

Group 2 - Solar Energy Calculator

Design description

Lukas Hamacek, Aliya Hussain, Charlie Höglund
Jonathan Larsson, Sebastian Lindgren, Avalika Podduturu Reddy

December 5, 2016

Contents

1	System overview	4
1.1	Background	4
1.2	Existing calculators	4
1.3	Current system	4
1.4	Our system	4
1.5	Main goals	5
2	Description of the system	5
2.1	Summarized	6
2.2	Extra functionality	7
3	Software architecture	7
3.1	MySQL	7
3.2	Scripts	7
3.3	Extra libraries	7
4	Detailed Software design	8
4.1	Coupling	8
4.2	Cohesion	8
5	Graphical User Interface	8

List of Figures

1	The use case diagram	6
2	The header of the start page	9
3	The calculator section	10
4	The about section	10
5	The contact section	11
6	Login	11
7	Uploading draft	12

1 System overview

1.1 Background

The interest for solar energy plants (i.e. photovoltaic plants) has been increasing a lot during the last years, but the general knowledge about them is still quite low. However, our client and his colleagues have developed detailed models for analyzing investment decisions in photovoltaic plants in Sweden. Now, our task is to develop a user-friendly web-calculator that is based on these models, which should help to determine what investments in solar energy that are most suitable for the potential investors. These potential investors are for example private persons and companies. See the project description document for more details.

1.2 Existing calculators

There are several already existing solar energy calculators, but most of the current ones are not very user-friendly. A majority of them are also in the form of excel sheets or websites that are not up to date in regard to for example energy-certifications and laws. Here are a few links to some available web-calculators that were given to us by the client:

- <http://www.elforsk.se/calculator/>
- <http://www.solelprogrammet.se/projekteringsverktyg/berakningsverktyg/>
- http://www.solkollen.nu/sv_SE/
- http://www.nrel.gov/analysis/tech_lcoe.html
- <http://www.energy101.com/free-educational-tools/calculators/>
- <http://re.jrc.ec.europa.eu/pvgis/apps4/pvest.php>

1.3 Current system

The system that is currently being used is in the form of an Excel file (i.e. one each for private persons and other users), in which the user has to input the necessary default values inside cells and then some built-in functions calculate the result. Here is a link to the available Excel files on MDH:s website: <http://www.mdh.se/forskning/inriktningar/framtidens-energi/investeringskalkyl-for-solceller-1.88119>

1.4 Our system

Our own web application will at least use the same functionalities as the ones in this already existing system, but bring it to the web. In other words, the already existing system will not be extended, but instead give us knowledge about the functional basis on which our web application will be built upon.

1.5 Main goals

The main goals of the calculator is for it to be user-friendly and also easily maintained by the administrator. Additionally, it should be able to provide both correct and easily understood results in the form of numerical values, tables and charts. How we will reach these goals and design the system, will be explained in detail in the rest of this document.

2 Description of the system

The idea is that you can either use the web-calculator as a registered or non-registered user. The only difference is that a registered user should be able to both save calculated results and compare two different calculated results. To calculate the solar energy costs the users should first decide, if they are a private person or other user (e.g. company or property owner), which will determine some of the default input values regarding for example how much taxes to pay. Then the users should enter all of the default input values that are necessary for the calculations. In addition to this, the users also get the opportunity to add extended/advanced input values, which are meant for the more experienced users, but is not a necessity. When all of the inputs have been entered, the users should get the calculated results of production cost, profitability and cash flow in the form of graphical tables and charts. At the end, the users should also be given the opportunity to download a report file containing both the input values and the results. Finally, there should be an administrator (i.e. a registered user) that also can upload an Excel file to update the input parameters (i.e. the default input values, minimum and maximum values, and guiding texts). The admin should also be able to download the previously uploaded Excel file just in case.

The desired functionalities of the system have been captured in a use case diagram which we have created in Figure 1. Besides this there is also a need for a user-friendly graphical user interface (e.g. through default input values and guiding texts), to make it easier for the users to navigate and use the web-calculator.

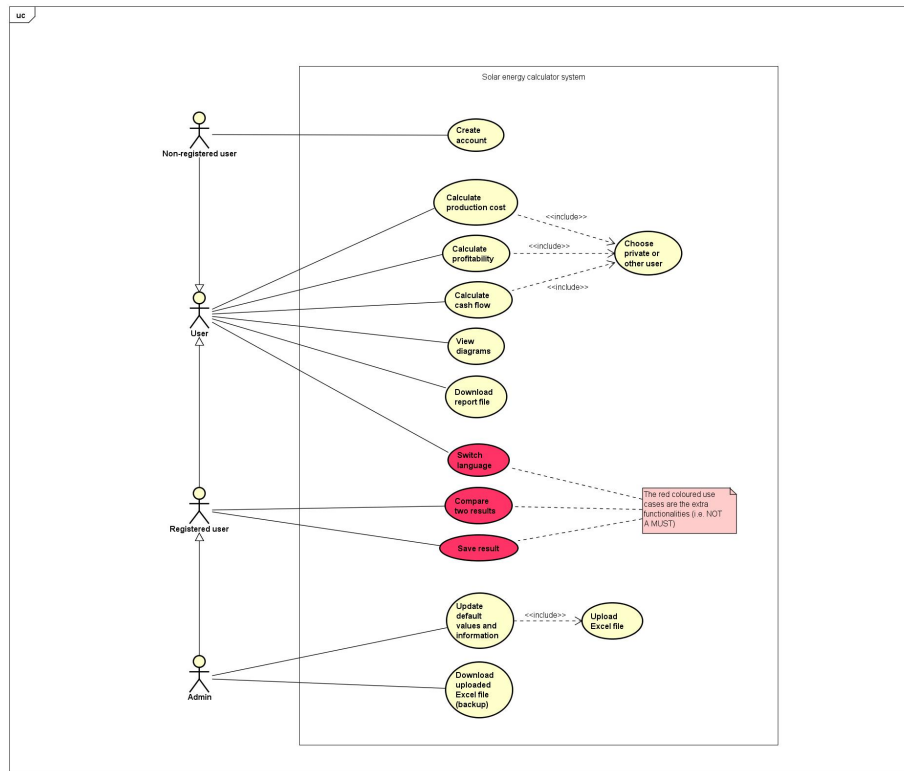


Figure 1: The use case diagram

The use case diagram covering the desired system functionalities.

2.1 Summarized

The functionality included in the calculator can be summarized as:

User:

1. Calculate production cost for the PV installations
2. Calculate profitability for the PV installations
3. Calculate cash-flow for the PV installations
4. Output diagrams and get a printable report file
5. Choose if you are a private person or company
6. Create account

Administrators can additionally upload a file to update the values on the site and the guiding texts. The administrator will also be able to download a backup file of the previous excel file.

2.2 Extra functionality

Furthermore, the extra desired functionalities of the system have been ranked after importance by the client (i.e. where 1 is the highest and 3 is the lowest). These functionalities are not a must but nevertheless would be appreciated by the client:

1. Compare the calculated values for two sets of input parameters
2. Save the used input values from one session to another session at a later time
3. Switch between Swedish and English language

3 Software architecture

Here we will talk about the software architecture and other related structure which the website will be built upon. Most important is that we right now aim for using a three-tier architecture for the client-server communication, which makes it easier to make changes to the different parts (i.e. the client, web and database server). For more software structure details see our Github repository (folder: Example for login and design).

3.1 MySQL

We will use a MySQL database (link: <https://www.phpmyadmin.net/>) where we store login data for the users, the default input parameters to be used (i.e. based on the latest upload by the admin) and the output report file.

3.2 Scripts

Since our website is a single page application except for the extra page that the logged in users and admins have (i.e. for the saved results, comparing results, uploading an excel file), then we will mostly use one large index file (index.php) where all of the HTML is being done. But the PHP and javascript code are probably going to be done in separate files and folders (i.e. to divide the work more evenly). The calculations will be done in JQuery since all of the calculations are static. For the design we will use Bootstrap, which will allow us to use already pre-made buttons, tables, a grid system and other helpful CSS classes.

3.3 Extra libraries

We will use the PHPExcel library for updating the databases default parameters through an Excel file, and the Google Charts javascript library for displaying the different output charts. Furthermore, we will also use Meedo (link: <http://medoo.in>) for the database handling in order to make the development

faster and more secure, since this comes with help for preventing any type of SQL injection. Lastly, we will also use the FPDF library for creating the output report files as PDFs.

4 Detailed Software design

The typical workflow between components is that we have a controller (index.js), api:s (in the api folder) and a view (index.php). The controller will get the information needed from the api and set values in the scope so that the view can use these.

4.1 Coupling

We have low coupling since we have a lot of different components that does not build upon each other. The different libraries that we use do not depend on each other, and we could easily switch different parts of the project without the other parts becoming unusable. This goes for most of the code as well, since for example if a good substitute were to be found for one part and is implemented, then the rest of the system should not be disrupted.

4.2 Cohesion

We have pretty high cohesion. This is because we have, with the exception for the registered users extra page, a one page web application. These two pages are thus filled with a lot of different functionality that have a clear task and responsibility, for example the separate file uploading functionality and login functionality (i.e. they are separate but depend on each others clear tasks) etcetera.

5 Graphical User Interface

Most of the functionality is described in previous sections. Here you'll instead get a taste of how the GUI is looking at the moment.

This website is not complete, it is missing the following things: A separate admin page (i.e. for uploading and downloading excel files), a separate page for saved and compared results for registered users... But these will be added in due time and follow the same graphical model.

Right now we don't have the calculations either but they will be there later. Soon there will be diagrams, tables etc. as well.

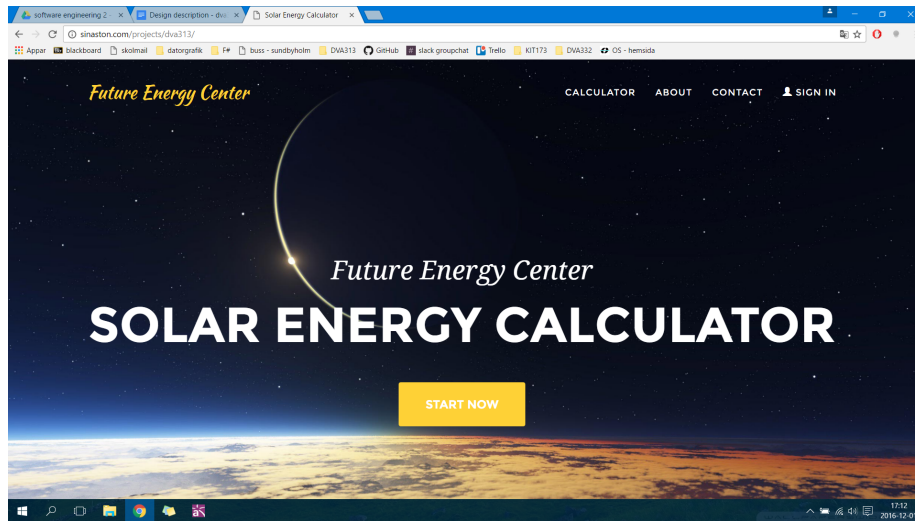


Figure 2: The header of the start page

Here you can see a navigation bar with the buttons for going to the calculator (i.e. start now button included), about and contact section of the start page. You can also see the sign in button for the registered users.

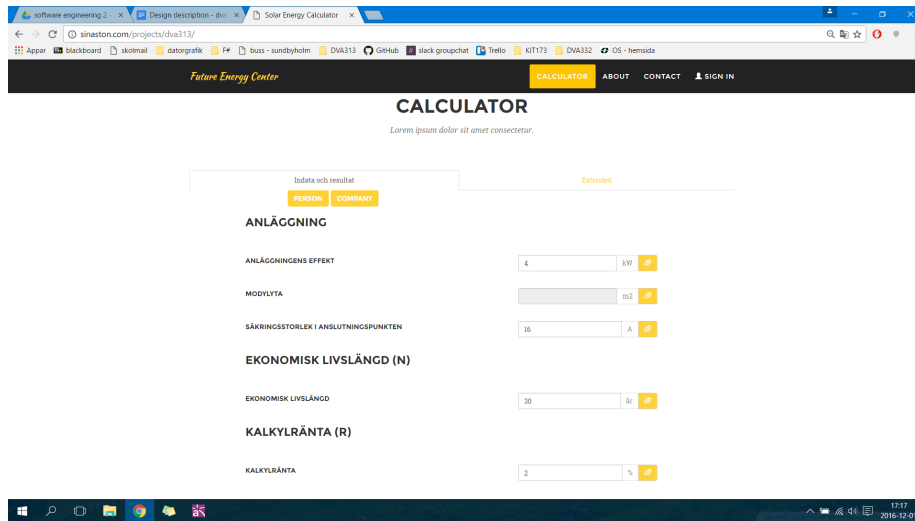


Figure 3: The calculator section

Here you can see the calculator section of the start page. It contains the two tabs: indata och resultat for entering the basic default values, and extended for entering the more advanced values. Both these tabs contains several input boxes where the users can enter the values. Also notice the person and company buttons where the users can decide what kind of customer they are.

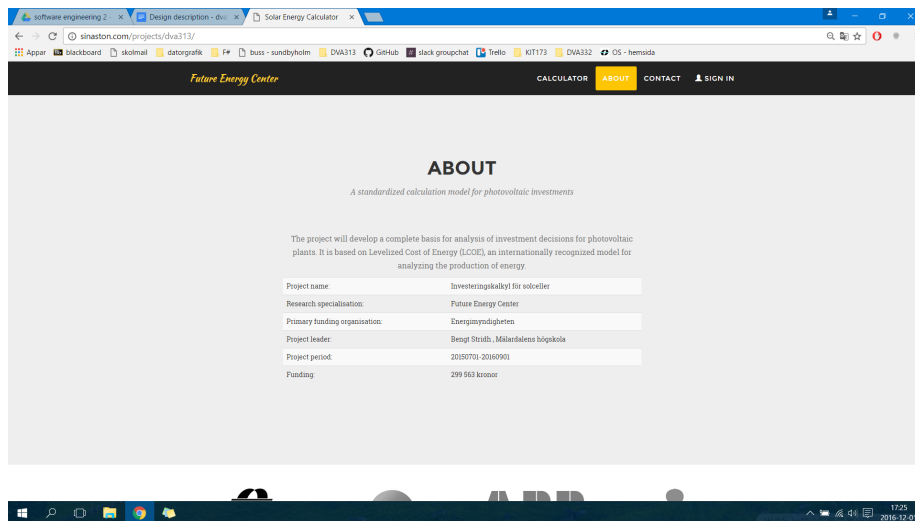


Figure 4: The about section

Here you can see the about section of the start page. It simply contains information about the researchers and their project, which will be updated later on.

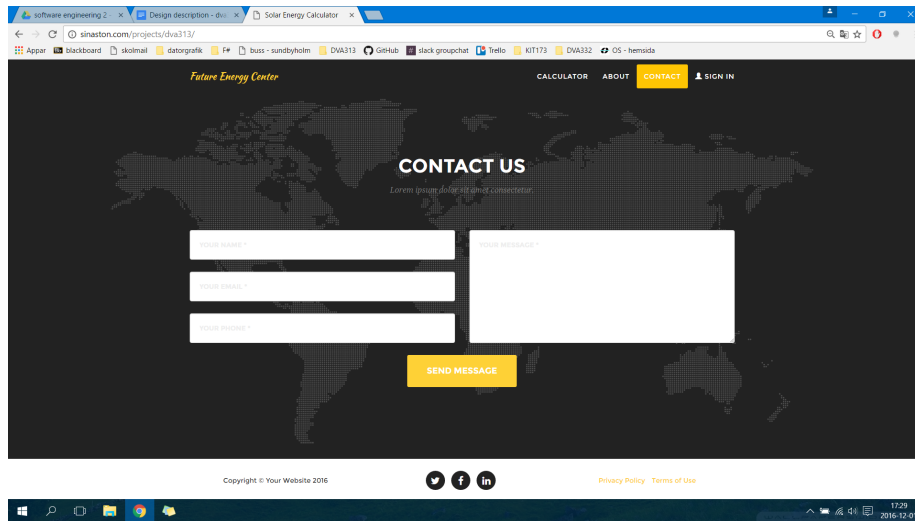


Figure 5: The contact section

Here you can see the contact section of the start page. It simply contains input boxes and a button for contacting the owners of the website.

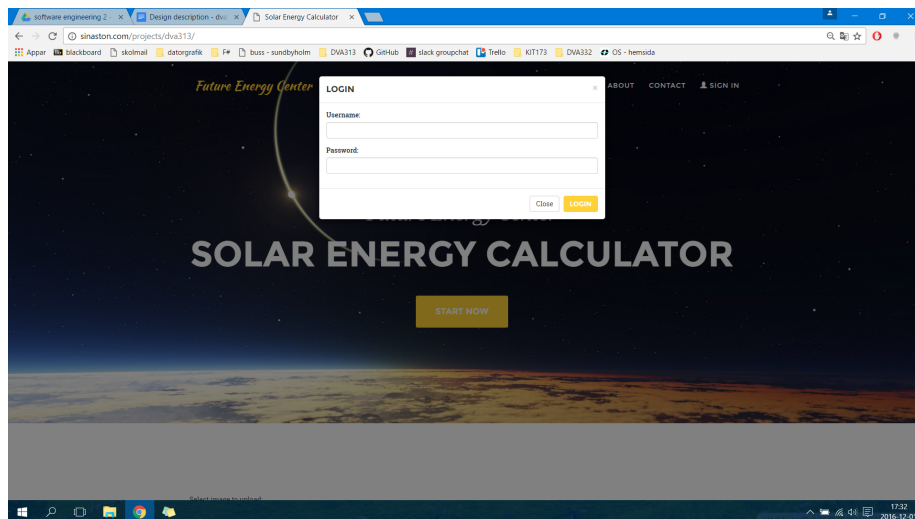


Figure 6: Login

Here you can see the login window that pops up after pressing the sign in button.

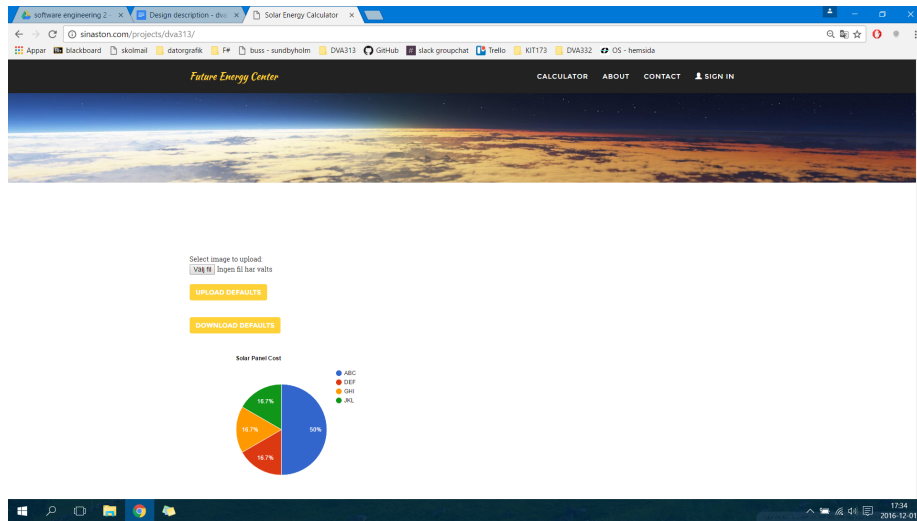


Figure 7: Uploading draft

A first draft of testing the uploading of an excel file, and downloading the backup excel file (I.e. only the admin). You can also see a pie chart that is supposed to show the result later on.