ELEVATOR WITH ZERO POWER CONSUMPTION

A PROJECT REPORT

Submitted by

SAI VIJAY ROHIT.P [Reg No: RA1511008010031]
AKSHAT KUMAR SINGH [Reg No: RA1511008010045]

Under the guidance of

Dr.M.Sayeekumar, Ph.D

(Assistant Professor, Department of Department of Information Technology)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

of

FACULTY OF ENGINEERING AND TECHNOLOGY

S.R.M. Nagar, Kattankulathur, Kancheepuram District

MAY 2019

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled "**ELEVATOR WITH ZERO POWER CONSUMPTION**" is the bonafide work of "SAI **VIJAY ROHIT.P [Reg No: RA1511008010031], AKSHAT KUMAR SINGH [Reg No: RA1511008010045]**", who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr.M.Sayeekumar, Ph.D
GUIDE
Assistant Professor
Dept. of Department of Information Technology

SIGNATURE

Dr. G.Vadivu
HEAD OF THE DEPARTMENT
Dept. of Information Technology

Signature of the Internal Examiner

Signature of the External Examiner

ABSTRACT

The main motive of this project is to provide with a simulation of an elevator which uses a dynamic weight array which changes according to the weight present in elevator.

For this weight sensor is used which collects the data and sends to web server (Raspberry Pi) where it is used to calculate a proper counter weight. So that the elevator will move up or down depending on the requirement of the person inside the lift.

For dynamic weight array formation knapsack algorithm will be used. So, depending up on the weight present in the lift the nearest counter weight value will be found and assign it to the weight array. This weight array has will have slots which will be filled according to the weight assigned.

The whole system will work on two pulleys. One at the top of the building. The other would be under the building with the counter weight system. Two pulleys diameter ratio is fixed according to the building height and threshold weight of the elevator.

For this simulation we are going to fix the ratio as 1:6 and building height as 8 meters. The whole system is in an ideal condition that is, Zero Friction and Gravity remains constant throughout all the floors.

ACKNOWLEDGEMENTS

The success and the final outcome of this project required guidance and assistance from different sources and we feel extremely fortunate to have got this all along the completion of our project. Whatever we have done is largely due to such guidance and assistance and we would not forget to thank them.

We express our sincere thanks to the Head of the Department, Department of Information Technology, Dr.G.Vadivu, for all the help and infrastructure provided to us to complete this project successfully and his valuable guidance.

I would like to express my deepest gratitude to my guide, Dr.M.SayeeKumar, Ph.D his valuable guidance, consistent encouragement, personal caring, timely help and providing me with an excellent atmosphere for doing research. All through the work, in spite of his busy schedule, he has extended cheerful and cordial support to me for completing this research work.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all the Teaching staff of the Department of Information Technology which helped us in successfully completing our minor project work. Also, we would like to extend our sincere regards to all the non-teaching staff of the department of Information Technology for their timely support.

Author

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

IoT	Internet of Things
PLC	Programmable Logic Controller
AC	Alternating Current
DC	Direct Current
IoWT	Internet of Wearable Things
GPIO	General-Purpose Input/output
PC	Personal Computer
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
GUI	Graphical User Interface

CHAPTER 1

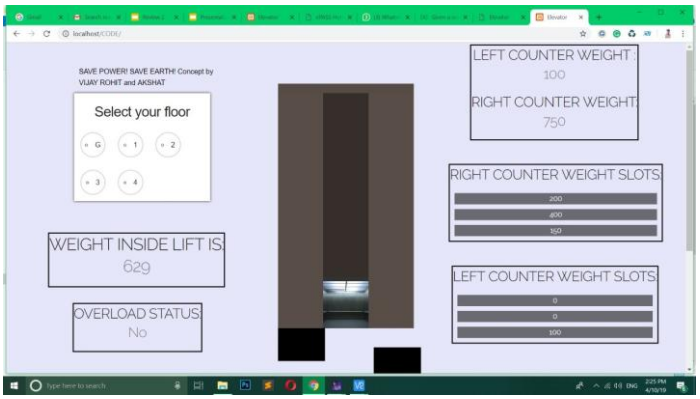
INTRODUCTION

1.1 Simulation

Simulation is typically a process to demonstrate a real world scenario on a virtual environment with the help of computer software and hardware.

The simulation here is an approximate imitation of the operation which is trying to manifest of an elevator; the act of simulating first requires a model is developed. This model is a well-defined description of the simulated subject, and represents its key characteristics, such as its behavior, functions and abstract or physical properties.

Figure 1.1: Simulation Figure



1.2 Our motive

Our main motive is to provide with a simulation of an elevator which uses a dynamic weight array which uses a dynamic weight array which changes according to the weight present in elevator. For this we are going to use a weight sensor which collects the data and sends to web server (Raspberry Pi) where it is used to calculate a proper counter weight. So that the elevator will move up or down depending on the requirement of the

person inside the lift. This elevator serves a better purpose than the existing elevators.

1.2.1 Too many Organizations

There are numerous organization which uses the conventional old method elevators that comprises of AC or DC motors. This results in huge amount of electricity consumption. Also, this raises a lot of concern towards efficient use of electricity.

1.2.2 Efficient Method

This simulation will change the way how present elevators are wasting the electric power. The project is dedicated towards a better efficient way which reduces the use of electricity and uses more of gravity.

1.2.3 Efficient Algorithm

Here in this method weight slots are used which will help in the movement of the elevator up or down. So, an effective algorithm was developed which can calculate the perfect weight to be used for the operation. The algorithm is basically a subset of the famous Knapsack problem.

1.2.4 Saves Money

Since the method reduces the dependency on electricity so, in the other way it saves money which is an essential commodity. This saved money can be used in better application for the enhancement of the industry.

1.3 Internet Of Things

Everyone knows about Internet of Things (IoT) and its application in today's world. The Internet of things (IoT) is the extension of Internet's connectivity into physical de- vices and everyday object. IoT is basically the combination of electrical devices and major technologies for the better applications in real life. Embedded with electronics, Internet connectivity, and other forms of hardware (such as sensors), these devices can communicate and interact with others over the Internet, and they can be remotely monitored and controlled.

1.3.1 Applications

There are lots of applications of IoT which are:

Consumer Application

These applications are business to consumer services like home automation, smart home appliances, smart homes etc. A growing portion of IoT devices are created for consumer use, including connected vehicles, home automation, wearable technology (as part of Internet of Wearable Things (IoWT)), connected health, and appliances with remote monitoring capabilities.

IoT devices are a part of the larger concept of home automation, which can include lighting, heating and air conditioning, media and security systems.Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off.

Commercial Application

These applications are business to business services medical and health care, transportation etc. The **Internet of Medical Things** (also called the **internet of health things**) is an application of the IoT for medical and health related purposes, data collection and analysis for research, and monitoring.

IoT devices can be used to enable remote health monitoring and emergency notification systems. These health monitoring devices can range from blood pressure and heart rate monitors to advanced devices capable of monitoring specialized implants, such as pacemakers, Fitbit electronic wristbands, or advanced hearing aids

Industrial Application

These applications are for heavy duty industrial purposes. The IoT can realize the seamless integration of various manufacturing devices equipped with sensing, identification, processing, communication, actuation, and networking capabilities. Based on such a highly integrated smart cyber physical space, it opens the door to create whole new business and market opportunities for manufacturing.

CHAPTER 2

LITERATURE SURVEY

2.1 Design and implementation of PLC based Elevator

Carter and A.Selvaraj (2013) explained about an elevator operation which uses an AC motor to drive the elevator-cabin and is fully automated using Programmable Logic Controller (PLC). Its control is based on the input that we get from the operator as well as from the sensors. Accordingly to that signal PLC will make the drive motor and door motor to work correspondingly. This paper described the entire circuit diagram and development of ladder logic. The developed ladder logic has been implemented by using VersaPro 2.02.

2.2 Design Of A PLC Based Elevator Control System

Anusha and Sutagundar (2015) introduced the basic structure of elevator control system design and the simulation of the design using Indra-works-engineering which is a Bosch-Rexroth group PLC simulation software. The main objective of this paper is to replace the traditional relay logic elevator control system with the modern PLC based elevator control system and explained block diagram of overall PLC system and signal control system. It explains elevator control system using PLC ladder logic for interfacing of various sensors, up-down movements etc.

2.3 Application of PLC for Elevator Control System

et.al (2011) explained research paper an attempt has been made to integrate the PLC with elevator for developing its control system. It is observed that the PLC based controller for elevator works better than the other control systems.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Problem Definition

The present model of elevators uses a static counter weight of certain maximum capacity which is controlled by PLC. So, every time someone uses the elevator the whole counter weight is moved up/down using an AC electric motor. This uses a lot of power every time. Thus leading to wastage of power everyday by a lot of organizations around the globe. To overcome this problem we need to replace the present static counter-weight with a dynamic counter weight. Which changes depending up on the weight present in the elevator and provided specifications by the operator. For this there is a need of an efficient algorithm to change the counter weight and proper infrastructure suitable to the weights in the counter weight array.

3.2 Proposed System

In order to access power continuously, we should also conserve energy at all costs. This new method is a befitting example to save our precious electrical energy which can be used in other useful areas. Also, the cost of the existing elevators is too much, which will be required.

3.3 Requirements

3.3.1 Hardware Requirements

Raspberry Pi

The Raspberry Pi is a little however full-included PC on a solitary board. It connects to a screen and you join a console, mouse and speakers. The Raspberry Pi can be utilized for perusing the web, making reports and spreadsheets, playing amusements, watching recordings and parcels more. It likewise gives an incredible domain to getting the hang of programming and advanced making. You can likewise associate up equipment to the Pi's GPIO (broadly useful info/yield) sticks and figure out how to program utilizing hardware segments. The Raspberry Pi can likewise be incorporated with custom tasks, for example, intelligent gallery displays or home robotization arrangements.

Weight sensor

A "load cell", which is a metal bar with a hole in the center (see picture on the right). This is available for different weight classes (up to 1kg, up to 5kg, up to 50kg, etc.). Even though some have a different form, all are provided with four cables. To read out the values, the HX711 weight sensor is also required. This sensor is available in two versions: red and green.

3.3.2 Software Requirements

Apache Webserver

Apache is the most broadly utilized web server programming. Created and kept up by Apache Software Foundation, Apache is an open source programming accessible for nothing. It keeps running on 67/100 of all webservers on the planet. Be that as it may, Word Press can keep running on other web server programming also.

HTML and CSS

The two are utilized for the advancement of recreation of the lift. The two are essential Web advancements and used to make basic website pages.

Python

Python is a universally useful programming language. Henceforth, you can utilize the programming language for creating both work area and web applications. Likewise, you can utilize Python for creating complex logical and numeric applications. Python is structured with highlights to encourage information investigation and perception.

3.4 Issues In Existing Methodology

There are numerous issues in the existing methodology. The prime issue is that, the present model of elevators use a static counter weight of certain maximum capacity which is controlled by a PLC. So, every time someone uses an elevator, the whole counter weight is moved up/down using an AC electric motor. There is an immense amount of electric energy which is used to operate the elevators.
Thus, leading to wastage of power every single day. Also, if there is a power outage in a certain organization, this affects the working of the elevators too.

3.5 New Methodology

It is obvious that during a power outage our life comes to a halt. In order to access power continuously, we should also conserve energy at all costs. This new method is a befitting example to save our precious electrical energy which can be used in other useful areas. Also, the cost of the existing elevators is too much, which will be required.

CHAPTER 4

SYSTEM DESIGN

4.1 Algorithm

4.1.1 Knapsack Algorithm

The knapsack issue or knapsack issue is an issue in combinatorial advancement: Given a lot of things, each with a weight and esteem, decide the quantity of everything to incorporate into a gathering so the all-out weight is not exactly or equivalent to a given breaking point and the absolute esteem is as vast as would be prudent. It gets its name from the issue looked by somebody who is obliged by a fixed-estimate backpack and should fill it with the most profitable things.

The rucksack issue has been examined for over a century, with early works dating as far back as 1897. The name "backpack issue" goes back to the early works of mathematician Tobias Dantzig (1884– 1956) and alludes to the typical issue of pressing the most important or valuable things without over-burdening the gear.

4.1.2 New Algorithm

A basic arrangement is to consider each pair and monitor nearest pair (supreme contrast between pair whole and x is least). At last print the nearest pair.s

1. Initialize a variable diff as unending (Diff is utilized to store the

   contrast among pair and x). We have to locate the base diff.

2. Introduce two record factors left and right in the given arranged cluster.

   (an) Initialize first to the furthest left record: left = 0

   (b) Initialize second the furthest right list: right = n-1

3. Circle while left < right.

(an) If abs(ar[left] + ar[right] - total) < diff at that point

update diff and result

(b) Else if(ar[left] + ar[right] < entirety) then left++

(c) Else right- -

4.2 Modules and Functionalities

4.2.1 Simulation

The whole idea is exhibited and it is working through a simulation of the idea using HTML, CSS and JavaScript. The working of the elevator is shown by demonstrating a visual elevator and floor selection panel, which can be used by the user to enter the desired floor he wants to reach. The weight present in the elevator on the left side corner and the working of the weight slabs according to the movement of the elevator will be shown.

4.2.2 Algorithm/Weights Slot Arrangement

In this module, the main motive is to get the perfect weight which will be used for the correct movement of the elevator as per the given constraints. Right counter weight slot and left counter weight slot which will comprise of the weight slabs as per the user requirements, and the algorithm will be used to put the correct weights in the slots.

4.2.3 Setting the Web server

As defined the raspberry pi will be used to sense the weight present on the load cell. The received data will be sent from the raspberry pi to the web server that is set up. Then, the data will be used to show it on the GUI part and also to calculate the weight slot arrangement.

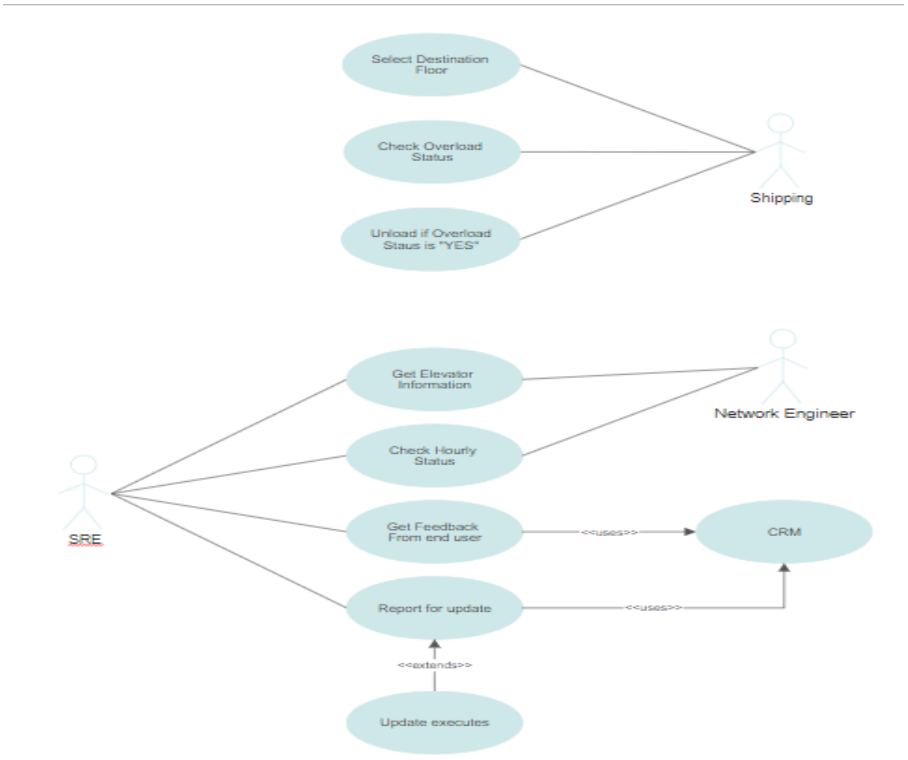
4.2.4 Coupling of the components

This includes coupling of the web server and the weight sensing module. System will use the data from the weight sensing module which basically collects the data which is the weight present in the elevator and then using the raspberry pi will send it to the web server so that it can show the weight on the screen and also calculate the weight required for the further operation.

4.2.5 Final result

Final result will be shown on the screen after computing the weight present in the elevator and then calculating the required counter weight for the further operation of going up or down.

4.3 Use Case Diagram



CHAPTER 5

CODING AND TESTING

5.1 Coding

5.1.1 index.py

```
#! /usr/bin/python2

import time
import sys

EMULATE_HX711=False

if not EMULATE_HX711:
    import RPi.GPIO as GPIO
    from hx711 import HX711
else:
    from emulated_hx711 import HX711

def cleanAndExit():
    print "Cleaning..."

    if not EMULATE_HX711:
        GPIO.cleanup()

    print "Bye!"
    sys.exit()


while True:
    try:

        if False:
            np_arr8_string = hx.get_np_arr8_string()
            binary_string = hx.get_binary_string()
            print binary_string + " " + np_arr8_string
        val = max(0,int(hx.get_weight(5)))
    #val = hx.read_long()
        print (val-485)
    f=open("load.php","w")
    f1=open("load.txt","w")
    f.write("%d \n" % (val-485))
```

```
    f1.write("%d \n" % (val-485))
    f.close()
    f1.close()
        hx.power_down()
        hx.power_up()
        time.sleep(0.1)
        f = open('output1.txt')
        fc = f.readline()
        f.close()
        fc=int(fc)
        T_force=(val-470)
        was=[10,50,100,200,400,800]
        MAX_VAL = 1000
        n=len(was)
        def printClosest(arr, n, x):
            re_left, re_right = 0, 0
            left, right, diff = 0, n-1, MAX_VAL
            while right > left:
                if abs(arr[left] + arr[right] - x) < diff:
                    re_left = left
                    re_right = right
                    diff = abs(arr[left] + arr[right] - x)

                if arr[left] + arr[right] > x:
                    right -= 1
                else:
                    left += 1
            return [arr[re_left], arr[re_right]]
        res=[]
        res=printClosest(was, n, T_force)
        print(res)
        rcwa=0
        lcwa=0
        if T_force>=900:
            err_mes="Yes"
            fx=open("error1.php","w")
            fx.write("%s" % err_mes)
        fx.close()
        else:
            err_mes="No"
            fx=open("error1.php","w")
            fx.write("%s" % err_mes)
        fx.close()
            if fc<0:
                if T_force>10:
                    lcwa=100
                else:
                    lcwa=10
                rcwa=res[0]+res[1]+150
                s1=open("slotr1.php","w")
                s2=open("slotr2.php","w")
```

```
                    s3=open("slotr3.php","w")
                s1.write("%d" % res[0])
                s2.write("%d" % res[1])
                s3.write("%d" % 150)
                s1.close()
                s2.close()
                s3.close()
                 s4=open("slotl1.php","w")
                 s5=open("slotl2.php","w")
                 s6=open("slotl3.php","w")
                s4.write("0")
                s5.write("0")
                s6.write("%d" % lcwa)
                s4.close()
                s5.close()
                s6.close()
                f2=open("loadl.php","w")
                f2.write("%d" % lcwa)
                f2.close()
                f3=open("loadr.php","w")
                f3.write("%d" % rcwa)
                f3.close()
            if fc>0:
                if T_force>10:
                    rcwa=100
                else:
                    rcwa=10
                lcwa=res[0]+res[1]+150
                 s1=open("slotl1.php","w")
                 s2=open("slotl2.php","w")
                 s3=open("slotl3.php","w")
                s1.write("%d" % res[0])
                s2.write("%d" % res[1])
                s3.write("%d" % 150)
                s1.close()
                s2.close()
                s3.close()
                 s4=open("slotr1.php","w")
                 s5=open("slotr2.php","w")
                 s6=open("slotr3.php","w")
                s4.write("0")
                s5.write("0")
                s6.write("%d" % rcwa)
                s4.close()
                s5.close()
                s6.close()
                f2=open("loadl.php","w")
                f2.write("%d" % lcwa)
                f2.close()
                f3=open("loadr.php","w")
                f3.write("%d" % rcwa)
```

```
                f3.close()

    except (KeyboardInterrupt, SystemExit):
        cleanAndExit()
```

5.1.2 index.php

```
<!DOCTYPE html>
<html lang="en" >

<head>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.
       com/ajax/libs/meyer-reset/2.0/reset.min.css">
    <link rel="stylesheet" href="css/style.css">
    <title>Elevator</title>
    <link href="//fonts.googleapis.com/css?family=Raleway
       :400,300,600" rel="stylesheet" type="text/css">
    <link rel="stylesheet" href="css/normalize.css">
    <link rel="stylesheet" href="css/skeleton.css">


</head>
<style>
#myContainer {
  width: 20%;
  height: 500px;
  margin: 0 auto;
  overflow: hidden;
}
#myAnimation {
  width: 100px;
  height: 70px;
  position: absolute;
  background-color: black;
}
#myAnimation1 {
```

```
  width: 100px;
  height: 70px;
  position: absolute;
  background-color: black;
  top: 648px;
  margin-left: 204px;
}
.text2{
text-align:center;
float:right;
position:relative;
top:-600px;
left:-100px;
border-width: unset;
border-style: solid;
border-color: black;
}
.text1{
    text-align: center;
    float: left;
    position: relative;
    top: -200px;
    right: -89px;
border-width: unset;
border-style: solid;
border-color: black;
}
.text3{
text-align: center;
    float: left;
    position: relative;
    top: -50px;
    right:  235px;
border-width: unset;
```

```
border-style: solid;
border-color: black;
}
.text4{
text-align: center;
float: right;
position: relative;
top: -450px;
left: 310px;
border-width: unset;
border-style: solid;
border-color: black;
}
.text4 div{
    margin: 10px;
}
.text5{
text-align: center;
float: right;
position: relative;
top: -246px;
left: 756px;
border-width: unset;
border-style: solid;
border-color: black;
}
.text5 div{
    margin: 10px;
}
</style>
<?php
$file="load.txt";
if(file_exists($file)) {
    $line = fgets(fopen($file, 'r'));
```

```
}
?>
<body style="background-color: lavender;">

  <div id="canvas">
    <p style="color:black;">SAVE POWER! SAVE EARTH! Concept
       by VIJAY ROHIT and AKSHAT</p>

   <div id="floorSelect">
     <h4>Select your floor</h4>
     <ul>

        <li data-floor="0" value="0" onclick="myMove()">G</
           li>
       <li data-floor="1" value="1" onclick="myMove()">1</
          li>
        <li data-floor="2" value="2" onclick="myMove()">2</
           li>
        <li data-floor="3" value="3" onclick="myMove()">3</
           li>
        <li data-floor="4" value="4" onclick="myMove()">4</
           li>

     </ul>

   </div>
  <div id="building">

    <div id="elevatorShaft">


      <div id="elevatorContainer" >
        <div id="strings"></div>
          <div id="elevator">
```

```
124             <div class="door active-left" id="leftDoor"></
                   div>
125             <div class="door active-right" id="rightDoor"></
                   div>
126
127
128          </div>
129
130
131       </div>
132
133
134   </div>
135
136 </div>
137   <div id ="myContainer">
138         <div id ="myAnimation"></div>
139         <div id ="myAnimation1"></div>
140       </div>
141 </div>
142     <script src='http://cdnjs.cloudflare.com/ajax/libs/
          jquery/2.1.3/jquery.min.js'></script>
143   <script src="js/test.js"></script>
144 <br>
145     <div class="text1" style=""><h3>WEIGHT INSIDE LIFT IS:
          <div id="auto"></div></h3></div>
146     <script type="text/javascript" src="jquery-3.3.1.js"></
          script>
147     <script type="text/javascript" src="cript.js"></script>
148     <div class="text2" style=""><h4>LEFT COUNTER WEIGHT :
          <div id="auto1"></div></h4><h4>RIGHT COUNTER WEIGHT:
          <div id="auto2"></div></h4></div>
149     <div class="text3"><h4>OVERLOAD STATUS:<div id="auto3"><
          /div></h4></div>
```

```
150     <div class="text4">
151       <h4> RIGHT COUNTER WEIGHT SLOTS: </h4>
152         <div id="auto4" style="background-color:black;color:
              white;"></div>
153         <div id="auto5" style="background-color:black;color:
              white;"></div>
154         <div id="auto6" style="background-color:black;color:
              white;"></div>
155
156     </div>
157     <div class="text5">
158       <h4> LEFT COUNTER WEIGHT SLOTS: </h4>
159         <div id="auto7" style="background-color:black;color:
              white;"></div>
160         <div id="auto8" style="background-color:black;color:
              white;"></div>
161         <div id="auto9" style="background-color:black;color:
              white;"></div
162
163     </div>
164     <script type="text/javascript" src="jquery-3.3.1.js"></
          script>
165     <script type="text/javascript" src="cript1.js"></script>
166     <script type="text/javascript" src="cript2.js"></script>
167     <script type="text/javascript" src="cript3.js"></script>
168         <script type="text/javascript" src="cript41.js"></
              script>
169         <script type="text/javascript" src="cript51.js"></
              script>
170         <script type="text/javascript" src="cript61.js"></
              script>
171     <script type="text/javascript" src="cript71.js"></
          script>
172         <script type="text/javascript" src="cript81.js"></
```

```
            script>
173         <script type="text/javascript" src="cript91.js"></
            script>
174
175 </body>
176
177 </html>
```

5.1.3 cript.js

```
1  $(document).ready( function(){
2  $('#auto').load('load.php');
3  refresh();
4  });
5  function refresh()
6  {
7    setTimeout( function() {
8      $('#auto').fadeOut('slow').load('load.php').fadeIn('slow
           ');
9      refresh();
10   }, 3000);
11 }
```

5.1.4 cript1.js

```
1  $(document).ready( function(){
2  $('#auto1').load('loadl.php');
3  refresh2();
4  });
5  function refresh2()
6  {
7    setTimeout( function() {
8      $('#auto1').fadeOut('slow').load('loadl.php').fadeIn('
           slow');
9      refresh2();
```

```
  }, 3000);
}
```

5.1.5 cript2.js

```
$(document).ready( function(){
$('#auto2').load('loadr.php');
refresh3();
});
function refresh3()
{
  setTimeout( function() {
    $('#auto2').fadeOut('slow').load('loadr.php').fadeIn('
       slow');
    refresh3();
  }, 3000);
}
```

5.1.6 cript3.js

```
$(document).ready( function(){
$('#auto3').load('error1.php');
refresh4();
});
function refresh4()
{
  setTimeout( function() {
    $('#auto3').fadeOut('slow').load('error1.php').fadeIn('
       slow');
    refresh4();
  }, 3000);
}
```

5.1.7 cript41.js

```
$(document).ready( function(){
$('#auto4').load('slotr1.php');
refresh5();
});
function refresh5()
{
  setTimeout( function() {
    $('#auto4').fadeOut('slow').load('slotr1.php').fadeIn('
       slow');
    refresh5();
  }, 3000);
}
```

5.1.8 cript51.js

```
$(document).ready( function(){
$('#auto5').load('slotr2.php');
refresh6();
});
function refresh6()
{
  setTimeout( function() {
    $('#auto5').fadeOut('slow').load('slotr2.php').fadeIn('
       slow');
    refresh6();
  }, 3000);
}
```

5.1.9 cript61.js

```
$(document).ready( function(){
$('#auto6').load('slotr3.php');
refresh7();
});
function refresh7()
```

```
{
  setTimeout( function() {
    $('#auto6').fadeOut('slow').load('slotr3.php').fadeIn('
       slow');
    refresh7();
  }, 3000);
}
```

5.1.10 cript71.js

```
$(document).ready( function(){
$('#auto7').load('slotl1.php');
refresh8();
});
function refresh8()
{
  setTimeout( function() {
    $('#auto7').fadeOut('slow').load('slotl1.php').fadeIn('
       slow');
    refresh8();
  }, 3000);
}
```

5.1.11 cript81.js

```
$(document).ready( function(){
$('#auto8').load('slotl2.php');
refresh9();
});
function refresh9()
{
  setTimeout( function() {
    $('#auto8').fadeOut('slow').load('slotl2.php').fadeIn('
       slow');
    refresh9();
```

```
10    }, 3000);
11  }
```

5.2 Testing

Testing means to undergo all the parameters that examine a particular project for the errors. Basically, it is a process of trying to discover every fault present in our system. It provides a way to examine the functionality of each module present in the project. It also checks each assembly and sub-assemblies in the project.

5.2.1 White-Box Testing

White Box Testing is a method to test a particular project by going through the knowledge of its whole inner working and structure. It is also used to check the language of the software and at least what purpose it serves. It is used to test all the areas that cannot be reached from a black box level.

5.2.2 Black-Box Testing

It is a method of testing particular software without the knowledge of its inner working and structure. Also, without the knowledge of its language and its purpose. There is a special requirement in black box testing, it can be written from a definitive source document such as to requirement document. As the name suggests, it is a testing method in which the software is treated as a black box which cannot be seen from outside. The tests give output without taking the notice of the fact about how the software works.

5.2.3 Alpha Testing

In the software development process, alpha test is a kind of test which is among the teams to confirm that your software is in working condition or not. To begin with, the term alpha relates to the first phase of testing in a software development process. The first phase comprises of unit testing, component testing and system testing. It also

enables the tester to test the product on the lowest common denominator machines and make sure download times are accepted.

5.2.4 Beta Testing

In the software development process, beta testing is the second phase in the software testing cycle. In which a sampling of the audience tries to test the product out. Beta testing basically means pre-release of the beta testing versions of the software are distribute to curriculum specialists and teachers and give the program a real-world test.

CHAPTER 6

CONCLUSION

In this project, the old conventional way of using the elevator will be challenged with a new method. There is a lot of electricity consumed while using a conventional elevator. So, a new method of using elevators is developed which will not use electricity and will majorly focus on the gravity-based dynamic counterweight system. A new paddle system is introduced which will use dynamic counterweight for the movement of the container which holds the passengers.

So, an algorithm was developed which is a subset of the knapsack algorithm and will deduce a pair of weights which is nearest to the load present in the container. The algorithm was developed using python language. Initially, getting an array of weights using the algorithm but instead of that in the revised version, it was decided to get a pair of weights in counter to the present weight in the elevator.

The simulation part which will manifest our idea will have an elevator system in the middle and floors up to Ground+4 floors. The overload condition which has a limit till 1000kgs is also shown. Also, the right counterweight and the Left counterweight showing the weights to counter the load present in the elevator. The simulation works perfectly fine and depicts all the conditions which are provided.

Overall, all the modules which are Simulation, Raspberry Pi, Weight Sensor, Algorithm, Presentation works perfectly. All the conditions including overload status and basic operations demonstrate our idea. The goal of the project was achieved.

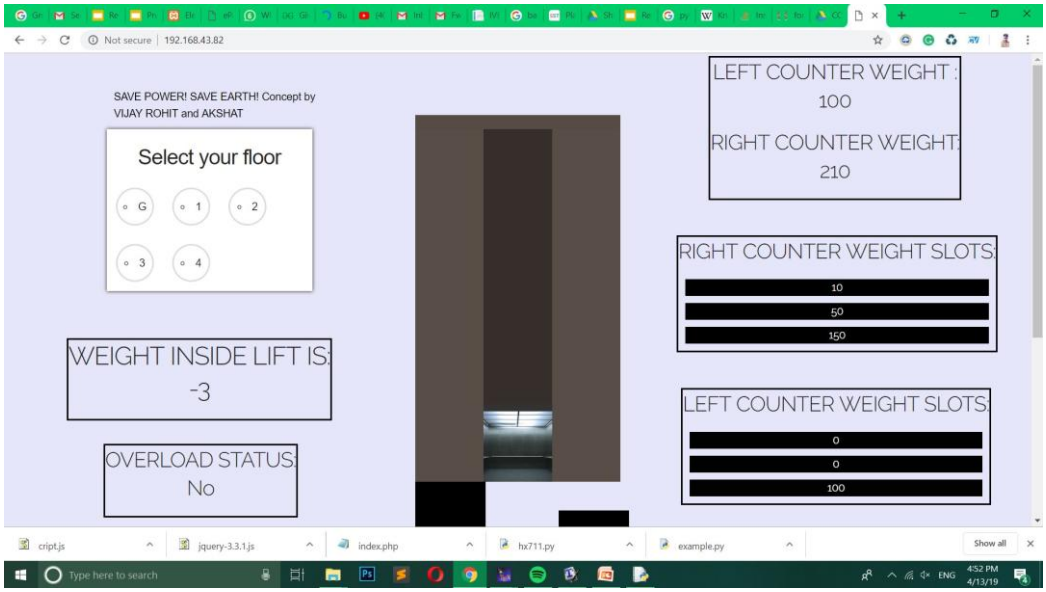


Figure 6.1: Empty Elevator at G Floor

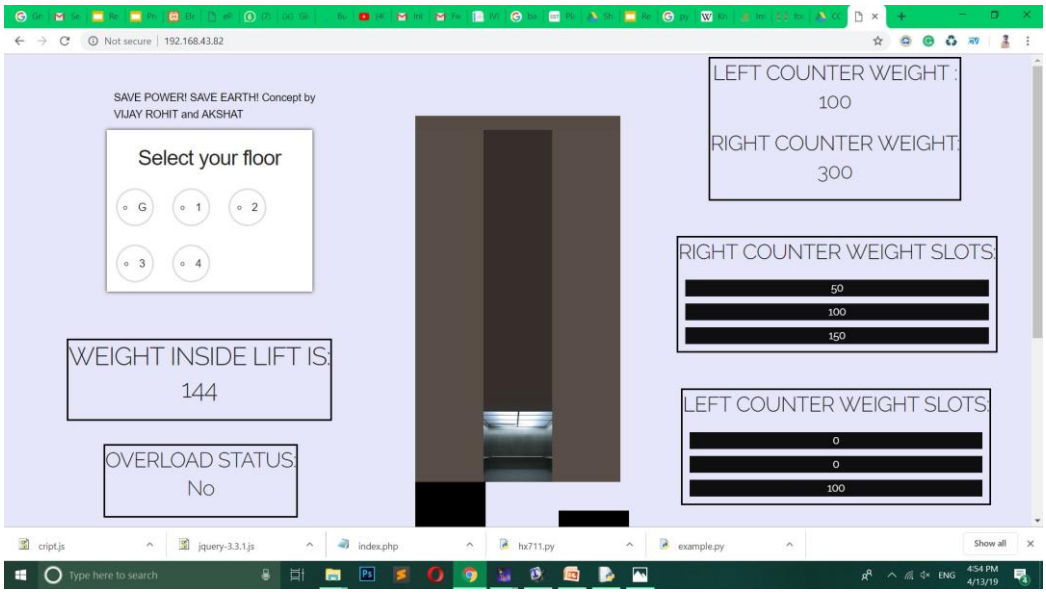


Figure 6.2: Weight Included

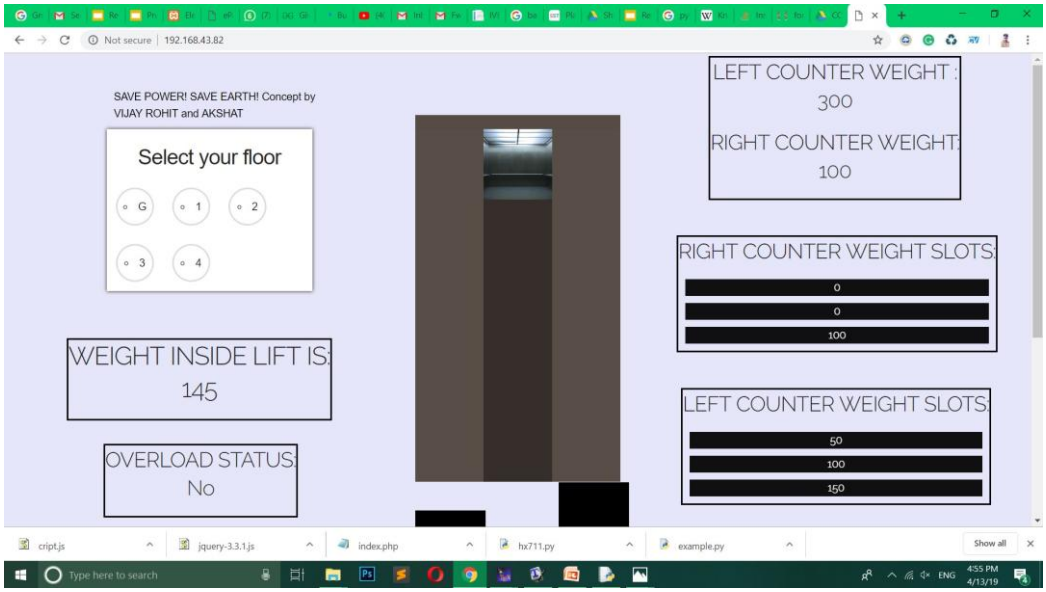


Figure 6.3: From G Floor to 4th Floor

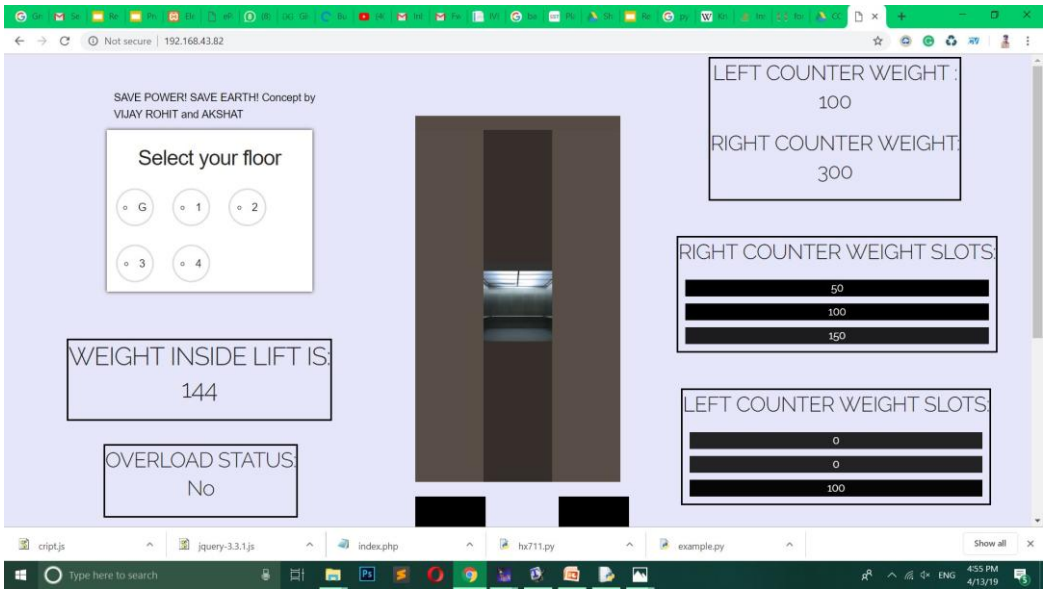


Figure 6.4: From 4th Floor to 2nd Floor

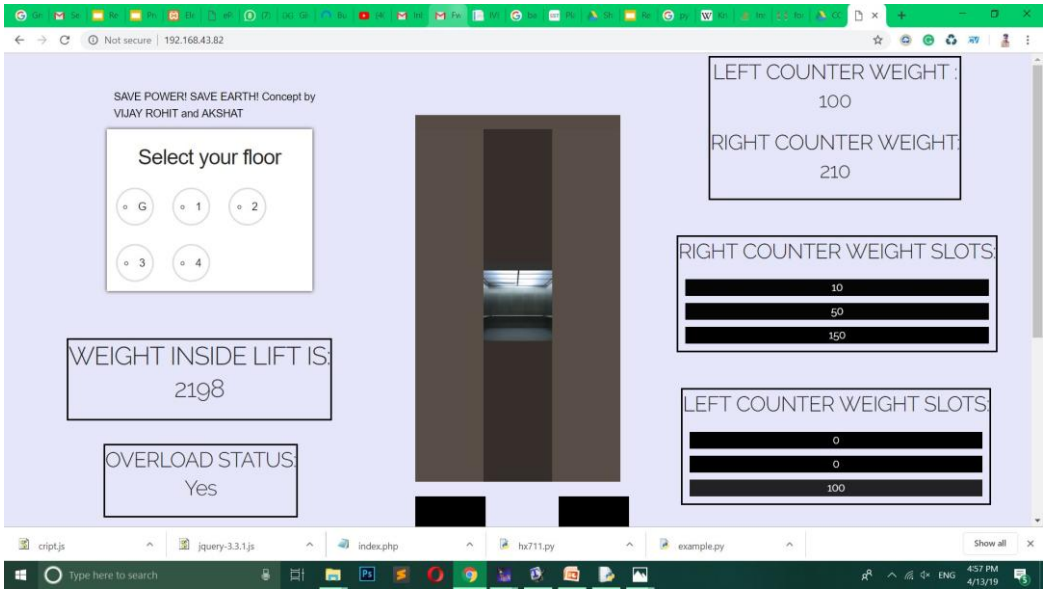


Figure 6.5: Overload Condition

REFERENCES

1. Anusha, A. M. and Sutagundar, A. V. (2015). “Design of a plc based elevator control system.” *International Journal on Emerging Technologies*, NA, 32–36.

2. Carter, S. R. and A.Selvaraj (2013). “Design and implemention of plc based elevator.” *International Journal of Computer Applications*, 68, 10–13.

3. et.al, T. (2011). “Application of plc for elevator control system.” *International Symposium on Devices MEMS, Intelligent Systems Communication*, 4, 267–274.

4. T. E. Barnum, "Elevator control," in Proceedings of the American Institute of Electrical Engineers, vol. 30, no. 7, pp. 1583-1603, July 1911.

5. Y. Zhang, Y. Yi and J. Zhong, "The Application of the Fuzzy Neural Network Control in Elevator Intelligent Scheduling Simulation," 2010 Third International Symposium on Information Science and Engineering, Shanghai, 2010, pp. 512-516.

6. S. Sato, T. Uchida, N. Kobayashi and T. Nakagawa, "Evaluation of an Elevator Emergency Stop Device With a Magnetorheological Fluid Damper Controlled in Conformity With the Elevator Safety Guide," in IEEE Transactions on Magnetics, vol. 51, no. 11, pp. 1-4, Nov. 2015, Art no. 4600304.

7. W. K. Tou, M. I. Vai and S. U. Cheang, "Elevator absolute landing position sensor using infrared interrupter and MEMS accelerometer," 2015 5th Australian Control Conference (AUCC), Gold Coast, QLD, 2015, pp. 366-370.

8. Z. Pan, F. Luo and Y. Xu, "Elevator Traffic Flow Model Based On Dynamic Passenger Distribution," 2007 IEEE International Conference on Control and Automation, Guangzhou, 2007, pp. 2386-2390.

Final Report

ORIGINALITY REPORT

2%	1%	2%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Kurt Cagle. "Chapter 10 SVG Components", Springer Nature, 2002 Publication	1%
2	www.tutorialswb.com Internet Source	<1%
3	Chris Apers, Daniel Paterson. "Beginning iPhone and iPad Web Apps", Springer Nature, 2010 Publication	<1%
4	www.sparknotes.com Internet Source	<1%
5	dev.ctor.org Internet Source	<1%
6	Submitted to University of Bedfordshire Student Paper	<1%

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		