Project Report on

Universal Control Panel

Submitted in partial fulfilment of the requirements
of Project Based Learning of

BACHELOR OF ENGINEERING

in

COMPUTER ENGINEERING

(A.Y. 2021-2022)

by

Raj Mazgaonkar (19)
Tejas Nair (33)
Kartik Nambiar (34)
Taranveer Nara (35)

Under the Guidance of

??????

Designation

???????

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

Raj Mazgaonkar (Roll No.19)

(Signature)

Tejas Nair (Roll No.33)

(Signature)

Kartik Nambiar (Roll No.34)

(Signature)

Taranveer Nanra (Roll No.35)

Date:

Abstract

We all know the number of deaths caused due to human and machine error in industrial workplaces, factories, due to radiations and chemical leaks is a lot. So, our application's sole purpose is to reduce these hazardous problems as much as possible by suggesting preventive measures before any major issues occur. We mapped out the problems we could possibly see, the challenges we could face, and the ways in which we could overcome them. With this knowledge, the control panel gives us all the necessary information that we need in order to overcome any sort of danger or even better, prevent any sort of disaster, that may have unfolded without the help of the control panel.

List of Contents

Introduction

Problem Definition:-

The first question that arises is, whats the need for this project/panel, we all know the number of deaths caused due to human and machine error in industrial workplaces, factories, etc. Due to radiations and chemical leaks is a lot. So our application's sole purpose is to reduce these hazardous problems as much as possible by suggesting preventive measures before any major issues occur. We mapped out the problems we could possibly see, the challenges we could face, and the ways in which we could overcome them.

Aim:-

Our application's sole purpose is to reduce these hazardous problems as much as possible by suggesting preventive measures before any major issues occur. We mapped out the problems we could possibly see, the challenges we could face, and the ways in which we could overcome them.

Scope:-

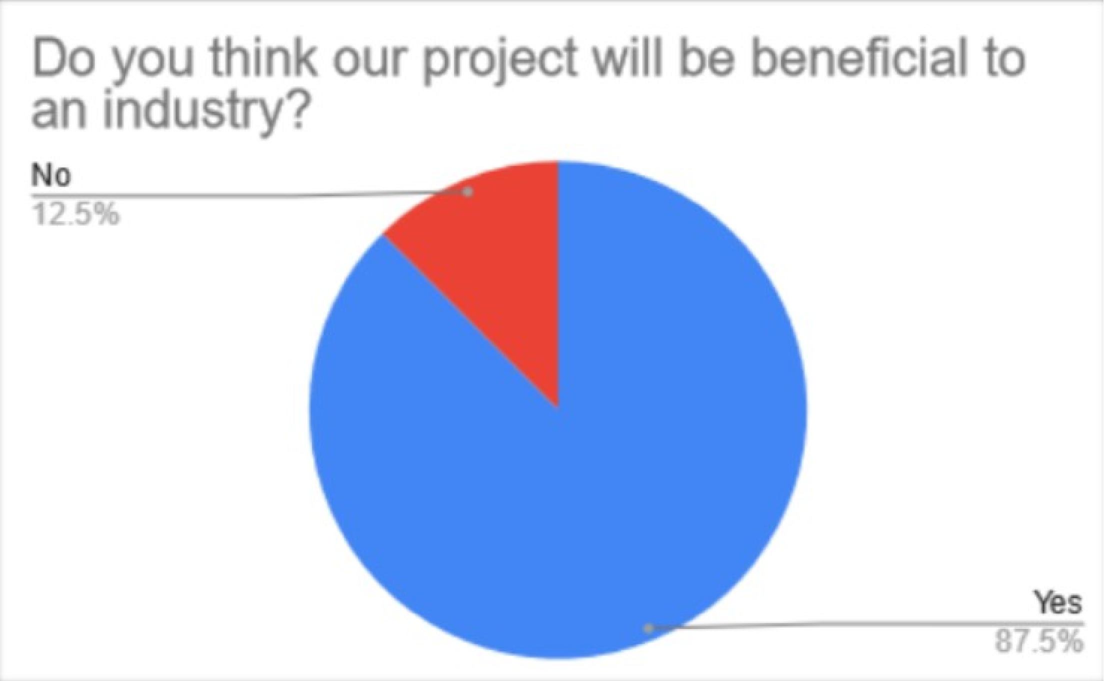
Our project aims to be used in the industrial sector, where most of the accidents related to chemicals, radiation, and anything mechanical happens. For example, nuclear power plants, biofuel plants, textile industries, etc.

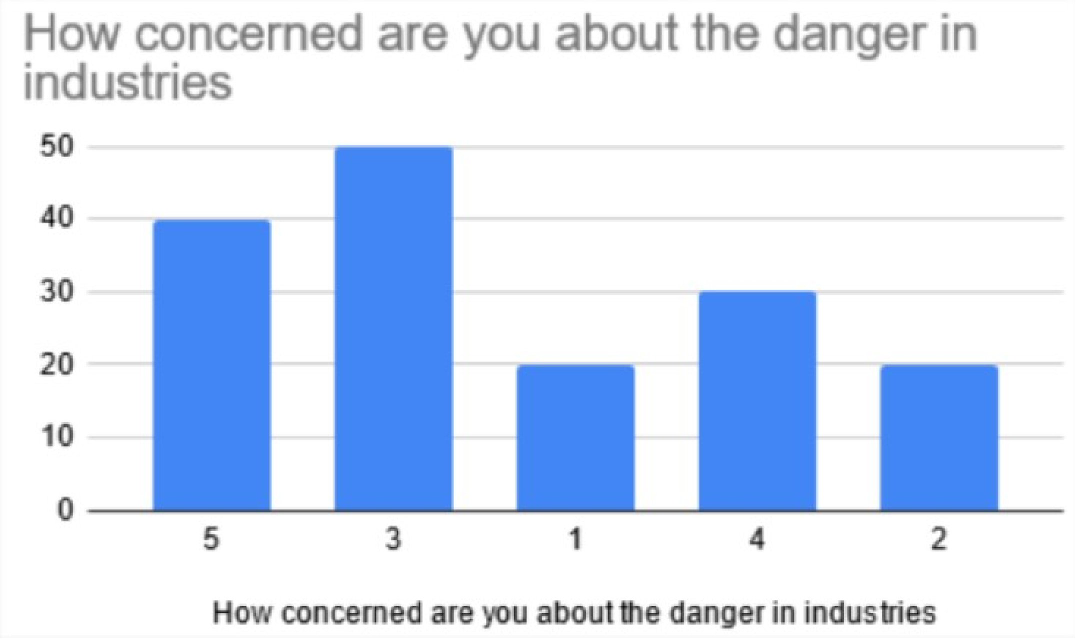
Motivation:-

The reason we want to do this project is because we recognize that deaths in industries are something that can be completely avoided, and making something that helps people, in of itself, is something that feels like an incredible feat.

Review of Literature

Literature Review

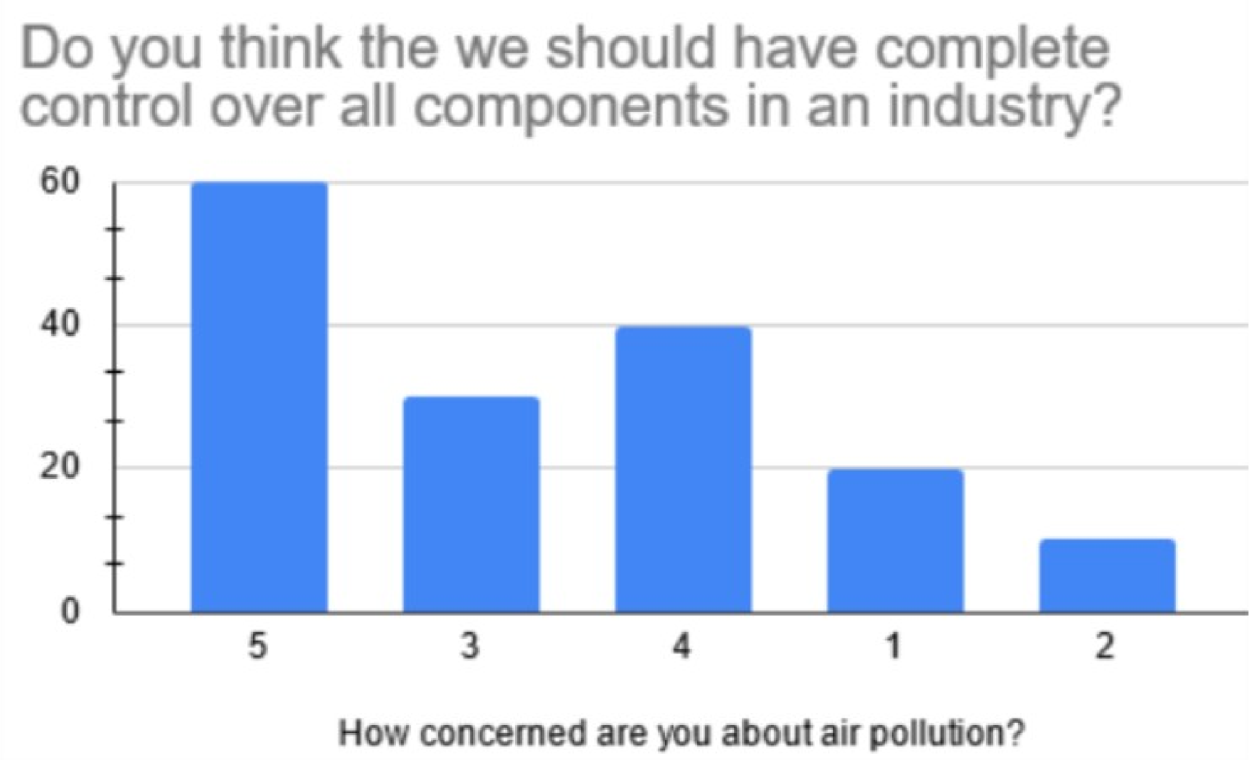


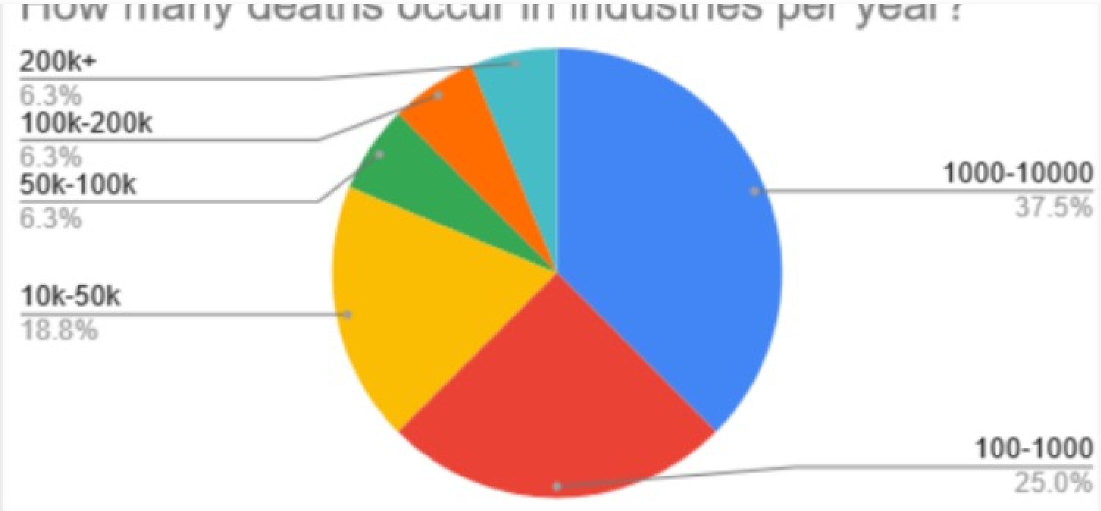


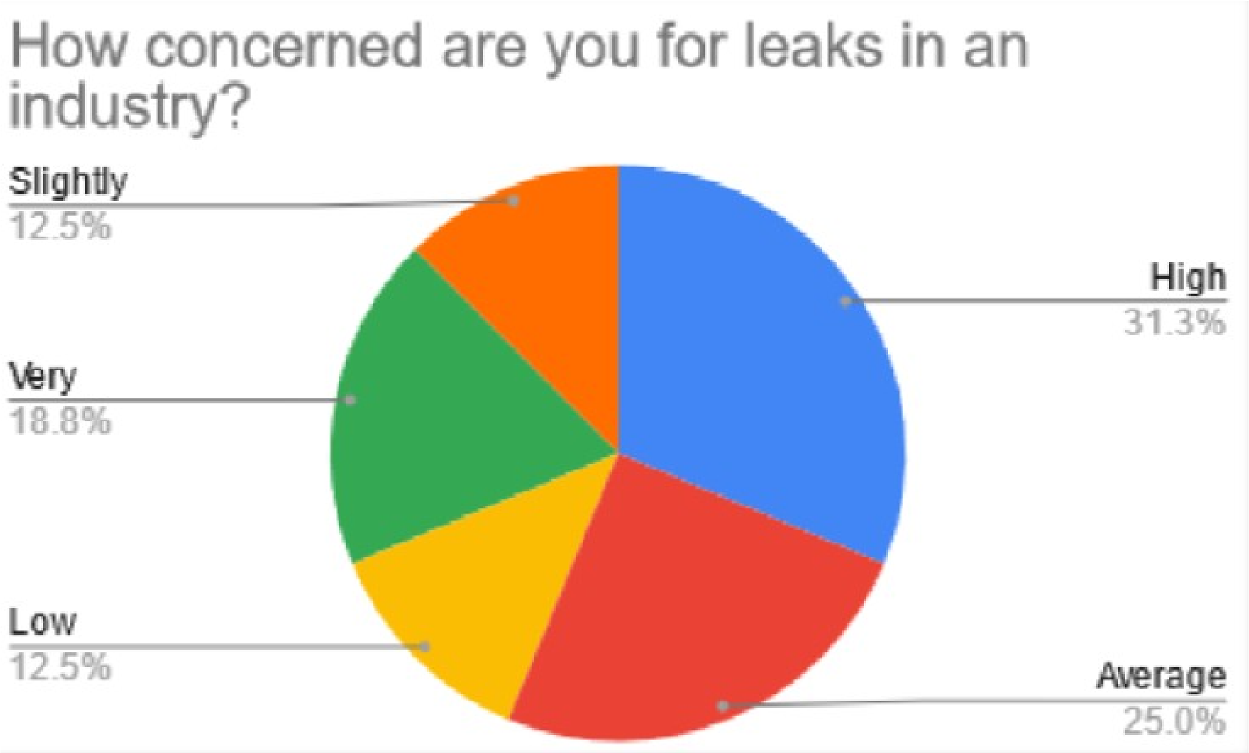
Gap Identified:-

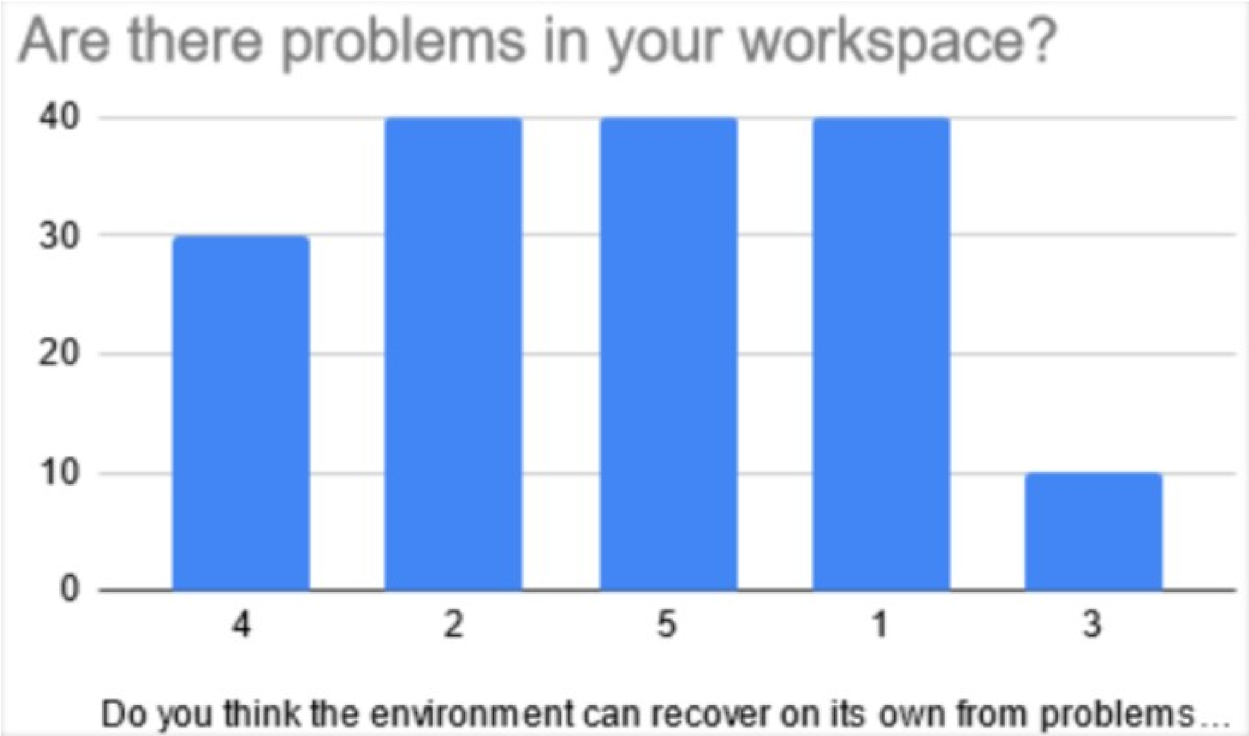
We created a google form to get the responses of people for this project. With this, we got the responses of:-

Almost everyone considered that this project was a great idea. Most people didn't know about how many deaths occur in industries. Most people thought that industry deaths are extremely concerning. Many people did care about complete control over everything, like machinery and tools in an industry. Many people were relatively concerned about leaks of pollutants. Many people say that they don't see many problems in their workspace.






How concerned are you for leaks in an industry?
Slightly
12.5%
High
31.3%
Very
18.8%
Low
12.5%
Average
25.0%


Are there problems in your workspace?
40
30
20
10
0
4
2
5
1
3
Do you think the environment can recover on its own from problems…

Related Theory:-

Qt:-

Qt (pronounced "cute") is a widget toolkit for creating graphical user interfaces as well as cross-platform applications that run on various software and hardware platforms such as Linux, Windows, macOS, Android or embedded systems with little or no change in the underlying codebase while still being a native application with native capabilities and speed.

Qt is currently being developed by The Qt Company, a publicly listed company, and the Qt Project under open-source governance, involving individual developers and organizations working to advance Qt. Qt is available under both commercial licenses and open-source GPL 2.0, GPL 3.0, and LGPL 3.0 licenses.

Qt is used for developing graphical user interfaces (GUIs) and multi-platform applications that run on all major desktop platforms and most mobile or embedded platforms. Most GUI programs created with Qt have a native-looking interface, in which case Qt is classified as a widget toolkit. Non-GUI programs can also be developed, such as command-line tools and consoles for servers. An example of such a non-GUI program using Qt is the Cutelyst web framework.

Qt supports various compilers, including the GCC C++ compiler, the Visual Studio suite, PHP via an extension for PHP5, and has extensive internationalization support. Qt also provides Qt Quick, that includes a declarative scripting language called QML that allows using JavaScript to provide the logic. With Qt Quick, rapid application development for mobile devices became possible, while logic can still be written with native code as well to achieve the best possible performance.

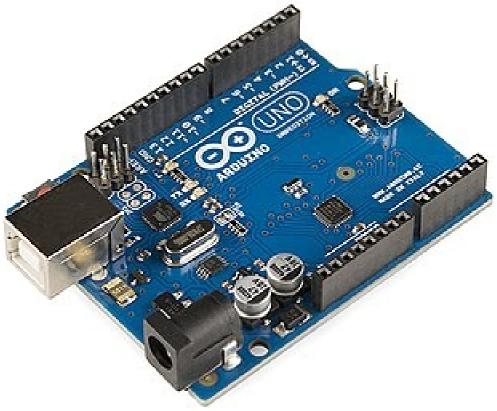
Other features include SQL database access, XML parsing, JSON parsing, thread management and network support.

Arduino:-

Arduino is an open-source hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. Its hardware products are licensed under a CC BY-SA license, while software is licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially from the official website or through authorized distributors.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. The microcontrollers can be programmed using the C and C++ programming languages, using a standard API which is also known as the Arduino language, inspired by the Processing language and used with a modified version of the Processing IDE. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) and a command line tool developed in Go.

The Arduino project began in 2005 as a tool for students at the Interaction Design Institute Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors.



Simulation:-

For the simulation of this project, to emulate the environments that would originally be tough to check in real life scenarios, we will use simulators to check if the control panel can handle those changes and combinations of scenarios.

We had to do extensive research on both of these applications and hardware, so that we could get a clear idea of how they work, and how they would react to certain scenarios. We finally, also had to do quantitative research on how changes should affect the control panel realistically, so that changes in the simulated devices are as close to real life scenarios as possible.

Design Methodology:-

Signals and Slots:-

This is the method in which we listen for and send signals, especially from one window to another. The most important part about having two windows is that we have the capability of communication between the two. Once we have established this, the rest of the project becomes quite easy, presuming that we already have the necessary data.

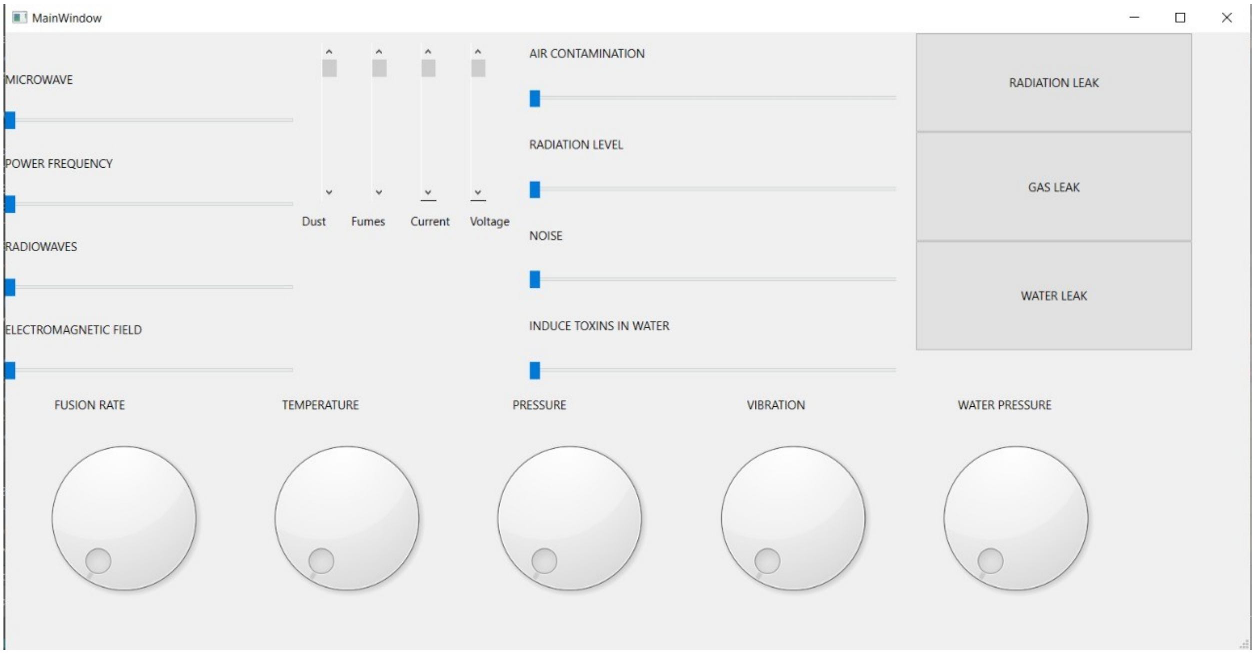
In Qt, we have an alternative to the callback technique: We use signals and slots. A signal is emitted when a particular event occurs. Qt's widgets have many predefined signals, but we can always subclass widgets to add our own signals to them. A slot is a function that is called in response to a particular signal. Qt's widgets have many pre-defined slots, but it is common practice to subclass widgets and add your own slots so that you can handle the signals that you are interested in.

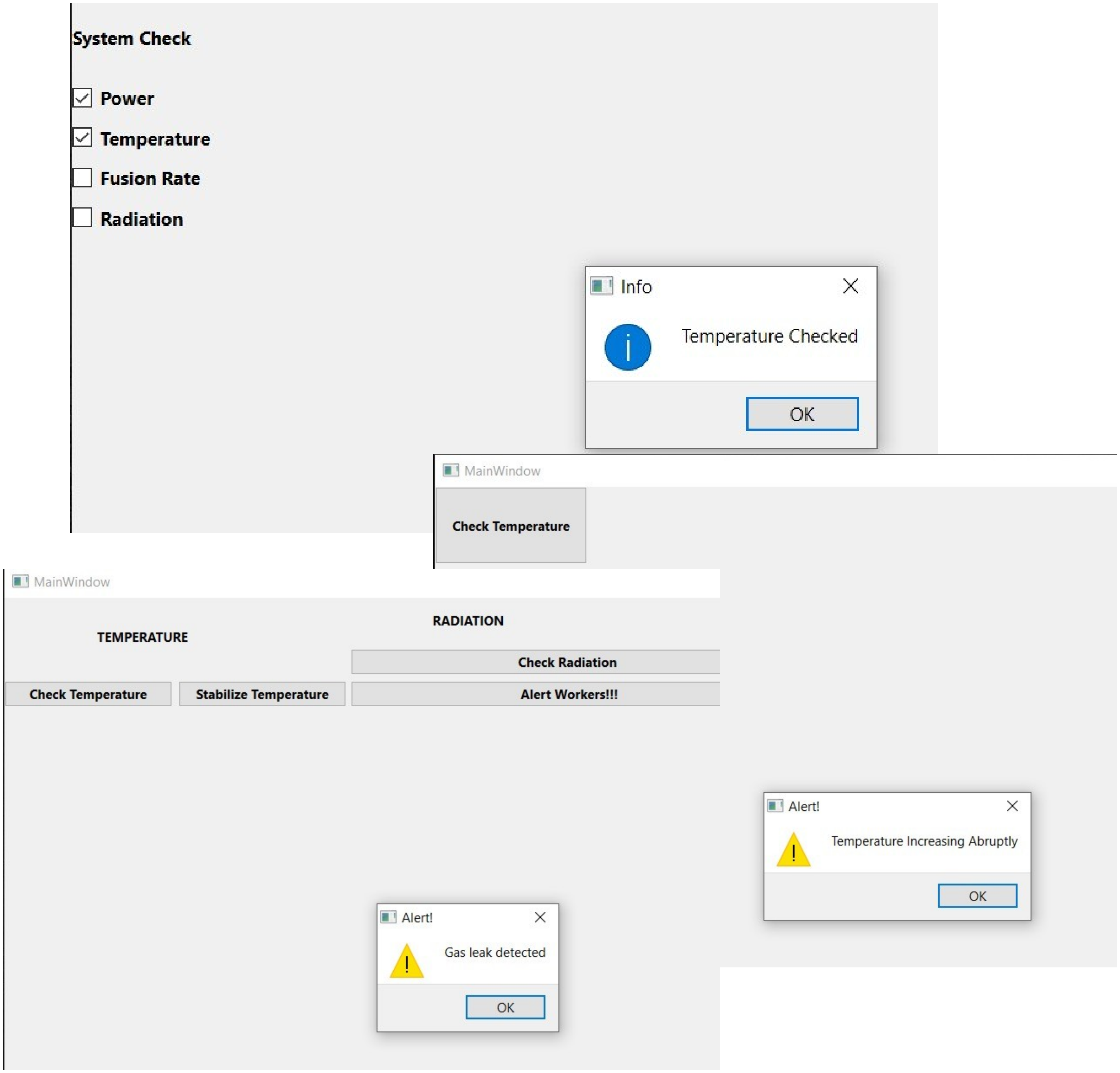
The signals and slots mechanism is type safe: The signature of a signal must match the signature of the receiving slot. (In fact a slot may have a shorter signature than the signal it receives because it can ignore extra arguments.) Since the signatures are compatible, the compiler can help us detect type mismatches when using the function pointer-based syntax. The string-based SIGNAL and SLOT syntax will detect type mismatches at runtime. Signals and slots are loosely coupled: A class which emits a signal neither knows nor cares which slots receive the signal. Qt's signals and slots mechanism ensures that if you connect a signal to a slot, the slot will be called with the signal's parameters at the right time. Signals and slots can take any number of arguments of any type. They are completely type safe.

QWidgets:-

We use this for making the second window for the simulation. This is the window that allows us to control the control panel, and check if it displays and handles the values correctly. The widget is the atom of the user interface: it receives mouse, keyboard and other events from the window system, and paints a representation of itself on the screen. Every widget is rectangular, and they are sorted in a Z-order. A widget is clipped by its parent and by the widgets in front of it.

A widget that is not embedded in a parent widget is called a window. Usually, windows have a frame and a title bar, although it is also possible to create windows without such decoration using suitable window flags). In Qt, QMainWindow and the various subclasses of QDialog are the most common window types.





System Check

☑ Power

☑ Temperature

□ Fusion Rate

□ Radiation

Info

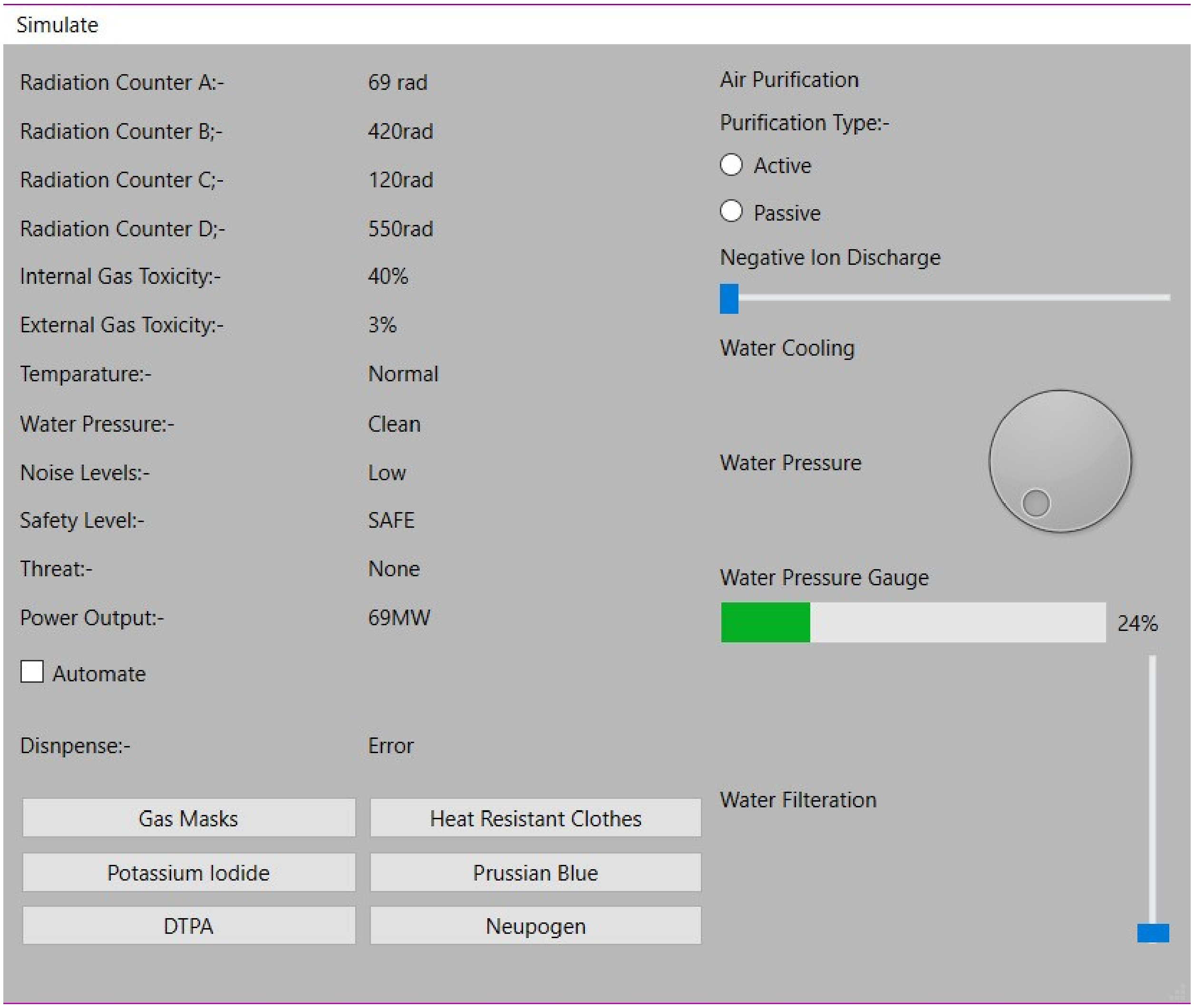
Temperature Checked

Alert!

Temperature Increasing Abruptly

Alert!

Gas leak detected



Simulate

Radiation Counter A:-	69 rad
Radiation Counter B;-	420rad
Radiation Counter C;-	120rad
Radiation Counter D;-	550rad
Internal Gas Toxicity:-	40%
External Gas Toxicity:-	3%
Temparature:-	Normal
Water Pressure:-	Clean
Noise Levels:-	Low
Safety Level:-	SAFE
Threat:-	None
Power Output:-	69MW

□ Automate

Disnpense:- Error

Purification Type:-

○ Active

○ Passive

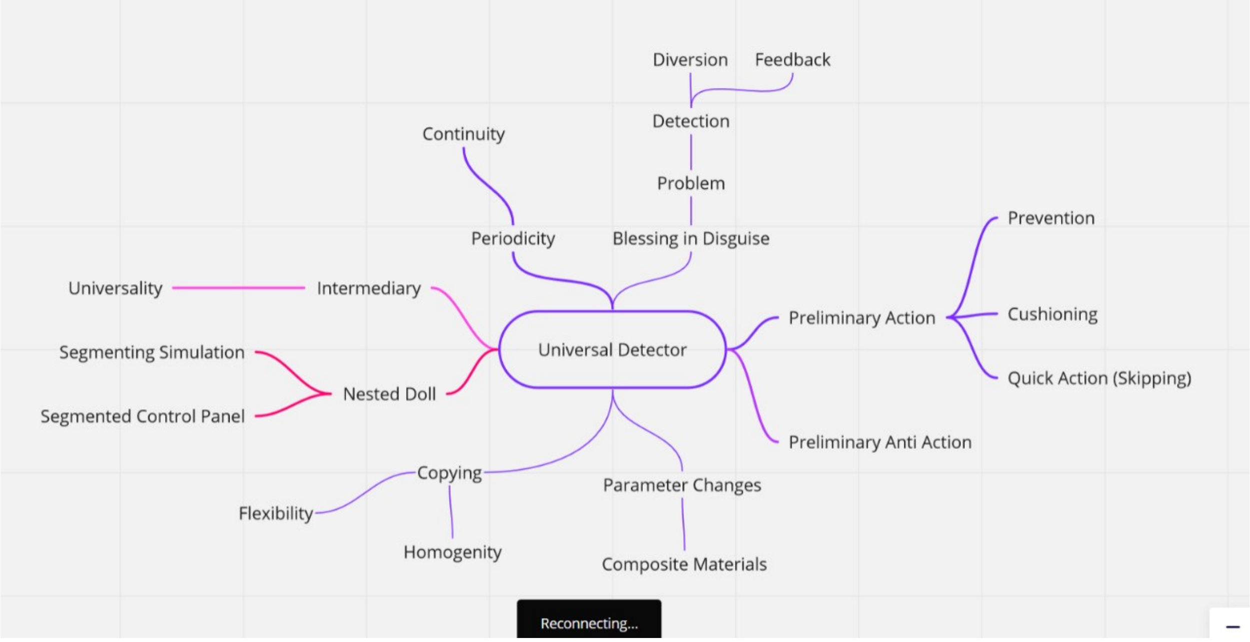
Water Pressure Gauge 24%

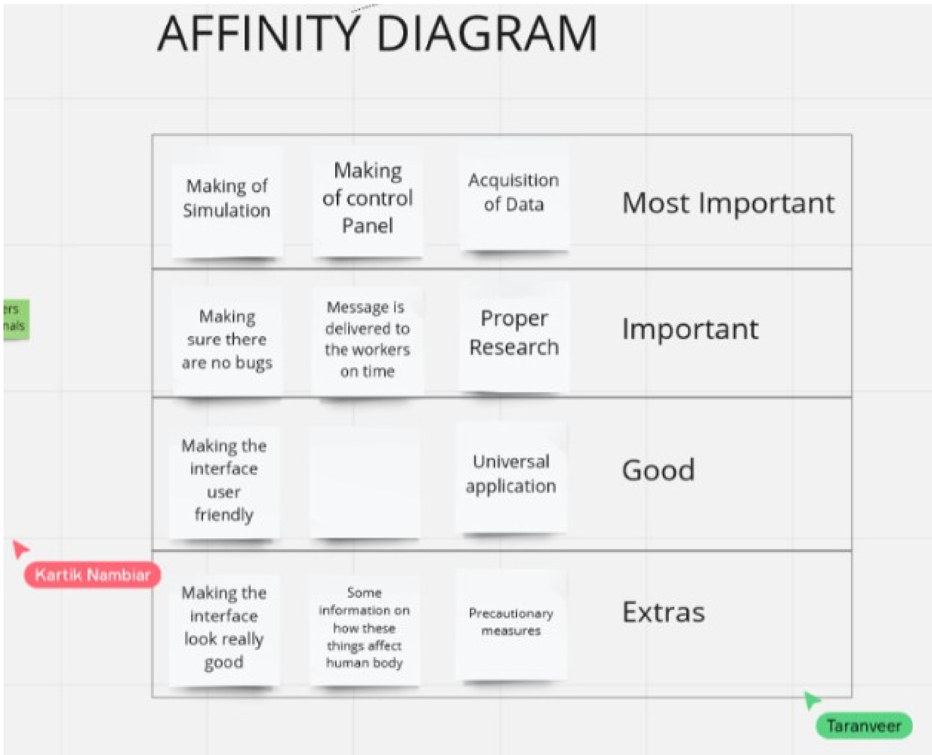
Simulate:-

This brings us to the topic of simulation. Data collection is the most important part of this, since getting the correct data is paramount to having the simulation behave like it should. The correct radiation values for the ultrasonic sound detector, and turbidity detectors, to get the purity of water, the pressure gauges to determine the flow of water, the thermocouples for the temperature, all of them must be accurately simulated for the control panel to be worth something.

Result and Discussion:-

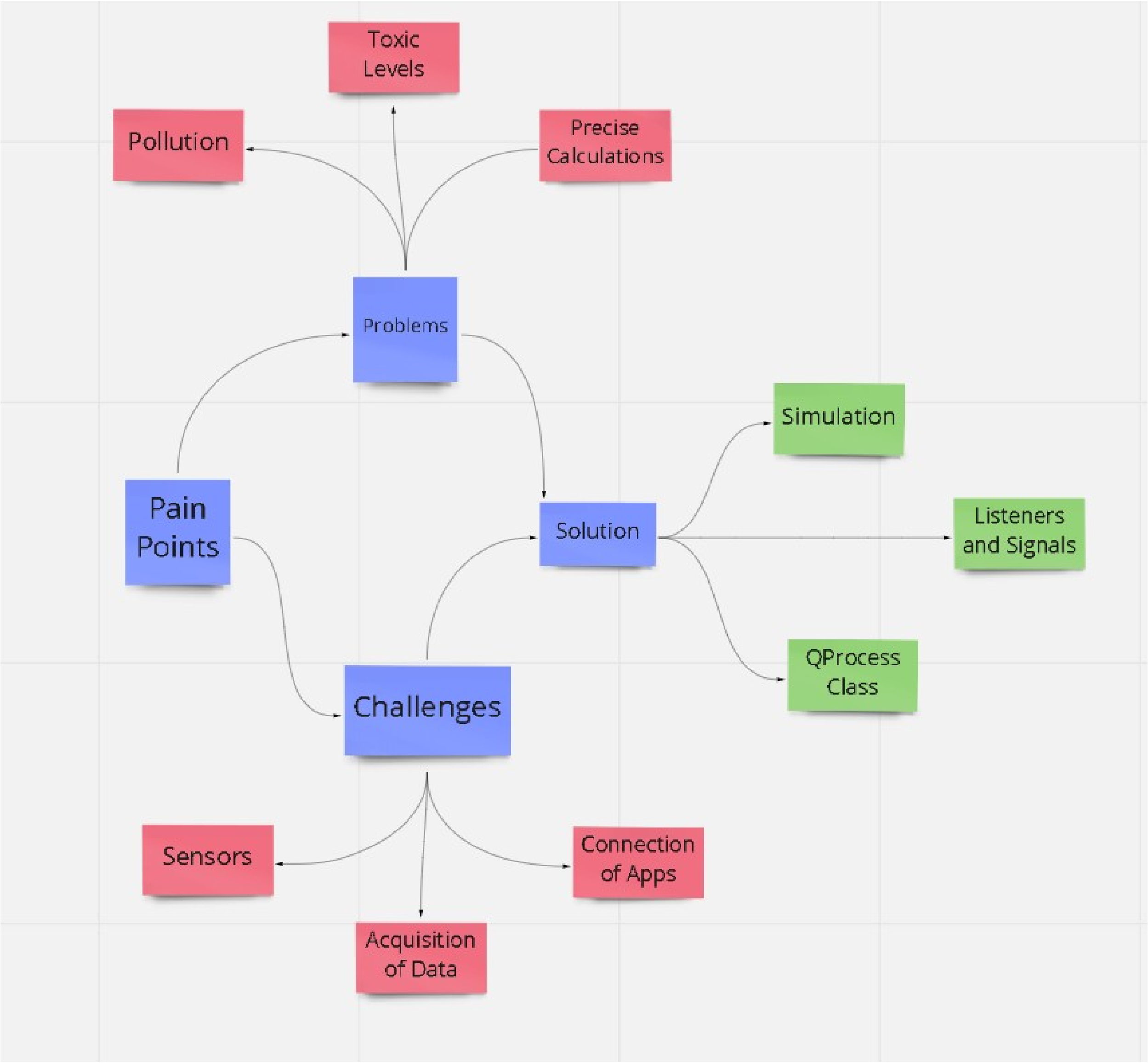
Implementation:-





AFFINITY DIAGRAM

Making of Simulation	Making of control Panel	Acquisition of Data	Most Important
Making sure there are no bugs	Message is delivered to the workers on time	Proper Research	Important
Making the interface user friendly		Universal application	Good
Making the interface look really good	Some information on how these things affect human body	Precautionary measures	Extras



Prediction of Results:-

Since all the data that we collected already tells us what the project needs to do, we already know what each configuration of inputs would result in our device state being. It usually would be a 1 to 1 model in real life but due to common error, we can say that its as close to reality as we can bring it.

Testing & Outcome:-

Screenshots of the project were provided above, and proves that this project is fully functional, and operational. It has stood every single test we throw at it and every corner or edge case we could give it, proving itself to be quite reliable in a real life scenario. We havent yet tried Arduino for our control panel yet but it is something that we have set our sights on.

Conclusion & Future Scope

This project is nowhere near complete, and we have made a path for this project, as stated before in the implementation, that we need to add multiple features to make this control panel truly ready to use in a work environment. As is, this control panel works great in the simulations, hence proves that it can work well in the industries, but of course, it still has a long way to go. We will continue working on this project, until we completely see it through.

References:-

https://en.wikipedia.org/wiki/

https://doc.qt.io/qt-5/

ACKNOWLEDGEMENT

With deep sense of gratitude we would like to thank all the people who have lit our path with their kind guidance. We are very grateful to these intellectuals who did their best to help during our project.

It is our proud privilege to express deep sense of gratitude to our Project Guide Prof.name for his continuous guidance and support throughout our project. It would never have been possible for us to complete this project successfully without his guidance and support.

We remain indebted to our H.O.D Dr. Rekha Sharma Computer Department, Principal Dr. B.K. Mishra for their comments and kind permission to complete this project. We would also like to thank them for their timely suggestions and valuable guidance.

And lastly we would like to thank our friends and the people who are directly or indirectly related to our project.