iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** IV **Month of publication:** April 2022

DOI: https://doi.org/10.22214/ijraset.2022.41307

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

Handicap Pickup and Drop System

Pratik Khawale[1], Himayatulla Qazi[2], Abdal Malik[3], Akshay Barua[4]
[1, 2, 3, 4]*Mehanical Engineering, Maharashtra State Board of Technical Education*

Abstract: ***The project titled "Handicap Pickup And Drop System" is a ramp travelling mechanism which can be used in railway station, metros, airports, and on other transportation system where there is no proper facility for the handicap and senior citizen on the platforms of the transportation system. The main elements of the project are sensors, micro – controller, conveyer belt, DC motor. Micro controller can be said as the heart of the mechanism as it controllers all the various functions and programs of the mechanism. The project deals with the automation of process for the handicap and senior citizen in their transportation system. This system gives a fast and effortless operation for the individual. The controller for this system is supplied by a single phase AC supply. The program defines the kind of task to be performed on the system corresponding to the inputs. This control program is feed into using a program device such as PC. Sensors are used to sense the presence of the individuals position. The signal from the sensor is as input to the controller. The controller gives the control signal to the actuators in the conveyer to move the chair to the appropriate position. The whole process can be done in a single operation and can be stopped after the process is completed. Thus this system gives a faster and reliable control to reduce the effort and ease the travelling experience for the handicap or the senior citizen. In the current scenario if a handicap or the senior citizen wants to go the train, it was a nightmare because first he/ she has to take the ticket from the ticket counter and then go to the appropriate compartment by himself or with the help of someone. The micro-controller controls the complete working of the system. It is common to use microcontrollers to make simple logical control decision.***

I. INTRODUCTION

In todays world as we travel in daily basis we can see the problems face by senior citizens and physically challenged people. As our country is still lacking on this things. To overcome this problem we have created a miniature model which can be used in railway stations, metro stations, monorail stations, bus stations, etc.

According to the Sensex which was held in 2011 there are nearly about 104 million senior citizens which are above 60 years. The senior citizens consist of 53 million females and 51 million males. Which is about 8.6% of India's total population. And considering about the physically challenged people it adds to about 10.7% of total India's population. As per the laws of India Physically Challenged people get quotas in colleges, schools, jobs, they also get scholarships. But the problem face by them in everyday life while travelling is not considered. Similarly Senior citizens who were ex-service man in government offices get their pensions. But the problems faced by them is on stations, bus stops, etc.

As we know that they have reserved seats in buses and reserved compartments in trains but what's the use if they cannot reach to their desired buses and trains. This can be surely implemented by our Indian Government in Future. Our Project will be a two floor station. It will consists of two control room, 1 wash room, 1 ticket counter, Food court. Which will be placed beside the moving ramp on the ground floor.

The ground floor and the 1st floor will be connected by a escalator for the handicap and senior citizen to rear the upper floors. On the upper floor there will be a control room controlling the system on the upper floor.

As the commuter reaches the control room the Individual will ask for the desired place for eg. Washroom, train station, food court, ticket counter, etc. The escalator will be connected to the moving ramp. As the end of the moving ramp there will be another ramp – 2 which will be controlled by the control room. As the train reaches the platform the controller will start the ramp. Similarly as the train leaves the platform the controller will turn the ramp – 2 off.

A. Project Statement

Control system implies that direct with the physical world. Control system include sensor and actuators, the critical pieces needed to ensure that our automatic system can help us manage our activates and environments in desired ways.

The sensor that will be implemented in this project is the infrared sensor. Now, infrared sensor is widely used in daily activities such as for security, motion detection and other purposes.

The infrared sensor consists of two basic parts, emitter and detector. In project, infrared sensor that is used to detect motion is used. The sensor detect the individual, the motor of conveyer will be stopped automatically.

Problem statement in this project are:

1) To design a system which can control conveyer by infrared sensor.
2) To design a chair which will moves in and out of the lift.

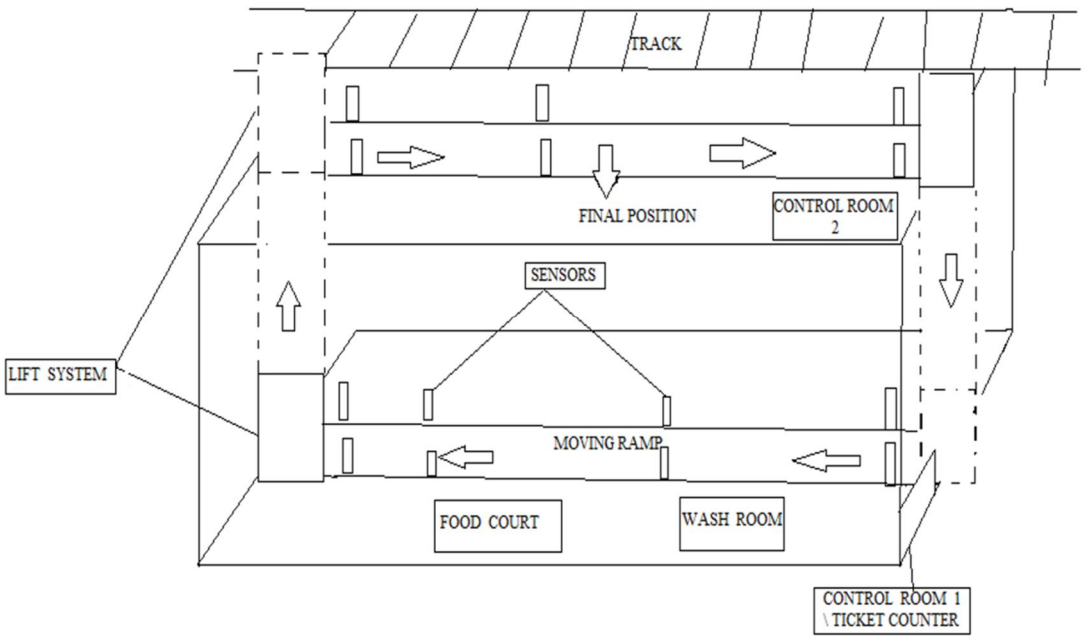
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

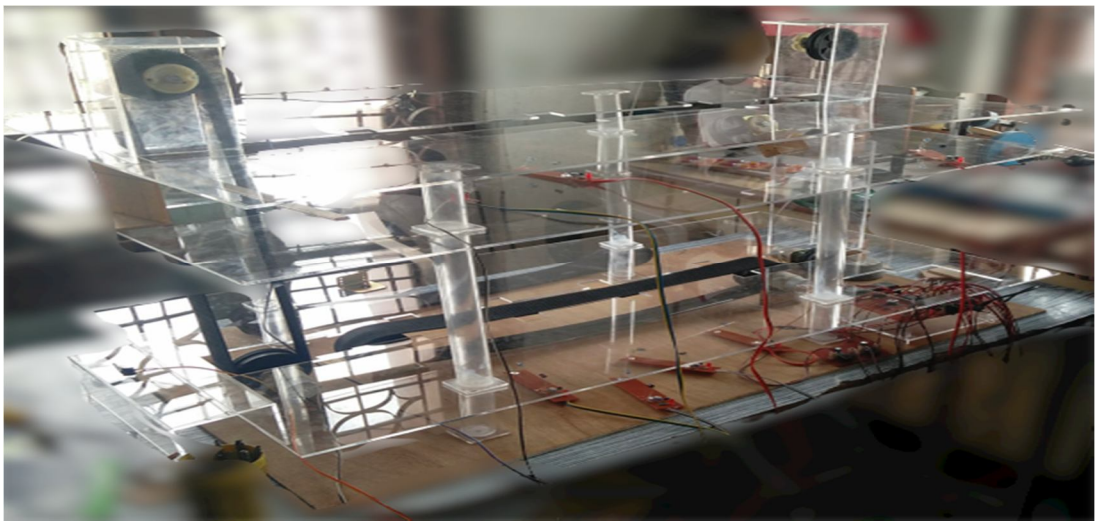
II. OBJECTIVE

The main objective of the project is to apply Micro Controller to design automatic pickup and drop system for the handicap and senior citizen. The fascination and wide application of Microcontroller has motivated to discover more about Microcontroller. To drive the main objective, there are several supporting goals need to be achieved as listed below.

1) To pickup and drop the handicap or senior citizen using Microcontroller SST 89E516RD.
2) To develop an infrared sensor to detect the position of the bottle.
3) To build a prototype which are light, low cost, user friendly and with transparent structure.

III. CONSTRUCTION

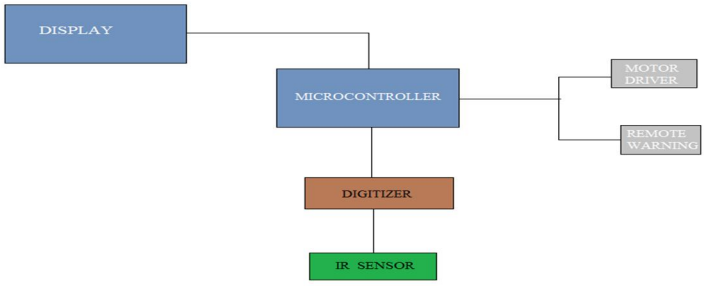




This figure shows the layout of the automated pickup and drop system for the handicap and senior citizen pickup and drop system. The conveyer will stop automatically after the infrared sensor detect the presence of the individual. The controller will switch on the conveyer system when the individual presses the start button on the remote. And the message will show on the display showing the position of the individual. This processes works continuously. And if the individual has any problem then if he presses stop the emergency message will show on the display and buzzer starts. The Handicap and Senior Citizen pickup and drop system will have the following components.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

IV. WORKING



In the automated pickup and drop system, in this when the route is selected, and the 1st sensor detects the individual then the motor starts running which gives transitional motion to the conveyer belt. The Microcontroller is used to run the Motor using motor drive system. Then the ramp stops at the particular stop which is set on the route. If the ramp doesn't stop on the position and moves further then the IR sensor detects it and show the EMERGENCY message on the display. The position of the individual is monitored and shown on the display. It involves assembly of all the components listed above. A simple conveyer belt will be stretched between two rollers, one roller which pushes it and other roller which pulls belt as the belt moves. The conveyer belt and rollers are accommodates on the fibre frame. The Infrared Sensors is fixed at certain position on the table frame in such way that it is able to detect the individual.
In this system when the route is selected, the motor starts running which gives translational motion to the conveyer belt. The microcontroller is used to continue to run the motor using motor drive system. And the parallel task is done by the infrared sensor. If during the process the individual has any problem then he has to press the button on the remote and the EMERGENCY message will show on the display. As the precaution the motor drive system are halted. The system also monitors and displays all the sensors parameters in real time.
It involves assembly of all the components listed above. A simple conveyer belt will be stretched between two rollers, one roller which pushes it and other which pushed the belt as the belt moves. The conveyer belt roller are accommodated on table frame. The Infrared sensor is fixed at certain position on the table frame in such a way that it able to detect the chair. When the individual enter from the entry position he will take the ticket and say where he wants to go, the ticker counter volunteer will set the required path R1, R2, R3.
The individual will sit on the chair and the ramp starts moving, first it stops on the washroom and the ramp will stop . And further it is decreed by the IR sensor. Second it stops on the Food Court, as the chair reached the ramp stops. The chair don't stop then the IR sensor will detect it and the message will show on the display.
Further there is a lift where the chair will move up. There is another ramp system on the top floor. And the chair will stop at the final position, If the individual that comes out from the train want to go to the exit. He will sit on the empty chair and further a lift will bring him down and in front of the exit.

V. COMPONENTS

Name of the Components And their Specification :-

1) Conveyer Belt
2) DC Motor
3) Infrared Sensor
4) Microcontroller
5) 16 X 2 LCD
6) Pushed On Switch
7) Reliment Connector 2 pin
8) Resistor 330 X3
9) Resistor 10k XL
10) Ceramic Resistor
11) Reliment Connector
12) Buzzer
13) Voltage Regulator
14) Filtration Capacitor

15) Capacitor
16) Rf module encoder
17) Reliment Connector 3 pin
18) Rf receiver
19) Variable Resistor
20) Limit Switch
21) Pulley
22) Motor Connector 100 uF 50V capacitor
23) Motor Driver Capacitor
24) Fiber sheets

VI. SAFETY PRECAUTIONS

A. Prevent the electrical circuit from water
B. Check the wire connections properly before starting the system.
C. Don't allow the wires to come in contact with the moving roller and metal base.
D. Conveyer belt should be properly attached with the rollers.
E. Prevent direct contact from circuit when it is in working condition.
F. Don't overload the roller conveyer.
G. The Infrared Sensor should be placed in proper location.
H. The IR Sensor, emitter and recover should be in front of each other.
I. Prevent the system from direct contact of sun as the IR Sensors will absorb the UV rays and not work properly.

VII. CODING

```
#include <at89x52.h>
#include <Lcd_16_2_4Bit_mode_at_upper_bits_8051.h>
# include <string.h>
sbit limit_SW_1=P0^0;
sbit limit_SW_2=P0^1;
sbit limit_SW_3=P0^2;
sbit limit_SW_4=P0^3;
sbit SW_1=P0^4;
sbit SW_2=P0^5;
sbit SW_3=P0^6;
sbit Buzzer=P0^7;
sbit VT=P1^0;
sbit D_1=P1^1;
sbit D_2=P1^2;
sbit IR_1=P1^3;
sbit IR_2=P1^4;
sbit IR_3=P1^5;
sbit IR_4=P1^6;
sbit IR_5=P1^7;
sbit IR_6=P3^0;
sbit IR_7=P3^1;
sbit MA_1=P3^2;
sbit MA_2=P3^3;
sbit MB_1=P3^4;
sbit MB_2=P3^5;
sbit MC_1=P2^0;
sbit MC_2=P2^1;
sbit MD_1=P2^2;
sbit MD_2=P2^3;

bit Rout_1=0,Rout_2=0,Rout_3=0;

void Rout_select();
```

```
void SM_ARU_process_1();
void SM_ARU_process_2();
void SM_ARU_process_3();
void SM_ARU_process_4();
void SM_ARU_process_5();
void emergency_loop();

void delay(unsigned int msec)
{
int i,j ;
for(j = 0;j<msec;j++)
                                        {
                                              for(i = 0;i < 120;i++)
                                              {
                                                        emergency_loop();
                                              }
                                        }
}
void main()
{
                                        P1=0XFF;
                                        P0=0xFF;
                                        P3=0XFF;
                                        P2=0xF0;
                                        MA_1=0;
                                        MA_2=0;
                                        MB_1=0;
                                        MB_2=0;
                                        lcdinit();
                                        Buzzer=0;
                                        delay(200);
                                        writecmd(0x01);
                                        lcd_string("Handicap Pick &");
                                        writecmd(0xc0);
                                        lcd_string("   Drop System");
                                        delay(5000);
                                        writecmd(0x01);

                                        while(1)
                                              {
                                                        Rout_select();
                                                        SM_ARU_process_1();
                                                        SM_ARU_process_2();
                                                        SM_ARU_process_3();
                                                        SM_ARU_process_4();
                                                        SM_ARU_process_5();

                                              }
}
void Rout_select()
{
                                        Rout_1=Rout_2=Rout_3=0;
                                        while(Rout_3==0)
                                        {
                                                        writecmd(0x80);
                                                        lcd_string(" Select Route");
                                                        if(SW_1==0)Rout_1=~Rout_1;
```

```
                        if(SW_2==0)Rout_2=~Rout_2;
                        if(SW_3==0)Rout_3=~Rout_3;

                        if(Rout_1==1)
                        {writecmd(0xc0);lcd_string("R1");}
                        if(Rout_1==0)
                        {writecmd(0xc0);lcd_string("  ");}

                        if(Rout_2==1)
                        {writecmd(0xc5);lcd_string("R2");}
                        if(Rout_2==0)
                        {writecmd(0xc5);lcd_string("  ");}

                        if(Rout_3==1)
                        {writecmd(0xca);lcd_string("R3");}
                        if(Rout_3==0)
                        {writecmd(0xca);lcd_string("  ");}
                }
                writecmd(0x80);
                lcd_string("   Welcome   ");
                delay(2000);
        while(VT==0)
                                {
                                        writecmd(0x80);
                                        lcd_string("Please On The");

                                        writecmd(0xC0);
                                        lcd_string("Conver System  ");
                                }

                        writecmd(0x01);
                        lcd_string("Conver System  ");
                        writecmd(0xC0);
                        lcd_string("On ");
}

void SM_ARU_process_1()
{
                        while(IR_1==1){emergency_loop();}
                        //if(IR_1==0)
                                {
                                        writecmd(0x01);
                                        lcd_string("System Start");
                                        MA_1=1;
                                        MA_2=0;
                                        delay(5000);
                                        if(Rout_1==1)
                                        {
                                                delay(1000);
                                                MA_1=0;
                                                MA_2=0;
                                                writecmd(0x01);
                                                lcd_string("Wash Room");
                                                writecmd(0xc0);
                                                lcd_string("Press S To Start");
                                                delay(1000);
```

```
                                                        while(D_1==1)
                                                        {
                                                                if(VT==0)
                                                                {
                                                                        writecmd(0x01);
                                                                        while(VT==0)
                                                                        {
                                                                        writecmd(0x80);
                                                                        lcd_string("No
Communication");
                                                                        }
                                                                }
                                                                else
                                                                {

                                                writecmd(0xc0);
                                                                        lcd_string("Press S To Start");
                                                                }
                                                        }
                                                        writecmd(0x01);
                                                        lcd_string(" System Start");
                                                        delay(1000);
                                                        MA_1=1;
                                                        MA_2=0;
                                                        Rout_1=0;
                                                }
                                        }
                                        while(IR_2==1);
//                                      if(IR_2==0 && Rout_1==0)
//                                      {
                                        if( Rout_1==0)
                                        {
                                                writecmd(0x01);
                                                lcd_string("Cross Point 1");
                                                delay(1000);
                                        }
                                        if(Rout_1==1)
                                        {
                                                MA_1=0;
                                                MA_2=0;
                                                writecmd(0x01);
                                                lcd_string("System Error !!!");
                                                while(1)
                                                {
                                                        Buzzer=1;
                                                        delay(1000);
                                                        Buzzer=0;
                                                        delay(1000);
                                                }
                                        }
}

void SM_ARU_process_2()
{
                                                MA_1=1;
                                                MA_2=0;
                                                if(Rout_2==1)
```

```
                {
                        delay(1000);
                        MA_1=0;
                        MA_2=0;
                        writecmd(0x01);
                        lcd_string("Food Court");
                        writecmd(0xc0);
                        lcd_string("Press S To Start");
                        while(D_1==1);
                        writecmd(0x01);
                        lcd_string(" System Start");
                        delay(1000);
                        MA_1=1;
                        MA_2=0;
                        Rout_2=0;
                }

        while(IR_3==1){emergency_loop();}
        //if(IR_3==0 && Rout_2==0)
        {
                if( Rout_2==0)
        {
                writecmd(0x01);
                lcd_string("Cross Point 2");
                delay(1000);
        }
}
        if(IR_3==0 && Rout_1==1)
        {
                MA_1=0;
                MA_2=0;
                writecmd(0x01);
                lcd_string("System Error !!!");
                while(1)
                {
                        Buzzer=1;
                        delay(1000);
                        Buzzer=0;
                        delay(1000);
                }
        }
}

void SM_ARU_process_3()
{
        while(IR_4==1){emergency_loop();}
        writecmd(0x01);
        lcd_string("Loop 1 Completed");
        MA_1=0;
        MA_2=0;
        delay(3000);
        writecmd(0x01);
        lcd_string("Loop 2 Start");
        delay(1000);
        MB_1=1;
        MB_2=0;
        while(limit_SW_1==0){emergency_loop();}
```

```
                                        MB_1=0;
                                        MB_2=0;
                                        writecmd(0x01);
                                        lcd_string("Loop 2 Completed");
                                        delay(2000);
}

void SM_ARU_process_4()
{
                                    while(IR_5==1);
                                    //if(IR_5==0)
                                        {
                                            writecmd(0x01);
                                            lcd_string("System  2 Start");
                                            MB_1=0;
                                            MB_2=1;
                                            while(limit_SW_2==0){emergency_loop();}
                                            MB_1=0;
                                            MB_2=0;
                                            delay(5000);
                                            MC_1=1;
                                            MC_2=0;
                                            if(Rout_3==1)
                                            {
                                                    Rout_3=0;
                                                    delay(1000);
                                                    MC_1=0;
                                                    MC_2=0;
                                                    writecmd(0x01);
                                                    lcd_string("Final Position");

                                                    writecmd(0xc0);
                                            }
                                        }

                                        while(IR_6==1);
                                            MC_1=1;
                                            MC_2=0;
//                                      if(IR_6==0 && Rout_3==0)
//                                      {
                                            if( Rout_3==0)
                                        {
                                            writecmd(0x01);
                                            lcd_string("Cross Point 3");
                                            delay(1000);
                                        }

//                                      if(IR_6==0 && Rout_3==1)
//                                      {
                                        if(Rout_3==1)
                                        {
                                            MC_1=0;
                                            MC_2=0;
                                            writecmd(0x01);
                                            lcd_string("System Error !!!");
                                            while(1)
                                            {
```

```
                                                Buzzer=1;
                                                delay(1000);
                                                Buzzer=0;
                                                delay(1000);
                                        }
                                }
                                while(IR_7==1);
                                MC_1=0;
                                MC_2=0;
                                writecmd(0x01);
                                lcd_string("System 2 Compilated");
                                delay(1000);
}
void SM_ARU_process_5()
{
                                writecmd(0x01);
                                lcd_string("Loop 3 Start");
                                delay(1000);
                                MD_1=1;
                                MD_2=0;
                                while(limit_SW_3==0);
                                MD_1=0;
                                MD_2=0;
                                writecmd(0x01);
                                lcd_string("Loop 3 Completed");
                                delay(1000);
                                MD_1=0;
                                MD_2=1;
                                while(limit_SW_4==0);
                                MD_1=0;
                                MD_2=0;
                                writecmd(0x01);
}
void emergency_loop()
{
                        if(D_2==0)
                        {
                        MA_1=0;
                        MA_2=0;
                        MB_1=0;
                        MB_2=0;
                        MC_1=0;
                        MC_2=0;
                        MD_1=0;
                        MD_2=0;
                                writecmd(0x01);
                                lcd_string("Emergency!!!");
                                while(D_1==1)
                                {
                                                Buzzer=1;delay(500);
                                        Buzzer=0;delay(500);
                                }

                                Buzzer=0;
                                while(1);
                        }
}
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

VIII. PERFORMANCE

Performance is an essential aspect of quality assurance since it represents the last examination of specification and design. The conventional perception of user testing is that it is undertaken to demonstrate that there are no problems in the system. The most beneficial and practical technique with the knowledge that performance is the process of displaying the project with the specific aim of detecting faults. If the performance test is successful, it will identify faults in the future. As a side advantage, testing will demonstrate that the system looks to be in accordance with the specification and that the performance criteria appear to have been reached.

A. Objectives Of Performance

1) Performance is the process of running the system in order to detect errors.
2) A excellent performance is one in which there is a significant likelihood of discovering an unknown mistake.
3) A successful performance is one that reveals previously unknown flaws.

IX. RESULTS & CONCLUSION

A. Testing And Discussion

System testing of software or system testing refers to testing done on an entire, integrated system to evaluate compliance with its set requirements. System testing is a type of black-box testing, which means that no knowledge of the underlying design of the code or logic is necessary. In general, system testing accepts all "integrated" software components that have passed integration testing, as well as the software system itself integrated with any acceptable system, as input. The purpose of integration testing is to identify any differences between the software components that have been integrated together (referred to as assemblages) or between any of the assemblages and the system. System testing is a more limited type of testing that seeks to identify flaws in both the "inter-assemblages" and the system as a whole. System testing is carried out on the entire system in the context of a Functional Requirement Specification(s) and/or a System Requirement Specification (SRS). System testing evaluates not just the design, but also the behaviour and even the customer's perceived expectations. It is also intended to test up to and beyond the boundaries specified in the software/system requirements description.

1) Objectives Of Testing

a) Identifying flaws that may have been introduced by the programmer when designing the software.
b) Developing trust in and giving information about the degree of quality.
c) To avoid flaws.
d) Ensure that the finished outcome fits the needs of the company and the user.
e) To ensure that it meets the BRS (Business Requirement Specification) and SRS (System Requirement Specifications).

2) Testing Consideration

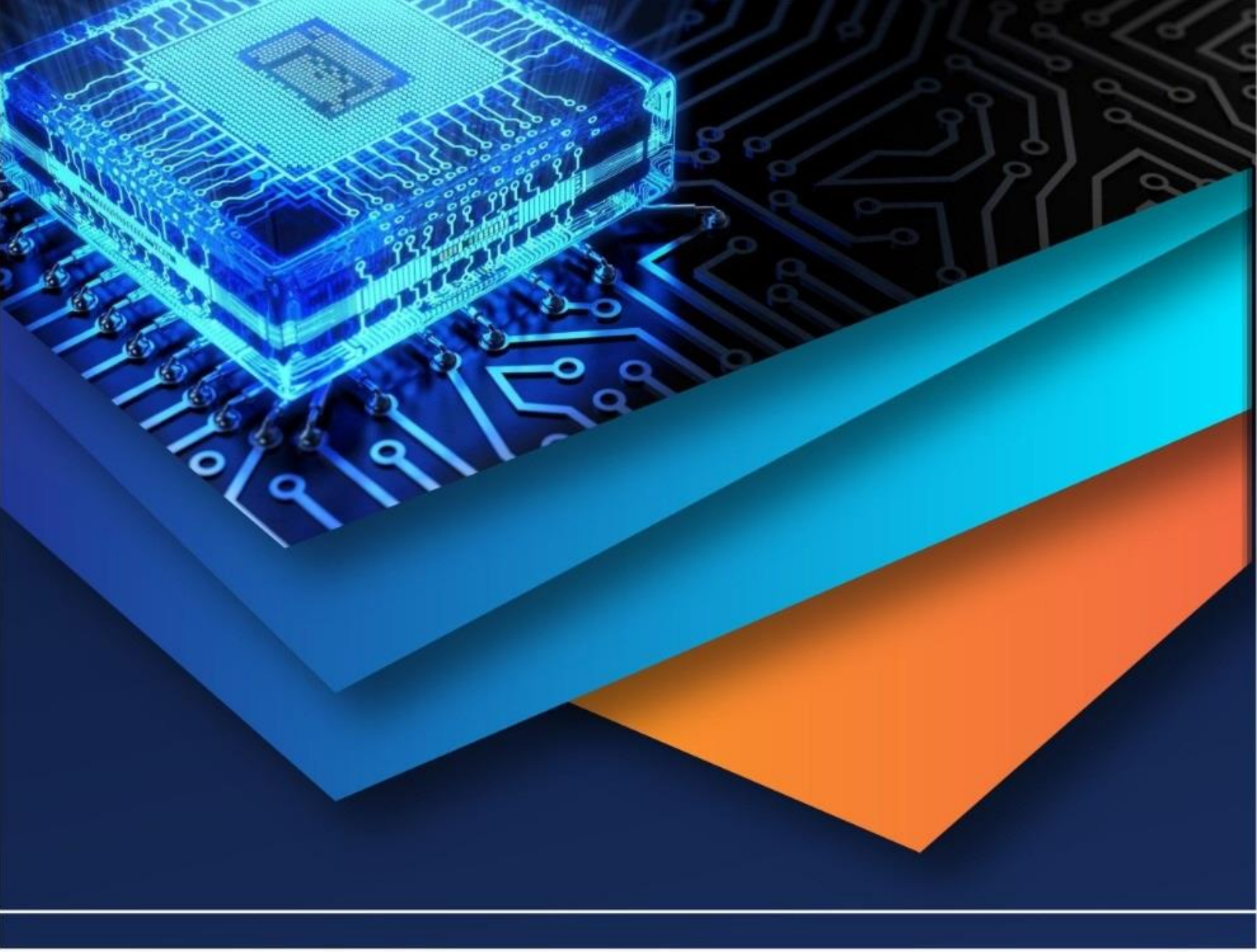
Testing is an inquiry that is carried out to provide stakeholders with information about the quality of the items or services being tested. It also gives certain objectives that are outside of the system to assist the company to recognise and comprehend the risks associated with system installation. A test strategy is an outline that specifies the system development cycle's testing methodology. It was developed to educate project managers, testers, and developers on certain critical aspects of the testing process. This contains the testing goals, new function testing technique, total time and resources necessary for the project, and testing environment. Test strategies describe how stakeholders' product risks are mitigated at the test level, what types of tests are to be performed, and what entry and exit criteria apply. They are built on the foundation of development design documents. System design documents are the most commonly used, but conceptual design documents may be referred to on occasion. The functionality of the system to be enabled in the upcoming release is described in design documents. To test the new features, a test strategy should be developed for each stage of development design. Performance testing is a recipe for testing disaster if it is treated as an afterthought. Instead of performing performance testing late in the development cycle, it should take an agile approach and be iterative throughout. Performance gaps can be identified faster and earlier in the development cycle this way.

B. Conclusion

"HANDICAP PICKUP AND DROP SYSTEM" is a comprehensive measure taken to reduce the effort of the handicapped or senior citizen from the ticket counter to the respected platform. This system employs a ramp system, in which the individual must sit on a chair, and as the chair approaches the lift, it will move and enter the lift. When the lift arrives, the chair will move outwards and the individual will reach the designated platform. The main goal of our project is to reduce the individual's effort while completely eliminating the need for the assistance of a second person.

REFERENCES

[1] Design Of Machine Elements, V.B. Bhandari.
[2] Machine Design, R.S. Khurmi, J.K. Gupta
[3] Design Of Machine Elements, Mrs. Nehal Mucchala, Vinod Thombre Patil
[4] Machine Design, Pandya & Shah
[5] Production Technology, Hazra & Choudhary
[6] Workshop Technology, R.K. Jain
[7] Design Data Book, P.S.G
[8] IEEE Papers



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY