

Developing a Web-based Application for Energy Supply Systems

Mohammed Faza, Maulahikmah Galinium, & Matthias Guenther

Faculty of Engineering and Information Technology, Swiss German University, Indonesia

mohammed.faza@student.sgu.ac.id, maulahikmah.galinium@sgu.ac.id, matthias.guenther@sgu.ac.id

Abstract: An energy supply system consists of a system of power plants and transmission and distribution systems that supply electrical energy. The present project is limited to the modelling of the generation system. Its objective is the design and implementation of a web-based application for simulating energy supply systems using the Laravel framework. The project focuses on six modules representing geothermal energy, solar energy, biopower, hydropower, storage, and fossil-based energy that are allocated to satisfy a given power demand. It is executed as a time series modelling for an exemplary year with hourly resolution. The development of the software is divided into four steps, which are the definition of the user requirements, the system design (activity, use case, system architecture, and ERD), the software development, and the software testing (unit testing, functionality testing, validity testing, and user acceptance testing). The software is successfully implemented. All the features of the software work as intended. Also, the software goes through validity testing using three different input data, to make sure the software is accurate. The result of the testing is 100% accuracy with respect to the underlying model that was implemented in an excel calculation.

Keywords: Energy Supply System, Power Plants, Renewable Energy, Laravel Framework, Simulation Software, Web-Based Application.

1. Introduction

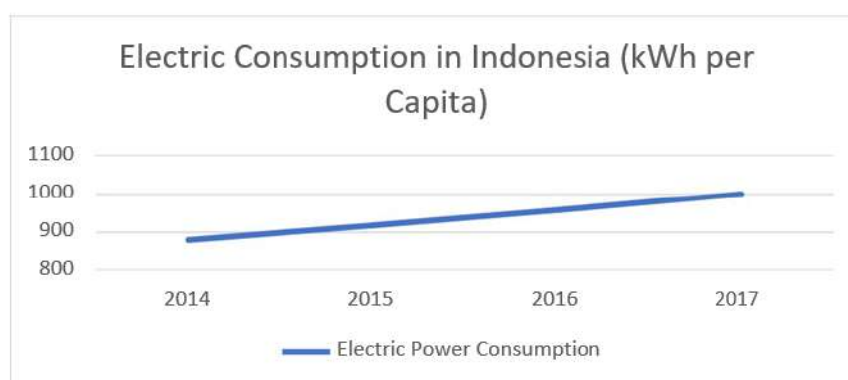


Figure 1. Electric Consumption (ESDM, 2017).

Figure 1 shows that in 2017, the electric power consumption in Indonesia is 1000 kWh per capita (ESDM, 2017). It steadily increased from 2014 until 2017. The installed power plant capacity in Indonesia is 60 GW (ESDM, 2017). It increased 7 GW from end of 2014 until 2017. In Indonesia non-renewable energy resources still dominate the generation of electricity, less than 10% is generated from renewable energy resources (Günther, Ganai, & Bofinger, 2018). If Indonesia continues relying on non-renewable energy resources, depletion of these resources will be a thread in the future. Also, the electricity prices tend to rise because the non-renewable energy resources are getting scarcer.

There are several factors why renewable energy is the best choice to replace the non-renewable energy. First, the price of renewable energy technologies, such as solar PV, has fallen (Tropitech, 2014). Second, renewable energy can be used also in the long run, by definition it does not get depleted. To use the renewable energy, it is required to have a supply system to manage the energy supply and to control the exceed energy from the power plants.

An energy supply system is a system that supplies electrical energy. The energy demand is not a fixed number, but it depends on the consumer. To simulate the energy supply system, a detailed calculation is required. This research focuses on six modules: solar energy, geothermal energy, biopower, hydropower, storage and fossil-based energy. Based on Matthias Guenther (personal communication, 16 November 2017), simple software that provides a calculation of the energy supply system is difficult to find. For each type of energy, there is a different data input and there are different formulas.

The objective of this research is to design and implement a web-based application for the modelling of energy supply systems. The software is developed to replace a previously existing excel tool by a web-based application. The study is significant in the sense of making an existing simulation strategy available to users.

2. Related Works

2.1. Homer Energy

Homer Energy is a software for finding optimal power mixes and for avoiding costly mistakes (HOMER Energy LLC, 2018). It has nine modules including Bioenergy, Hydropower, Combined Heat & Power, Advanced Load, Advanced Grid, Hydrogen, Advanced Storage, Multi-year, MATLAB Link (HOMER Energy LLC, 2018). The advantage of this software is that it has all energy modules. It has many features and it gives a tour to a new user. The disadvantage of this software is users are required to buy the software in order to get the full version, otherwise, the users only get the trial version.

2.2. Energy Plan

EnergyPlan is a software to simulate the energy systems for each hour (Aalborg University, 2015). It has been developed since 1999. EnergyPlan has five categories: electricity, heating, cooling, industry, and transport sectors for all national or regional energy system (Aalborg University, 2015). EnergyPlan's main purpose is to analyse the energy and analyse the economic impact of various energy strategies (Aalborg University, 2015). In EnergyPlan, each module can be compared with another module. It focuses on how the future energy system will be operating. The output of EnergyPlan is the energy balance, the total cost, the annual production, and the fuel consumption.

3. Methodology

3.1. Research Overview

There are four steps to developed the software. First, the software is developed with the user requirements. The second step is the system design. In the system design there is ERD, architecture diagram, flowchart, activity diagram, and use case diagram. The third step is the software development, in this software development there are software framework and calculation flowchart. In this step, the cycle goes to the user requirements because maybe there are new user requirements. The forth step is software testing to test the accuracy of the result from the software by comparing the software result with the excel result. In software testing there are unit testing, functionality testing, validity testing and user acceptance test (UAT). This step is to make sure the software works well as intended. Also, in this step, the cycle goes back to the user requirements because maybe there are new user requirements.

3.2. Architecture Diagram

The architecture diagram of this system software is the user accesses the website from a browser that sends a request to the server. The server calls the right route. Then the route calls the controller and the

controller calls the right view. Before the controller calls the right view, it goes to the database to retrieve the required data. After that, it goes back to the user's browser.

3.3. Activity Diagram

Figure 2 shows the Activity Diagram of the software. First, the user inputs the data of electricity used per hour. Then the user saves it into the database. After that, the user can choose the energy module. Next the user inputs all the required fields that show on the page. Each energy module has its specific input fields. After the user click submit, the data goes to the system and the system calculates the data and saves them into the database. Finally, the user can choose whether he wants to choose another energy module or whether he wants to view the result.

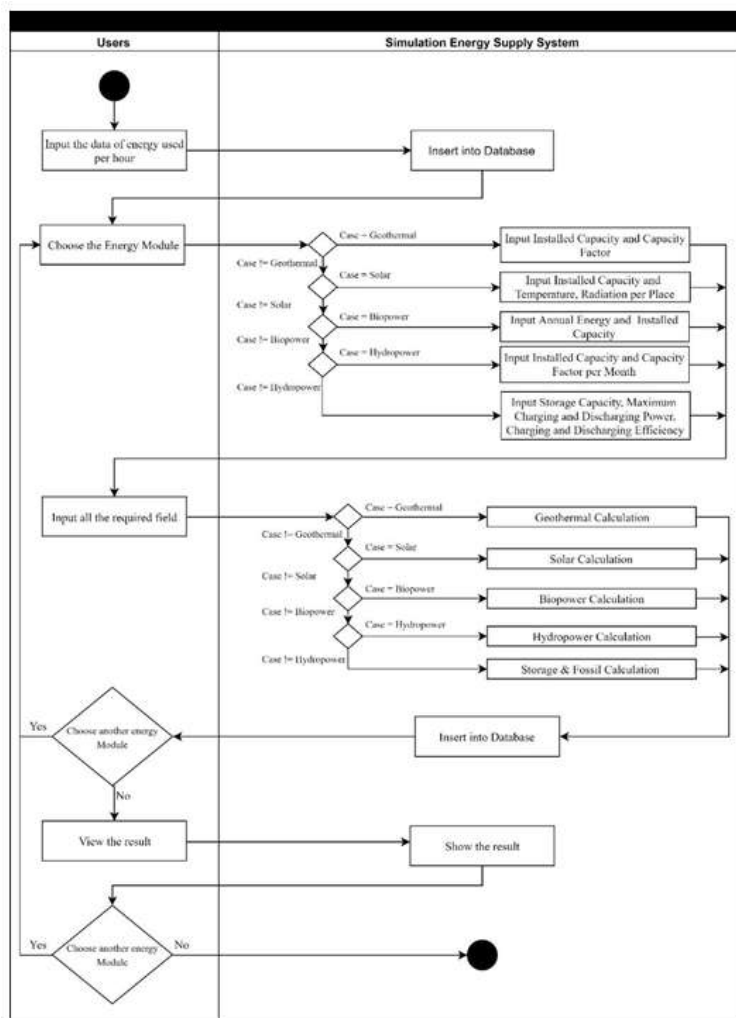


Figure 2. Activity Diagram

3.4. Software Framework

The software is built with Laravel Framework and the programming language of Laravel is PHP. There are several factors that makes Laravel Framework preferable to other frameworks. First, Laravel Framework already separates three folders, which are Model, View, and Controller (Nguyen, 2015). This makes the Laravel Framework easy to configure and to maintain. Second, Laravel Framework can have custom validations for the input fields. The software is a web-based application, which is easy to maintain and which is accessible both from mobiles or desktops.

Table 1: Validity Testing Input

Testing No.	Geothermal		Solar	Biopower		Hydropower												Storage					
	Installed Capacity (MW)	Capacity Factor (%)	Installed Capacity (MW)	Annual Energy (MWh)	Installed Capacity (MW)	Installed Capacity	Capacity Factor in Month (%)												Storage Capacity (MWh)	Max. Charge Power (MW)	Max. Discharge Power (MW)	Charging Efficiency (%)	Discharging Efficiency (%)
							1	2	3	4	5	6	7	8	9	10	11	12					
1	15000	85	300000	5000000	10000	10000	80	80	70	70	60	60	50	60	60	60	50	50	400000	20000	20000	80	80
2	5500	85	62200	12000000	5000	5800	70	75	65	50	40	32	26	22	30	40	50	60	20000	4000	3500	85	80
3	3000	85	50000 & 300000	5000000	2000	3000	40	50	50	40	30	30	40	50	60	50	40	40	20000	4000	3500	85	80

3.5. Validity Testing

Validity testing is done to check whether the algorithm that is used in the software gives the same results as the formulas in excel. In this paper, there are three validity tests with different inputs.

3.5.1. Load

The three validity tests use two different load time-series comprising one year, i.e. 8760 hours.

3.5.2. Modules

Table 1 shows the input for each module. In the solar energy module, there are name, irradiance, and temperature for two places. Each place has 8760 hours.

3.6. User Acceptance Testing

User Acceptance Testing (UAT) is a test to make sure that the software satisfies the user requirements before the software is released (Rouse, 2010). In this paper, the UAT was made by one person only who is an expert in supply systems. Only the user authentication and main features that is tested with the UAT.

4. Results

4.1. Validity Testing Result

Table 2: Comparison Result

No.	Module Name	Testing Number	Software		Excel		Accuracy
			Result 1 (in MWh)	Result 2	Result 1 (in MWh)	Result 2	
1	Geothermal	1	111.690.000	32.9 %	111.690.000	32.9 %	100 %
		2	40.953.000	12 %	40.953.000	12 %	100 %
		3	22.338.000	19.6 %	22.338.000	19.6 %	100 %
2	Solar	1	110.967.780	32.6 %	110.967.787	32.6 %	99.99 %
		2	121.166.290	35.6 %	121.166.295	35.6 %	99.99 %
		3	47.656.935	41.9 %	47.656.937	41.9 %	99.99 %
3	Biopower	1	5.000.000	1.5 %	5.000.000	1.5 %	100 %
		2	12.000.000	3.5 %	12.000.000	3.5 %	100 %
		3	5.000.000	4.4 %	5.000.000	4.4 %	100 %
4	Hydropower	1	69.608.725	20.5 %	69.608.731	20.5 %	99.99 %
		2	23.392.061	6.9 %	23.392.068	6.9 %	99.99 %
		3	11.328.580	10 %	11.328.583	10 %	99.99 %
5	Storage	1	42.733.464	12.6 %	42.733.482	12.6 %	99.99 %
		2	5.533.607	1.6 %	5.533.609	1.6 %	99.99 %
		3	10.676.238	9.4 %	10.676.550	9.4 %	99.97 %
6	Fossil	1	0	0 %	0	0 %	100 %
		2	135.981.480	40.3 %	135.981.484	40.3 %	99.99 %
		3	16.731.598	14.7 %	16.731.629	14.7 %	99.98 %

Table 2 compares the simulation result from the software and from the excel tool for scenario 1, 2, and 3. Result 1 specifies the total of electricity that is covered with the different energy modules and the Result 2 specifies the percentage the respective type of energy contributes to the power mix. For some types of energy, the accuracy is 100%. But there are other types where the difference amounts to 0.01% to 0.03%.

The difference occurs because in the software the result values for each formula are not rounded, only the final result is rounded. On the contrary, in excel, all values are rounded. For example, in the solar energy module there are four functions, and one function is a loop through 8760 iterations. In the software, the value of each iteration is not rounded so that the final result is not 100% accurate, even if the difference is only 0.01% to 0.03%. For the geothermal energy module, which contains only one function, the accuracy of the result in the software with respect to the one of the excel tool is 100 %. For the fossil energy module, even if it contains only one function has an accuracy below 100% because the function depends on the storage function that has sixty-nine functions. For biopower, which has nine functions, the accuracy is 100%.

For all three validity tests the results are almost 100%, which means that the algorithm of the software mirrors very well the formulas in excel. These results can be used for the decision makers.

4.2. User Acceptance Testing Result

The result of UAT for each feature of the software is all the features are acceptable by the user. There are some remarks on some features of the software. For the user authentication it was commented that the gender field is not necessary for the usage of the software. For the simulation system it was commented that the example file in load and solar page should contain 8760 lines because the system should execute simulations for one year. For the simulation results it was commented that the charts should be in the right position with respect to the load curve.

5. Conclusion

The result of this paper is to replace the existing excel tool by a web-based application. The software is successfully designed and implemented as explained in the Results section. The software is developed using Laravel framework. The programming language is php. The software is tested with four different test types: unit testing, functionality testing, validity testing, and user acceptance testing. The result of unit testing and functionality testing is that all the features of the software work perfectly. The result of the validity test is that the calculation algorithm is accurate as explained in the Result section. The result of the UAT is that the software is acceptable for doing the simulation of energy supply systems.

References

- Aalborg University. (2015). Introduction to EnergyPLAN. Retrieved January 2, 2018, from <http://www.energyplan.eu/training/introduction/>
- ESDM. (2017). Berapa Konsumsi Listrik Perkapita Indonesia? Retrieved June 5, 2018, from <https://databoks.katadata.co.id/datapublish/2017/09/28/berapa-kapasitas-pembangkit-listrik-nasional>
- Günther, M., Ganai, I., & Bofinger, S. (2018). A 100 % Renewable Electricity Scenario for the Java-Bali Grid, 7(1), 13–22.
- Nguyen, Q. H. (2015). Building a web application with Laravel 5. Oulu University of Applied Sciences. Retrieved from http://theseus32-kk.lib.helsinki.fi/bitstream/handle/10024/113987/thesis_laravel5-final_31_05_2016.pdf?sequence=1
- Rouse, M. (2010). user acceptance testing (UAT). Retrieved June 22, 2018, from <https://searchsoftwarequality.techtarget.com/definition/user-acceptance-testing-UAT>
- Tropitech. (2014). Why do we need energy storage? Retrieved May 14, 2018, from <http://www.tropitechsolutions.com/news/ask-tropitech/renewable-energy/140-why-do-we-need-energy-storage>