QUEENSLAND UNIVERSITY OF TECHNOLOGY

IFN 702 – Project report

Online trade web application of Peer to Peer energy sharing platform with Photovoltaic Panels battery systems

Supervisor: Jinglan Zhang

Project Type: Development

Student Name: Yixi Zhou

Student Number: n9599291

Contents

Executive Summary 3

Introduction 4

Environmental scan and review of prior related work 8

Project Methodology 12

Outcomes or Results 16

Login system 17

Solar energy Commerce system 22

Discussion 31

Conclusion 33

Reflections on your learning 34

Reference 35

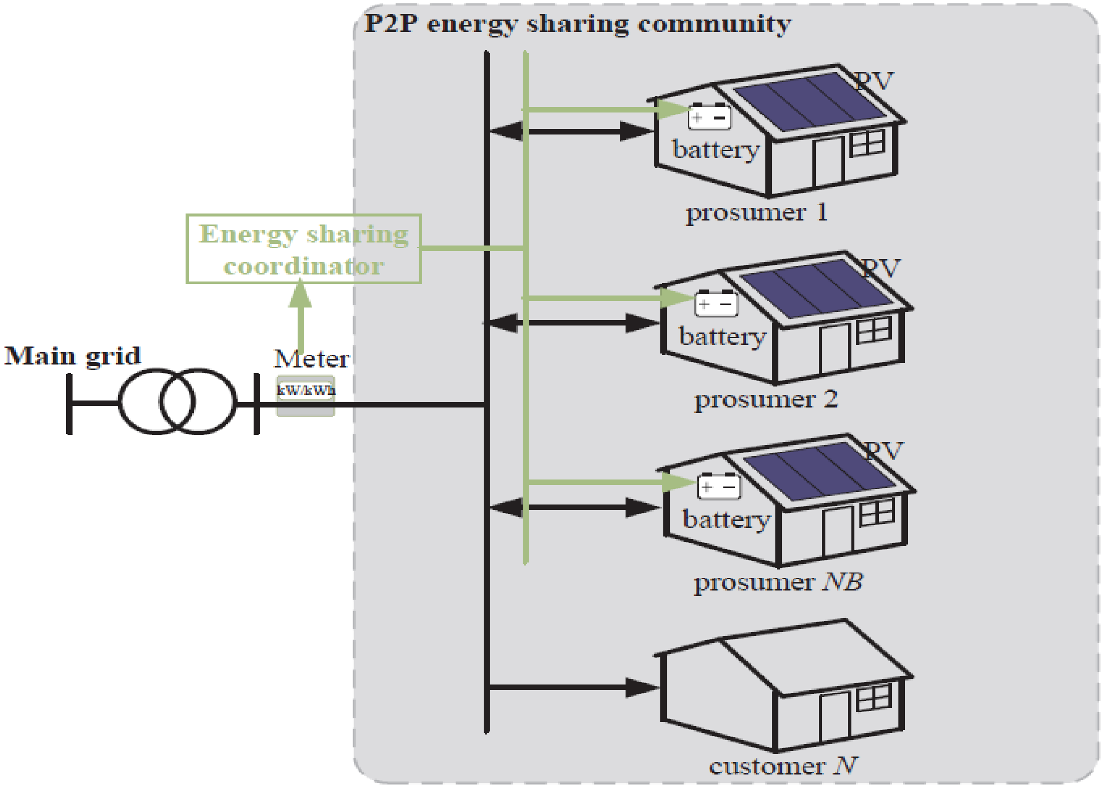
Executive Summary

With rapidly developing in renewable energy, traditional energy systems have been challenged and influenced, where the distributed generators into electrical power systems has been commonly encouraged by many areas across the globe. Especially, solar photovoltaics (PV) become a popular type of Distributed Generators (DGs). Thus, Peer-to-Peer energy sharing platform is brought forward as an efficiently approach to manage PV distributed energy resources (DERs) by many researchers(Ruotsalainen, Karjalainen, Child, & Heinonen, 2017). This paper will introduce the battery based P2P energy sharing architecture and design an online trade web application to contract producers and consumers. The HTML, CSS, MYSQL and PHP will be used to write the web programming. The key deliverable is a root file folder containing HTML, CSS and PHP source code for electricity trade web application. This web comprises by login system and commerce system. The login system can check the input format and give feedback about error type. The transaction system can trade solar energy by duration of time to use solar panels. Another outcome is some journal article reviews for P2P solar power sharing system architecture. A basic Peer-to-Peet solar energy sharing system consist of web application interface, scheme system and power management unit.

Introduction

Due to the increasing of carbon emissions and air pollution, traditional energy systems have caused more and more issues to today’s society. Meanwhile, with rapidly developing in renewable energy, a new architecture named Distributed Generators (DGs) has emerged into contemporary electrical power systems. In conventional energy systems, electrical energy production has been considered as a centralised generating procedure consisting of energy consumers and producers(Zhang, Wu, Long, & Cheng, 2017). The electrical energy producers are concentrated on several centres and consumers buy electricity from centre power plant. In contrast, solar photovoltaics (PV) as a type of representative renewable energy present decentralised energy supplying architecture to society. For instance, many houses have been installed solar panels on their rooftop area in Australia and become small energy suppliers. Furthermore, governments encourage lots of energy companies to invest resources on residential-scale PV systems because of financial and environmental concerns. Therefore, community energy systems will play a critical role in the future energy systems. And, how to effectively organize small-scale distributed energy resources (DERs) in community Microgrids has come to be an important industry problem. To manage DERs, Peer-to-Peer energy sharing platform has been proposed by researchers as a regional market solution. The core idea for above innovation is that building a platform allows customers who have installed the PV system could sell their surplus electricity to their neighbours. It can create a Microgrid solar energy sharing community and electricity marketplace(Long et al., 2017). Especially, Australia have sufficient illumination which make PV technology become very popular. There is a huge potential community solar electricity marketplace that give many opportunities for P2P solar energy sharing platform.

During this project, we will choose the battery based P2P energy sharing architecture. The below picture will demonstrate the major structure of this system, containing Photovoltaic Panels, Battery, Main Grid and Energy Sharing Coordinator.



To establish the whole P2P system, electronic engineering and information technology disciplines knowledge is required. However, the physical solutions, like installed the PV or battery and design the suitable circuit, are out of this project scope. The main aim for this project is to focus on IT area, where our team concentrate on giving the software solution for this P2P platform via building an Energy sharing coordinator system. There are four objectives for the Energy sharing coordinator system. One objective is to make a web application to control in-house appliance. The purpose for this application is to allow prosumers to sell more solar energy during the electricity peak consumption time. Through turning off some house appliance, such as air conditioner or dishwasher, more surplus energy could be used for trading to increase the financial profit. Another objective is to build an online trade web application that contract with individual generators and consumers and calculate the electricity price. The third objective is a scheme system that allocate the solar power from prosumers to different customers. The final objective is power management unit which estimate batteries charge states and determine supply power or load to server.

Since the above Energy sharing coordinator system is complex, this semester project scope is to implement online trade web application that able to sell or buy solar power. The main function is to contract with individual generators and consumers, which ensure the transaction could happen in website.

Out-scope

1. Power Management Unit to control charge or discharge in battery
2. Web application to manipulate the house appliance.
3. Design and install the Physical device, such as solar panels and battery.
4. Scheme system to distribute solar energy.
5. Electric circuit design (micro grid network)

The overall project approach is composed of two phase. At the initial phase, the literature review had been done to acquire the knowledge about the basic architecture of Peer-to-Peer solar power sharing system. Next, the relevant programming languages (HTML, CSS, SQL and PHP) are learned for obtaining web development technique skills .In second phase, above languages are used in programming phase to create the online trade web application. During the programming phase, XAMPP and ATOM text editor are employed to create the database and write code. HTML, CSS and Bootstrap library are utilized to design a simple interface. The back-end code PHP (Hypertext Pre-processors) is applied to establish login system and transaction function for electricity trade website. To manage this project, Scrum framework had been picked from agile management method. The academic supervisor schedule regular week meeting on Wednesday every week to check project progress and give the feedback for weekly deliverables.

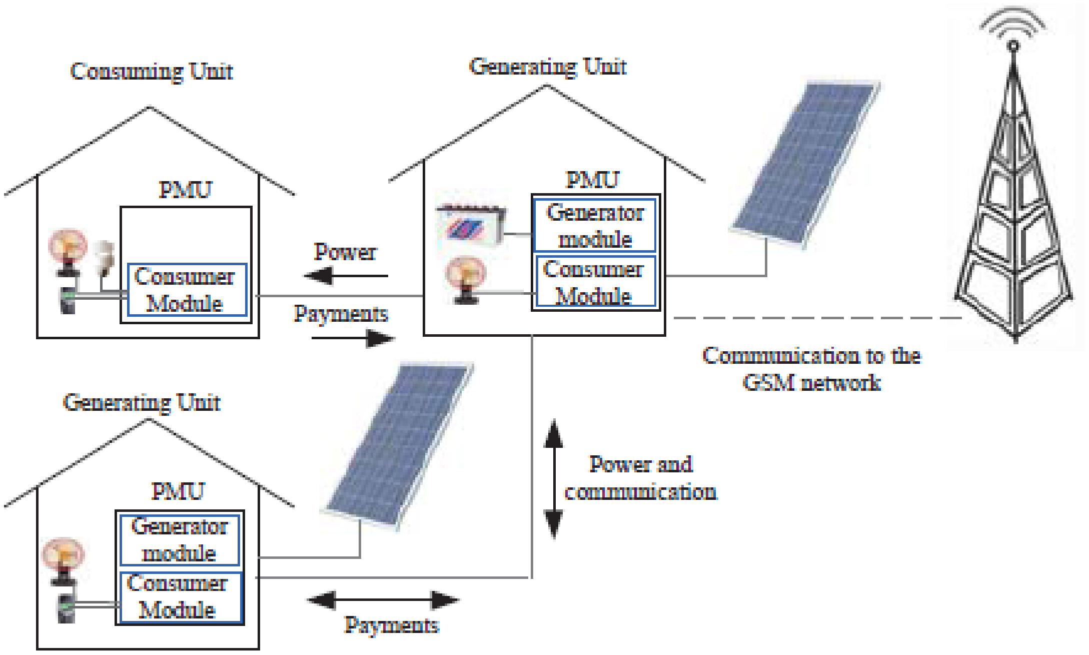
The key deliverable is a root file folder containing HTML, CSS and PHP source code for electricity trade web application. Another deliverable is some literature reviews for P2P solar power sharing system architecture. Through using this trade web application, customer can create an account in the energy sharing platform to buy or sell their surplus solar power. As a buyer, solar power transaction information will display in the Buy website, including the time frame to use neighbours' solar power and the corresponding price. As a seller, they can input the beginning and end electricity vending time to decide moment to provide solar power. In addition, this platform allow sellers to determine the vending price based on the time of employing solar panel. Via previously trade web application, energy consumers are successfully contracted with energy generators for trading the solar power. Furthermore, the significance for doing literature review is to demonstrate a blueprint of P2P power sharing system and show direction to sketch the structure of whole system. According to previous researches about P2P solar power sharing system, it assist our team to design a basic system structure including Online Trade Web Application Interface, Power Allocation Scheme System, Power Management Unit and House Appliance Controlling Web Application. It could help us to recap the project scope to each group member and assign the specific tasks to individuals. In conclusion, this online trade web application could bring financial profit to energy generators and consumers. Energy producers can earn money by selling the solar power and energy consumers can use electricity in a cheaper price.

Environmental scan and review of prior related work

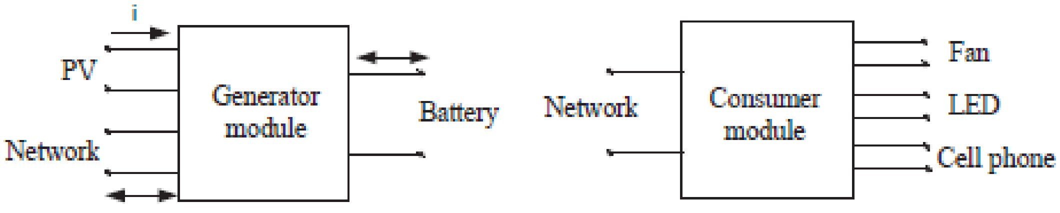
According to the environment of energy market, there are five motivations for researchers to develop P2P solar power sharing system. Falling costs in PV and growth of retail electricity prices is the most essential motivations. Concerning for climate change caused by fossil energy is another critical reason. Next, P2P solar power sharing may provide greater control and autonomy for energy consumption and generation. Also, bringing financial profit encourage people to participate P2P solar power sharing community. The final incentive is the Peer to peer PV system could supply electricity access for off-grid areas, such as mountain or rural district. It will cost a great deal of resources to extend electrical grid to those small remote areas. With increasing number of participators on the P2P solar Power sharing platform, many researchers consider that exist three anticipated significances. Initially, raising the self-consumption ratio can increase affordability for community. The definition of self-consumption is the proportion between the power consumption and total energy produced by solar panels. To raise self-consumption, Chao long's team moderate 40% customers from a community to use electricity produced by PV systems. Afterwards, it reduces 30% energy cost of the whole community compared to use electricity from power station(Inam, Strawser, Afridi, Ram, & Perreault, 2015). Moreover, PV systems perhaps enhances energy security. The reason is the voltage produced by PV system usually around 24V. According to "Use of conventional touch voltage limits – application guide ", the voltage below 50V is safe to do operation. Furthermore, P2P power sharing system is a good method to manage distributed energy resources, which could increase the coefficient of energy utilization(Long, Wu, Zhou, & Jenkins, 2018b).

In previous research, there are three types of P2P solar energy sharing architecture has been developed. The first category is block chain based, which offers a decentralised database to recognize the ownership of energy and organize multiple transaction agreements between sellers and consumers. Block chain based P2P platform has been trailed in local communities. For example, the Perth corporation Power Ledger, in Australia, has constructed a system that essentially permit surplus renewable energy to be sell directly in the zoom who required it at same time. Any surplus solar power generated can go into the local electricity network. Then, it distributes to the closest area that is currently demanding power. As the second kind of P2P energy sharing system, Online matching platform enables clients to buy electricity directly from DERs (small-scale distributed energy resources) sellers. The consumers may obtain a lower rate because the electricity is purchased from nearby resources. There are trials of this type of system in Piclo UK and Vandebron Netherlands. The final category is battery storage based, which allows individual to store their surplus energy and sell to other community members in peak power consumption(Long, Wu, Zhou, & Jenkins, 2018a).

Since the battery storage based architecture has been selected for this project, we do more literature review about the structure of battery storage based P2P solar energy sharing platform. In foregoing work, a basic battery storage architecture has been innovated which show below picture.

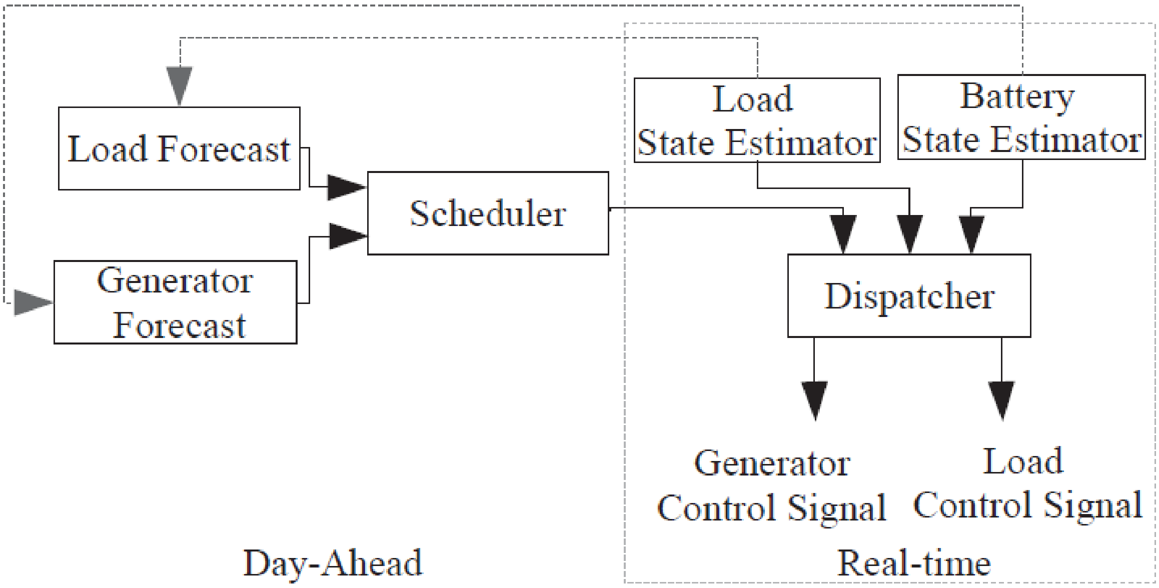


It is made up by Power Management Unit (PMU) and day-ahead scheduling algorithm. The PMU is working on a P2P Direct current Microgrid network. Not only it offer the conversions obligatory to power particular loads but also help construct and control the electricity grid. Cooperated with day-ahead scheduling algorithm to dispatch power, such systems could supply dependable electricity to customers who live in the same energy network. In Details, the PMU is constituted of a generator and consumer module that demonstrate on below graph.



(a) Generator module. (b) Consumer module.

The generate model, which takes power from the solar panels, utilize a charge controller to charge the battery and a bidirectional converter to connect with network for providing electricity. The consumer module, which get the electricity from network, powers the electric devices in the house. To sum up, the PMU need to allocate and dispatch power to network, battery and home appliances(Ruotsalainen et al., 2017). However, when numerous people join this P2P solar power sharing platform, the microgird could be instability due to various contingencies. Such as, there are not enough energy generators in the community network. Hence, day-ahead scheduling algorithm is required to learn about user power demand in order to make proper dispatch decisions for the PMU. Thereby, this scheme system need to offer three functions including load forecast, generator forecast and real-time dispatch (show in below graph).



1) Load Forecast: Historical electricity consumption data is utilized to predict the power requirement for each day. This approach allows the whole system make smart decisions about how to distribute power to each consumer.

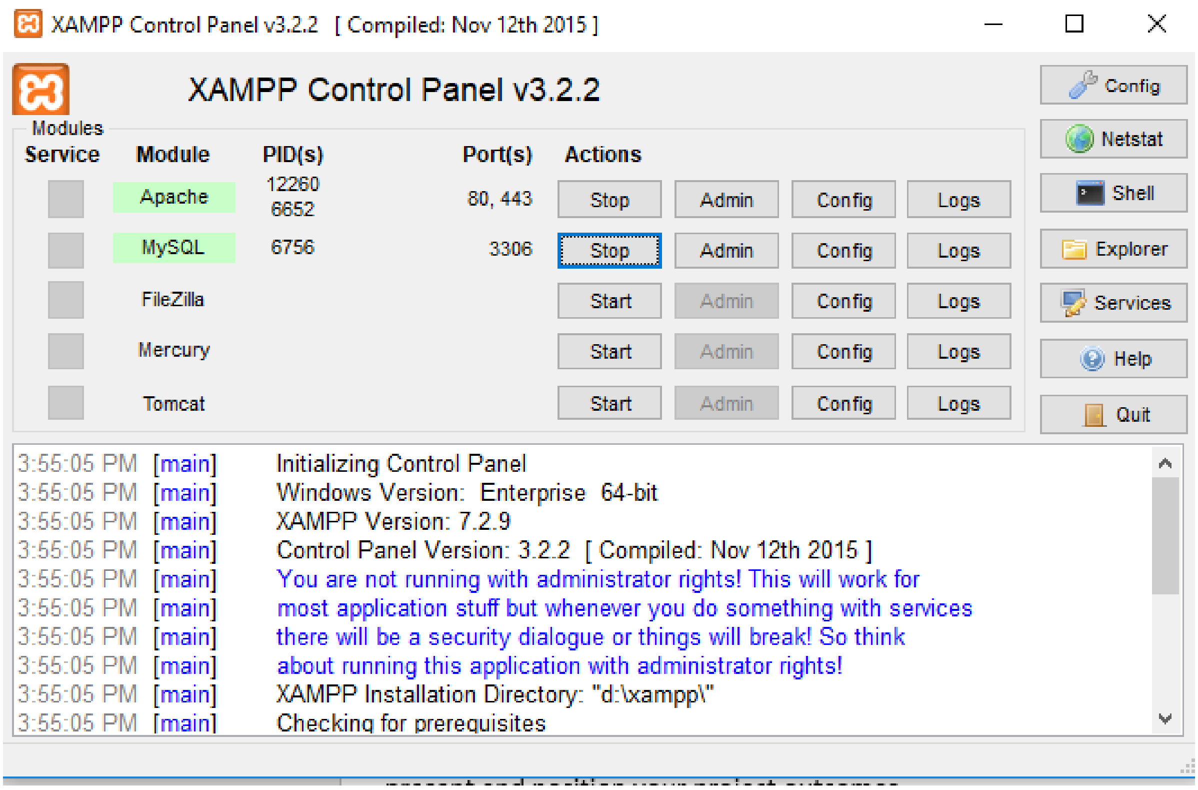
2) Generator Forecast: Historical solar irradiation data is collected in various scenarios such as sunny and cloudy. Then, gather the electricity generating capacity data of solar panel in corresponding weather scenarios. The electricity producing capacity in different weather can be predicted by utilizing above data.
3) Real-time Dispatch: The load state estimator is used to evaluate the loads' current power consumption. The battery state estimator is employed to evaluate the batteries' charge states. Based on preceding evaluation, the dispatcher determine which batteries load to serve and which batteries supply power.

To summarize, most of prior work is focus on the architecture of Peer-to-Peer solar energy sharing platform. Some researchers have already build the system and test it in community. Nevertheless, the system still do not implement into electricity market, so there is a gap about how to sell and buy solar power in this P2P platform. That leads to the purpose of this paper, which build an online trade web application for solar energy transaction.

Project Methodology

Because I am still not familiar with the P2P Solar Power sharing system, literature review will become the first phase to study required skills and knowledge for the project. Based on the previous work, our team separate the whole system to four parts, including online trade web application, house appliance control web application, power management unit and scheme system. Each group member are responsible for one part. My individual project is to build online trade web application. At beginning of programming solar energy trade web application, YouTube online courses teach me the grammar of HTML, CSS and PHP language and how to establish a web application.

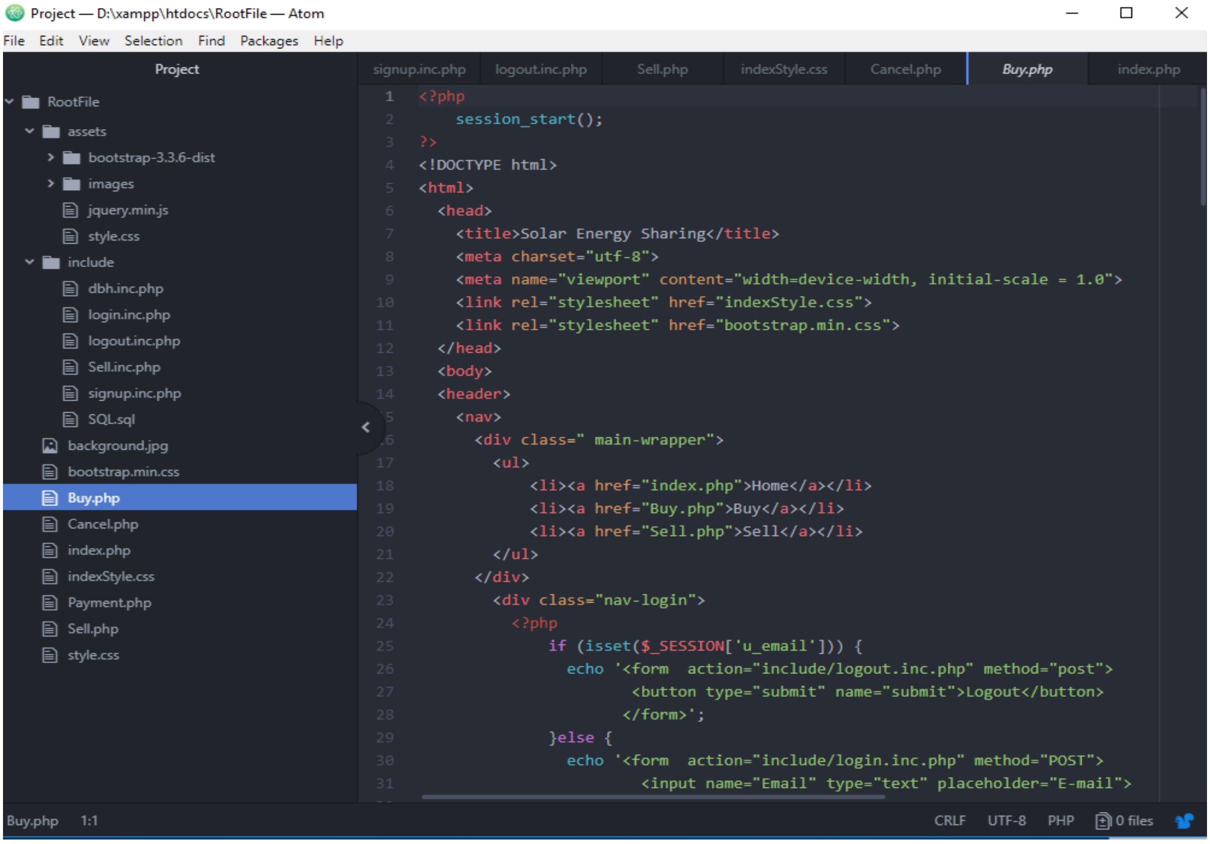
To achieve this project, two tools are employed during this semester. The first tool is XAMPP that is a free and open-source cross-platform web server constituted by Apache HTTP Server, Maria DB database, PHP interpreters and Perl programming languages. It can be used to create the relevant database and become to local sever to run the web PHP code. The below graph shows the control panel of XAMPP(Kumari & Nandal, 2017).


XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]
XAMPP Control Panel v3.2.2
Config
Netstat
Shell
Explorer
Services
Help
Quit
Modules

Service	Module	PID(s)	Port(s)	Actions			
	Apache	12260 6652	80, 443	Stop	Admin	Config	Logs
	MySQL	6756	3306	Stop	Admin	Config	Logs
	FileZilla			Start	Admin	Config	Logs
	Mercury			Start	Admin	Config	Logs
	Tomcat			Start	Admin	Config	Logs

```
3:55:05 PM  [main]    Initializing Control Panel
3:55:05 PM  [main]    Windows Version:  Enterprise  64-bit
3:55:05 PM  [main]    XAMPP Version: 7.2.9
3:55:05 PM  [main]    Control Panel Version: 3.2.2  [ Compiled: Nov 12th 2015 ]
3:55:05 PM  [main]    You are not running with administrator rights! This will work for
3:55:05 PM  [main]    most application stuff but whenever you do something with services
3:55:05 PM  [main]    there will be a security dialogue or things will break! So think
3:55:05 PM  [main]    about running this application with administrator rights!
3:55:05 PM  [main]    XAMPP Installation Directory: "d:\xampp\"
3:55:05 PM  [main]    Checking for prerequisites
```

For writing the HTML, CSS and PHP code, the ATOM Text editor is utilized, which is a free source code editor to develop the web technologies (interface shown in below graph).



```
<?php
    session_start();
?>
<!DOCTYPE html>
<html>
  <head>
    <title>Solar Energy Sharing</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale = 1.0">
    <link rel="stylesheet" href="indexStyle.css">
    <link rel="stylesheet" href="bootstrap.min.css">
  </head>
  <body>
  <header>
    <nav>
      <div class=" main-wrapper">
        <ul>
            <li><a href="index.php">Home</a></li>
            <li><a href="Buy.php">Buy</a></li>
            <li><a href="Sell.php">Sell</a></li>
        </ul>
      </div>
        <div class="nav-login">
          <?php
            if (isset($_SESSION['u_email'])) {
              echo '<form  action="include/logout.inc.php" method="post">
                     <button type="submit" name="submit">Logout</button>
                    </form>';
            }else {
              echo '<form  action="include/login.inc.php" method="POST">
                      <input name="Email" type="text" placeholder="E-mail">
```

To build this web application, Design is the first phase before the programming. Five web pages are determined to achieve the main function. The first web page is home page, which allow users to login their account or register new account via Email address. Next, the Buy web page is employed to demonstrate electricity vending information. Since selling electricity produced by solar panel is different from marketing common products that can be merchandised one by one, time frame is considered as unit to sell the electricity. Hence, the Buy web page will display several windows which include the price, beginning and end time to use neighbours' solar power. All the windows can be seen as "products" and customers could buy them via adding them to the cart. For the Sell page, suppliers are able to input the transaction start and stop time to decide the period to merchandise electricity. The fourth web page called "Cancel", which allows sellers to drop the transaction. The final Payment page will display the receipt information after consumers purchase electricity.

The second phase focus on programming. For achieving before functions, HTML and CSS are wielded to design five simple web interfaces. In particular, HTML form is major approach to get users input information and pass them to back-end code by URL. The corresponding back-end code is written by PHP (Hypertext Pre-processors)(MacIntyre, 2010). Also, MySQL language use in the back-end which manipulate the related database. The first reason to use PHP to build the web application is PHP can be run on various platforms and have many powerful libraries. The second benefit is MYSQL is utilized with PHP as back-end tool and can be interfaced well with PHP. The third advantage is that PHP is dynamic which could work in combination of HTML to demonstrate dynamic elements on the web pages. In summary, there are two main systems in this web application incorporated by Login system and commerce system.

For efficiently managing IF702 project, Scrum framework has been picked from agile management method due to two benefits. Since this web application is composed by several web pages, though using Scrum framework we are able to get the feedback from product owner (academic supervisor) via numerous sprint review. Furthermore, because I work with team members to accomplish the whole project, it requires team members to communicate with each other. The Scrum framework could supply sprint meet to obtain advice from other group members. On the basic of the Cynefin leadership framework, the complex method including probe, sense and respond, is chosen in the project(Burman & Aphane, 2016). The reason is that there are many unknown knowledges during development, like the lack of experience to develop the web application and how to sell and buy solar electricity online. Moreover, as the final deliverable is software artefact, the empirical process control approach is proper to manage an IT area project. The below table show its three critical principle:

Transparency	• Create product backlog demonstrate the obligatory assignments • Make monthly time planning scheme • Prioritize the assignment • Meet academic supervisor in weekly meeting
Inspection	• Collect feedback and comment form academic supervisor in regular week meeting • Academic supervisor inspects the deliverables at sprint review and sprint retrospective
Adaption	• According to academic supervisor advice, focus on the web application interface developing • Add or cancel item in Backlog based on product owner advice

According to the principle of Scrum framework, a sprint backlog is created to help me organize the whole project.

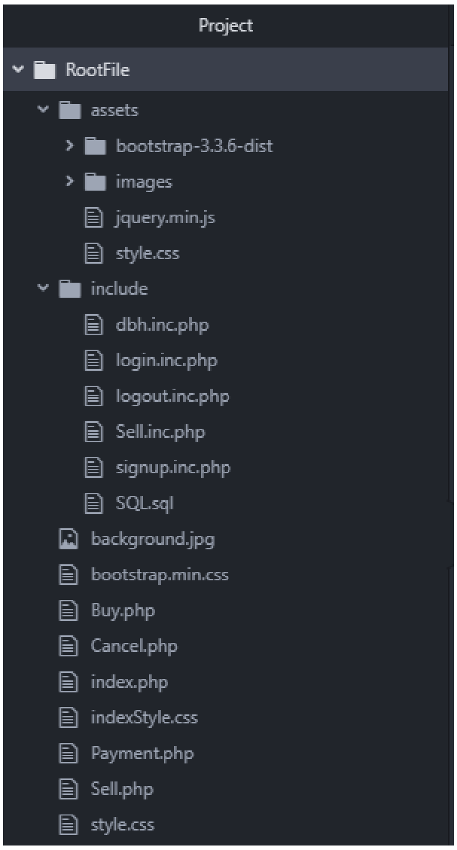
ID	Requirement	Priority	Estimate points
1	learn PHP programming language	Must	8
2	Learn HTML and CSS language	Must	8
3	Design the structure of the trade web application	Must	5
4	Create the home page index.php	Must	3
5	Create the style.css to decorate web page	Should	2
6	Import bootstrap library to decorate web page	Should	5
7	Import Jquery.min.js to show the cartoon	Should	3
8	Use MYSQL to create database in XAMPP	Must	2
9	Use PHP to achieve signup up function	Must	3
10	Use PHP to achieve login and logout function	Must	5
11	Create the Cancel.php web page	Must	3
12	Build the Buy.php web page	Must	8
13	Build the Payment.php web page	Must	5
14	Make the interface become more user friendly	Should	8
15	Write final report	Must	8
16	Add security system to the transaction	Could	8
17	Insert more image to the website	Could	1
18	Create the Sell.php web page	Must	8
19	Debug and test Energy web application	Must	5
20	Lean JavaScript language	Could	8

Based on above requirement, a weekly plan is made to breakdown these tasks. The regular week meeting is every Wednesday 2.45 pm to 3.15, where the academic supervisor checks the deliverables and give feedback.

Time	Product Backlog	Deliverables
		Sprint 1
Week 5	2	Understand HTML and CSS language
Week 6	1	Understand the PHP language
Week 7	3,8	Basic design for website and login_sytem database
Week 8	4,5	Index.php file and style.css file
Week 9	9,10	Finish the login system
		Sprint 2
Week 10	12,13	Buy.php file and Payment.php file
Week 11	18, 11,17	Sell.php and Cancel.php file
Week 12	14,6,7	Import bootstrap, image and jQuery
Week 13	15	Write final report

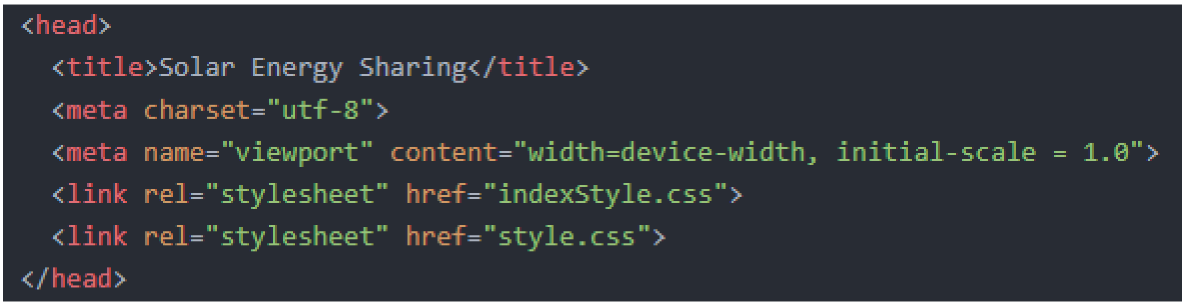
Outcomes or Results

After finishing the project, the outcome is a root file folder which contain HTML, CSS and PHP source code file. the function php file is put into the ‘include’folder. The assets folder is used to store bootstrap library, images and jQuery. The Root File content is shown in below picture:



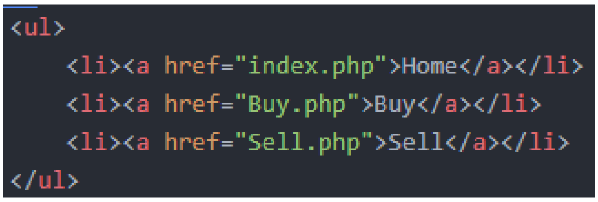
```
Project
RootFile
  assets
    bootstrap-3.3.6-dist
    images
    jquery.min.js
    style.css
  include
    dbh.inc.php
    login.inc.php
    logout.inc.php
    Sell.inc.php
    signup.inc.php
    SQL.sql
  background.jpg
  bootstrap.min.css
  Buy.php
  Cancel.php
  index.php
  indexStyle.css
  Payment.php
  Sell.php
  style.css
```

At the start step to build a website is to create a index.php file as home page. When server try to run a web application, it will consider the website named index.php as the home page. Therefore, it is important to set our name of home page to index.php. To depict the colours, layout and fonts of web pages, we create a indexStyle.css file to manipulate presentation of HTML. The link tag is written between the head tag for citing the CSS file.



```
<head>
  <title>Solar Energy Sharing</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale = 1.0">
  <link rel="stylesheet" href="indexStyle.css">
  <link rel="stylesheet" href="style.css">
</head>
```

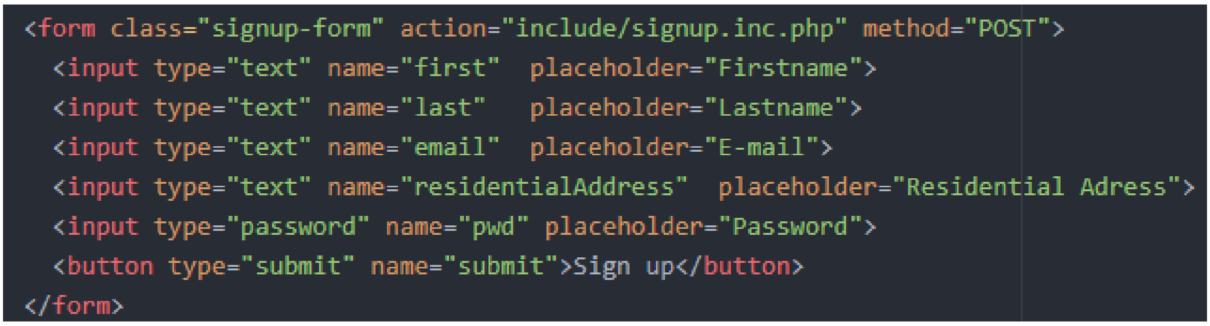
Then, three major hyperlinks are added to the navigation bar, which connect to home, buy and sell web pages.



```
<ul>
    <li><a href="index.php">Home</a></li>
    <li><a href="Buy.php">Buy</a></li>
    <li><a href="Sell.php">Sell</a></li>
</ul>
```

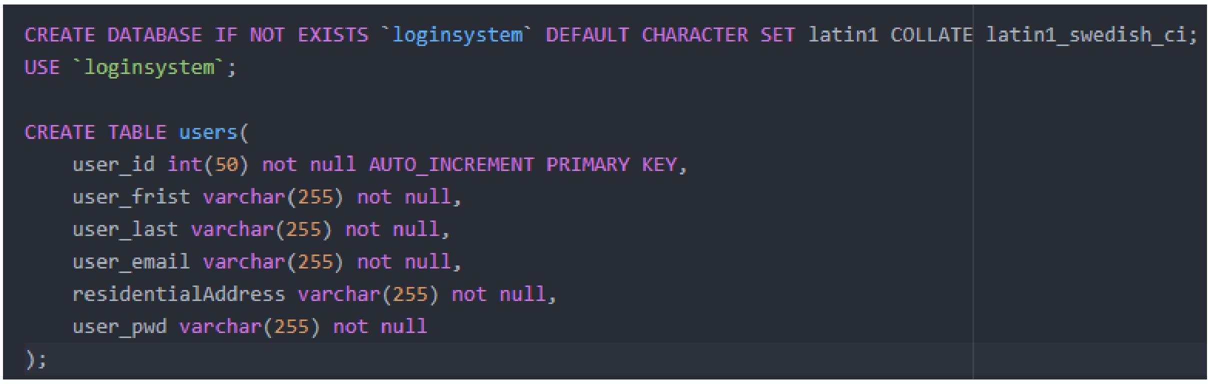
Login system

In the home page, a login system is constructed for customers to create their account in P2P solar energy sharing platform. To establish a login system, firstly HTML form is utilized to make a user interface that allow customers to input their account information. The relevant code is shown in below picture.



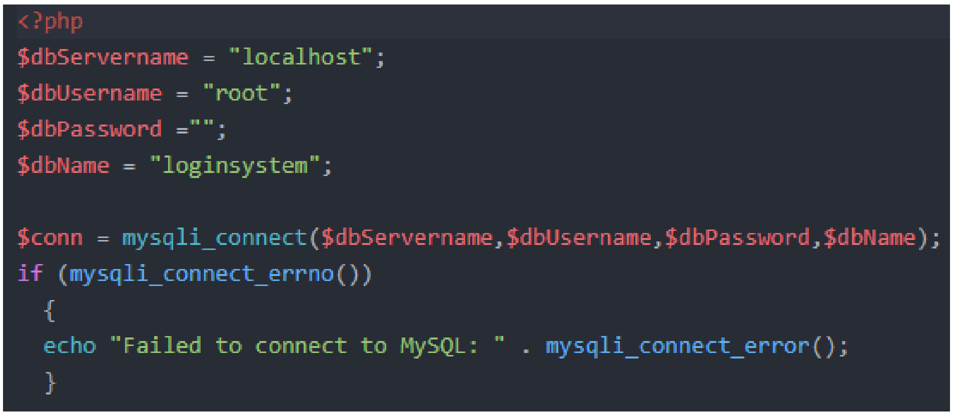
```
<form class="signup-form" action="include/signup.inc.php" method="POST">
  <input type="text" name="first"  placeholder="Firstname">
  <input type="text" name="last"   placeholder="Lastname">
  <input type="text" name="email"  placeholder="E-mail">
  <input type="text" name="residentialAddress"  placeholder="Residential Adress">
  <input type="password" name="pwd" placeholder="Password">
  <button type="submit" name="submit">Sign up</button>
</form>
```

In web development, HTML form is a common way to transfer the information from front-end to back-end via URL. The action parameter in HTML form is employed to write the file path that have the back-end code. Thereby, the user register information has delivered to the signup.inc.php file located on “include” folder. The function of signup.inc.php file is to check the correctness of register information and save it to the related database. Next, we need to construct the required database and “users” table for memorizing form information. The below picture shows relevant code:



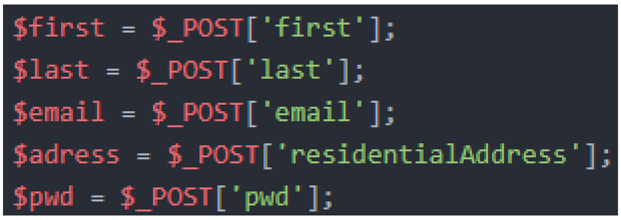
```
CREATE DATABASE IF NOT EXISTS `loginsystem` DEFAULT CHARACTER SET latin1 COLLATE latin1_swedish_ci;
USE `loginsystem`;

CREATE TABLE users(
    user_id int(50) not null AUTO_INCREMENT PRIMARY KEY,
    user_frist varchar(255) not null,
    user_last varchar(255) not null,
    user_email varchar(255) not null,
    residentialAddress varchar(255) not null,
    user_pwd varchar(255) not null
);
```

To interact with database, it should be connected by below PHP code:



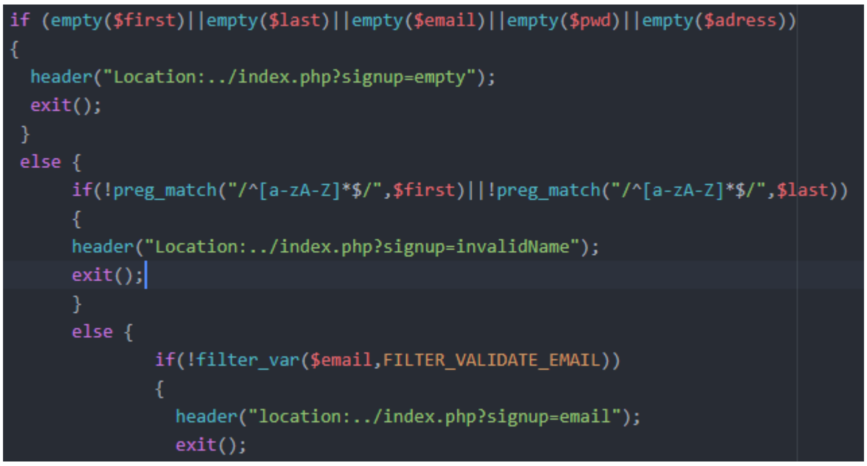
```
<?php
$dbServername = "localhost";
$dbUsername = "root";
$dbPassword ="";
$dbName = "loginsystem";

$conn = mysqli_connect($dbServername,$dbUsername,$dbPassword,$dbName);
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
```

In the back-end, $_POST method is utilized to catch the message from HTML form and assign them to PHP variable, which is proper for PHP manipulate the user input information.



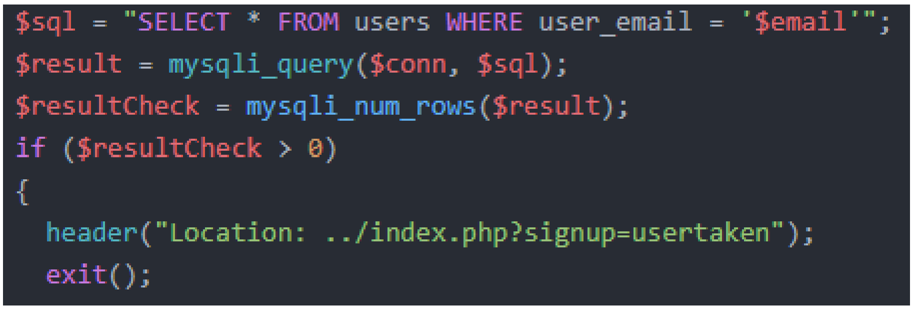
```
$first = $_POST['first'];
$last = $_POST['last'];
$email = $_POST['email'];
$adress = $_POST['residentialAddress'];
$pwd = $_POST['pwd'];
```

Then, “if else” statement is applied to check the correctness of input data. There are three useful PHP function to assist us to check different type of input mistakes. The first is empty() function which confirm users input required information to the interface. If it exist empty value, the header() function will be executed for returning the home page and the error message will display in the URL. The second function is preg_match() function, which can verify do the input messages have the same format with setting up the regular expression. The final function is filter_var($email,FILTER_VALIDATE_EMAIL), which inspect whether the email address is right format.



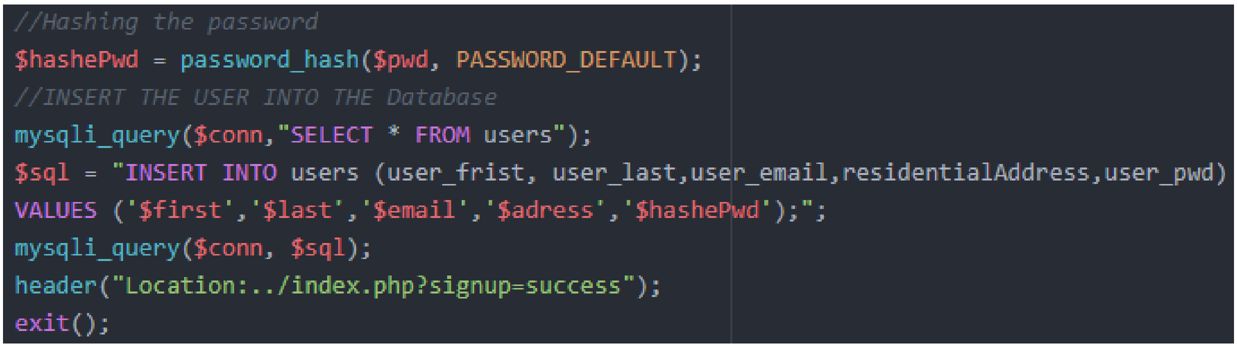
```
if (empty($first)||empty($last)||empty($email)||empty($pwd)||empty($adress))
{
  header("Location:../index.php?signup=empty");
  exit();
 }
 else {
     if(!preg_match("/^[a-zA-Z]*$/",$first)||!preg_match("/^[a-zA-Z]*$/",$last))
     {
     header("Location:../index.php?signup=invalidName");
     exit();
     }
     else {
             if(!filter_var($email,FILTER_VALIDATE_EMAIL))
             {
               header("location:../index.php?signup=email");
               exit();
```

However, since our login system use the email address as login name, we need to ensure that each customer utilizes a unique email address to create the account. Thereby, the under code is to check whether the email address has already taken.



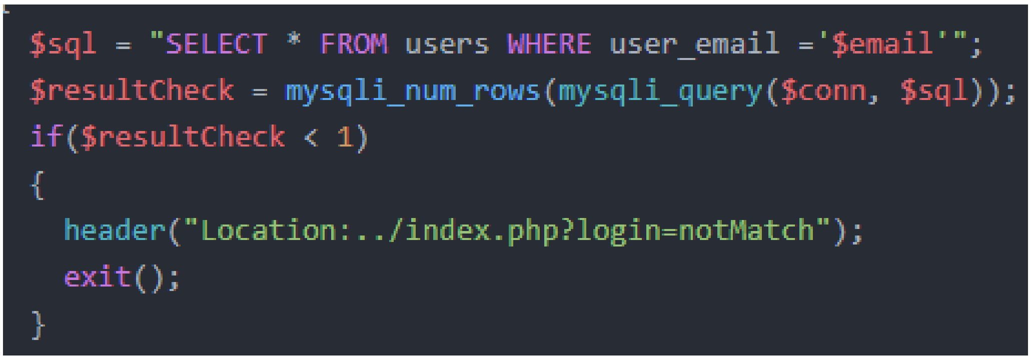
```
$sql = "SELECT * FROM users WHERE user_email = '$email'";
$result = mysqli_query($conn, $sql);
$resultCheck = mysqli_num_rows($result);
if ($resultCheck > 0)
{
  header("Location: ../index.php?signup=usertaken");
  exit();
```

Finally, we can store the register information to the “users” table. In addition, since the password is sensitive information, we should hash the password when it is saved in the database.



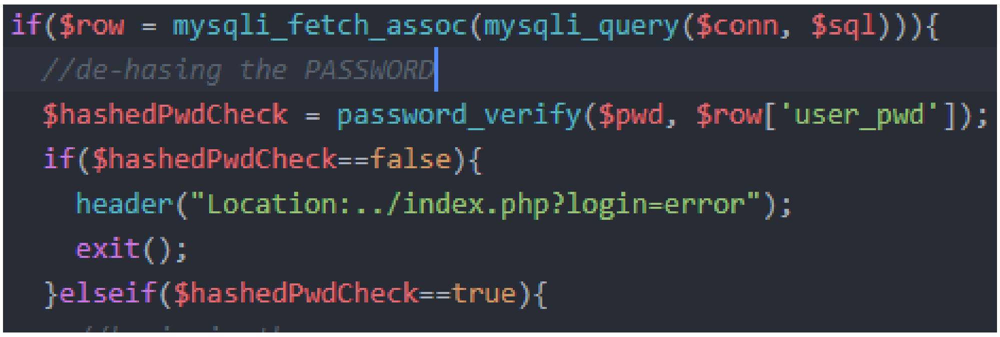
```
//Hashing the password
$hashePwd = password_hash($pwd, PASSWORD_DEFAULT);
//INSERT THE USER INTO THE Database
mysqli_query($conn,"SELECT * FROM users");
$sql = "INSERT INTO users (user_frist, user_last,user_email,residentialAddress,user_pwd)
VALUES ('$first','$last','$email','$adress','$hashePwd');";
mysqli_query($conn, $sql);
header("Location:../index.php?signup=success");
exit();
```

With finishing the register function, the next step is to achieve login function. As database has already stored email address and hashing password, the login function is to compare whether the input email and password are equal to the email and password saving in the database. Hence, the first step is to use input email to find whether it is exciting same email in database (see below code).



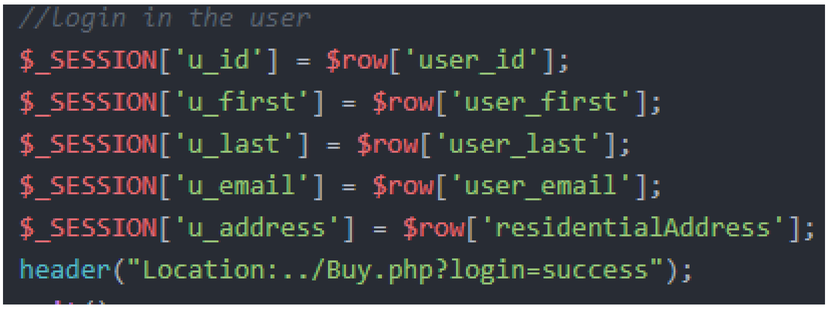
```
$sql = "SELECT * FROM users WHERE user_email ='$email'";
$resultCheck = mysqli_num_rows(mysqli_query($conn, $sql));
if($resultCheck < 1)
{
  header("Location:../index.php?login=notMatch");
  exit();
}
```

If there exist a same email, the second step is that our programming will verify whether the password is identical. However, before comparing input password with database password, we need to de-hash the database password because it is stored hashing format.



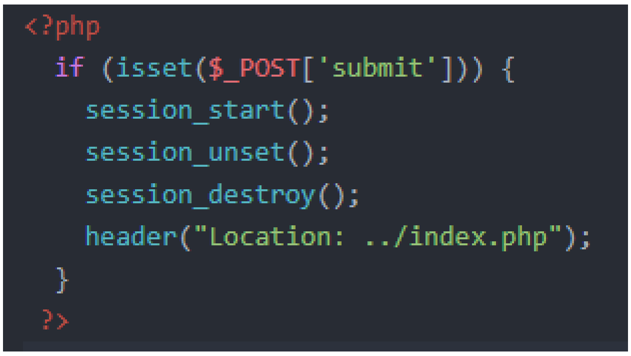
```
if($row = mysqli_fetch_assoc(mysqli_query($conn, $sql))){
  //de-hasing the PASSWORD
  $hashedPwdCheck = password_verify($pwd, $row['user_pwd']);
  if($hashedPwdCheck==false){
    header("Location:../index.php?login=error");
    exit();
  }elseif($hashedPwdCheck==true){
```

The third step is to assign the account information to $_SESSION variable. The SESSION variable is a global variable that make server knows when you start the web application and when you end. Thereby, the server knows who is using the web application via evaluate the status of $_SESSION variable. In each page of solar energy trade website, the session_start() method is added to top position of PHP code, so it could inspect whether website have been login or not.



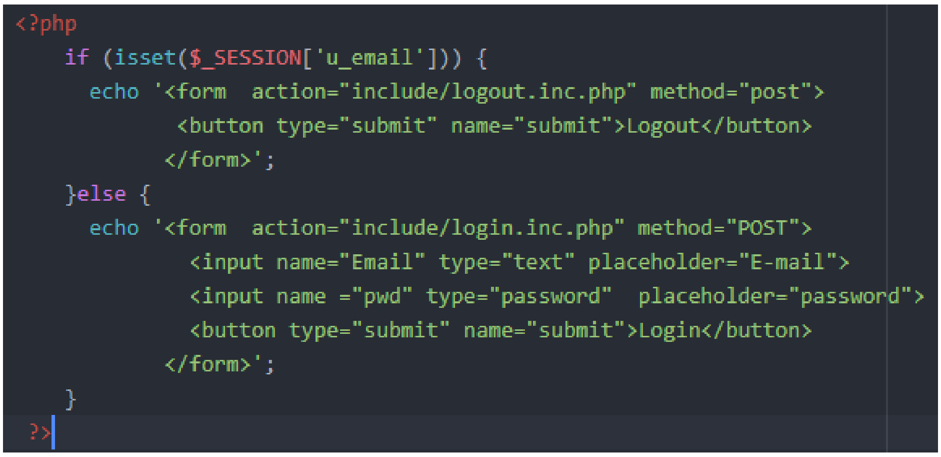
```
//login in the user
$_SESSION['u_id'] = $row['user_id'];
$_SESSION['u_first'] = $row['user_first'];
$_SESSION['u_last'] = $row['user_last'];
$_SESSION['u_email'] = $row['user_email'];
$_SESSION['u_address'] = $row['residentialAddress'];
header("Location:../Buy.php?login=success");
```

The logout function is to clear the session variable by session_unset() and session_detroy() function.



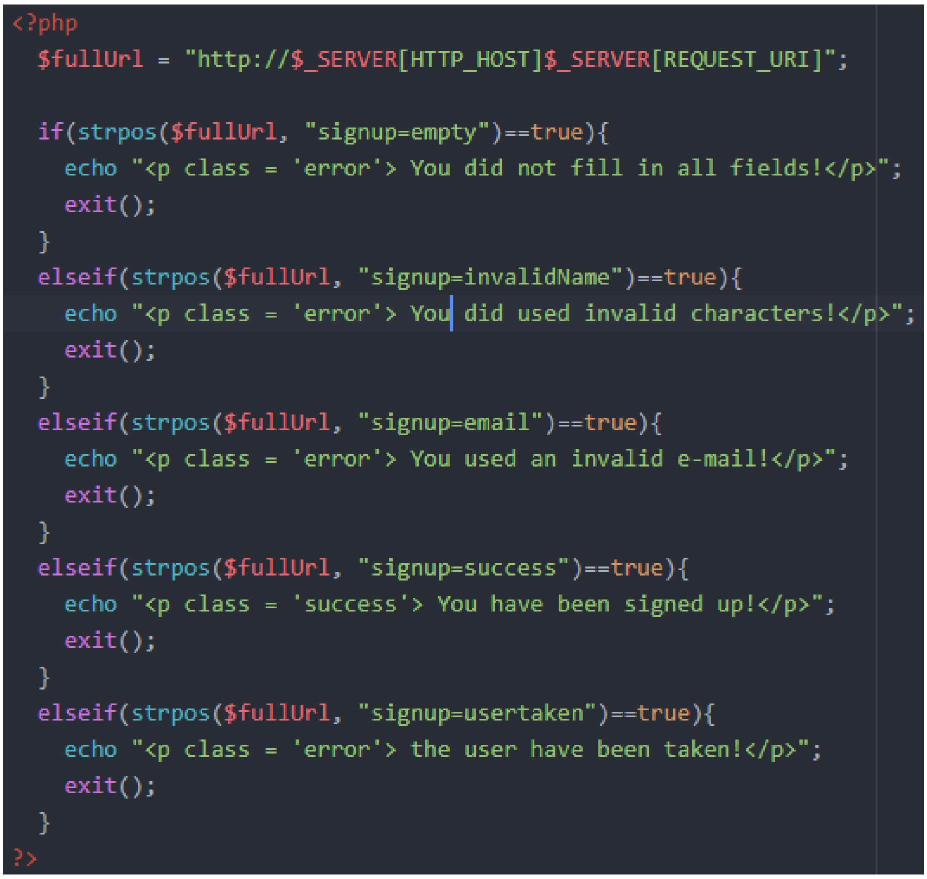
```
<?php
  if (isset($_POST['submit'])) {
    session_start();
    session_unset();
    session_destroy();
    header("Location: ../index.php");
  }
 ?>
```

After finishing the back-end code for login system, we operate “if else” and “echo” statements to verify the status of session variable. If the server finds there is a $_SESSION['u_email'] variable, the web application is on the login in status. Therefore, the “logout” button will be displayed otherwise the “login” button will be shown in the navigation bar.



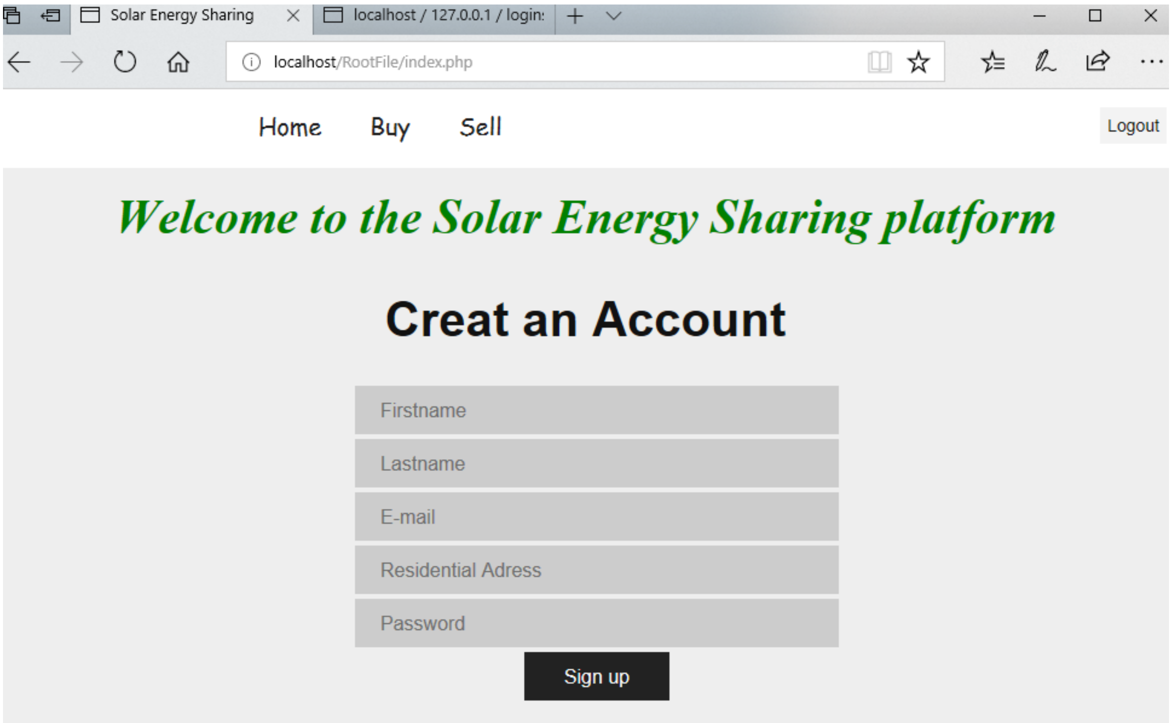
```
<?php
    if (isset($_SESSION['u_email'])) {
      echo '<form  action="include/logout.inc.php" method="post">
             <button type="submit" name="submit">Logout</button>
            </form>';
    }else {
      echo '<form  action="include/login.inc.php" method="POST">
              <input name="Email" type="text" placeholder="E-mail">
              <input name ="pwd" type="password"  placeholder="password">
              <button type="submit" name="submit">Login</button>
            </form>';
    }
  ?>
```

At the end of login system, we add a function to inform users which information they input wrong.



```
<?php
  $fullUrl = "http://$_SERVER[HTTP_HOST]$_SERVER[REQUEST_URI]";

  if(strpos($fullUrl, "signup=empty")==true){
    echo "<p class = 'error'> You did not fill in all fields!</p>";
    exit();
  }
  elseif(strpos($fullUrl, "signup=invalidName")==true){
    echo "<p class = 'error'> You did used invalid characters!</p>";
    exit();
  }
  elseif(strpos($fullUrl, "signup=email")==true){
    echo "<p class = 'error'> You used an invalid e-mail!</p>";
    exit();
  }
  elseif(strpos($fullUrl, "signup=success")==true){
    echo "<p class = 'success'> You have been signed up!</p>";
    exit();
  }
  elseif(strpos($fullUrl, "signup=usertaken")==true){
    echo "<p class = 'error'> the user have been taken!</p>";
    exit();
  }
?>
```

To summarise, we make a basic login system which achieve three functions, including check the user input information, insert user information to database and sign in function. The below picture shows the interface for the login system.



Welcome to the Solar Energy Sharing platform

Creat an Account

Firstname

Lastname

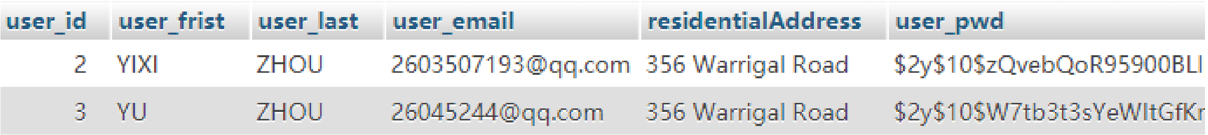
E-mail

Residential Adress

Password

Sign up

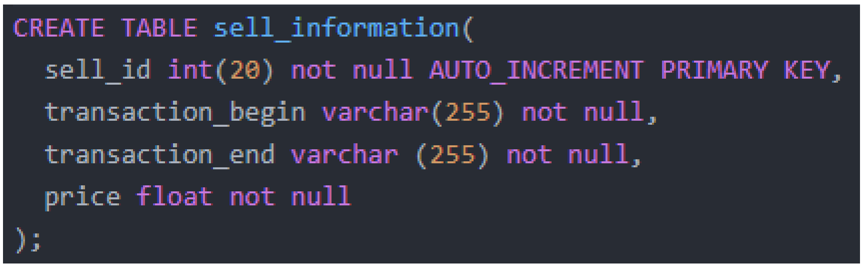
The below picture show how database store the user information: The password is hashed, so we cannot directly see it in the database.



user_id	user_frist	user_last	user_email	residentialAddress	user_pwd
2	YIXI	ZHOU	2603507193@qq.com	356 Warrigal Road	$2y$10$zQvebQoR95900BLI
3	YU	ZHOU	26045244@qq.com	356 Warrigal Road	$2y$10$W7tb3t3sYeWItGfKr

Solar energy Commerce system

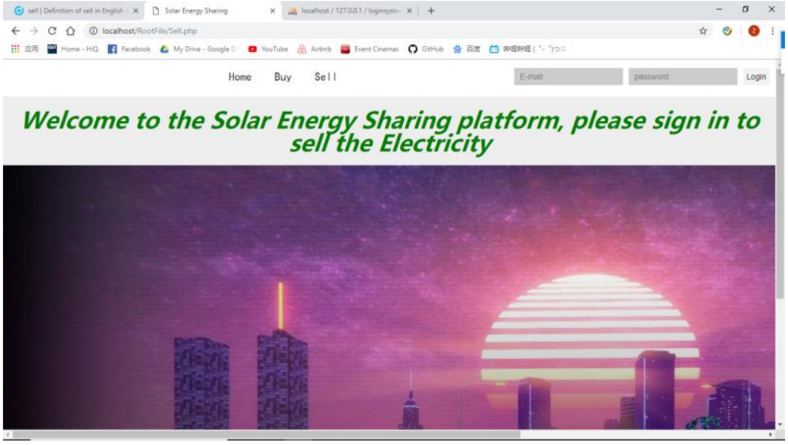
To establish the commerce system, the first step is to build the related database to store the sell information.



```
CREATE TABLE sell_information(
  sell_id int(20) not null AUTO_INCREMENT PRIMARY KEY,
  transaction_begin varchar(255) not null,
  transaction_end varchar (255) not null,
  price float not null
);
```

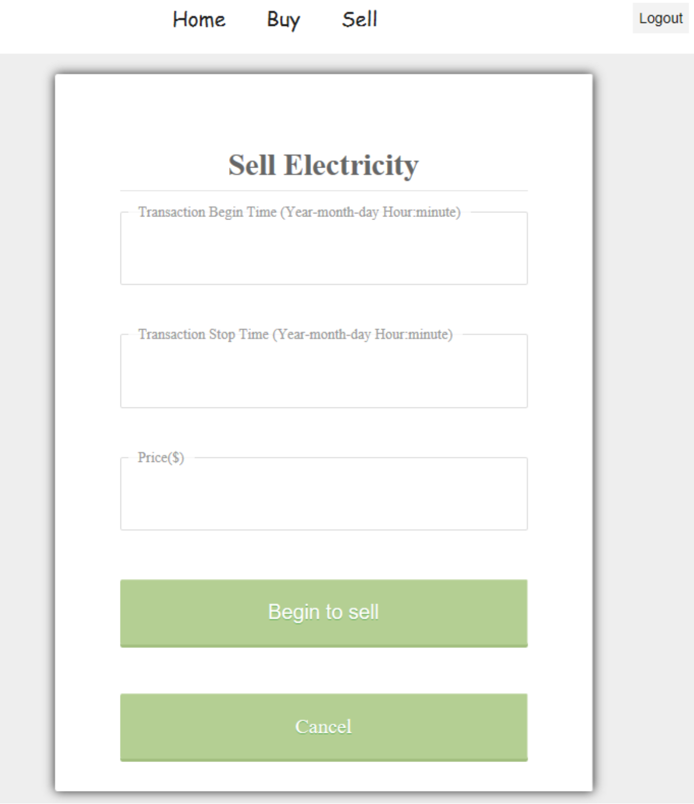
Above code create four column table to save the transaction information. Both “Sell” and “Buy” website can access this sell information table. Especially, “Buy” web page only have the authority to read this information from database. “Sell” website can delete and insert information to this table.

For sell web page, login is the prerequisite to access the “Sell” website. If users do not login to the website, the sell website would remind users to sign in the solar energy sharing platform (shown in below picture). The reason is that the session_start() function is added in all top of php file. Therefore, a “if” statement can be used to verify status of $_SESSION [“E-mail”].



Welcome to the Solar Energy Sharing platform, please sign in to sell the Electricity

After login to the account, users can vend their electricity via Sell website. The web page is going to change to the under graph:



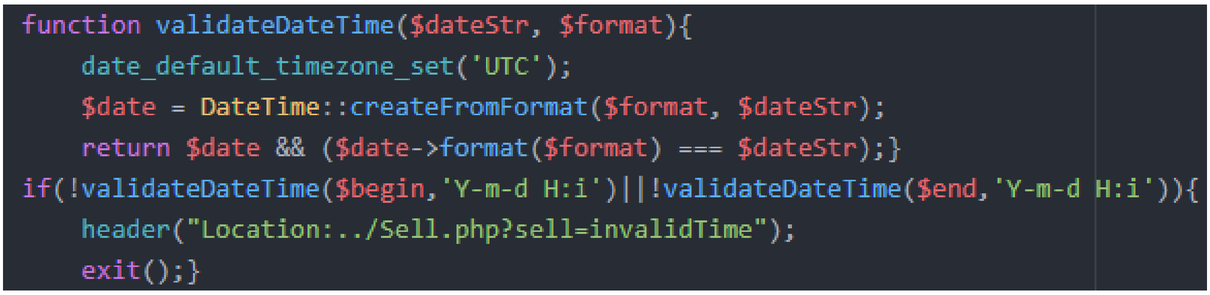
Sell Electricity

Transaction Begin Time (Year-month-day Hour:minute)

Transaction Stop Time (Year-month-day Hour:minute)

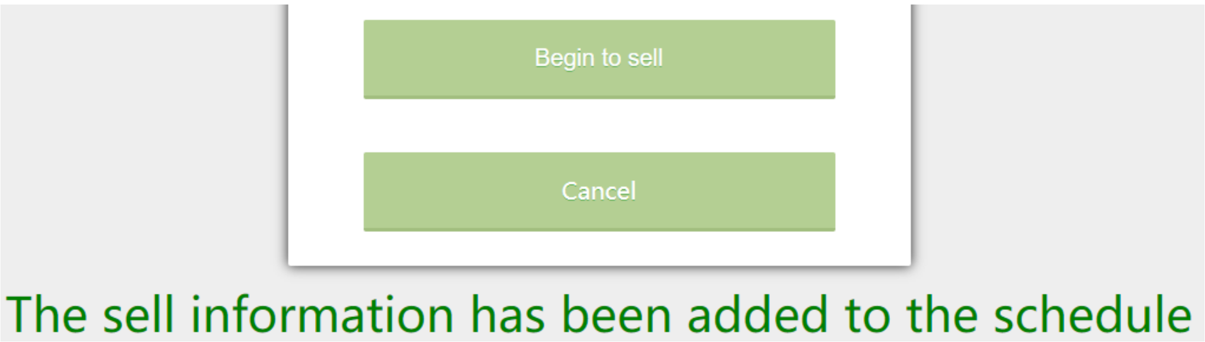
Price($)

Sellers can input transaction begin and stop time to decide when they prefer to merchandise the solar power. Also, sellers can decide the price. Since the time format is critical, validateDateTime() function is constructed to inspect time format. The code is show in under graph:



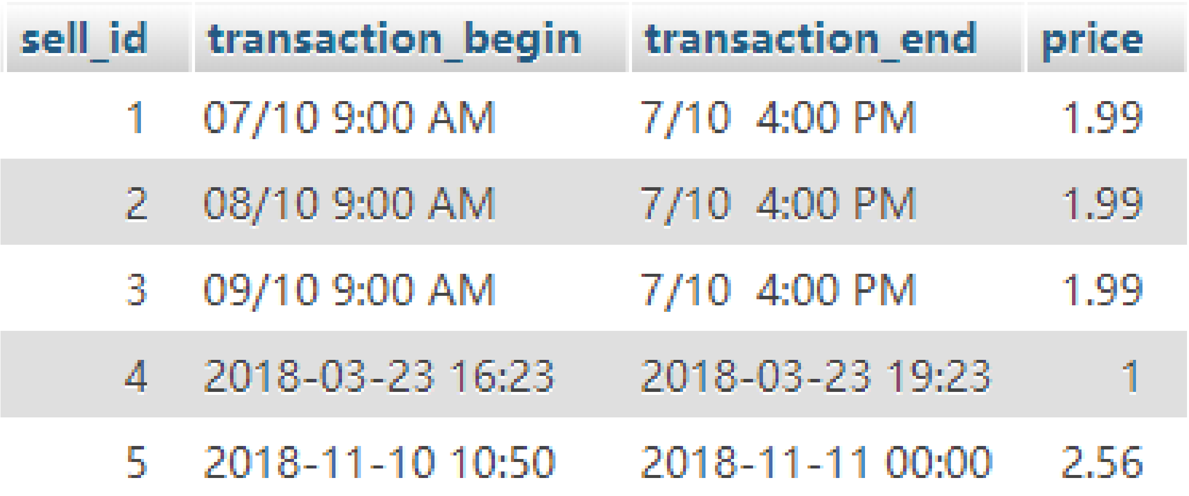
```
function validateDateTime($dateStr, $format){
    date_default_timezone_set('UTC');
    $date = DateTime::createFromFormat($format, $dateStr);
    return $date && ($date->format($format) === $dateStr);}
if(!validateDateTime($begin,'Y-m-d H:i')||!validateDateTime($end,'Y-m-d H:i')){
    header("Location:../Sell.php?sell=invalidTime");
    exit();}
```

When users fill all sell information, the sell data will be insert to database flowing clicked the “begin to sell” button. Meanwhile, users are going to be informed by this web page.



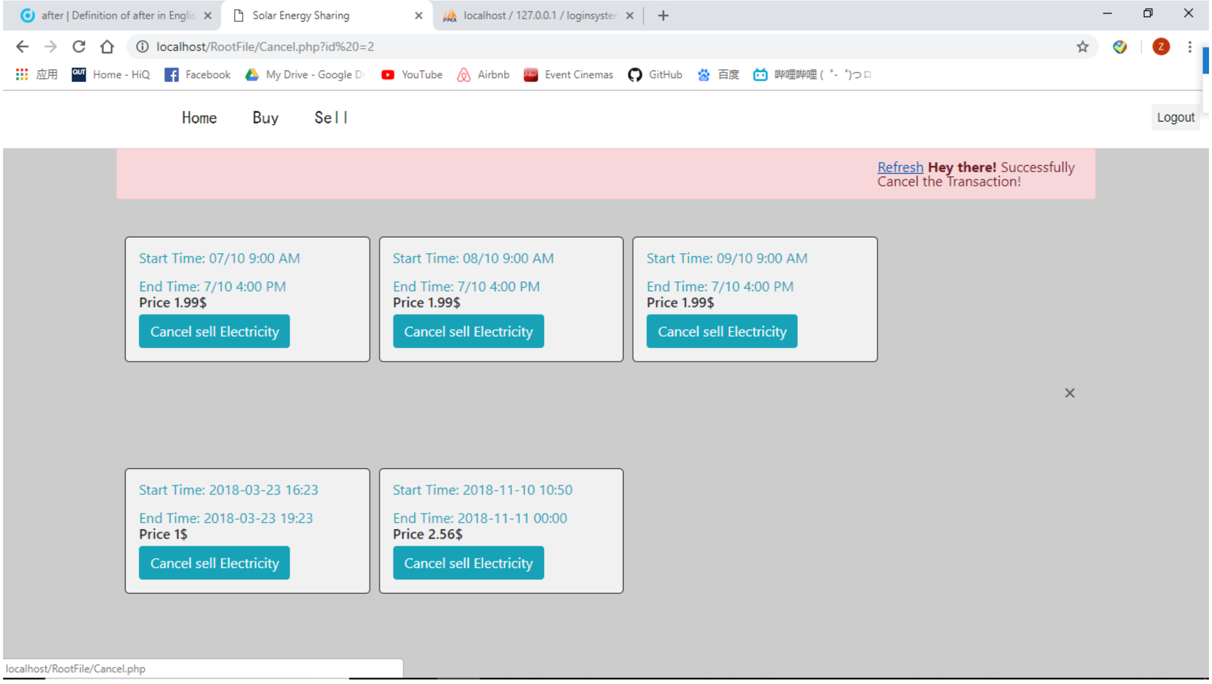
The sell information has been added to the schedule

Accessing the local/phpMyAdmin local website can show the detail information for “sell_information” table, which demonstrate in under graph.



sell_id	transaction_begin	transaction_end	price
1	07/10 9:00 AM	7/10 4:00 PM	1.99
2	08/10 9:00 AM	7/10 4:00 PM	1.99
3	09/10 9:00 AM	7/10 4:00 PM	1.99
4	2018-03-23 16:23	2018-03-23 19:23	1
5	2018-11-10 10:50	2018-11-11 00:00	2.56

In addition, if sellers want to cancel the transaction, clicking the “Cancel” button would open the “Cancell.php” (show in under graph).


after | Definition of after in Englis
Solar Energy Sharing
localhost / 127.0.0.1 / loginsystem
localhost/RootFile/Cancel.php?id%20=2
应用
Home - HiQ
Facebook
My Drive - Google D
YouTube
Airbnb
Event Cinemas
GitHub
百度
Home
Buy
Sell
Logout
localhost/RootFile/Cancel.php

Refresh **Hey there!** Successfully Cancel the Transaction!

Start Time: 07/10 9:00 AM
End Time: 7/10 4:00 PM
Price 1.99$
Cancel sell Electricity

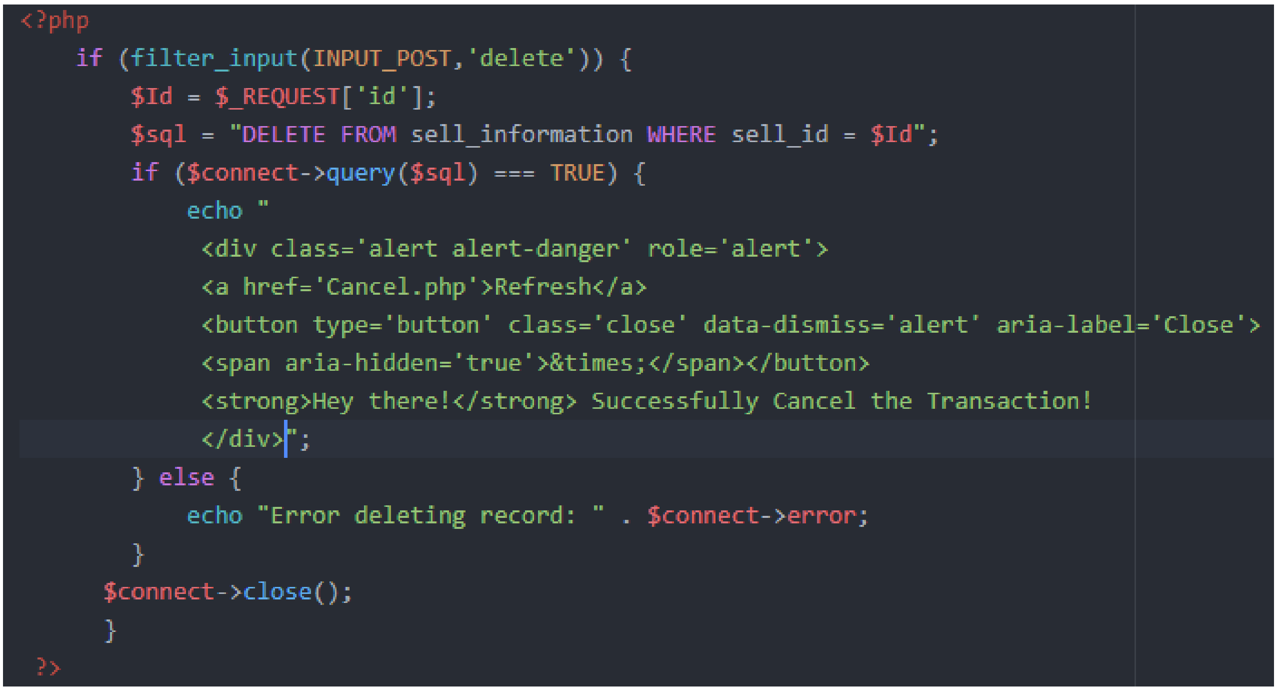
Start Time: 08/10 9:00 AM
End Time: 7/10 4:00 PM
Price 1.99$
Cancel sell Electricity

Start Time: 09/10 9:00 AM
End Time: 7/10 4:00 PM
Price 1.99$
Cancel sell Electricity

Start Time: 2018-03-23 16:23
End Time: 2018-03-23 19:23
Price 1$
Cancel sell Electricity

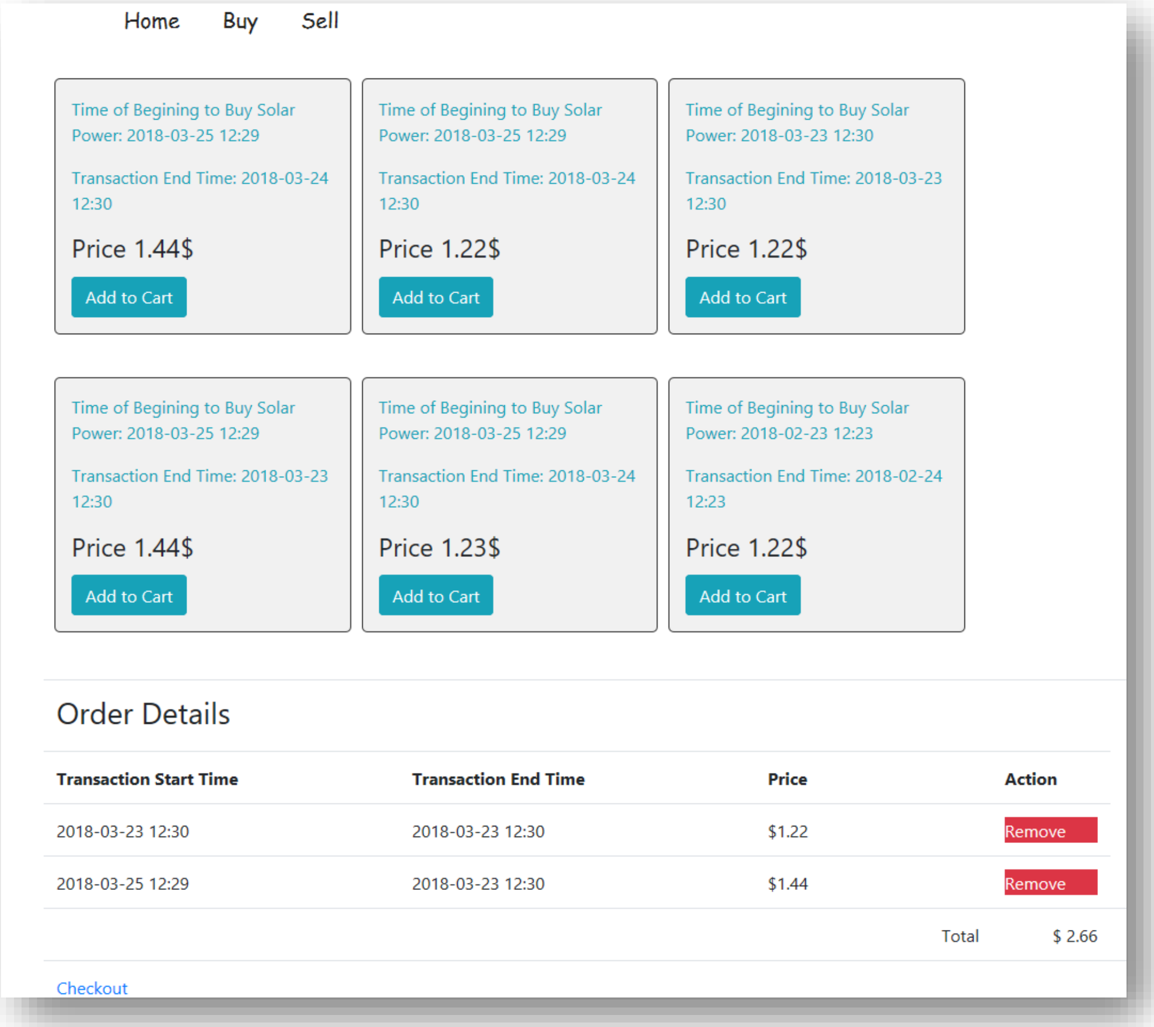
Start Time: 2018-11-10 10:50
End Time: 2018-11-11 00:00
Price 2.56$
Cancel sell Electricity

This website will demonstrate all your selling information. If cancelling a transaction is required, click the blue button named “Cancel sell electricity”. A red alert information is shown in top of website and announce to sellers the transaction has already been cancelled. The related code is shown below:



```
<?php
    if (filter_input(INPUT_POST,'delete')) {
        $Id = $_REQUEST['id'];
        $sql = "DELETE FROM sell_information WHERE sell_id = $Id";
        if ($connect->query($sql) === TRUE) {
            echo "
             <div class='alert alert-danger' role='alert'>
             <a href='Cancel.php'>Refresh</a>
             <button type='button' class='close' data-dismiss='alert' aria-label='Close'>
             <span aria-hidden='true'>&times;</span></button>
             <strong>Hey there!</strong> Successfully Cancel the Transaction!
             </div>";
        } else {
            echo "Error deleting record: " . $connect->error;
        }
      $connect->close();
      }
 ?>
```

For Buy website, there are four functions which are display sell information, display order details, add “remove” button and check out function. The interface is shown in under screen shot. Each white window can be view as an electricity product, which record the time to start use solar energy and time to stop use it. Furthermore, if user do not login, he or she cannot buy the electricity.



Time of Begining to Buy Solar Power: 2018-03-25 12:29

Transaction End Time: 2018-03-24 12:30

Price 1.44$

Time of Begining to Buy Solar Power: 2018-03-25 12:29

Transaction End Time: 2018-03-24 12:30

Price 1.22$

Time of Begining to Buy Solar Power: 2018-03-23 12:30

Transaction End Time: 2018-03-23 12:30

Price 1.22$

Time of Begining to Buy Solar Power: 2018-03-25 12:29

Transaction End Time: 2018-03-23 12:30

Price 1.44$

Time of Begining to Buy Solar Power: 2018-03-25 12:29

Transaction End Time: 2018-03-24 12:30

Price 1.23$

Time of Begining to Buy Solar Power: 2018-02-23 12:23

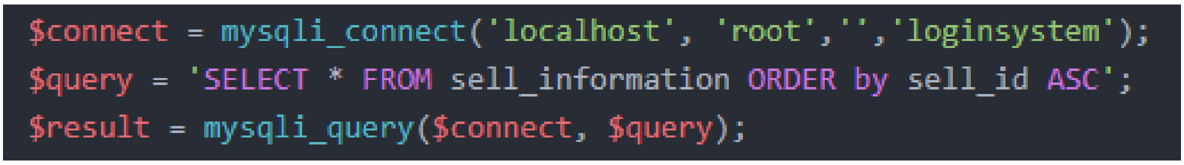
Transaction End Time: 2018-02-24 12:23

Price 1.22$

Order Details

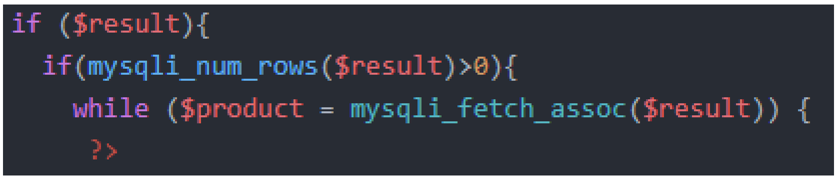
Transaction Start Time	Transaction End Time	Price	Action
2018-03-23 12:30	2018-03-23 12:30	$1.22	Remove
2018-03-25 12:29	2018-03-23 12:30	$1.44	Remove
		Total	$ 2.66

To show sell information, the first step is to connect to the database and run MYSQL query to select all electricity products from “sell_information” table. The code is shown below:



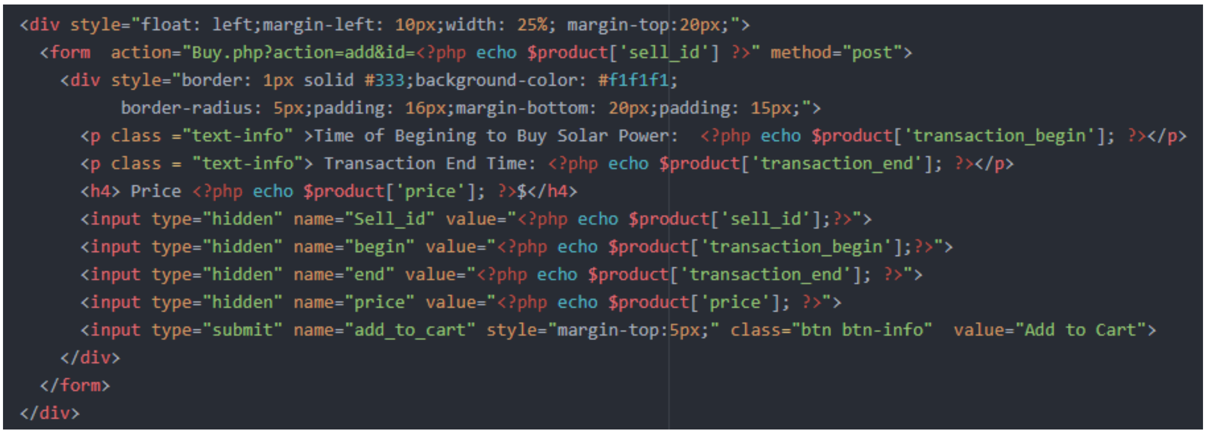
```
$connect = mysqli_connect('localhost', 'root','','loginsystem');
$query = 'SELECT * FROM sell_information ORDER by sell_id ASC';
$result = mysqli_query($connect, $query);
```

The second step is to use while loop and mysqli_fetch_assoc() function to acquire all the results in sell information table and store them to $product list.



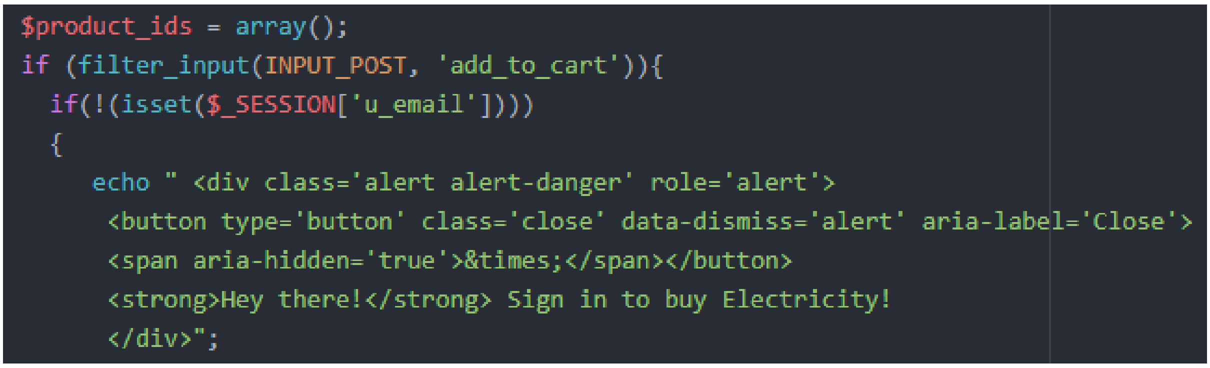
```
if ($result){
  if(mysqli_num_rows($result)>0){
    while ($product = mysqli_fetch_assoc($result)) {
      ?>
```

The third step is to demonstrate sell information data inside “add to cart form” with information about Time of beginning to buy solar power, transaction end time, price and add to cart button. The code is displayed in under picture:



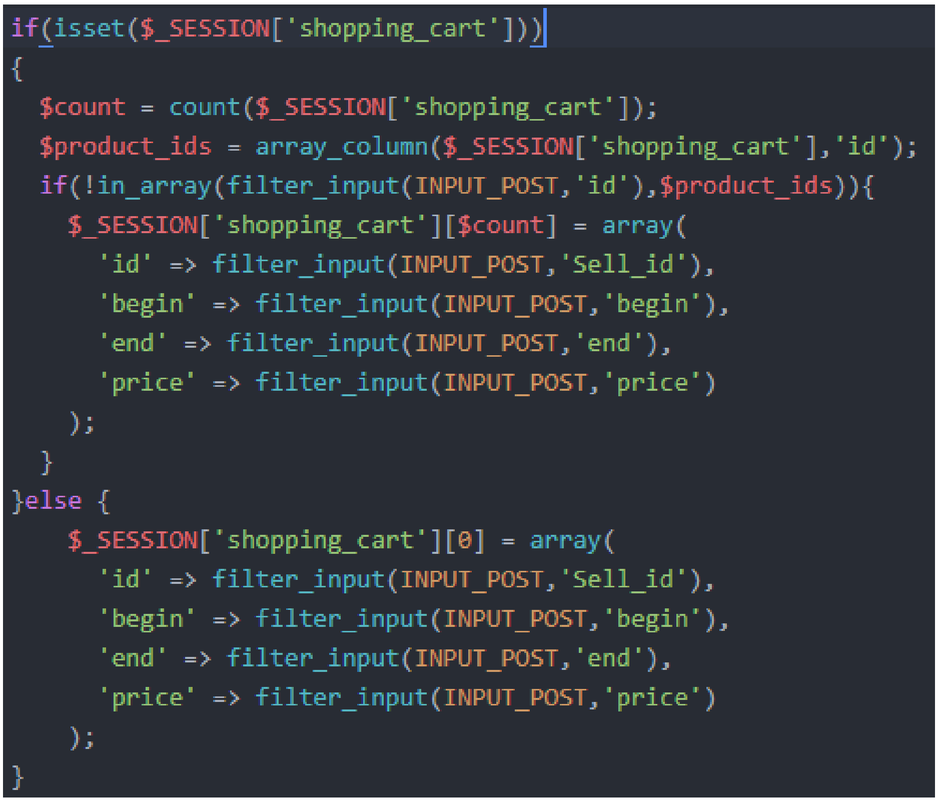
```
<div style="float: left;margin-left: 10px;width: 25%; margin-top:20px;">
  <form  action="Buy.php?action=add&id=<?php echo $product['sell_id'] ?>" method="post">
    <div style="border: 1px solid #333;background-color: #f1f1f1;
          border-radius: 5px;padding: 16px;margin-bottom: 20px;padding: 15px;">
      <p class ="text-info" >Time of Begining to Buy Solar Power:  <?php echo $product['transaction_begin']; ?></p>
      <p class = "text-info"> Transaction End Time: <?php echo $product['transaction_end']; ?></p>
      <h4> Price <?php echo $product['price']; ?>$</h4>
      <input type="hidden" name="Sell_id" value="<?php echo $product['sell_id'];?>">
      <input type="hidden" name="begin" value="<?php echo $product['transaction_begin'];?>">
      <input type="hidden" name="end" value="<?php echo $product['transaction_end']; ?>">
      <input type="hidden" name="price" value="<?php echo $product['price']; ?>">
      <input type="submit" name="add_to_cart" style="margin-top:5px;" class="btn btn-info"  value="Add to Cart">
    </div>
  </form>
</div>
```

To display order details, the first step is to check whether the add to cart form has been submitted and whether users have signed to their account.



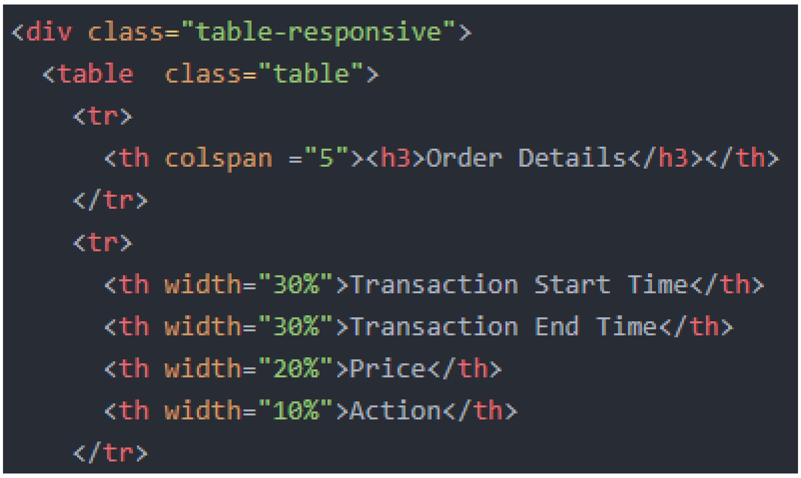
```
$product_ids = array();
if (filter_input(INPUT_POST, 'add_to_cart')){
  if(!(isset($_SESSION['u_email'])))
  {
     echo " <div class='alert alert-danger' role='alert'>
      <button type='button' class='close' data-dismiss='alert' aria-label='Close'>
      <span aria-hidden='true'>&times;</span></button>
      <strong>Hey there!</strong> Sign in to buy Electricity!
      </div>";
```

The second step is to conform if the $_SESSION shopping cart exists. If there is no shopping cart, a $_SESSION shopping cart array should be produced with submitted sell information data start from key equal to 0 and fill it with values. If the shopping cart has existed, a counter should be created to keep track the number of products in the shopping cart. Then, using sequential numeric $_product_ids array to track array keys and match them to product ids. Subsequently, Next product is added to existing $_SESSION shopping cart array via $count variable as next array key. The related code is demonstrated below:



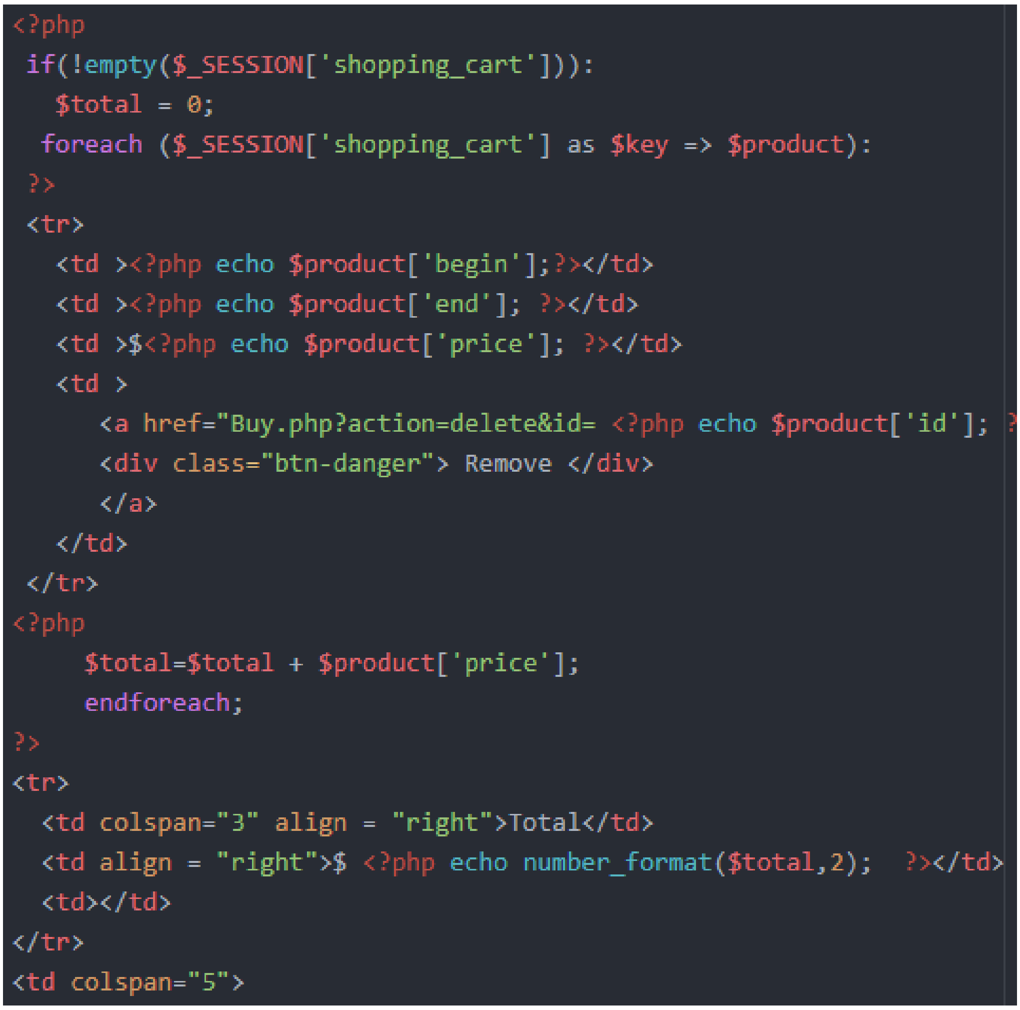
```
if(isset($_SESSION['shopping_cart']))
{
  $count = count($_SESSION['shopping_cart']);
  $product_ids = array_column($_SESSION['shopping_cart'],'id');
  if(!in_array(filter_input(INPUT_POST,'id'),$product_ids)){
    $_SESSION['shopping_cart'][$count] = array(
      'id' => filter_input(INPUT_POST,'Sell_id'),
      'begin' => filter_input(INPUT_POST,'begin'),
      'end' => filter_input(INPUT_POST,'end'),
      'price' => filter_input(INPUT_POST,'price')
    );
  }
}else {
    $_SESSION['shopping_cart'][0] = array(
      'id' => filter_input(INPUT_POST,'Sell_id'),
      'begin' => filter_input(INPUT_POST,'begin'),
      'end' => filter_input(INPUT_POST,'end'),
      'price' => filter_input(INPUT_POST,'price')
    );
}
```

The third step is to calculate the total price for customers order and display the order details. The table is used to record the product information such as transaction start time, transaction end time, price and action.



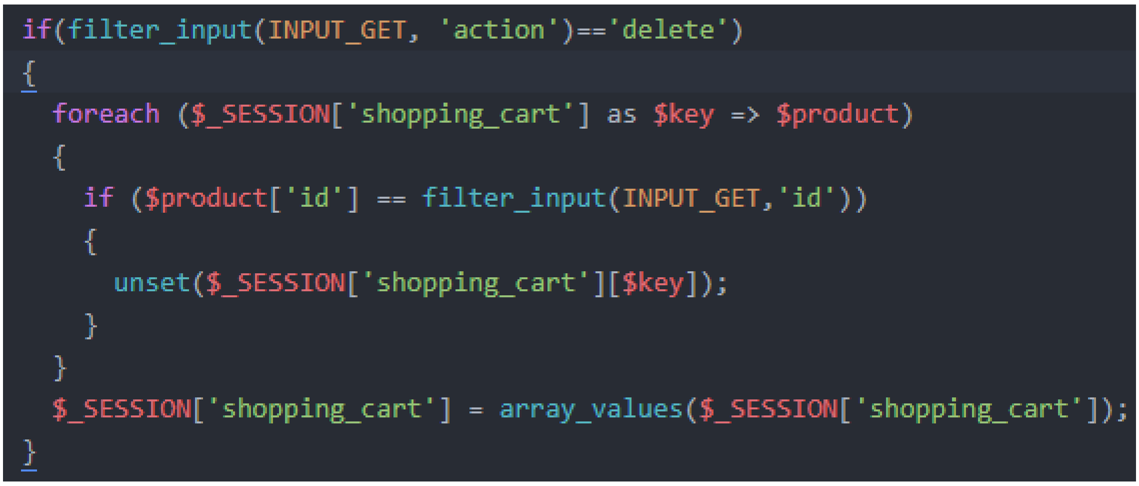
```
<div class="table-responsive">
  <table  class="table">
    <tr>
      <th colspan ="5"><h3>Order Details</h3></th>
    </tr>
    <tr>
      <th width="30%">Transaction Start Time</th>
      <th width="30%">Transaction End Time</th>
      <th width="20%">Price</th>
      <th width="10%">Action</th>
    </tr>
```

Then, using foreach loop is applied to demonstrate sell information on solar power in shopping cart. Furthermore, calculate the total price for buying solar energy and create remove button hyperlink that pass the deleted product id to URL.



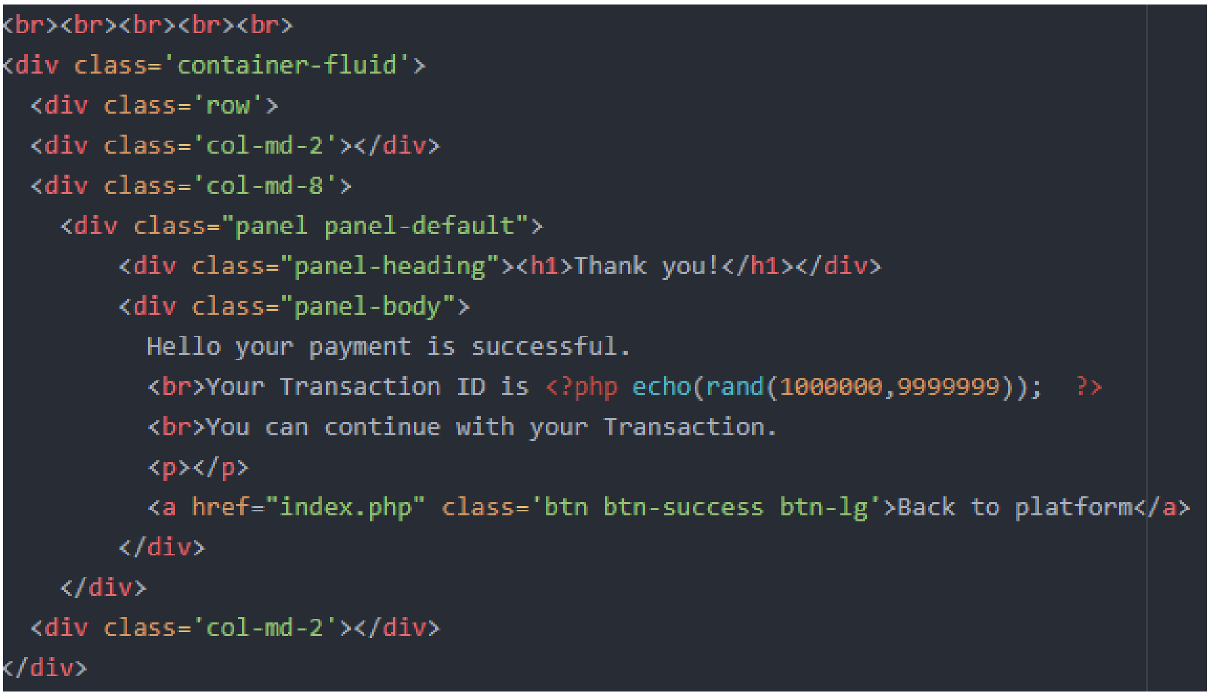
```
<?php
 if(!empty($_SESSION['shopping_cart'])):
   $total = 0;
  foreach ($_SESSION['shopping_cart'] as $key => $product):
 ?>
 <tr>
   <td ><?php echo $product['begin'];?></td>
   <td ><?php echo $product['end']; ?></td>
   <td >$<?php echo $product['price']; ?></td>
   <td >
      <a href="Buy.php?action=delete&id= <?php echo $product['id']; ?
      <div class="btn-danger"> Remove </div>
      </a>
   </td>
 </tr>
<?php
     $total=$total + $product['price'];
     endforeach;
?>
<tr>
  <td colspan="3" align = "right">Total</td>
  <td align = "right">$ <?php echo number_format($total,2);  ?></td>
  <td></td>
</tr>
<td colspan="5">
```

To achieve the “remove” button function, “if” statement is employed to judge whether URL contain the word delete. If URL have word delete, foreach loop is wielded to read each element in $_SESSION [‘shopping_cart’]. At same time, find the value of $product [‘id] which is equal with ‘id’ value extracted from URL. Afterwards, unset() function is operated to delete it.

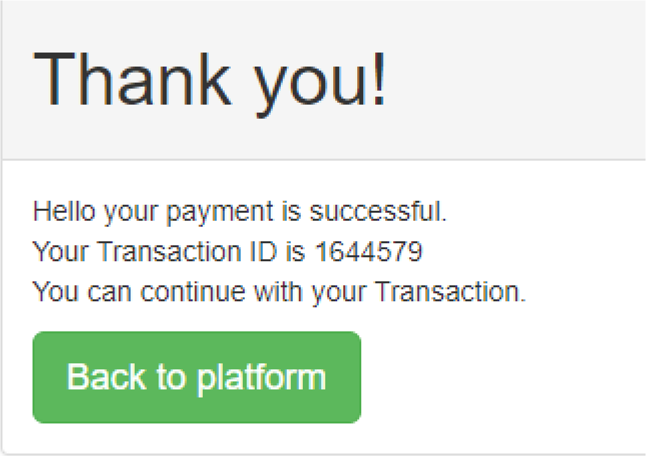


```
if(filter_input(INPUT_GET, 'action')=='delete')
{
  foreach ($_SESSION['shopping_cart'] as $key => $product)
  {
    if ($product['id'] == filter_input(INPUT_GET,'id'))
    {
      unset($_SESSION['shopping_cart'][$key]);
    }
  }
  $_SESSION['shopping_cart'] = array_values($_SESSION['shopping_cart']);
}
```

In checkout function, the bootstrap library is utilized to design the layout, font and colours of receipt. The rand() function is applied to generate the transaction ID. The code and result is shown in below. However, use rand() function may get the same receipt number.



```
<br><br><br><br><br>
<div class='container-fluid'>
  <div class='row'>
  <div class='col-md-2'></div>
  <div class='col-md-8'>
    <div class="panel panel-default">
        <div class="panel-heading"><h1>Thank you!</h1></div>
        <div class="panel-body">
          Hello your payment is successful.
          <br>Your Transaction ID is <?php echo(rand(1000000,9999999));  ?>
          <br>You can continue with your Transaction.
          <p></p>
          <a href="index.php" class='btn btn-success btn-lg'>Back to platform</a>
        </div>
    </div>
  <div class='col-md-2'></div>
</div>
```



Thank you!

Hello your payment is successful.
Your Transaction ID is 1644579
You can continue with your Transaction.

Discussion

During the 702 project, there are three key finding about web developing. The first finding is back-end and front-end code position do not have an explicitly boundary. In the past, I have prejudice, which consider the front-end code must separate with back-end code. Therefore, at the begin of this project, a folder named ‘include’ are prepared to put all back-end code into it. When the Buy.php was being developing, I realize HTML and CSS can be embedded into PHP code. Conversely, PHP code also could be embedded into HTML. Both on Buy.php and Sell.php, the while loops are utilized to produce many HTML forms. There, there are two type of web programming. One is to spilt back-end and front-end code during programming. Another is mix them together. Another finding is that URL play a significant role for deliver information from front- end to back-end. For instance, the $_POST[] variable is the most general method to obtain the value from front-end. Furthermore, even information is transferred on the same page, PHP and HTML still use URL to deliver messeage. For, example filter_input() function is utilized to get the information from <Input type = “hidden”> element. The third key finding is that the SESSION variable is import for website. The SESSION variables are utilizing the user-specific information and can be accessed by multiple pages. The login system is based on the feature of SESSION variables. In addition, another key finding is one feature on transaction of electricity generated by solar panel. The first feature the transaction of solar power may continue for a long time. The reason is the solar energy is not a common product, which more like rent business. You actually like to borrow the neighbour’s PV system. Thus, time become the measurement unit to sell or buy the solar energy in this trade web application(Zhang, Wu, Zhou, Cheng, & Long, 2018).

At the end of semester, the basic functions of online trade web application are accomplished. This web application consists of two system, incorporating by login system and trade system. Users are able to do solar energy business on this platform. However, there are several limitations in this web application. The first limitation is the interface is too simple, which cannot attract people to use this website. Thereby, we need more user-friendly interface such as add more graphs or videos. The third limitation is user input check. This web application will let user retype all the information if customer make a single error, so it will make users be annoyed when they input many information. The input check method should be modified, which can keep the right input information for users. The fourth limitation is about security. The main purpose of Online trade web application of Peer-to-Peer energy sharing platform is transaction. Nevertheless, this web application does not have a completed system to ensure the security of trading. Meanwhile, there is no any measures to protect the user's information like name, residential address and email address. Finally, the payment system is not completed yet. Current payment website only can generate a receipt, so it is still not an eligible payment system.

In the future work, we will focus on three area to improve this web application. The first is more user-friendly. We need to add other background website to introduce the Peer-to-Peet solar power sharing system and list the benefits for participating this platform. the second area is to create more original table in database. The current website only has two table in database, which record electricity vending information and user's information. The new table is required to save the trade information. The Third improvement is to make this application can acquire distance between sellers and buyers via google API. The reason to add this function is to help the scheme system calculate the minimum distance between generator and consumers. When there are many generators and consumers, schedule system can contract the closet generator and consumer, which increase the energy utilization. In addition, since I only implement one part of P2P solar energy sharing system, there are still three parts including Power management unit, scheme systems, and appliance control web application. Hence, our team can assemble these four parts and establish a real Peer-to-Peet solar energy sharing platform.

Conclusion

In this paper, firstly we mention three type of Peer-to-Peer energy sharing platform architecture including blockchain based, Online matching and battery-based platform. Blockchain based P2P platform provide a decentralised database to identity the ownership of electricity and arrange numerous trading agreements between suppliers and buyers. Online matching P2P platform authorizes clients to purchase energy straight from small-scale distributed energy resources. Battery storage based P2P platform permit client to reserve their excess energy and market to other neighbours in peak energy consumption. our team select battery based P2P platform, which is composed of Photovoltaic Panels, Battery, Main Grid and Energy sharing Coordinator. Since our team only focus on software solution, we concentrate on energy sharing coordinator. According to previous research, there are two basic structure for battery based P2P platform, which are Power Management Unit (PMU) and day-ahead scheduling algorithm. Power Management Unit consist of consume and generator model, which estimate the batteries' charge situation and load's current energy consumption. In the basic of batteries' states, Power management unit do determine which battery should charge and which should discharge. Day-ahead scheduling algorithm is used to do load forecast and generator forecast based on the electricity consumption demand and Historical solar irradiation data. There are some significances for researching P2P solar power sharing platform. Increasing of retail electricity prices is an essential motivation. Climate changing and bringing financial profit also encourage clients to join P2P energy sharing platform.

According to above literature review, our team design a battery based P2P platform comprised by online trade website, Power management Unit, home appliance control web application and scheme system. my project scope is to implement online electricity web application. To develop web application, ATOM editor and XAMPP are used to program and run the code. The outcome is a Root File folder that contain HTML, CSS and PHP source code. This web application has Five web page. The home page is used to achieve Login/login out function, which can check the format of input information, register user account to database and sign in to existed account. Buy, Sell, Payment and Cancel web page make up commerce system, which support buy or sell electricity. The basic online trade web application is accomplished, but there still several limitations. Such as, the interface is not user-friendly and lack of security system. In the future, it is better to add background website to introduce the benefit of P2P solar power sharing system. In addition, more tables in database should be created to store transaction information. This electricity trade application is a significant prototype to discuss how to trade the solar energy in the future.

Reflections on your learning

Project Challenge

In the process of doing the project, I have a few challenges.

In the first place, because the system is a new system that has not been developed and implemented before. I need to understand and research the relevant background. In this case, it has taken me much effort to do the literature research and industrial study.

Second, I have not learnt PHP language before the project. Hence, it has taken me much time to study the language.

Strength & Weakness

There are a few strengths regarding the project.

Firstly, this system can help to trade and deal the solar power, which can reuse the renewable source and make profit on it. Secondly, to trade the solar power online has considered the natural condition and situation of Australia, which makes use of the advantages of the solar energy based in Australia.

During my work, it has some weaknesses. Firstly, because I have not learnt PHP language before, it was going slow at the beginning of the project.

What I have achieved from Project

I have gained a lot from this project.

In the first place, I have learnt PHP programming, which has enforced my programming skills.

Second, I have understood the importance to manage my timetable during multitasks. In this semester, I have 3 units. All of three units have a lot of assignments and heavy content. I did not manage my schedule very well at the beginning, which made me stressed when all of the tasks come together. Especially this project, it has double credits, which means it twice heavy compared with the other two units. I would manage the tasks one by one, instead of leaving everything in the last second.

Reference

Burman, C. J., & Aphane, M. A. (2016). Leadership emergence: the application of the Cynefin framework during a bio-social HIV/AIDS risk-reduction pilot. *African Journal of AIDS Research, 15*(3), 249-260. doi:10.2989/16085906.2016.1198821

Inam, W., Strawser, D., Afridi, K. K., Ram, R. J., & Perreault, D. J. (2015). Architecture and system analysis of microgrids with peer-to-peer electricity sharing to create a marketplace which enables energy access (pp. 464-469): Korean Institute of Power Electronics.

Kumari, P., & Nandal, R. (2017). A Research Paper OnWebsite Development OptimizationUsing Xampp/PHP. *International Journal of Advanced Research in Computer Science, 8*(5).

Long, C., Wu, J., Zhang, C., Thomas, L., Cheng, M., & Jenkins, N. (2017). Peer-to-peer energy trading in a community microgrid (Vol. 2018-, pp. 1-5): IEEE.

Long, C., Wu, J., Zhou, Y., & Jenkins, N. (2018a). Aggregated battery control for peer-to-peer energy sharing in a community Microgrid with PV battery systems. *Energy Procedia, 145*, 522-527. doi:10.1016/j.egypro.2018.04.076

Long, C., Wu, J., Zhou, Y., & Jenkins, N. (2018b). Peer-to-peer energy sharing through a two-stage aggregated battery control in a community Microgrid. *Applied Energy, 226*, 261-276. doi:10.1016/j.apenergy.2018.05.097

MacIntyre, P. (2010). *PHP* (1. ed.). Sebastopol: O'Reilly Media.

Ruotsalainen, J., Karjalainen, J., Child, M., & Heinonen, S. (2017). Culture, values, lifestyles, and power in energy futures: A critical peer-to-peer vision for renewable energy. *Energy Research & Social Science, 34*, 231-239. doi:10.1016/j.erss.2017.08.001

Zhang, C., Wu, J., Long, C., & Cheng, M. (2017). Review of Existing Peer-to-Peer Energy Trading Projects. *Energy Procedia, 105*, 2563-2568. doi:10.1016/j.egypro.2017.03.737

Zhang, C., Wu, J., Zhou, Y., Cheng, M., & Long, C. (2018). Peer-to-Peer energy trading in a Microgrid. *Applied Energy, 220*, 1-12. doi:10.1016/j.apenergy.2018.03.010

Hirayama, M., Yamamoto, T., Chang, M., Hwang, W., Chen, M., & Müller, W. (2011). Web Programming Education through Developing Online Shop Web Application. In *Edutainment Technologies. Educational Games and Virtual Reality/Augmented Reality Applications: 6th International Conference on E-learning and Games, Edutainment 2011, Taipei, Taiwan, September 2011. Proceedings* (Vol. 6872, pp. 397–398). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-23456-9_69

Gonçalves, L. (2018). Scrum. Controlling & Management Review, 62(4), 40-42. doi: 10.1007/s12176-018-0020-3

Cramer, L. (2012). *Building an Ecommerce Web Site Using Dreamweaver with PHP.* Carpenteria, CA: lynda.com.

Boiser, M. E. (Singer-songwriter). (2018). Agile series : Scrum Master toolkit. On Scrum Master toolkit. Place of publication not identified]: Technics Publications.