# COMP3004A Team 6 Project documentation:

## Summary:

Denas-Simulator project is basically following the design and description of Assignment 2.

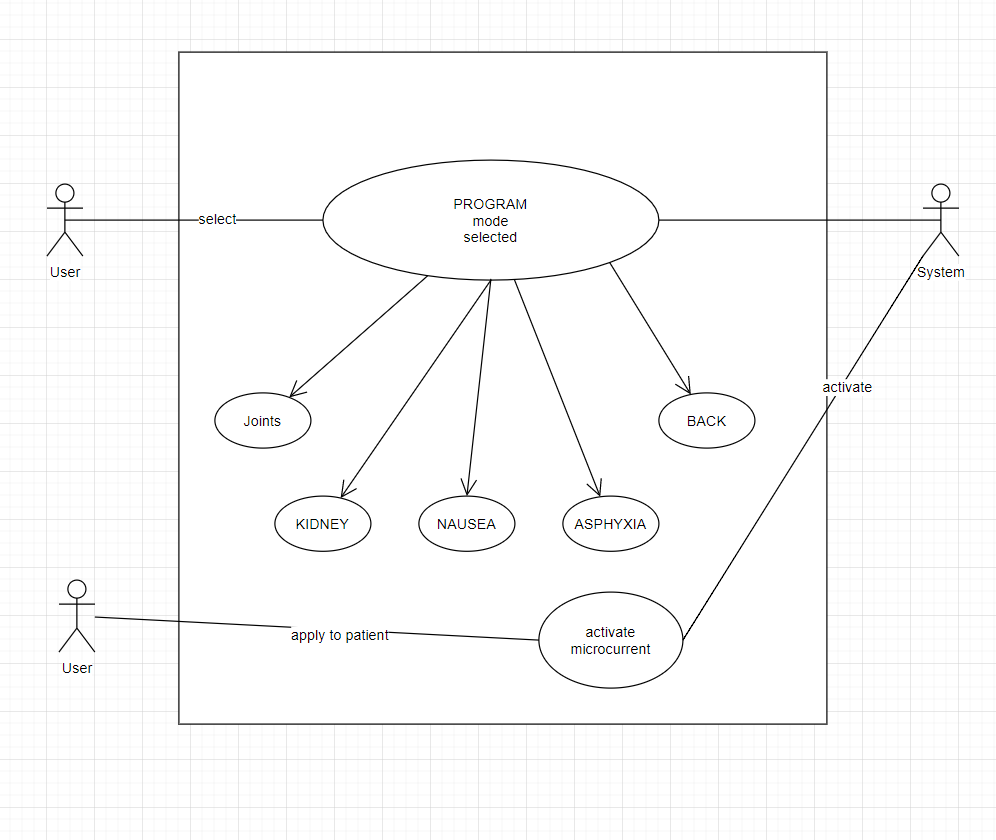
The simulator will cover 4 basic use cases:

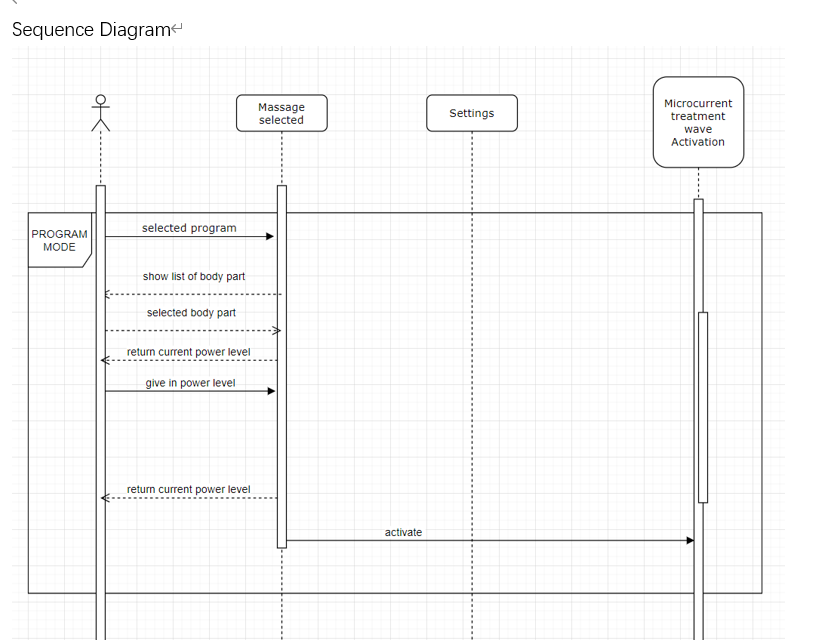
1. Program Therapy treatment
2. Frequency Therapy Treatment
3. MED Therapy Treatment
4. Settings changing language between English, French, Spanish

This document is meant to explain the use cases, providing basic procedure flows along with diagrams that we used in Assignment 2. Each sub section will contain basic use case diagram, basic explanation, and sectional code snapshot with explanations.

## Use Cases:

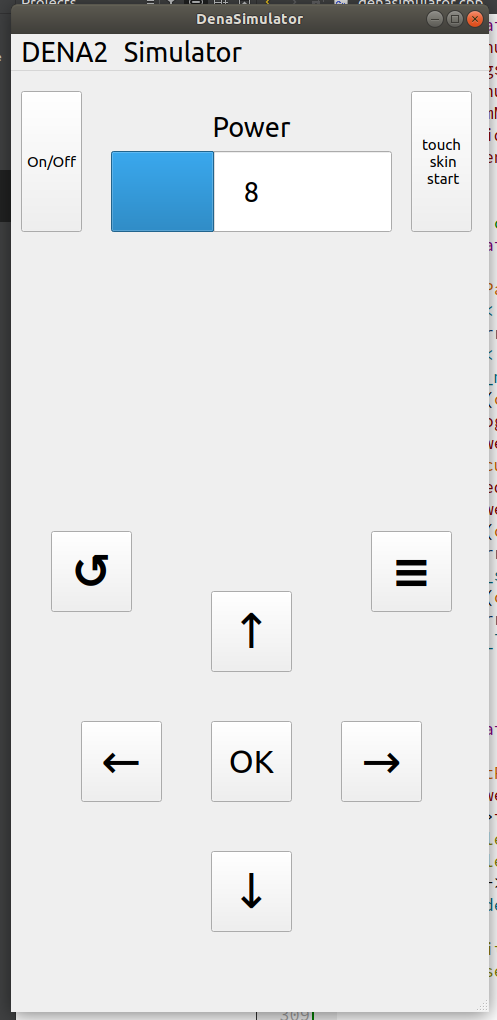
1. Program Treatment

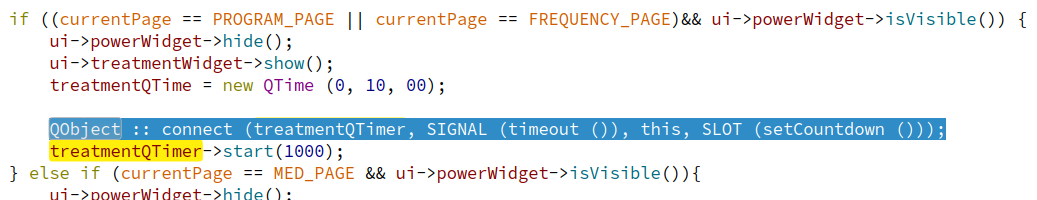


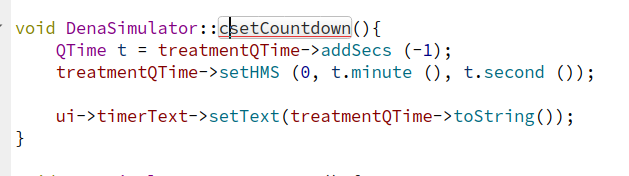
First case is the Pre-Programed Therapy. Basically there is a list of predefined settings (of frequency) for treating dedicated part of body or in some cases, certain type of symptoms such as Nausea. To start the treatment: 

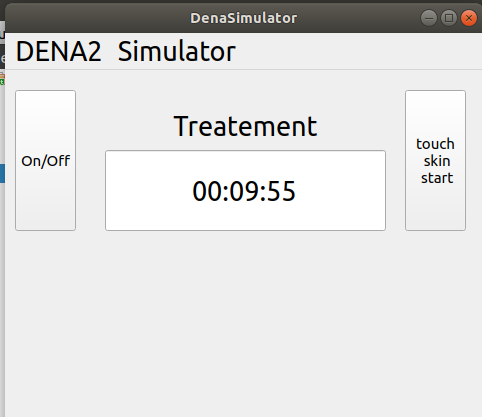
There will be a few steps:

1. At main menu, select program mode, in code, this is handled by handle\_main\_page\_selection(int) function, which is called when confirm button is clicked at main menu.
2. When in program selection menu, a list of body part / symptom will be available for selection. User can pick one according to his/her needs.
3. Upon selection, the power adjustment page will show up, and clicking left or right button will allow user to adjust the power level. As shown below



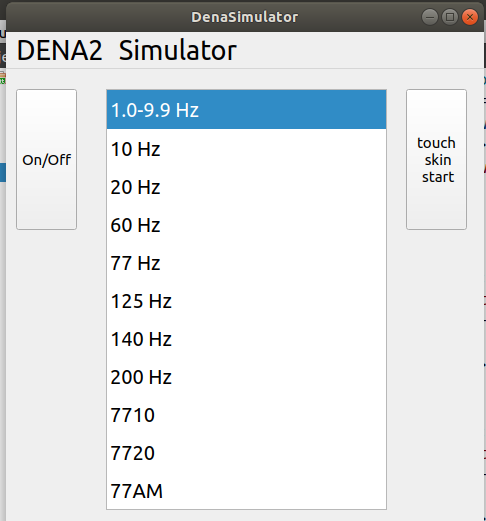
1. In real world, there will be a sensor on Dena2 to detect whether the machine has contacted human skin at the power selection page. In this simulator, we have a “touch skin start” button to simulate the behavior of that sensor. So when “touch skin start” button is clicked, the machine will start treatment by “emitting electro pulse”.
2. A count down timer will show up providing treatment durance. In this simulator, we used QTimer to do time out count. QTimer will have a slot(callback function) attached to it and triggered by timeout signal. 

Thus upon start, the setCountdown call back function will be triggered every 1 second (1000 milliseconds) and will decrease 1 second on QTime, which is a time formatter, and then display. 

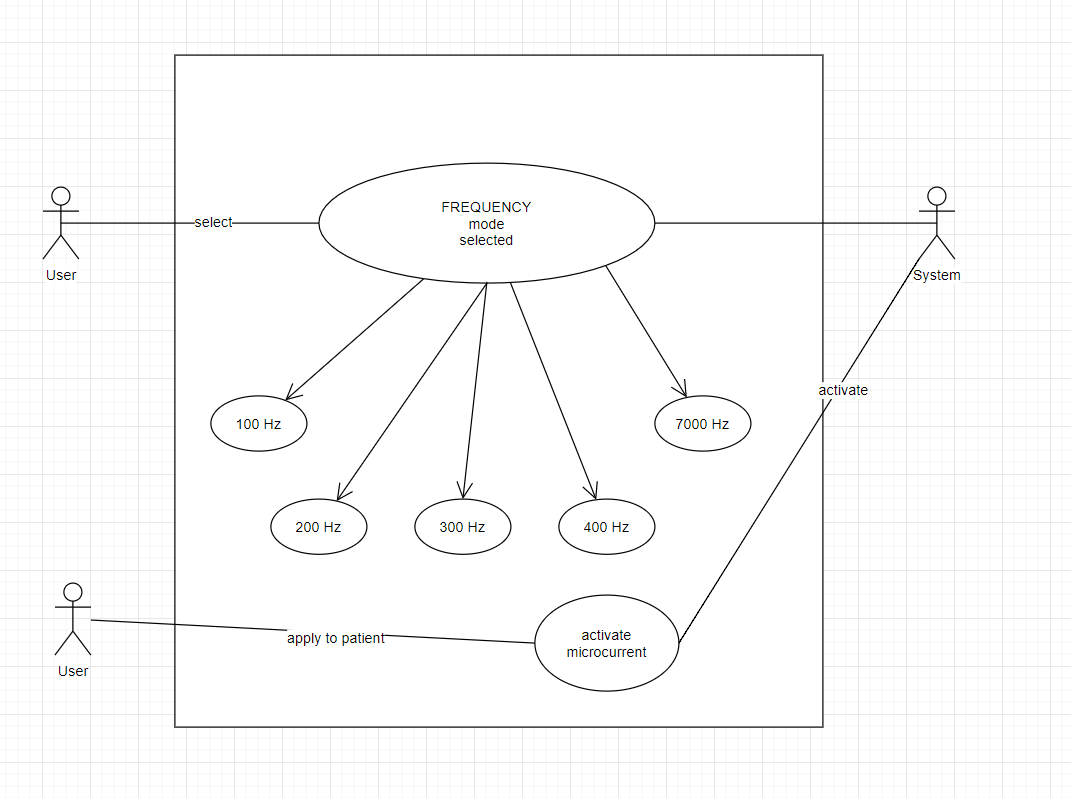


1. Frequency Treatment

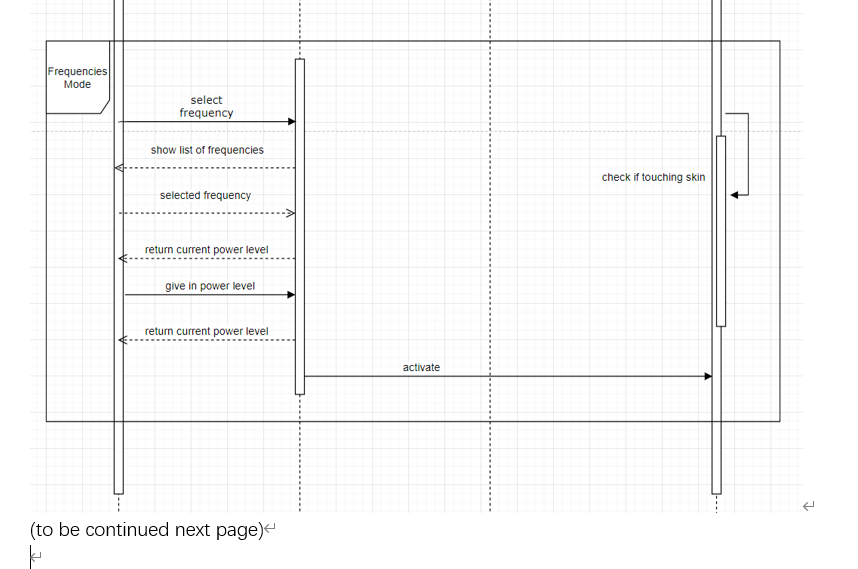
This function is very similar to Program Treatment, except that when offering selection, it’s not giving specific body part or symptom, but giving direct frequency selection.



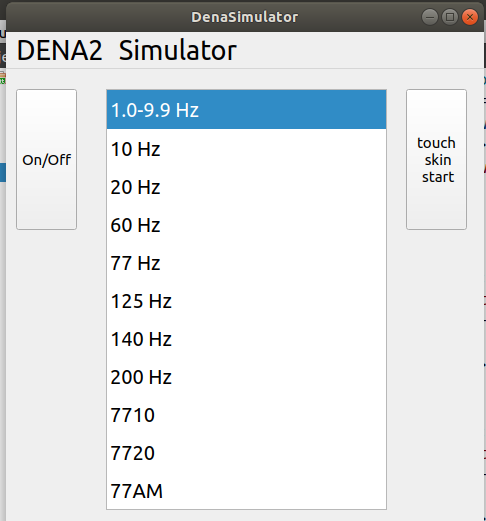
Such is to give users a more versatile way to treat themselves with different frequencies.



The steps are very similar to those of Program Treatment



1. At main menu, select freqency mode, in code, this is handled by handle\_main\_page\_selection(int) function, which is called when confirm button is clicked at main menu.
2. When in program selection menu, a list of frequencies will be available for selection. User can pick one according to his/her needs.



1. Upon selection, the power adjustment page will show up, and clicking left or right button will allow user to adjust the power level.
2. Clicking touch skin start button, and the countdown will show up as we’ve seen in Program Treatment.
3. MED Treatment

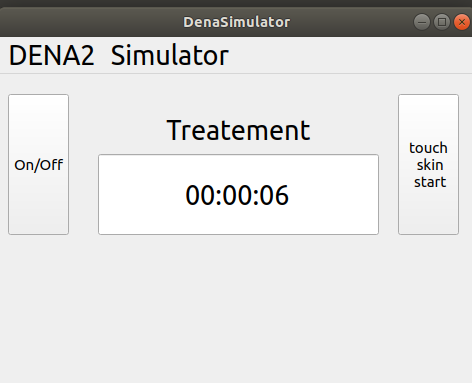
This one is much simpler in terms of user interface.

MED mode allows machine to automatically determine frequency according to on-board sensor feedback. So user only has to choose power level and apply the machine on skin to start treatment.

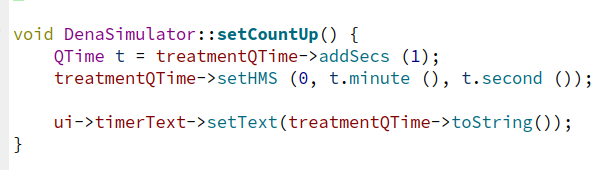
Steps are :

1. Choose MED on Main Menu.
2. Select Power level by clicking left or right for adjustment
3. Apply touch skin start, to start the counting.

The difference here is that the counting will be applied incrementally rather than counting down.



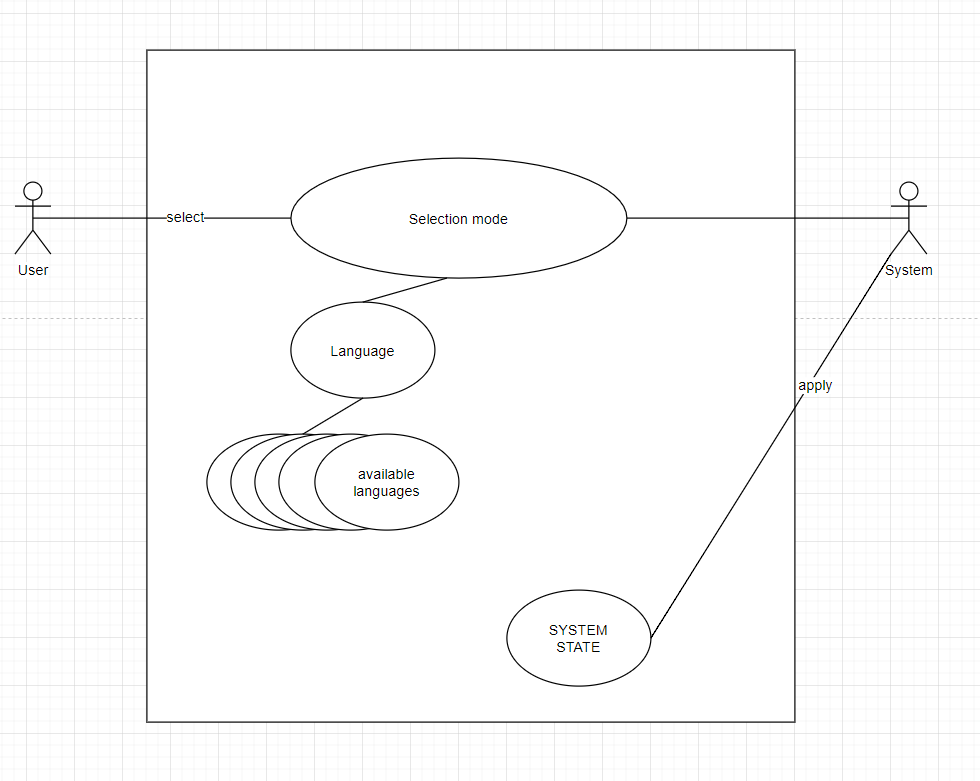
So in code, we cannot attach the “countdown()” to the QTimer this time. Instead, we must use

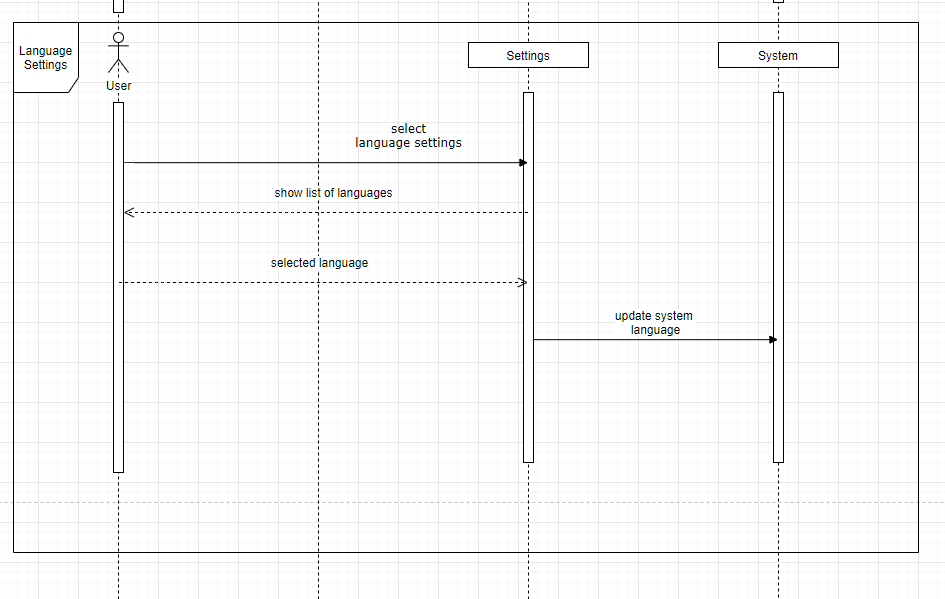


To add positive number to the QTime to generate an increasing counter.

1. Language setting

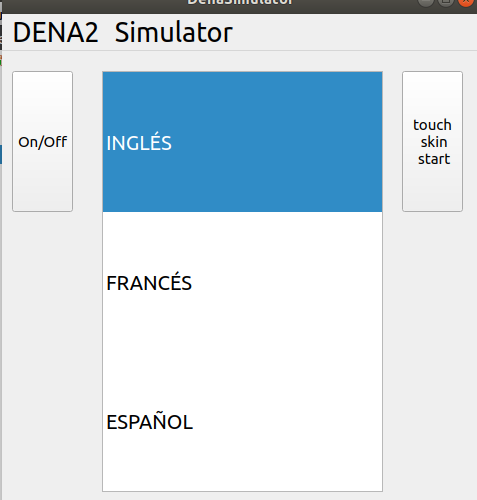
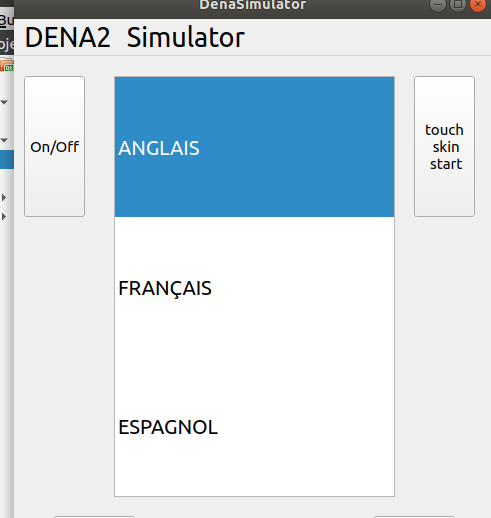
Denas is sold worldwide so it’s necessary to provide multi-language support to the machine. In our simulator, we are simulating language changes between English, French and Spanish to demonstrate that we can extend the range of languages and sell the machine outside of North America.





1. User will browse through the Main Menu and go to “settings”
2. User will select “Language” option in Settings Menu page
3. User will then select one of the language that’s available in this simulator.

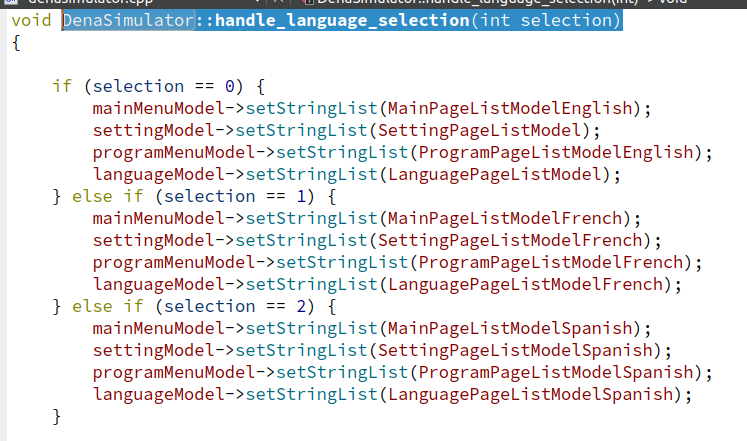
Upon selection, the system will update the entire machine and make all menu in the indicated language. The following screenshot has shown that the system language is changed accordingly (French and Spanish)



In Code, this is done by

DenaSimulator::handle\_language\_selection(int selection)

In which we change the string list for the menu model so that all languages are changed

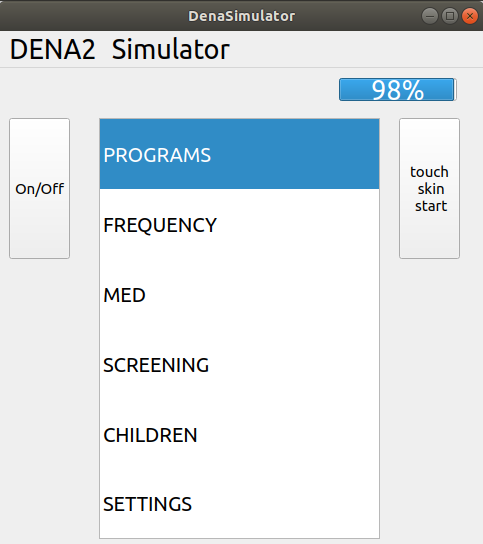


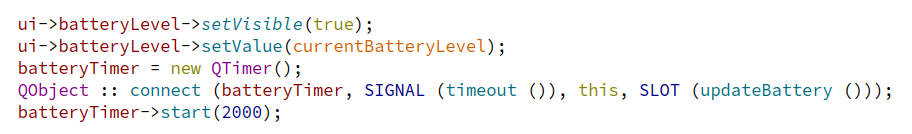
1. Confirm / Return buttons

These two buttons allow user to navigate through different menus and go to different menu pages. In the code, the system is managed in a state-machine fashion, i.e. “currenPage” keeps track which page (state) that the user is currently at. Upon clicking “confirm” or “return”, through handling codes (please look into the source code for more details), the system will be able to navigate to pages accordingly.

6. Battery Level

The simulator also simulates a battery state. A timer is associated with the battery, when the power is on, 1% of battery will be drained every 3 seconds. Once the power is out, machine will be automatically shutdown. WE ASSUME THAT WHEN TURNED ON THE MACHINE IS FULLY CHARGED. When power button is clicked, the battery level will be back to 100.





## Conclusion:

Our Denas simulator has successfully simulated the functionalities of what we can see in the demo video. Due to the hardware limitation, we can only use “touch skin start” button to simulate the actual sensor. We’ve also demonstrated the system language change in the simulator.