Team 5 Project Report

Engineering Foundation Year – Computer Applications Group Project

Team 5

Rasit Ahmetrasit - ra2u16

Matteo Trombini – mt4g16

Thomas Grantham – trg1u16

Will Hough – wh2g16

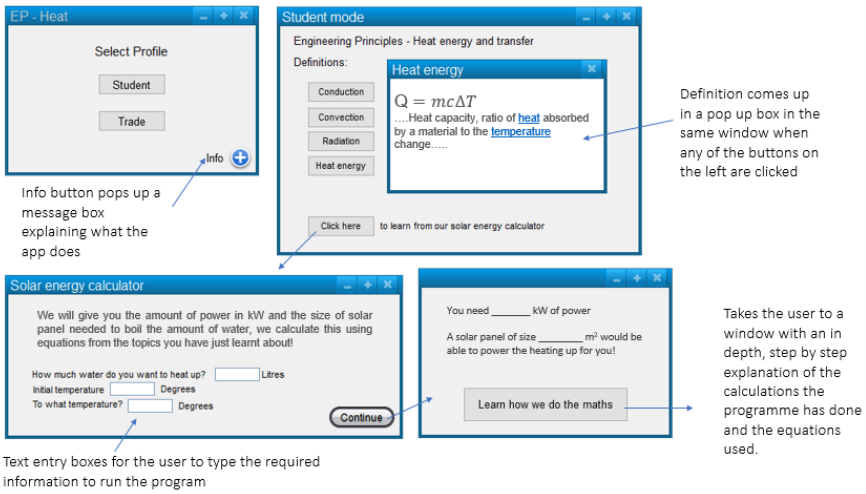
Miguel Torre – mat3g16

Evaluation of the application delivered

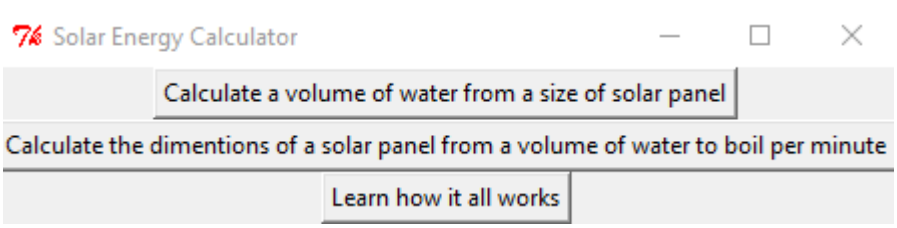
The development of our application was the final choice of many initial ideas we had in mind. At the beginning of the project we knew that collaboration of all participants was needed so we set up a group project file on GitHub to put our ideas into and make a time table to manage our time well.

Our goal was to build a calculator application that would help an Engineering Foundation Year student understand a theory from the course. From many ideas we had, we found out that many were already present on the internet so we wanted to create something different, therefore we chose to do an energy calculator that converts energy created by the solar panel to heat energy that is used to heat up the water in the tank and how much energy will be needed to boil certain litres of water at the room temperature.

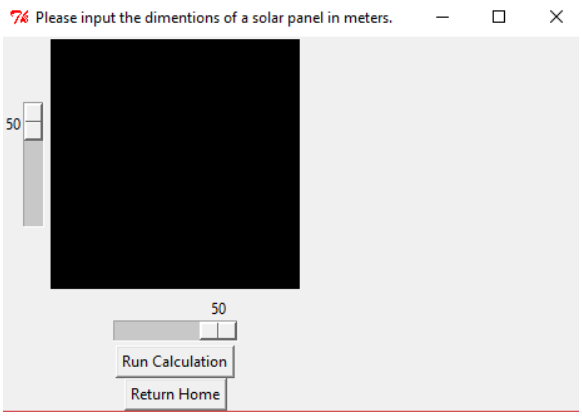
For increment 1, we designed a storyboard and a few personas and scenarios that would show how people can benefit from our application.



For increment 2, the idea of creating calculator was extended to make a working version of our idea. By the end of increment 2 we had developed a code that could turn the area of the solar panel into how much water will be boiled by the energy it produced and a code that determines how much energy is needed to boil a given amount of water that is at room temperature. To calculate how much heat energy the solar panel would produce, we took values from the internet to find out the efficiency of a solar panel and typical energy outputs.



Since our idea depends on the position of Sun and the weather conditions we assumed that the weather was always clear and the Sun perpendicular to the panel. For the other part of our code we added another section that calculates the energy needed to boil a certain amount of water from room temperature and this gives the user an estimate of the size of solar panel they need. In increment 3, GUIs were created to enhance the code and make good visual representation of our idea, as well as GUIs we created message boxes that the user can look at to understand the math theory behind our calculations. From the feedback that we got from increment 3 we changed the program to make it look visually nicer while keeping it simple.



We have added a home page where the user can choose which part of the code they want to use. We also separated the codes from each other and added them to the home page individually.

The message boxes were removed from the calculator section so the user wouldn't have to deal with message boxes during calculations and an extra section is created just for the theory part where the user can understand the theory behind it.

Test results:

Criteria	Mark (1 to 5)	Comments
Usability	4	It is easy to understand how to use the application
Effectiveness	3	The app is effective, the outputs given are correct, but the student might not gain much insight on the theory behind it due to the lack of detailed explanations.
Accuracy	3	The outputs given by the application are accurate, but the assumptions made to create the algorithms make it slightly unrealistic

Controls	4	The application includes good controls such as buttons and sliders.
Aesthetics	2	There are no background colours or attractive designs
Ease of use	5	The application is very easy to use
Graphics	4	The sliders create graphics of either the size of solar panel or amount of water
App flow	2	The flow of the application is not very good as too many pop-up windows appear, the theory also appears as message boxes. The user might find it slightly annoying to navigate through our app.
Text	3	The app contains an adequate amount of text in the interest of making it simpler and more effective, however, some instructions to the user are given in the window titles, which is not very convenient.
Improvements	The way new windows appear and the design and aesthetics of our application were definitely areas that we would have liked to improve. We did not manage to deliver an application as good looking as we would have liked to, however, the usability is reasonably good and the product is fit for its purpose.	

Evaluation of Team 11 application

This program is designed to teach the user about the charging and discharging of a capacitor, while also calculating different aspects of the capacitor as well as its transients. The program had four main sections, two theory sections explaining the concepts of charging and discharging a capacitor respectively, a calculator and graphing section where voltage, current and charge at any given time can be calculated and a test section where the user can test their new gained knowledge.

The application was fairly easy to understand. The home window was easiest to follow with clear and precise labels to the buttons. This made it obvious where the button would take you with each button opening a new window for the specific function requested. The theory pages were clear and concise with the explanation of the theory as the main feature of the window in one block of text in the centre of the window. The theory window also featured buttons that would take you elsewhere such as the calculator and home windows which was extremely useful as it made the program much easier to use. The calculator section was the least easy to use. It featured multiple sliders for the components of the calculations with buttons below for the different calculations and graphs. The buttons however, were not easy to use and understand as there was no labelling as to what the calculation was doing. The button opened another window which displayed the calculated answer as well as the inputted variables to calculate said answer, however this is not explained and the variables are simply displayed as numbers without much indication as to why they are there. Finally, the test section was easier to use, as it gave a standard list of buttons, each opening a question

for the user to solve. The program also featured a help section which offered a basic description of the program.

On the whole, the program ran as expected with few bugs or unexplained/unwanted functions or processes. The issues that were found were all in the calculation/ graphing section. Firstly, the calculation for the charging and discharging voltage had an issue with the time variable, displaying it as a set of numbers not even close to the slider input and thus causing a false voltage value calculated. The other main faults came with the graphs, due to the same issue with the time input for the voltage calculation. Another issue with the program was that the graphs were not displaying correctly in some of the sections, with some graphs showing a filled area, others showing just the data points as single pixels and others showing simply a rough line.

The program generally was very good though, with a large majority of the functions working and explaining the concepts with ease. The use of colour in the program was a nice addition as it made the program more aesthetically pleasing. Possible improvements would be fixing the bugs in the program, adding labels to the graphs so the user knows exactly what they are looking at, and providing more explanation on how to use the calculator section.

Evaluation of the experience of working on this project

How the team worked as a whole

The overall effort of the team was good with all team members contributing good ideas at the start of the project. Moreover, the work was spread quite equally with everyone either writing code, creating the storyboard or coordinating group efforts. Towards the end of the project some group members did not complete their responsibilities, which resulted in an overall decline in the quality of the result. This was overcome by the rest of the group picking up the slack.

What went well?

The efforts of coordinating went very smoothly and efficiently. The group used GitHub to delegate responsibilities and little jobs. We used a Facebook messenger chat to inform each other of deadlines and remind ourselves of work that still needed to be done. This not only helped us coordinate efforts, but also keep tabs on how much work the others had done.

What could have gone better?

The splitting of group work towards the last increments (when more coding was involved) was not balanced which resulted in the members of the group more proficient at coding having to do extra work to fill in the gaps. The final product was not exactly as intended in that some of the elements in the storyboard were absent, or simplified.

Were the XP values followed?

As we were creating the storyboard for the first increment submission we assigned jobs based on everyone's abilities. For example, Will and Matteo were in charge of writing the code as we believed them to be the most competent at coding - the others were in charge of the rest of design tasks such as testing and presenting the work and outcome of each increment.

Burn down

The table below shows a rough estimate of hours per increment spent by each team member:

Member	Increment 1	Increment 2	Increment 3	Increment 4
	Hours			
Matteo	2	5	4	2
Will	2	5	5	2
Tom	2	3	2	2
Rasjit	2	2	3	2
Miguel	3	3	0	1

Use of collaborative tools

The team used GitHub and Facebook Messenger to coordinate what work had to be done. Messenger was used more to keep tabs on people whilst we were not physically meeting or in the computer applications labs. Github on the other hand was used to share iterations of code we had created so that others could see and improve on it.

Communication/Co-operation

The team met physically two or three times, although in some occasions team members who did not respond to the messages directed at the group did not turn up. As stated above facebook was the best tool for us to keep in contact with each other.

Individual reflections

Matteo – 'Overall I felt as though the complete lack of input of some of the team members towards the end of the group was very demotivating and frankly unfair. If I could make suggestions to the computer applications teaching team next year is to allow groups to be formed on their own, so that those who wish to actually do the work will, and those that don't won't interfere with the other's. I very much enjoy coding, so even though half the team refused to code I still learned something. I do believe however the product could have been much better than what it is.'

Will – 'My experiences of the process have been generally positive. Despite not really knowing each other at the process we, for the most part, worked well together and created a high quality product. We did have some frustrating bumps in the road where people did not put in the effort to complete their allocated tasks, but these were quickly and calmly solved. Personally I have gained valuable management skills, such as learning when to let lack of effort slide and when to clamp down on it. As well as more advanced coding techniques that I had to learn to produce a quality, bug free, program.'

Rasit – 'In this group project of ours, over the period of 9 weeks I can say that I have gained many skills and realised that I had more potential in coding than I did at the beginning of this project. The most important skill I have gained is the ability to work with others and share the work load accordingly. Over this time, I had many struggles to keep up with the code that we were working on but writing the code for the theory part of our code and making small GUIs helped me to understand the code better thus I was more motivated to do more and contribute to my team.'

Miguel – 'The team effectively managed to deliver a working application that met the design specification of the project. I feel like the team showed good management skills and was able to solve most issues that appeared during the process. I was very involved during the first two increments, I designed most of the storyboard but fell behind after Increment 2 when I saw that the coding was being started without following the structure of the storyboard, due to this confusion I then found it hard to catch up, ending up not really knowing what I had to do for Increment 3 and not being able to provide any helpful work for that increment. I admit that I should have made myself more available for my team during the last stages, I felt that I lacked some coding skills and understanding of what was being done.'

Tom - 'The team worked very well together on this project in general. All of us knew that we had to work together effectively to be able to create a working application and as such did our best to help the team as best we could. As my coding ability is not as good as others, I contributed to the team more with the theory, testing and the non-coding side of the project. This included testing the code for increment two and part of three. Also, I had a hand in finding and implementing the theory behind our application, focussing mainly on the calculations from solar energy to heat energy in water. I was also responsible for the theory page in the main application. In my opinion, the group worked together very well with no personality clashes and no major conflicts. Also, none of the group members did their part of the project without complaining and in a timely manner. I feel that the group could have been improved by having more of us at a higher skill level in coding. This would not have limited us both time wise and with our possibilities for our application. Group communication could also have been improved as we had a group chat and meetings however, at times some group members would not check the group chat or come to the meetings, thereby limiting our abilities. However, in general, the majority of the group communicated quite well and managed to create a working application.'