of the building 11 equaled zero.

Case II: Developing Power Exchange Software

The software was developed on the AngularJS bases in the model-view-controller (MVC) platform using PHP language. MySQL was used as database for recording different parameters like load profile, RE production (kWh), status of the building, net metering, real-time cost, etc. in 10 second intervals but the parameters on the software interface would be refreshed every 5 minutes intervals.

1. **Photovoltaic system installation**

In this work, eleven buildings at the University of Phayao were the case-study buildings for modeling of the real-time energy flows among multiple loads and sources including renewable energy. All buildings were installed with Photovoltaic systems on the rooftop of the building. The building name and power producing capacity has been shown in Table 3.2 and Figure 3.6.

Table 3.2 Building name and production capacity

Building No.	Building Name	Peak Capacity (kWp)
1	School of Energy and Environment	90
2	School of Engineering	60
3	School of Dentistry	60
4	Civil Engineering Building	80
5	Automotive Technology Development Building	30
6	Industrial Engineering Building	30
7	Mechanical Engineering Building	30
8	School of Allied Health Sciences	30
9	Pharmaceutical Sciences Laboratory	30
10	Food Science and Technology Laboratory	30
11	Anatomy Laboratory	30
Total capacity		500

Figure 3.6 Eleven buildings at University of Phayao installed with 500 kW Photovoltaic rooftop

All of Photovoltaic modules in use were of SHARP model: ND-AA250. The capacity was equal to 250 Wp / module. The specifications of the Photovoltaic module are shown in Table 3.3 below.

Table 3.3 Electrical data of SHARP model: ND-AA250

Maximum power	P_{max}	250	Wp
Torerance of Pmax		+5% / -0%	
Open-circuit voltage	V_{oc}	37.39	V
Short-circuit current	I_{sc}	8.81	A
Voltage at point of maximum power	V_{mpp}	30.68	V
Current at point of maximum power	I_{mpp}	8.15	A
Module efficiency	η_m	15.2	%

2. Server specification

The server was a center of the information system. It would calculate the electrical power flow and power unit identification algorithm. The resulted electrical data had to be stored in the server such as: building name, building source/target, power consumption/production, net metering, status, energy cost, etc. The specifications of the server had been shown in Table 3.4.

Table 3.4 Server specification of the system

Processor / Chipset	
CPU	Intel Xeon E5-2620V3 / 2.4 GHz
Max Turbo Speed	3.2 GHz
Number of Cores	6-core
Chipset Type	Intel C612
Cache Memory	
Installed Size	15 MB
Cache Per Processor	15 MB
RAM	
Memory Speed	1866 MHz
Configuration Features	1 x 16 GB
RAM Supported	1.5 TB - Load-Reduced 384 GB - registered

Storage	
RAID Level	RAID 0, RAID 1, RAID 10
Capacity	2 x 300 GB
Storage Controller	
Type	1 x SAS
Interface Type	Serial ATA-600 / SAS 3.0
Storage Controller Name	ServeRAID M5120
Processor	
Installed Qty	1
Max Supported Qty	2
Upgradability	upgradable
Monitor	
Monitor Type	18.5 Inch
Networking	
Data Link Protocol	Ethernet, Fast Ethernet, Gigabit Ethernet
Form Factor	integrated
Ethernet Ports	4 x Gigabit Ethernet
Ethernet Controller(s)	Broadcom BCM5719

Networking	
Remote Management Controller	Integrated Management Module v2.1 (IMM2)
Graphics Controller	
Form Factor	integrated
Graphics Processor	Matrox G200eR2
Video Memory	16 MB
Video Interfaces	VGA
Power	
Type	power supply - hot-plug
Power Redundancy	optional
Power Redundancy Scheme	1+1 (with optional power supply)
Installed Qty	1
Max Supported Qty	2
Operating System / Software	
OS Provided: Type	Microsoft Windows Server 2012 R2
OS Provided	Microsoft

3. Database design

MySQL database, an open source software, was used for recording the data in this research. When the power flow and power unit identification algorithm finished the calculation, the result had to be stored in this database. The software would use the information from the database for reporting the power flow and source of electricity in the system. The ER diagram in the database design has been shown in the Figure 3.7 below.

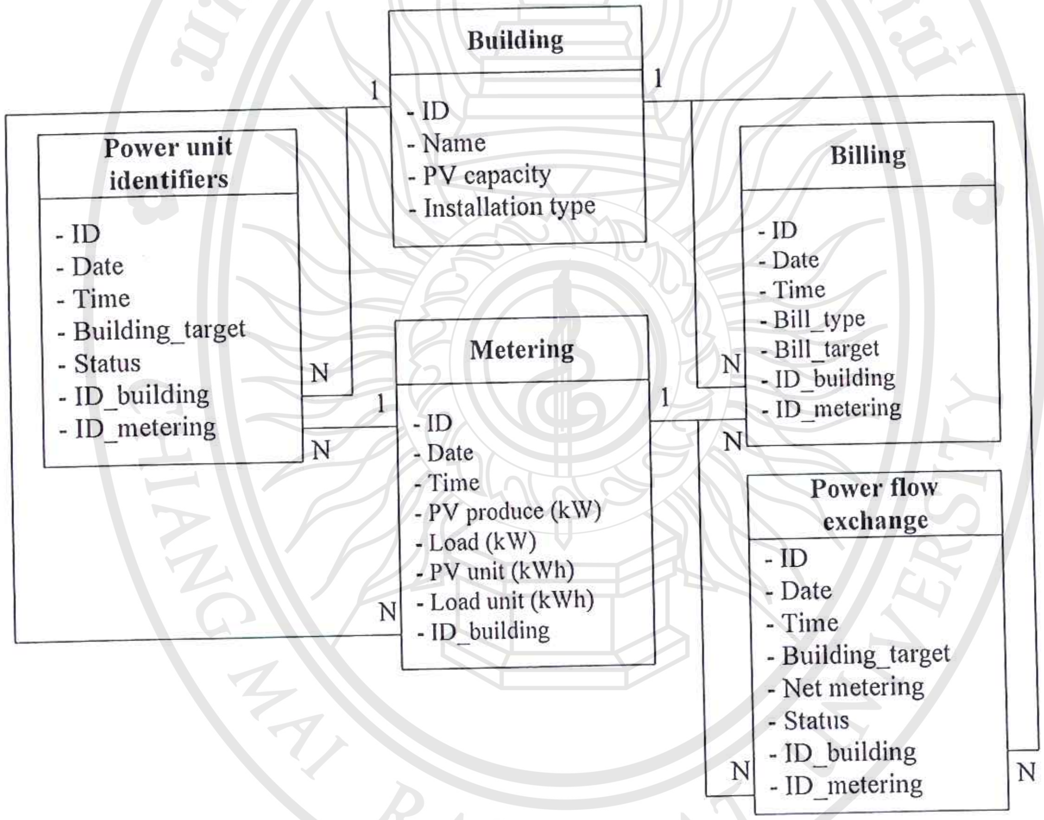

Figure 3.7 ER Diagram of the database design

It consisted of 5 databases for recording the data as follows.

3.1 Building_DB was a database for recording the details of building such as: name of building, Photovoltaic capacity and type of Photovoltaic installation.

3.2 Metering_DB has been created for recording the data from Photovoltaic production and power consumption in kW. At the same time, it calculates the energy production (kWh) and energy consumption (kWh) that is very important data for use in the power flow and power unit identification algorithm.

3.3 Power flow exchange_DB was a database for recording the results of the power flow algorithm that is a building source and destination. It will calculate every 5 minutes and report the power flow in the software.

3.4 Power unit identifiers_DB was a database for reporting the source of electricity that is a result of the power unit identification algorithm.

3.5 Billing_DB was to create for reports for the monthly energy cost, including source of electricity and a type of the billing.

4. System configuration

The system components consisted of Photovoltaic module, inverter, data logger, server, database and software. The Photovoltaic module could produce the DC current and inverter is used to convert into AC. The data logger was used to store the scanned power quality data such as voltage (V), current (A), frequency (Hz) and sent it to the server. All the data stored in the database. The software would report the information by using the algorithm developed in this research. The system configuration is shown in the Figure 3.8.

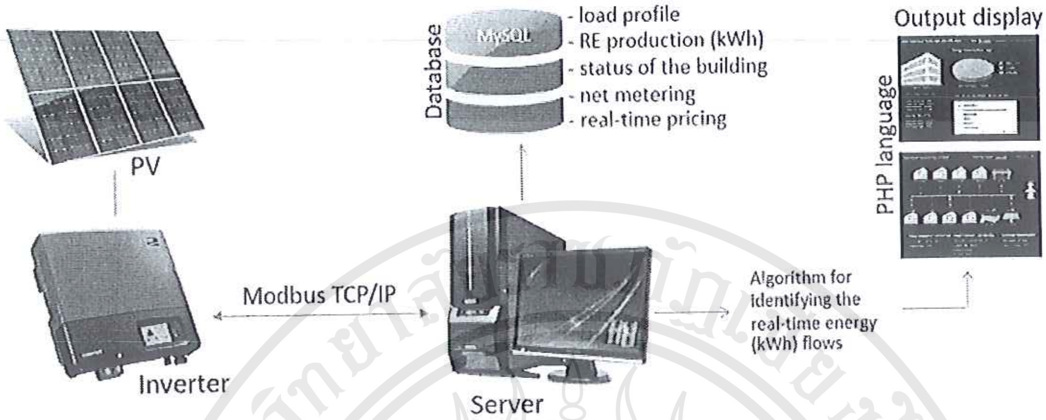

Figure 3.8 Diagram of the software system

5. **Data collecting**

Power analyzer had been installed in the eleven buildings for measuring the power consumption and Photovoltaic production. It sent the data to a data logger after every 10 seconds. The server was supposed to receive the data from the data logger by sending the command from programming to store in the database. Eleven buildings were installed with the DL-NET-1 data logger using the TCP/IP network communication. Table 3.5 and Figure 3.9 show the specifications of the DL-NET-1 data logger configuration.

Table 3.5 Modbus data logger specification in the model DL-NET-1

Voltage recommend	Transformer 220 Vac 50 Hz
Maximum power	9Vdc 400 mA
AC Input Protection	Varistor 275 Vac 7 KA, Fuse 1 Amp
Terminal	Unplugable
Working Temp	10-55 °C

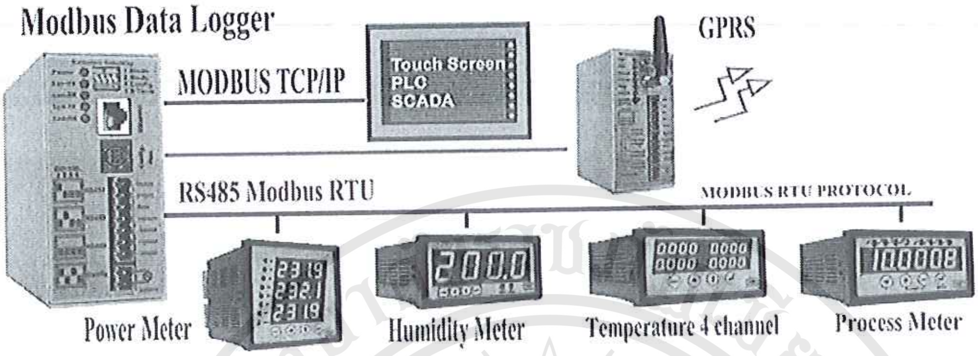

Figure 3.9 Typical of Modbus data logger installation in the system

Source: (SME International, 2009)

6. Programming Language

PHP programming language was used for developing the software. Software design followed the concept of web base technology. User could connect and used the software by opening the web browser, hence it could use the software from mobile, tablet, notebook and personal computer. The Figure 3.10 shows the basic concept of the client-server connection.

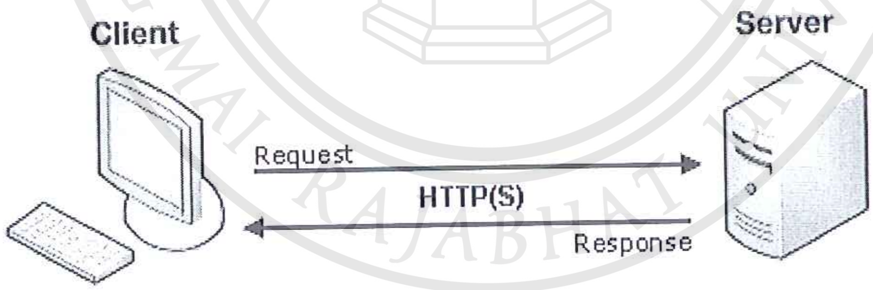

Figure 3.10 Client & Server connection concept in the web base technology

Source: (Mozilla Development Network, 2015)