AUTOMATION OF HOSPITAL MANAGEMENT SYSTEM

Submitted by: DEEKSHA MAURYA/ IMPANA H C / LEYA SUSSAN / MANISHA SAH/ PRIYANKA SONI/ SNEHA G R/ VISHAL SINHA(Team Leader)
Batch 6

Submitted on: 24/10/2020

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to SIEMENS for providing us an opportunity to work with this project.

We would also like to extend our gratitude to our mentor “Mr. DEEPAK SINGH” for his able guidance and support in completing this project.

Also, we would like to express our acknowledgement to GROWTH CENTRE team for constantly keeping our morale high during the project.

CONTENTS

1. Problem Statement
2. Our Solution/ Introduction to project
3. Process flowchart of automated Hospital
4. Components Required
5. Detailed Working of every subparts
6. Estimated cost of project
7. Who will be benefitted (market)
8. Challenges faced during working with this project
    a. What went right
    b. What went wrong
9. Faults/ Mistakes Committed
10. Alternate Possibilities/ Improvement Options
11. References and Studies done to complete the project
12. Summary and Conclusion
13. Project Documentation
14. Work flow of team members

PROBLEM STATEMENT

Due to increase devastation caused by covid-19 worldwide, there is an urgent need to come up with a solution which can reduce contamination in public places. And one such place which is hit worst are Hospitals . Due to improper medication facilities, especially in small scale Hospitals there are more probability of getting infected if we are in vicinity of any of their wards.

Also, overcrowding is another challenge which these Hospitals are facing during this pandemic and there is really a need of a proper system which can eradicate the cause of contamination and can insure less crowdedness.

OUR SOLUTION / INTRODUCTION TO PROJECT

To deal with this problem, we thought of making an automated model of Hospital which will insure less contamination and will also avoid the hospital premises from getting crowded.In order to achieve this we have divided the whole autonomous process into following subparts, the brief introduction of each is as given below:

1. **AUTOMATED MONITORING OF HEAD COUNT**: As soon as the patient/staff enters the hospital premises there is a system designed using which would detect the person and increases the head count by 1. Similarly, if anybody leaves then head count would decrease by 1. And as soon as the head count exceeds a certain safe limit there is a security alarm buzzer which would ring and alert the guards.
2. **SANITIZATION**: Once the person enters inside the hospital campus, he has to undergo an automated sanitization process .

3. OFFLINE REGISTRATION/ READING PATIENT's DATA FROM OCR: Online registration is not done through any pen pare mode, the patient has to provide his/her documents or id which would be scanned through OCR and will be send to hospital's website to keep patient's record.

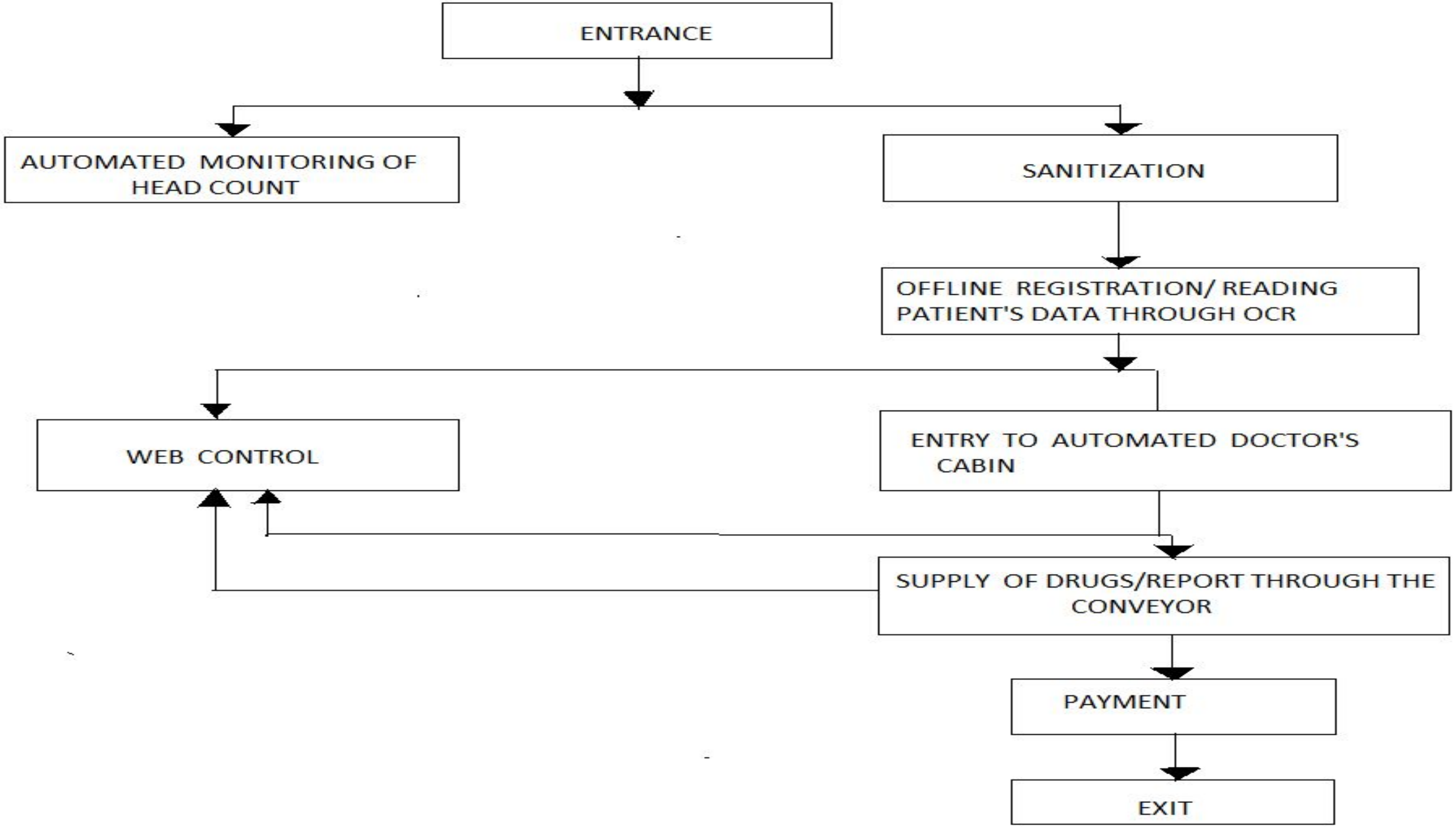
4.WEBSITE DESIGNING: Hospital's website is also designed to keep a track of every patient's data, including his personal details, assigned doctor, report generated by doctor and list of drugs/medicines he has been prescribed. However, there is also an option of online registration through the web in which patient can book his desired slots with the concerned doctor.

5.AUTOMATED DOCTOR'S CABIN: Once the patient is allowed access to doctor's cabin, we need to insure that the doors are not contaminated anyhow. For this, we will install automated door which would sense the person and automatically closes/open the door. Further, we have divided the cabin by a thin glass film and the patient and doctor will be either side of it. Also, we have installed a system which would take a health report of patient including his body temperature and heart rate and would display the output to doctor via a screen. Also, the cabin is sterilized using UVGI sterilizer to kill harmful germs.

6.SUPPLY OF DRUGS/REPORTS THROUGH CONVEYOR: Reports and medicines will not be handled to patient manually. Rather, there is a conveyor installed which would pass through 3 stations, namely: doctor's subordinate, pharmacy and basement.Once the report is prepared by doctor then that same report will be send to subordinate and pharmacy to gather patient's medicines/reports in a box which the patient has to collect directly from basement. So, this would insure no crowdedness in pharmacy area and outside subordinate's cabin.

6. PAYMENT AND EXIT: After collecting his belongings the patient can move out of the premises after duly payment of fees. Both online and offline mode of payments is available.

PROCESS FLOWCHART OF AUTOMATED HOSPITAL



ENTRANCE

AUTOMATED MONITORING OF HEAD COUNT

SANITIZATION

OFFLINE REGISTRATION/ READING PATIENT'S DATA THROUGH OCR

WEB CONTROL

ENTRY TO AUTOMATED DOCTOR'S CABIN

SUPPLY OF DRUGS/REPORT THROUGH THE CONVEYOR

PAYMENT

EXIT

COMPONENTS REQUIRED

1.	PIR SENSOR	:	12v
2.	RELAY	:	7805
3.	IC	:	BC547
4.	TRANSISTOR	:	10uf /16V ,100uf/16V
5.	CAPACITOR	:	IN4007
6.	DIODE	:	1K ohm
7.	RESISTOR	:	
8.	PIN SCREW CONNECTORS	:	
9.	HEAT SINK	:	
10.	CABINET		
11.	WATER PUMP		
12.	PNEUMATIC	:	T,L
13.	SPRAY NOZZLE		
14.	ADAPTOR	:	12V,3A
15.	CABLE CLIP	:	6mm

16. BUFFER (DAC)
17. ARDUINO SENSOR
18. LED INDICATOR
19. INVERTER
20. COUNTER
21. DC MOTOR
22. ACTIVE IR SENSOR
23. MOTOR DRIVER : L2930
24. TEMP SENSOR : MLX90614
25. PULSE OXIMETER : MAX30102
26. BUZZER
27. OLED DISPLAY
28. DOTE BEARDBOARD
29. MICROCONTROLLER : 20 pin AT89C2051

DETAILED WORKING OF EVERY SUBPARTS

The following module will include detailed working of each subpart in the given sequence:

1. AUTOMATED MONITORING OF HEAD COUNT
2. SANITIZATION
3. WEB DEVELOPMENT
4. OFFLINE REGISTRATION/READING PATIENT’S DATA THROUGH OCR
5. AUTOMATED DOCTOR’S CABIN
6. SUPPLY OF DRUGS/REPORTS THROUGH CONVEYOR
7. PAYMENT AND EXIT

AUTOMATED MONITORING OF HEAD COUNT IN HOSPITAL PREMISES

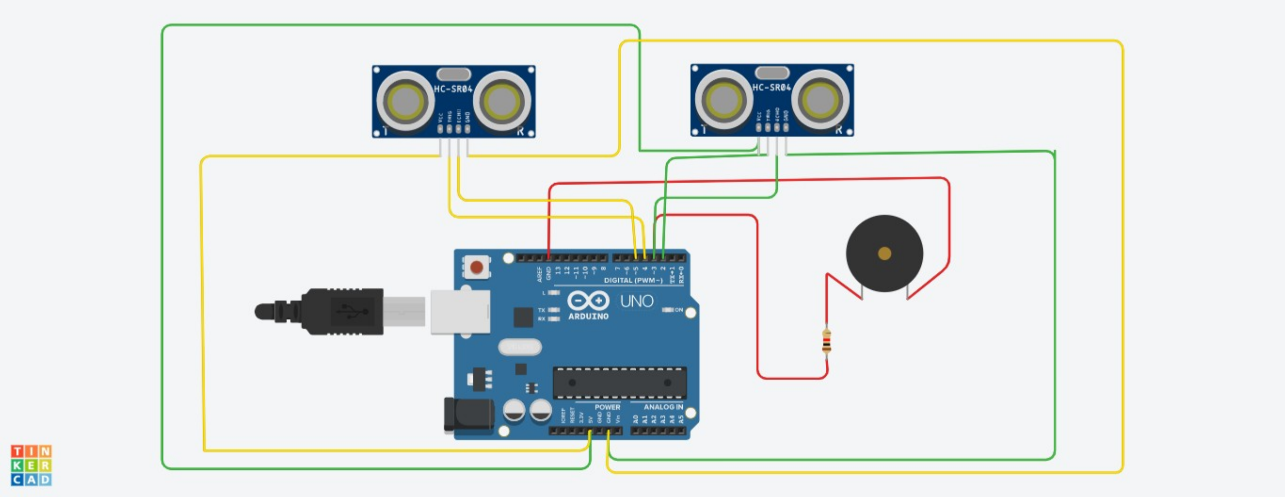
COMPONENTS USED:

1. Ultrasonic sensor(HC-SR04) – 2 units
2. Security alarm buzzer
3. Arduino mini

CONSTRUCTION:

1. Connect echo and trigger pin of 1st HC-SR04 to D3 and D2 pins of arduino respectively.
2. Similarly, connect echo and trigger pins of 2nd HC-SR04 to D5 and D4 pins respectively.
3. Connect their Gnd and VCC terminals with Gnd and 5V pins of arduino respectively.
4. Connect security alarm buzzer with D3 and Gnd pins .

CIRCUIT DIAGRAM:



The following circuit has been drawn from TinkerCad software.

(However, due to unavailability of arduino mini in components list, I have used here arduino uno.) The circuit has 2 HC-sr04 sensors and 1 buzzer as mentioned above.

WORKING :

1. Arduino Code:

```
int head_count,dist1,dist2,prev_sensor;
unsigned long t1,t2,wait_1,wait_2;
float select_dist;
unsigned long sense(int trig,int echo){
      digitalWrite(trig,0);
      delay(2);
      digitalWrite(trig,1);
      delayMicroseconds(10);
      digitalWrite(trig,0);
      return pulseIn(echo,1);
}
float distance(unsigned long t){
      return .5*343*t/10000;//Speed of sound taken = 343 m/s
```

```
}
int isProperDist(int trig,int echo){
      float d=distance(sense(trig,echo));
      return (d>0&&d<=select_dist);
}
void setup(){
      pinMode(2,OUTPUT);//Sensor 1 Trigger Pin
      pinMode(3,INPUT);//Sensor 1 Echo Pin
      pinMode(4,OUTPUT);//Sensor 2 Trigger Pin
      pinMode(5,INPUT);//Sensor 2 Echo Pin
      pinMode(7,OUTPUT);//LED Pin
      digitalWrite(7,0);
      wait_1=1000;//Time to wait for ENTRY/EXIT
      wait_2=500;//TIme to wait after ENTRY/EXIT
      head_count=0;
      select_dist=100;//in cm
      prev_sensor=0;
}
void loop(){
      if(prev_sensor==0){
            dist1=isProperDist(2,3);//Ultrasonic Sensor 1
            dist2=0;
      }
      else if(prev_sensor==1){
            dist2=isProperDist(4,5);//Ultrasonic Sensor 2
            dist1=isProperDist(2,3);//Ultrasonic Sensor 1
      }
      else{
            dist1=isProperDist(2,3);//Ultrasonic Sensor 1
            dist2=isProperDist(4,5);//Ultrasonic Sensor 2
      }
outer1:
      while(dist1){
            dist1=0;
            prev_sensor=1;
            if(head_count==0)
                  digitalWrite(7,1);
            while(isProperDist(2,3));//Ultrasonic Sensor 1
            unsigned long start=millis(),elapsed=0;
            int pass_sensor2=0;
            while((elapsed=millis()-start)<wait_1){
                  if(isProperDist(4,5)){//Ultrasonic Sensor 2
                        pass_sensor2=1;
                        break;
                  }
            }
            if(pass_sensor2){
                  while(isProperDist(4,5));//Ultrasonic Sensor 2
                  start=millis();
                  elapsed=0;
                  while((elapsed=millis()-start)<wait_2){
                        if(dist1=isProperDist(2,3))//Ultrasonic Sensor 1
                              goto outer1;
                  }
                  head_count++;
            }
            else{
                  if(head_count==0)
```

```
                        digitalWrite(7,0);
                  prev_sensor=2;
            }
      }
outer2:
      while(dist2){
            dist2=0;
            prev_sensor=2;
            while(isProperDist(4,5));//Ultrasonic Sensor 2
            unsigned long start=millis(),elapsed=0;
            int pass_sensor1=0;
            while((elapsed=millis()-start)<wait_1){
                  if(isProperDist(2,3)){//Ultrasonic Sensor 1
                        pass_sensor1=1;
                        break;
                  }
            }
            if(pass_sensor1){
                  while(isProperDist(2,3));//Ultrasonic Sensor 1
                  start=millis();
                  elapsed=0;
                  while((elapsed=millis()-start)<wait_2){
                        if(dist2=isProperDist(4,5))//Ultrasonic Sensor 2
                              goto outer2;
                  }
                  head_count--;
                  if(head_count==0){
                        digitalWrite(7,0);
                        prev_sensor=0;
                  }
            }
            else
                  prev_sensor=1;
      }
if (head_count >=30 && <=35)
play_alert();
else if (head_count>=40)
play_highAlert();

}

void play_alert() {
  tone(3, 3000, 2000);  // pin,frequency,duration
  delay(200);
  tone(3, 4000, 2000);
  delay(100);
  tone(3, 5000, 2000);
  delay(200);
  noTone(3);
}

void play_highAlert()
{
  tone(3, 6000, 2000);
  delay(1000);
  tone(3, 7000, 3000);
  delay(1000);
  tone(3, 8000, 3000);
  delay(1000);
  tone(3, 9000,4000);
  delay(1000);
  tone(3, 10000,4000);
  delay(500);
  noTone(3);
}
```

2. WORKING EXPLAINED:

a. Here we have used 2 ultrasonic sensors installed one after another.
b. If any person walks from outside to inside then 1st sensor would trigger first and then 2nd sensor and the code is designed in a way that it count this as increase of head count by 1.
c. Similarly, if any person walks out then 2nd sensor is triggered first then 1st sensor and arduino will read it as decrement of head count by 1.
d. The variable head_count stores the value of no. Of person inside the hospital.
e. If head count becomes>=30 and <=35 then security alarm will play alert(), for alerting the guards and other staffs.
f. When head count becomes >=40 ,then security alrm plays highAlert(), to insure main gates are closed then .
g. **This system is designed to prevent overcrowding in the hospital premises and hence to reduce the probability of contamination.**

AUTOMATIC SANITIZER MACHINE

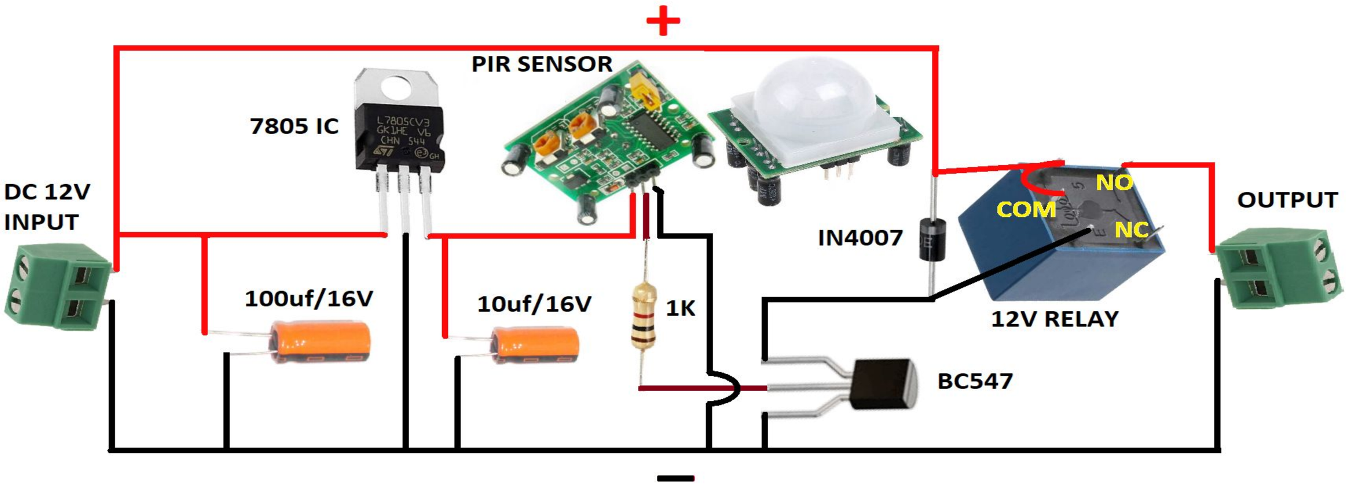
COMPONENTS

- PIR SENSOR
- 12 V RELAY
- 7805 IC
- BC547
- 100uf/16V CAPACITOR
- 10uf/16V CAPACITOR
- IN4007 DIODE
- 1K RESISTOR
- PIN SCREW CONNECTORS – 2
- HEAT SINK
- CABINET
- DATA BOARD

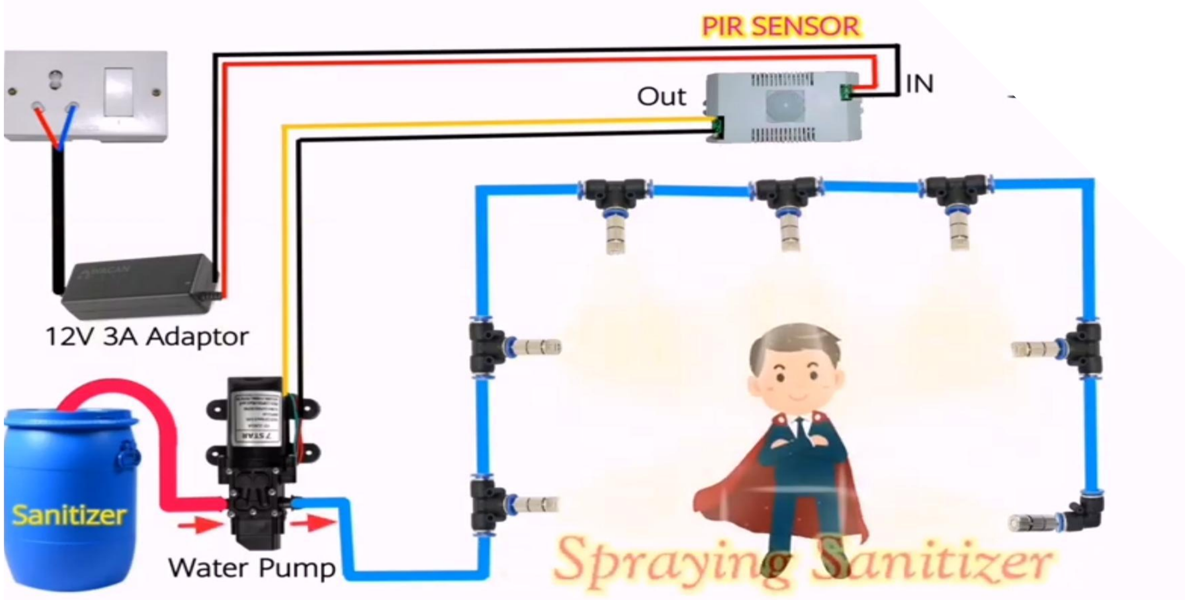
MATERIALS REQUIRED

- WATER PUMP
- PNEUMATIC –T ,PNEUMATIC-L
- SPRAY NOZZLE
- 12V,3A ADAPTOR
- 6mm PIPE,12mm PIPE
- 6 mm CABLE CLIP
- DOTE BEARDBOARD

CIRCUIT DIAGRAM



PICTORIAL REPRESENTATION



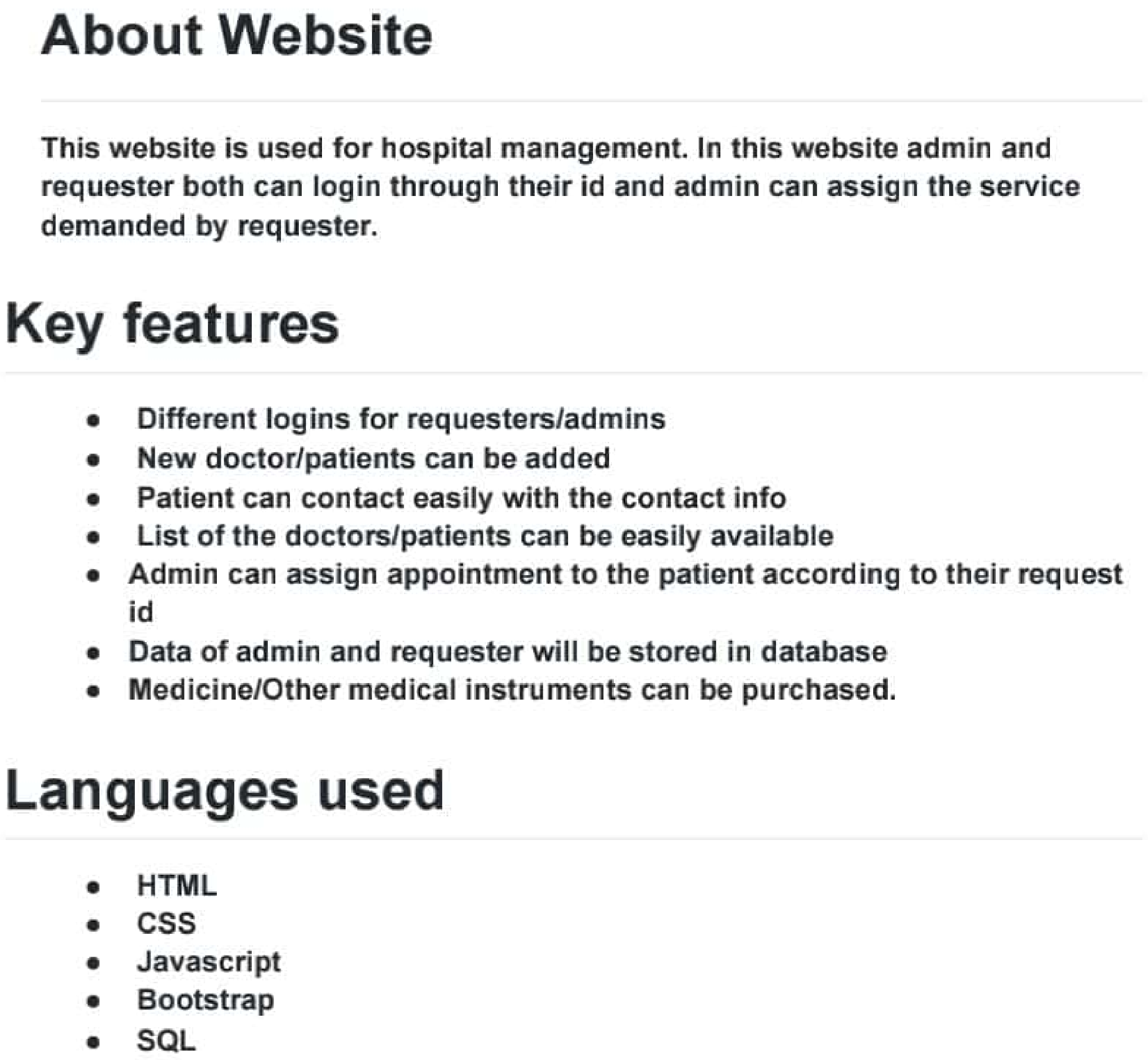
CONSTRUCTION

- Sanitizer dispenser are majorly of 3 types
- Setting the circuit as shown in the circuit diagram on to breadboard.
- Automatic ,Manually hand operator and foot operated dispenser.
- Fixing breadboard on the cabinet by adjusting the sensitivity and timer of PIR sensor.
- PIR is placed above the passage track so whenever the motion occur it will be detected by the sensor.
- The output of the sensor cabinet is connected with the water pump. its start pumping sanitizer within fraction of second .
- Attach dc to the base make sure the wood attached to the dc motor will be close enough to it to the dispenser pump of the hand sanitizer bottle .

WORKING

- Infrared sensor detect the infrared energy that is emitted by one’s body heat
- When the hands are placed in the proximity of the sensor, The infrared energy quickly fluctuates . This fluctuation triggers the pump to activate and dispenses the designated amount of the sanitizer .
- When we have sanitize our hand, the uses hands are placed under the nozzle and before the sensor.
- Motion detection sensor detects the presence of people passing through tunnel pneumatic – T and L with nozzle spares are connected with 6mm(10 pipes of 24 inches pipe)

Website Development Part



About Website

This website is used for hospital management. In this website admin and requester both can login through their id and admin can assign the service demanded by requester.

Key features

- Different logins for requesters/admins
- New doctor/patients can be added
- Patient can contact easily with the contact info
- List of the doctors/patients can be easily available
- Admin can assign appointment to the patient according to their request id
- Data of admin and requester will be stored in database
- Medicine/Other medical instruments can be purchased.

Languages used

- HTML
- CSS
- Javascript
- Bootstrap
- SQL

- PHP(Core+Advanced)

Tools/Software Used

- Visual Studio Code/Brackets Code Editor
- XAMPP

Browser

- Chrome

Code Link

https://github.com/Deeksha-Maurya/Automation-In-Hospital-Management

OCR Scanner :

It is the technology that is used to convert characters or that is either handwritten or printed in the form of paper,scanned document ,photos etc . into machine encoded text or we can say in the digital form.The steps involved in oce is basically processing the input,recognizing the text and processing it further for the required purpose.

Work process of the ocr :

Now while dealing with images for character recognition,

1. The first step is image preprocessing. What to do in image preprocessing is converting the input image into proper format and doing like a clean-up where if the image has some defects or unnecessary things that would hinder the process of recognition.
   Also, the image is configured dimensionally to give it as the input to the model for recognition.

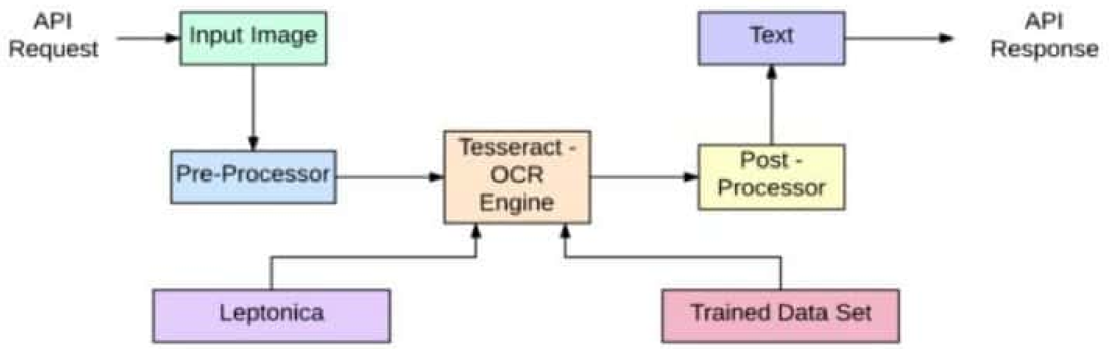
2. Processes like image or document alignment are done in the proper way i.e. if the image is tilted or reversed, it is fixed. Color coding is changed like converting to grayscale which helps is better feature extraction and hence efficient recognition. Layout analysis is done which involves finding paragraphs or columns etc. Many more operations are also involved and finally after all these preprocessing, image is fit for recognition.

3. This step involved is the recognition step. The most important part in recognition is extraction of features in an efficient and reliable way such that required results can be achieved.

4. The final step can be called as post-processing. After the character is identified, it can be converted into required types or forms such as ASCII etc. which can be used by computers for further purposes like storing data, verifying information, decoding message etc.

Control process flow

OCR Process Flow



Software requirements:

This software's required to convert the scanned image to text files

- Python :-is the general- purpose coding language.Python is a free open source programming language that is available to use .It also has huge and growing ecosystem with a variety of open source packages and libraries

- V S Code :- VS code stands for Visual Studio Code is a free source code editor made by microsoft for all types of operating system.Features include support for debugging,syntax highlighting ,intelligent code,co Vishal Studio is code editor redefined and optimized for building and debugging modern web and cloud applications.

- OpenCv :- Open source computer vision library,OpenCv is a library of programming functions mainly aimed at real-time computer vision.

- Docpraser :-Docparser is a document data capture solution built for today's modern cloud stack. Automatically fetch PDFs or scanned documents from various sources, extract specific data fields or tables, and move the parsed data to where it belongs in real-time.

- Pytesseract : It is an optical character recognition engine for various operating systems.It is free software released under the Apache License,Version 2.0,pytesseract is considered as one of the most accurate open-source OCR engines currently available.

Libraries required:

1.OpenCv

- OpenCv (open source Computer Vision Library) is a library of programming functions mainly aimed at real time computer vision.
- OpenCv-python makes use of Numpy ,which is a highly optimised library for numerical operations with MATLAB style syntax.
- By using it,one can process images to identify object ,faces or even handwriting of human

2.Numpy

- Numpy stands for Numerical python.
- Numpy is a library for python programming language ,adding support for large,multi dimensional arrays and matrices ,along with a large collection of high level mathematical functions to operate on these arrays.
- It also has functions for working in the domain of linear algebra,Fourier transform and matrices.

3.FPDF

- FPDF is a PHP library which is used to generate PDF.
- FPDF is an open source library .we can use it for any kind of usage and modify it to suit our needs.that is to say without using the PDFlib library.
- It's benefits are high level functions, choice of measure unit ,page format,margins,automatic page break,page header and footer management.

Data detection,extraction and converting text :

Steps are followed in data detection with this program:

1 . First the user gives an image or other format as the input.
2. Next the image is preprocessed and it is converted into the format such that the model in the recognition
3. Next, The model does the work of feature extraction .This is a very essential step as as the recognition is based on the efficiency of features extraction and preservation.

Program :

Code for detection of character ,words and digits

```
import cv2
import pytesseract
import numpy as np
from PIL import ImageGrab
import time


pytesseract.pytesseract.tesseract_cmd = 'C:\\Program
Files\\Tesseract-OCR\\tesseract.exe'
img = cv2.imread('1.png')
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
pytesseract
# Image to String
# print(pytesseract.image_to_string(img))
# Detecting Characters
hImg, wImg,_ = img.shape
boxes = pytesseract.image_to_boxes(img)
for b in boxes.splitlines():
    print(b)
    b = b.split(' ')
    print(b)
    x, y, w, h = int(b[1]), int(b[2]), int(b[3]), int(b[4])
    cv2.rectangle(img, (x,hImg- y), (w,hImg- h), (50, 50, 255), 2)
    cv2.putText(img,b[0],(x,hImg-
y+25),cv2.FONT_HERSHEY_SIMPLEX,1,(50,50,255),2)
```

```
#Detecting Words
# #[   0        1         2         3         4         5        6       7       8        9        10       11 ]
# #['level', 'page_num', 'block_num', 'par_num', 'line_num', 'word_num', 'left', 'top',
'width', 'height', 'conf', 'text'] boxes = pytesseract.image_to_data(img)
 for a,b in enumerate(boxes.splitlines()):
        print(b)
       if a!=0:
           b = b.split()
           if len(b)==12:
              x,y,w,h = int(b[6]),int(b[7]),int(b[8]),int(b[9])

cv2.putText(img,b[11],(x,y-5),cv2.FONT_HERSHEY_SIMPLEX,1,(50,50,255),2)
#               cv2.rectangle(img, (x,y), (x+w, y+h), (50, 50, 255), 2)
```

```
#Detecting ONLY Digits

 hImg, wImg,_ = img.shape
 conf = r'--oem 3 --psm 6 outputbase digits'
 boxes = pytesseract.image_to_boxes(img,config=conf)
 for b in boxes.splitlines():
     print(b)
     b = b.split(' ')
     print(b)
     x, y, w, h = int(b[1]), int(b[2]), int(b[3]), int(b[4])
     cv2.rectangle(img, (x,hImg- y), (w,hImg- h), (50, 50, 255), 2)
```

```
   cv2.putText(img,b[0],(x,hImg-
y+25),cv2.FONT_HERSHEY_SIMPLEX,1,(50,50,255),2)
cv2.imshow('img', img)
cv2.waitKey(0)
```

Code for extracting and converting to pdf

```
import cv2
import pytesseract
import numpy as np
from fpdf import FPDF

pytesseract.pytesseract.tesseract_cmd = "C:\program
Files\tesseract-OCR\tesseract-exe"

image = cv2.inread("Result")
grayscale = cv2.cvtcolor(image, cv2.COLOR_RGB2GRAY)
adaptive =
cv2.adaptiveThreshold(grayscale,255,cv2.ADAPTIVE-THRESH_GAUSSIAN_C,cv2.TH
RESH_BINARY, 11)

text = pytesseract.Image_to_string(image)

print(text)
#cv2.inshow('image'.image)
cv2.inshow('adaptive'.adaptive)
cv2.waitkey(0)

# A Variable called PDF is created with assigning FPDF () Class to it
pdf = FPDF()
pdf.add_page()
pdf.set_font("Times", size = 20)
fd = open("Text.txt"."r")

for i in fd:
   pdf.cell(200,10,txt = i,lh = 1,align ='C')
pdf.output("test_File.pdf")
```

Uploading the pdf from to software :

- PDF documents are sent to the Docparser API
- Docparser extracts data from PDF files
- The extracted data fields are then sent back to server with a HTTP webhook or our native Salesforce integration

// change the following variables according to your use-case

```
String fileName = 'MY_FILE.txt';
String targetURL = 'https://your.target.com/api';
String separationString = 'A_RANDOM_STRING';

// assemble the body payload
String header = '--' + separationString + '\nContent-Disposition: form-data; name="file";
filename="' + fileName + '"\nContent-Type: application/octet-stream\n\n';
String body = EncodingUtil.base64Encode(fileContent) + '\n';
String footer = '--' + separationString + '--';
String bodyPayload = header + body + footer;

// send out the request
HttpRequest req = new HttpRequest();
req.setHeader('Content-Type', 'multipart/form-data; boundary=' + separationString);
req.setHeader('Content-Length', String.valueof(bodyPayload));
req.setMethod('POST');
req.setEndpoint(targetURL);
req.setBody(bodyPayload);

Http http = new Http();
http.send(req);
```

AUTOMATED DOCTOR's CABIN

Components used:

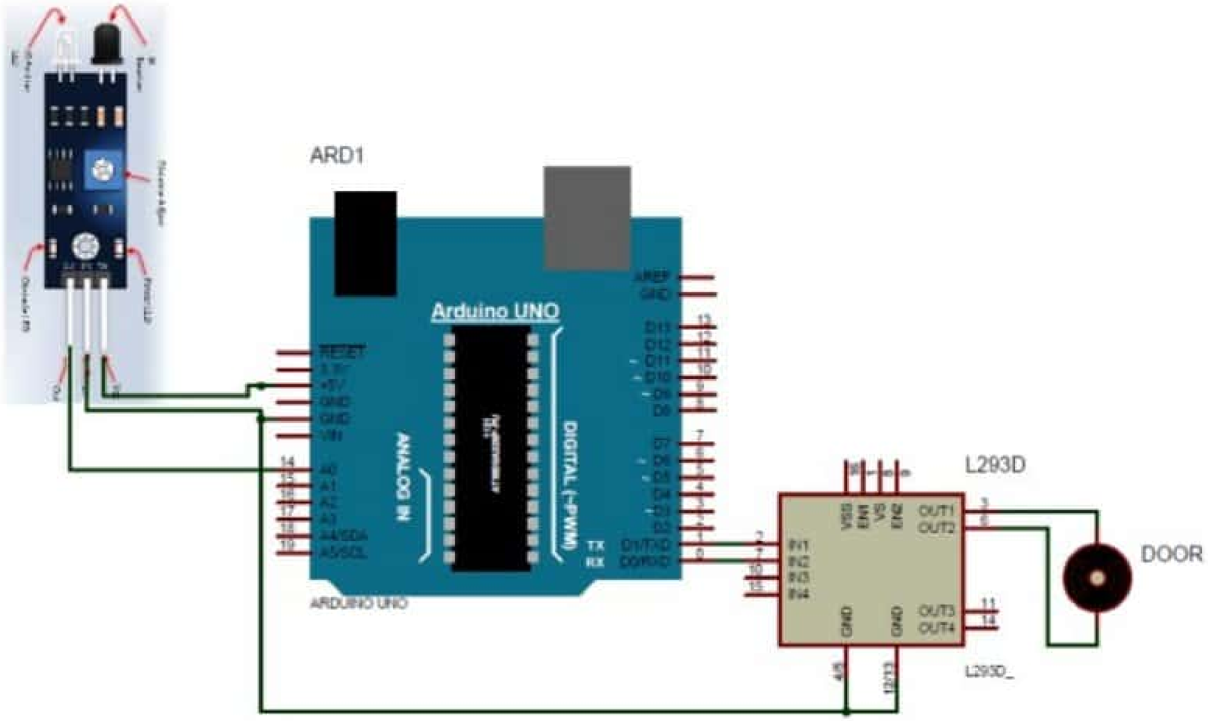
a. For automated door:
1. Active IR sensor
2. Motor driver (L293D)

b. For health report:
1.Temperature sensor (MLX90614)
2.Oled display screen (128*64 pixels)
3.Buzzer
4. Ultrasonic Sensor (HCSRO4)
5. Pulse Oximeter (MAX30102)
6. Oled display screen(128*32 pixels)
7. UVGI sterilizer
8. BCS2300
9. Arduino Uno

CONSTRUCTION FOR AUTOMATED DOOR:

1. Connect the VCC pin of IR sensor with 5V supply pin of arduino.
2. Connect the Gnd pin of IR sensor, arduino board and L293D together.
3. Connect the output pin of IR sensor with A0 pin of arduino.
4. Connect D0 and D2 pin of arduino with IN2 and IN1 respectively of L293D.
5. Connect OUT1 and OUT2 of L293D with door.

CIRCUIT DIAGRAM OF AUTOMATED DOOR:



This is just a sample circuit for automated door control only, the final circuit diagram(project's main circuit) is shown in the next module.

WORKING OF AUTOMATED DOOR:

1. Arduino Code:

```
#define doorsensor A0
 #define m1 0   // D0 pin of arduino;
 #define m2 1   // D1 pin of arduino;
void setup()
 {
pinMode(m1, OUTPUT);
 pinMode(m2, OUTPUT);
pinMode(doorsensor, INPUT);
 delay(3000);
 }
 void loop()
{
if(digitalRead(doorsensor))
 {
digitalWrite(m1, HIGH); // gate opening digitalWrite(m2, LOW);
delay(350);
digitalWrite(m1, LOW); // gate stop for a while digitalWrite(m2, LOW);
delay(5000);
 digitalWrite(m1, LOW); // gate closing digitalWrite(m2, HIGH);
 delay(350);
digitalWrite(m1, LOW); // gate closed digitalWrite(m2, LOW);
 delay(1000);
 }
else
 {
 digitalWrite(m1, LOW);
digitalWrite(m2, LOW);
 }
 }
```

2. Working Explained:

a. Here m1 represent D0 pin and m2 represent D1 pin of arduino respectively.
b. As soon as any person is detected by IR sensor, it sends the signal to arduino through A0 analog pin and sets m1 HIGH and M2 LOW, which sends further
commamd to motor driver to rotate in counterclockwise sense to open the door.
c. Once the door is open , it stops for a while there(delay of 5000 msec)with m1 amd m2 at LOW .
d. Then after this m1 is set to LOW and m2 to HIGH which triggers the motor driver to rotate in clockwise sense to shut the door.
e. When no person is detected, m1 and m2 is set at LOW.

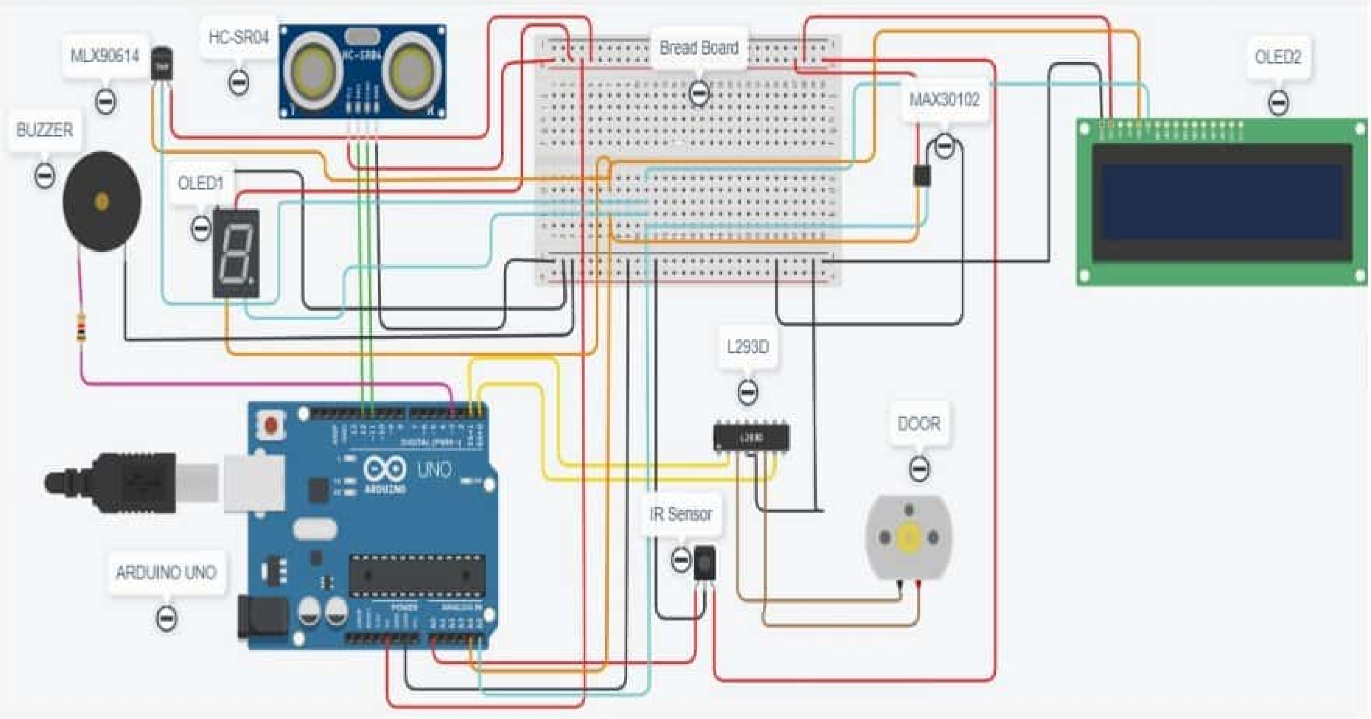
CONSTRUCTION FOR AUTOMATED HEALTH CHECK UP:

1. Connect Gnd and VCC terminal of buzzer with corresponding terminals of arduino, connect its input terminal with D3 pin of arduino.
2. Connect trig and echo pin of HC-SR04 with D12 and D11 respectively, VCC and Gnd connected as usual manner.

3. Connect SCL AND SDL pins of OLED's (128*64 and 128*32) with A4 and A5 pins of arduino respectively, Gnd pin connected in same manner and VCC pins of both OLED's connected to 5V and 3.3V pins of arduino respectively.

4. Similarly connect SCL and SDL pins of MLX90614 and MAX30102 with A4 and A5 pins,5V and Gnd connected in similar manner.

CIRCUIT DIAGRAM OF AUTOMATED HEALTH CHECK UP:



MAIN DESIGN OF WHOLE AUTOMATED CABIN

(The following circuit has been drawn using TinkerCad software)

Common Colour notationa used are as follows:
1. **Red** - Power supply (5V)
2. **Black** – Ground
3. **Orange**- A4 pin of arduino (SDA pin)
4. **Tortoise**- A5 pin of arduino (SCL PIN)

left side: Temp. monitor
right side: Heart Rate monitor
bottom: Automated door

WORKING OF AUTOMATED HEALTH CHECK UP:

1. Arduino Code:

```
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include "MAX30105.h"          //MAX3010x library
#include "heartRate.h"         //Heart rate calculating algorithm

Adafruit_MLX90614 mlx = Adafruit_MLX90614();
MAX30105 particleSensor;
```

```
//for oled1
Adafruit_SSD1306 display(-1);
//Adafruit_SSD1306 display(128,64, &Wire, -1);

for oled2
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 32 // OLED display height, in pixels
#define OLED_RESET    -1 // Reset pin # (or -1 if sharing Arduino reset pin)

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
//Declaring the display name (display)



//    code for pictures to be displayed in oled 2 which shows heart rate
static const unsigned char PROGMEM logo2_bmp[] =
{ 0x03, 0xC0, 0xF0, 0x06, 0x71, 0x8C, 0x0C, 0x1B, 0x06, 0x18, 0x0E, 0x02, 0x10, 0x0C, 0x03,
0x10,            //Logo2 and Logo3 are two bmp pictures that display on the OLED if called
0x04, 0x01, 0x10, 0x04, 0x01, 0x10, 0x40, 0x01, 0x10, 0x40, 0x01, 0x10, 0xC0, 0x03, 0x08, 0x88,
0x02, 0x08, 0xB8, 0x04, 0xFF, 0x37, 0x08, 0x01, 0x30, 0x18, 0x01, 0x90, 0x30, 0x00, 0xC0, 0x60,
0x00, 0x60, 0xC0, 0x00, 0x31, 0x80, 0x00, 0x1B, 0x00, 0x00, 0x0E, 0x00, 0x00, 0x04, 0x00,  };

static const unsigned char PROGMEM logo3_bmp[] =
{ 0x01, 0xF0, 0x0F, 0x80, 0x06, 0x1C, 0x38, 0x60, 0x18, 0x06, 0x60, 0x18, 0x10, 0x01, 0x80,
0x08,
0x20, 0x01, 0x80, 0x04, 0x40, 0x00, 0x00, 0x02, 0x40, 0x00, 0x00, 0x02, 0xC0, 0x00, 0x08, 0x03,
0x80, 0x00, 0x08, 0x01, 0x80, 0x00, 0x18, 0x01, 0x80, 0x00, 0x1C, 0x01, 0x80, 0x00, 0x14, 0x00,
0x80, 0x00, 0x14, 0x00, 0x80, 0x00, 0x14, 0x00, 0x40, 0x10, 0x12, 0x00, 0x40, 0x10, 0x12, 0x00,
0x7E, 0x1F, 0x23, 0xFE, 0x03, 0x31, 0xA0, 0x04, 0x01, 0xA0, 0xA0, 0x0C, 0x00, 0xA0, 0xA0,
0x08,
0x00, 0x60, 0xE0, 0x10, 0x00, 0x20, 0x60, 0x20, 0x06, 0x00, 0x40, 0x60, 0x03, 0x00, 0x40, 0xC0,
0x01, 0x80, 0x01, 0x80, 0x00, 0xC0, 0x03, 0x00, 0x00, 0x60, 0x06, 0x00, 0x00, 0x30, 0x0C, 0x00,
0x00, 0x08, 0x10, 0x00, 0x00, 0x06, 0x60, 0x00, 0x00, 0x03, 0xC0, 0x00, 0x00, 0x01, 0x80, 0x00
};

float roomTemp; // ambient temperature
float objectTemp, stemp; // object temperature

int readcount = 0;
float threshold= 5.6 ;

// --------- Ultrasonic Sensor preparation
#define echoPin 11 // Echo Pin
#define trigPin 12 // Trigger Pin
int maximumRange = 25; // Maximum range needed
int minimumRange = 15; // Minimum range needed
long duration, distance; // Duration used to calculate distance
int dtime;
// MAX 30102 sensor preparation
const byte RATE_SIZE = 4; //Increase this for more averaging. 4 is good.
byte rates[RATE_SIZE]; //Array of heart rates
byte rateSpot = 0;
long lastBeat = 0; //Time at which the last beat occurred
float beatsPerMinute;
int beatAvg;



void setup() {
//for temp. sensor
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial.begin(9600);
  // setting up the OLED display
  display.begin(SSD1306_SWITCHCAPVCC,0x3C) ;
  delay(1000);
  display.clearDisplay();
  display.setTextColor(WHITE);

// for heart rate sensor
 display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //Start the OLED display
  display.display();
  delay(3000);
  // Initialize sensor
  particleSensor.begin(Wire, I2C_SPEED_FAST); //Use default I2C port, 400kHz speed
  particleSensor.setup(); //Configure sensor with default settings
  particleSensor.setPulseAmplitudeRed(0x0A); //Turn Red LED to low to indicate sensor is running
}






void loop()

{

// code for temp. sensor
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  //Calculate the distance (in cm) based on the speed of sound.
  distance= duration*0.034/2;

  // reading object and ambient  temperature
  objectTemp = threshold + mlx.readObjectTempC() ;
 roomTemp = mlx.readAmbientTempC() ;

 // print to Serial port
Serial.println("Object:" + String(objectTemp) + ", Ambient:" + String(roomTemp));
Serial.println(distance);

 // display on OLED
 display.clearDisplay();
 display.setTextSize(1);
 display.setCursor(0,25);
 display.print("Dis:" + String(distance) + "cm");
 display.setCursor(65,25);
 display.print("Room:" + String(roomTemp).substring(0, 4)  + "C");
 display.display();
 display.setTextSize(2);
display.setCursor(0, 0);
if (distance > maximumRange) {
   display.print("GET CLOSER");
 }
 if (distance < minimumRange) {
   display.print("TOO CLOSE!");
 }
 if ((distance >= minimumRange) && (distance <= maximumRange))
  {
   if (readcount == 5) {   // after reading 5 consecutive time
     disptemp();

   } else {
     display.print("HOLD ON"); // when in range, ask user to hold position
     stemp = stemp + objectTemp;
     readcount++;
     dtime = 200;      // until approx. 5 x 200 ms = 1 sec
   }
 } else {     // if user is out of range, reset calculation
   dtime = 100;
   readcount = 0;
   stemp = 0;
 }
 display.display();
 delay(dtime);
 Serial.println("count  :"+String(readcount));
}

void disptemp()
{
     objectTemp = stemp / 5;       // get the average reading of temp
     display.setTextSize(1);
     display.print("YOUR TEMP:");
     display.setTextSize(2);
     display.setCursor(60,5);
     display.print(String(objectTemp).substring(0, 4) + "C");
     display.display();
```

```
      readcount = 0;
      stemp = 0;
      if (objectTemp >= 37.5) {
        play_alert();
      } else {
        play_ok();
      }
      dtime = 5000; // wait for 5 seconds
 }

void play_ok() {  // play three sequential notes when object temperature is below 37.5C
  tone(3, 600, 1000);  // pin,frequency,duration
  delay(200);
  tone(3, 750, 500);
  delay(100);
  tone(3, 1000, 500);
  delay(200);
  noTone(3);
}

void play_alert() { // beep 3x when object temperature is >= 37.5C
  tone(3, 2000, 1000);
  delay(1000);
  tone(3, 3000, 1000);
  delay(1000);
  tone(3, 4000, 1000);
  delay(1000);
  noTone(3);
}




//  code for heart rate sensor
long irValue = particleSensor.getIR();    //Reading the IR value it will permit us to know if there's a
finger on the sensor or not
                                           //Also detecting a heartbeat
if(irValue > 7000){                                         //If a finger is detected
    display.clearDisplay();                                  //Clear the display
    display.drawBitmap(5, 5, logo2_bmp, 24, 21, WHITE);       //Draw the first bmp picture (little
heart)
    display.setTextSize(2);                                  //Near it display the average BPM you can display
the BPM if you want
    display.setTextColor(WHITE);
    display.setCursor(50,0);
    display.println("BPM");
    display.setCursor(50,18);
    display.println(beatAvg);
    display.display();

  if (checkForBeat(irValue) == true)                        //If a heart beat is detected
 {
  display.clearDisplay();                          //Clear the display
  display.drawBitmap(0, 0, logo3_bmp, 32, 32, WHITE);    //Draw the second picture (bigger
heart)
  display.setTextSize(2);                          //And still displays the average BPM
  display.setTextColor(WHITE);
  display.setCursor(50,0);
  display.println("BPM");
  display.setCursor(50,18);
  display.println(beatAvg);
  display.display();
  tone(3,1000);                                //And tone the buzzer for a 100ms
  delay(100);
  noTone(3);                                  //Deactivate the buzzer to have the effect of a "bip"
  //We sensed a beat!
  long delta = millis() - lastBeat;                //Measure duration between two beats
  lastBeat = millis();

  beatsPerMinute = 60 / (delta / 1000.0);          //Calculating the BPM

  if (beatsPerMinute < 255 && beatsPerMinute > 20)            //To calculate the average we strore
some values (4)
  {
   rates[rateSpot++] = (byte)beatsPerMinute; //Store this reading in the array
   rateSpot %= RATE_SIZE; //Wrap variable

   //Take average of readings
   beatAvg = 0;
   for (byte x = 0 ; x < RATE_SIZE ; x++)
    beatAvg += rates[x];
   beatAvg /= RATE_SIZE;
  }
 }

}
 if (irValue < 7000){       //If no finger is detected it inform the user and put the average BPM to 0
or it will be stored for the next measure
   beatAvg=0;
   display.clearDisplay();
   display.setTextSize(1);
   display.setTextColor(WHITE);
   display.setCursor(30,5);
   display.println("Please Place ");
   display.setCursor(30,15);
   display.println("your finger ");
   display.display();
   noTone(3);
   }
}
```

WORKING EXPLAINED:

1. For Temperature Sensor(MLX90614):

a. Here OLED1 is displaying object temperature along with other warning messages.
b. Also, we made use of ultrasonic sensor to measure the person's distance from the sensor as the temp depends on the distance from the object also.
3. The range we kept is from 15cm to 25 cm (for efficient calculation of temp.) , if the person is within this range then the OLED1 would display a message " HOLD ON" and waits for 1000 msec to display body temperature.
4. We have kept a threshold temperature of 5.6degree celcius after callibrating with digital thermometer.
5. If the person is very near to sensor(<15cm) ,it will display " TOO CLOSE" and if he is at >25cm then it will dispay " GET CLOSER" .
6. If the object temp. is>37.5 degree celcius then the buzzer will play alert_tone(beep 3*) or else it will play ok_tone (3 sequential notes).

2. For Heart rate sensor (MAX30102):

a. Here, OLED 2 is displaying heart rate of the person(in BPM) along with other warning message and picture of heart.
b. If a finger is detected the screen first displays the smaller heart picture(logo_2) and shows the average heart rate inBPM.
c. If heart rate is detected the screen then displays the bigger heart picture(logo_3) and still shows the average heart rate in BPM(beats per minute).
d. Also, whenever a heart rate is detected the buzzer produces the beep tone of 100 msec.
e. When no finger is detected, the screen shows the message " PLEASE PLACE YOUR FINGER".

CONSTRUCTION AND WORKING OF UVGI STERILIZER AND BCS 2300:

1. UVGI Sterilizer and BCS 2300 need to be connected directly with mains.
2. UVGI Sterilizer would emit UV rays to eradicate and kill harmful bacterias of doctor's cabin.
3. We will separate doctor's cabin with a thin glass film , where patient and doctor will be either side of it.
4. BCS2300 is used for communicating between doctor and patient through microphone an amplifier speakers.

UVGI Sterilizer (Source: Internet)

BCS-2300 (Source: Internet)

Conveyor System :

It is a mechanical handling equipment that moves materials from one location to another.Conveyors are especially useful in applications involving the transport of heavy or bulky materials .Conveyors systems allow quick and efficient transport for a wide variety of materials ,which make them very popular in the material handling.

Hardware Requirements:

1. 20 pin AT89C2051 microcontroller : It is a low voltage,high performance he AT89C2051 is a low-voltage, high-performance CMOS 8-bit microcomputer with 2K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmels high-density nonvolatile memory technology and is compatible with the industry-standard MCS-51 instruction set.

2. Buffer : Op-amps have a variety of uses. One use is as a so-called *buffer*. A buffer is something that *isolates* or *separates* one circuit from another. In order to explain this more precisely, let's take a closer look at our 3-bit DAC.

3. Arduino sensors : A component that measures one form of energy (like light or heat or mechanical energy) and converts it to voltage or current.

4. LED indicator : The LED (light-emitting diode) Indicator is a device which can provide visual feedback. The LEDs come in red, yellow, and green colors.

5. Inverter : An inverter circuit outputs a voltage representing the opposite logic-level to its input. Its main function is to invert the input signal applied. If the applied input is low then the output becomes high and vice versa.

6. Counter : Counter (digital) In digital logic and computing, a counter is a device which stores (and sometimes displays) the number of times a particular event or process has occurred, often in relationship to a clock signal.

Software Requirement :

1. C program :is a general-purpose, procedural computer programming language supporting structured programming, lexical variable scope, and recursion, with a static type system.

2. <u>KIEL (ODE) Compiler</u> :The Keil C51 C Compiler for the 8051 microcontroller is the most popular 8051 C compiler in the world. It provides more features than any other 8051 C compiler available today.

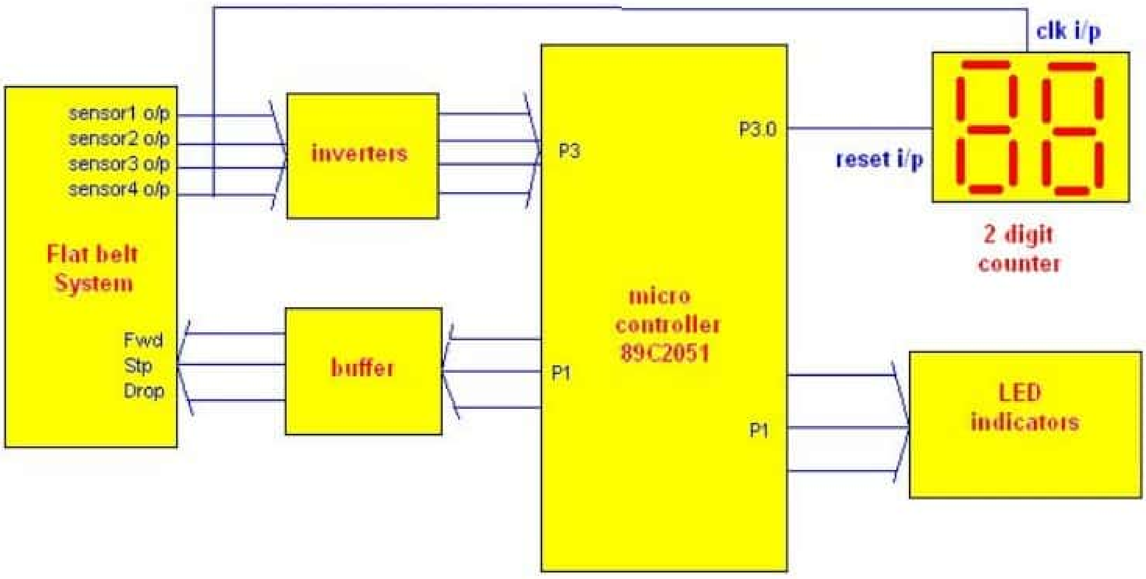
<u>Construction</u> :

This is a very small system and doesn't require more input output pins.. So we are using 20 pin **89C2051** micro controllers. As shown in figure there are inverters, buffers, indicators and a 2 digit counter, the building blocks of the system

89C2051 – it performs the following tasks.

- It controls the belt motor by giving fwd and stp pulses.
- Also it gives a drop command to fill the container.
- It counts no of items in the container and resets the counter with every new container
- Indicates different actions on LEDs.

1. Inverters - It inverters the positive pulses into negative pulses.
2. Buffers - it provides isolation between digital system(micro controller) and analog system(conveyer belt)
3. LED indicators - It indicates various events currently going on

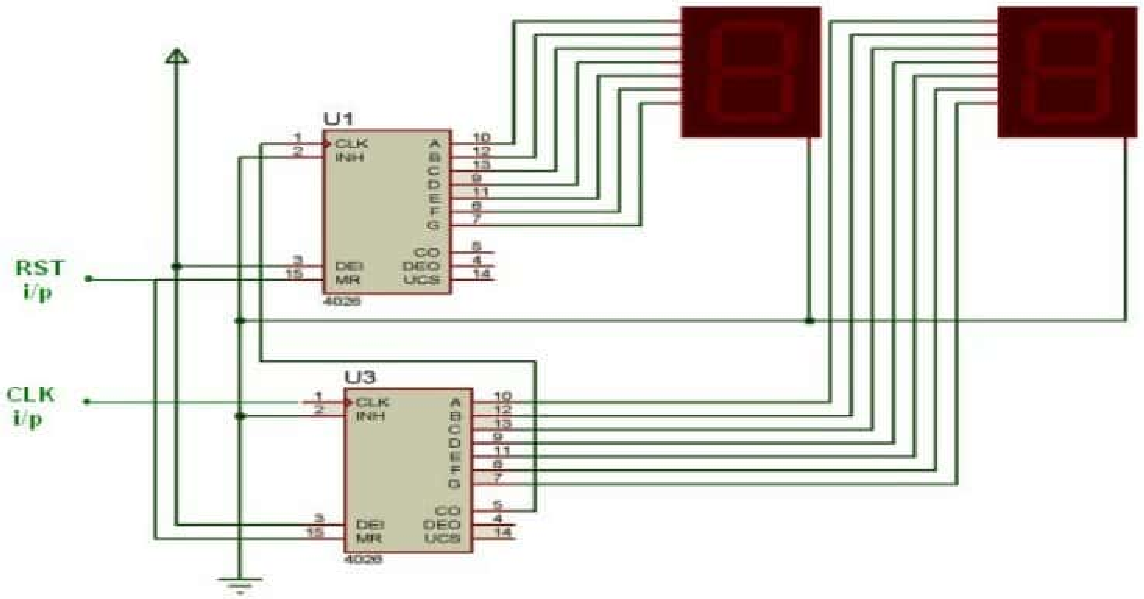


Conveyor Belt Controller Circuit Connections and Operation

Connections -

As shown in circuit tab 1 two ports of 89C2051 P1 and P2 controls entrie system . Pins P1.0 , P1.1,P1.2 are connected to forward ,stop and control buffer 7407 . P1.5 , P1.6, and P1.7 drives 3 LEDs as shown .SO P1 is completely output port.All four sensor outputs are connected with P3 pins P3.2,P3.3,P3.5 and through inverters 74LS04.One LED indicator is also connected with each output.pin p3.0 is connected with reset input of counter .Also the sensor 4 output is directly connected with reset input of counter .A 12 MHz crystal along with 2 33 pf capacitors is connected with XTAL pins to provide clock signal .A push button switch along capacitor c1 forms power on reset circuit.

Circuit diagram for conveyor :



Above figure shows 2 digit counter using CMOS chip CD4026, that is counter cum 7 segment display driver. Its outputs can be directly connected to common cathode type 7 seven segment displays. As shown in figure the clock signal from sensor 4 is given to chip U3. Its carry out signal is connected with the clock signal of chip U1. Clock inhibit signals (INH) of both chips are tied to Gnd and display enable signals (DEI) are tied to Vcc. The master reset (MR) of both chips is connected with reset output from 89C2051.

Operation :-

- Controller will apply fwd pulse to motor and motor will start rotating
- At the same time one timer is started. Container will move forward and this is indicated by D1
- As document passes through sensor 1 the timer is reset
- As a document reaches sensor 2 immediately a stop pulse is applied and the document is stopped or waited until the pharmacy person takes a receipt around 15sec . D2 glows to indicates this event
- Now If he took then the stop signal is made high to wait for items from the pharmacy person to keep using the timer and counter upto required time.then Now the forward pulse is applied.
- OR
- If the pharmacy person doesn't do it then again it gets started to move end Now again the forward pulse is applied.
- As the container reaches sensor 4. Again motor is stopped for 3 sec
- Again this cycle repeats
- Now if there is no document is the belt. There is no output from sensor 1.
- So timer overflows and that means there will be no more documents to be carried .
- So the motor will be automatically stopped.

Software Program :

Delay function generates random fix amount delay (<10 ms) to provide pulse to LEDs. Bigly function generates a fixed amount of delay of 2 sec to hold the conveyor belt when the container reaches sensor 4 positions.Mildly function is again random fix amount delay (<1 ms) provided in between two item counts.

Timer function is an interrupt enabled function and it is called automatically when timer overflows from all 1's to all 0's. It is called after every 50 ms. It increases the count (b) and calculates up to 200. If this count is reached that means a total 10 sec delay is over then it will apply a stop pulse and stop rotating the motor and belt.

Interrupt function is also an interrupt enabled function and it is called when the container passes through it. It just resets the count (b) every time as containers are passing through the belt.

Main function performs following tasks

- Initializes ports, timer and enables interrupts
- Then it applies start pulse to motor and reset pulse to counter
- Indicates motor is running on LED
- Waits for container to reach at sensor 2. When it reaches stops the motor
- Now it applies high logic to drop pin to drop item into container
- Counts number of items to 50. And then stops dropping items.
- Again applies start pulse to motor and now waits for o/p from sensor 3
- As container reaches to sensor 3 position sends stop pulse
- After 2 sec again the same above cycle repeats.

Project Code :

```
#include<reg51.h>
 sbit strt = P1^0;    // defining port pins
 sbit stp = P1^1;
 sbit drop = P1^2;
 sbit led1 = P1^7;
 sbit led2 = P1^6;
 sbit led3 = P1^5;
 sbit obj = P3^4;
 sbit cntr = P3^5;
 sbit end = P3^7;
 sbit rst = P3^0;
unsigned int c=0,b=0;
void main()
 {
    P1=0x00;   // P1 as output port
    P3=0xFE;         // P3 as input port
    TMOD=0x01;       // timer initialization
    TL0 = 0xAF;      // load timer value
    TH0 = 0x3C;
    TR0 = 1;         // start timer
    IE=0x83;         // enable timer and external interrupt
    back:strt = 1;   // apply start pulse to motor
    rst=1;       // and reset pulse to 2 digit counter
    delay();
    strt = 0;
    rst=0;
    led1 = 1;        // indicate on LEDs
    led2=0;
    while(obj==1);   // wait until sensor 2 output
    stp = 1;     // apply stop pulse to motor
    delay();
    stp = 0;
    led1=0;      // indicate on LEDs
    led2=1;
    wait = 1;    // start  taken from patient
    led3=1;
    wait =0;     // when item is kept items
    led3=0;
    strt = 1;     // start motor
    delay();
    strt = 0;
    led2=0;
    led1 = 1;
    while(end==1);   // wait until sensor 3 output
    stp = 1;     // apply stop pulse again
    rst=1;
    delay();
    stp = 0;
    rst=0;
    led1=0;
    led2=1;
```

```
   bigdely();  // wait for 2 second
   obj=1;
  end=1;
  EA=1;
  goto back;           // again start process


}
```

PAYMENTS AND EXIT:

To pay for all services patients will receive e-bill on registered mobile number/email with the link of payment modes which include net banking, debit/credit card and various UPI and online wallet. To avoid cash flow which is also one of the channel of spread of corona virus, patients are advised to use cashless transition features as much as possible . But in case , he/she will unable to pay through online mode then he/she pay cash at cash counter.

Also, we can make use of pre deposit kiosk system which would work similar as credit card. In this system patients /staffs has to pre-deposit some amount of money to the Hospital and everytime they visit hospital their bill would be automatically detucted from this deposit. The disadvantage of this system is that it is useful only for patients who are under regular treatment ao who periodically visits the hospital for regular check up.

Cost Estimation

Sanitation part :

PIR sensor	-	₹ 139
12V relay	-	₹ 109
7805 IC	-	₹ 9
BC547trasistor	-	₹ 10
100uf/10v capacitor	-	₹ 7
IN4007 diode	-	₹ 6
Heat sink	-	₹ 64
Cabinet	-	₹ 245
Water pump	-	₹ 350
Pneumatic T & L	-	₹ 191
Spray Nozzle	-	₹ 350
12V3A adaptor	-	₹ 355
6mm pipe 12mm	-	₹ 283
pipe 6mm cable	-	₹ 283
clip	-	₹ 85
Total Amount	-	₹ 2506

Conveyor System :

AT89C2051 Microcontroller	-	₹ 89
Arduino Sensors	-	₹ 35
Buffer (dac)	-	₹ 64
Inverter(+ to - pulse)	-	₹ 245
Counter	-	₹ 139
LED indicator	-	₹ 89
DC Motor	-	₹ 1000
Total Amount	-	₹ 1661

Doctor's Cabin :

Active IR Sensor	-	₹	35
Motor driver (L2930)	-	₹	99
MLX90614(Temp sensor)	-	₹	779
MAX 30102(Pulse oximeter)	-	₹	190
Buzzer	-	₹	36
OLED display(128 * 64 pixel)	-	₹	230
OLED display(128 * 32 pixel)	-	₹	230
Arduino uno	-	₹	335
Ultrasonic sensor	-	₹	75
Total Amount	-	₹	2009

Automated Surveillance:

Ultrasonic sensor*2	-	₹	150
Security Alarm Buzzer	-	₹	36
Arduino mini	-	₹	165
Total Amount	-	₹	351

_Total Cost Estimation :

Sanitation part	-	₹	2506
Conveyor System	-	₹	1661
Doctor Cabinet	-	₹	2009
Automated Surveillance	-	₹	351
Total Cost	-	₹	6527

WHO WILL BE BENEFITTED / MARKET

There are many benefits to healthcare providers through automation. Not only will the end to end processing of customer records would be made easier, automation would also result in the actualization of efficiencies and synergies across the entire value chain of activities that healthcare organizations provide. When we mean healthcare providers, we include the entire gamut of service providers including day care providers, clinics, full fledged hospitals and pharmaceutical outlets. The practice of storing medical records of patients and referring to them on subsequent visits would greatly benefit the doctors and the paramedics who would not only have access to the patients' history but also have details about the patients in terms of medications and allergies to any specific drug. Further, by automating pharmacy it makes contactless interaction with the patient as well as with the doctor.It will impact effectively to whole area whether it is urban or rural.Our project is based on small scale so that it could be easily implemented in villages.Rural areas of Bihar and Uttar Pradesh are hit worst. A recent report says that Bihar has 0.12 beds per thousand infected people which is really appalling and raises eyebrows. So, our project can help in a way to deaccelerate the spreading of virus in INDIA.

CHALLENGES FACED DURING WORKING WITH THE PROJECT

WHAT WENT RIGHT	WHAT WENT WRONG
1. Our idea was to make the project so effective with the low cost.	1. First of all our very first idea was to implement this project using PLC software, but since the cost of PLC hardware is too much and it would not be useful for small scale hospitals, so our very challenge was to make it cost effective.
2. Kiosks -we thought to use kiosks which gives a self-service access to products and services	2. But kiosks are expensive in cost.
3. Shoes are also the potential carrier of COVID virus ,so we thought of to use sanitiser mats and we left out with it because it comes self-made from market ,so there is no chance of further implementation.	3. Leg operated sanitizer availability not mentioned and refilling of sanitizer tank .

WHAT WENT RIGHT	WHAT WENT WRONG
4. . In place of kiosks we thought of using the ocr scanner which scans the patient data and converts to digital form automatically and thereby we can reduce some contact of person to device	4. In providing a report we thought let the patient get their report through mail but when it comes to offline and same patients are not aware of this technology.
5. We thought of using a conveyor in which we can transfer their report and also which pass through pharmacy where they get a report along with medicine if required.	5. In the case of sanitizing vehicles we did nothing.Another method to be formulated to sanitizer Vehicle parts.
6. The website has been developed with so many features in less time. It is user friendly too and depicts the hard work of every member.	6. The web is not developed on large scale and can’t handle more users. Also, to implement every team member’s idea is also not possible.
7. Use of cheap sensors for automatic surveillance in the hospital premises is a great approach to monitor the head count.	7. For the temp. Sensor part the challenge was to get the exact reading of patient’s body temp. and since it changes with distance , so we needed a system which can work in specific range only, that’s why we made use of ultrasonic sensor.

FAULTS/ MISTAKES COMMITTED

1. Firstly we are thought of using plc in all automated fields but it is very costly and then we change to all less cost technology.In this i made some mistake in using some different technique then we thought of using an ocr scanner and then conveyor by some code and some open source software.
2. In the case of sanitizing vehicles we did nothing.Another method to be formulated to sanitizer Vehicle parts.
3. For the heart rate sensor ,it was really very difficult to design a non-contact sensor using pulse oximeter, our system makes use of sensor which activates on pressing the finger(contact sensor). However, we can use thin film of fibre through which the patient can touch the sensor and after use the film sticks to his finger which can be further removed.
4. The idea of developing web was adopted by team after significant amount of time. If it has been adopted a bit earlier , then we would have excelled in making website with added features and more options.
5. First we thought of using the kiosk where people can fill their details .But in offline way this may not turn up to our concept of automated technology and even it may include contact of humans.

ALTERNATE POSSIBILITIES/ IMPROVEMENT OPTIONS

1. Advanced future technologies may be used to keep the patient updated through website/app.For patients who are not having smartphones ,updates can be send to them through text message using IOT.
2. Also, in the project we are unable to design a non contact sensor which can measure blood pressure(although CamBP is available in market which performs the same function but that is too expensive). So, probably we can design a system using piezoelectric Ultrasonic transducers to measure blood pressure and this would be more accurate and efficient than presently available sphygmomanometer.
3. Since the available cash deposit kiosk is too expensive, so perhaps we can design a cheaper machine which would accept cash as well as coins.

5. We can use this graphic scanner to convert text, drawings, and photographs into digital form and store it to the computer system for further processing. Or a barcode scanner for adhere code in place of ocr . Other than conveyor belts we can use some small robots which follow some instruction like Goods-to-Person technology(G2P).

6. Future automated hand sanitizer helps to monitor and control and spread of coronavirus from using a same sanitizer bottle from a suspected carrier to a safe and healthy carrier.

7. It eliminates the need of soap and water for sanitization of hands.

8. It minimises the the touch points and thus achieving effective sanitization.

9. Also, we will try to make the website on large scale , we will also try to make an alternative for rural area.

10. Further we can use automation in labs of Hospitals to collect samples.

REFERENCES AND STUDIES DONE

1. www.w3schools.com
2. Getbootstrap.com
3. Fontawesome.com
4. https://www.engineersgarage.com/contributions/conveyor-belt-controller/
5. https://docparser.com/blog/post
6. https://www.murtazahassan.com/courses/opencv
7. https://www.open-electronics.org/simplify-covid19-diagnosis-measuring-heart-rate-and-spo2-with-max30102/
8. https://learn.sparkfun.com/tutorials/mlx90614-ir-thermometer-hookup-guide/all

9. Basically for completion of this project, we made use of you tube tutorials .
   For sanitizer part, to make DIY Sanitation Disinfection Tunnel we referred to “Electronics Adda”.

10. For knowledge of various sensors in the project, we took help of the pdf provided by Coursera Community during an online training on UAV(Unmanned Automated Vehicles).

11. For web development part, we referred to studies of HTML, CSS, Bootstrap, SQL,PHP, nodeJS from you tube tutorials and nptel videos.

SUMMARY & CONCLUSION

Technologies used: IOT/ Digital Electronics/ Sensors and transducers/ arduino programming/ microcontrollers/ C/C++ used in arduino coding/ PHP(core + advanced)/SQL/Bootstrap/Javascript/SQL/ CSS

The project was very helpful for us in many ways. We learnt different technologies, worked more on our known skills to make this project successful.Some takeaways from this project are as follows:

1.The project enhanced our knowledge about different sensors and actuators,also we simulated complex circuit using arduino in TinkerCad software which boosted our confidence.

2. We also learnt to design our own website using SQL/PHP/HTML and CSS, which was completely new for non coding branch students, so we developed new skills also.

3. Designing automated monitoring of head count using ultrasonic sensors was completely new idea which enhanced our critical thinking ability.

PROJECT DOCUMENTATION

PROJECT PHASE	DATE	DESCRIPTION
Pre- Initiation	20/04/2020	Team formation
Initiation 1.0	02/05/2020	Project details submitted to Growth Centre
Initiation 2.0	14/05/2020	Mentor Assigned
Planning 1.0	06/07/2020	Idea submission and discussion with mentor
Planning 2.0	18/07/2020	Progress Report 1.0 submitted
Planning 3.0	27/07/2020	Project Overview submitted
Monitor & Control 1.0	29/07/2020	Project Proposal submitted
Monitor & Control 2.0	26/08/2020	Implementation in report by updated guidelines given by mentor
Monitor & Control 3.0	17/09/2020	Project Update to Growth Centre
Closure 1.0	01/10/2020	Pre Final Report submitted
Closure 2.0	24/10/2020	Final Report submitted

WORK FLOW OF TEAM MEMBERS

1. SANITIZATION	LEYA SUSSAN / SNEHA G R
2. WEB DESIGNING	DEEKSHA MAURYA / MANISHA SAH
3. AUTOMATED MONITORING OF HEAD COUNT	VISHAL SINHA
4. READING DATA THROUGH OCR	IMPANA H C
5. AUTOMATED DOCTOR’S CABIN	VISHAL SINHA
6. SUPPLY OF DRUGS/REPORT THROUGH CONVEYOR	IMPANA H C
7. PAYMENTS AND EXIT	PRIYANKA SONI

THANK YOU