

Smart Trolley System Using IoT

Project done by:

Group Name: **RansomWhere**

Presented to **Dr. Reena
Monica**



R VIGNESH	- 18BEC1064
N T SRIHARI	- 18BEC1180
VIGNESH M	- 18BEC1223
SUSHIL KUMAR B	- 18BEC1227
PRAMOD K	- 18BEC1257
ANDREW JOHN	- 18BEC1278
ABINESH VEL S	- 18BEC1340

OBJECTIVE

The team's objective is to develop a project that will help with the tedious and time consuming process of grocery shopping by greatly reducing the billing time and shopping time. It is targeted towards customers of all types for easy and seamless shopping. This promotes contactless shopping and transaction which is need of the hour now. We are planning to achieve the same by using a smart trolley system that uses RFID tags to help ease the hassle. This benefits both the customer and the shopkeeper in various ways.

MOTIVATION

- This project has aimed at the concept of “**Smart Shopping**”, promotes **Social Distancing** and tries to adhere to all the Government norms that pertain to the Covid’19’s Lockdown Regulations and Policies, given that we still have the Covid-19’s threat amongst us.
- There has been one and only one way to tag the given products to their cost, **the Barcode**, and sometimes the Barcode fails to scan the given the product given that it will be printed over the product's packaging and it might be damaged.
- The **RFID tag** process proposed in this Project and Product works smoothly given that it needn’t only gives the cost of the product when the tag is scanned, but in turn gives about a **small, brief information about the product too**, to educate the customer and the queue near the Billing Counter can be drastically reduced since the Bill is **automated** and can be paid using a Secure Payment Gateway.
- Our “Smart Shopping” is also motivated at **reducing the labour cost** and the cost of the labour can be used in servicing the Smart Trolley after getting a good use out of it over the years.

LITERATURE SURVEY & FIELD SURVEY

Customer Questionnaire

Shopkeeper Questionnaire

Literature Survey



Field survey



HARDWARE COMPONENTS

➤ **ESP8266 NodeMCU x 1 -**

- NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. controlling unit which contains the information regarding the products, calculates the bill amount and sends to LCD for display

➤ **EM18 RFID Reader Module x 1 -**

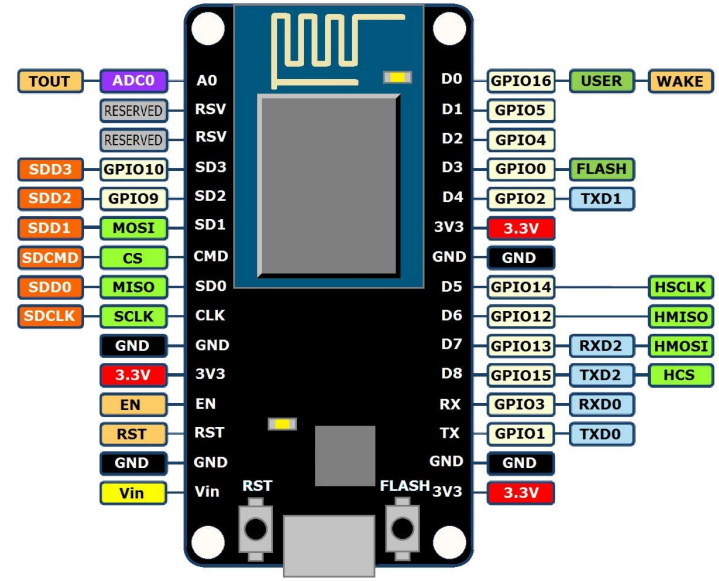
- To transmit and receive radio waves in order to communicate with **RFID** tags. used to read RFID tags

➤ **RFID Tags**

HARDWARE COMPONENTS - NodeMCU

NodeMCU ESP8266 Specifications & Features

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna



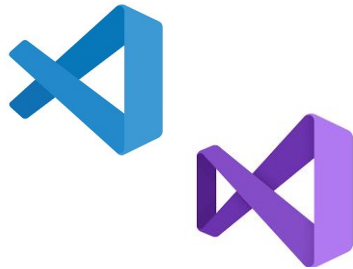
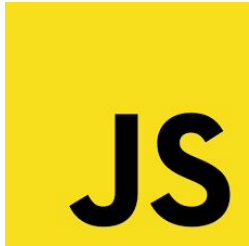
HARDWARE COMPONENTS - RFID

EM-18 Features and Specifications

- Operating voltage of EM-18: +4.5V to +5.5V
- Current consumption: 50mA
- Can operate on LOW power
- Operating temperature: 0°C to +80°C
- Operating frequency: 125KHz
- Communication parameter: 9600bps
- Reading distance: 10cm, depending on TAG
- Integrated Antenna



SOFTWARE USED



SOFTWARE COMPONENTS

- **XAMPP - Modules : Apache and MySQL** - XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.
- **Arduino IDE for Hardware** - The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards.
- **IDE - Visual Studio Code** - Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

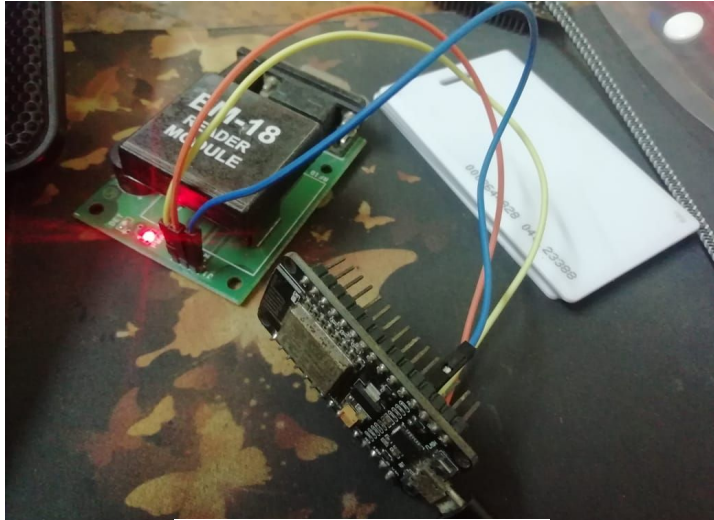
SOFTWARE COMPONENTS

➤ **Language (Website) :**

- **HTML (Hypertext Markup Language)** - The language for building web pages
- **CSS (Cascading Style Sheets)** - The language for styling web pages
- **JS (JavaScript)** - The language for programming web pages
- **PHP (Hypertext Preprocessor)** - A web server programming language

➤ **Language (Hardware) : C++** - C++ is a general-purpose programming language created by Bjarne Stroustrup as an extension of the C programming language, or "C with Classes".

REAL TIME IMPLEMENTATION



Hardware used for RFID Rx/Tx:

User gives input from the RFID tag for giving the product credentials and bills it to the final receipt.

- The Signal from the RFID tag is tagged to a specific product and the reader picks it up once the tag happens.
- The information from the tag is sent to the database and a copy is made visible to the customer's UI where the customer can increase the count.
- The consolidated bill is sent to Razorpay and the payment is commenced.
- After the successful payment, the customer proceeds to check out aisle and drops the trolley.

REAL TIME IMPLEMENTATION



Prototype UI as perceived by the user.

- This is the prototype where the user scans the product to the RFID Reader which is attached to the Tablet (to be installed on the Smart Trolley along with the reader)
- The user will access a tab which is the UI where they will be able to see the purchases they made throughout the shopping

CODE SNIPPET

```
sketch_apr10a
/*
 * HTTP Client POST Request
 * Copyright (c) 2018, circuits4you.com
 * All rights reserved.
 * https://circuits4you.com
 * Connects to WiFi HotSpot. */

#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266HTTPClient.h>

int count = 0;
char input[12];

/* Set these to your desired credentials. */
const char *ssid = "Mohan"; //ENTER YOUR WIFI SETTINGS
const char *password = "08121110";

//Web/Server address to read/write from
const char *host = "192.168.0.107"; //https://circuits4you.com website or IP address of server

//=====
//                               Power on setup
//=====

void setup() {
  delay(1000);
  Serial.begin(9600);
  WiFi.mode(WIFI_OFF); //Prevents reconnection issue (taking too long to connect)
  delay(1000);
  WiFi.mode(WIFI_STA); //This line hides the viewing of ESP as wifi hotspot

  WiFi.begin(ssid, password); //Connect to your Wifi router
  Serial.println("");
```

```
sketch_apr10a
Serial.print("Connecting");
// Wait for connection
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}

//If connection successful show IP address in serial monitor
Serial.println("");
Serial.print("Connected to ");
Serial.println(ssid);
Serial.print("IP address: ");
Serial.println(WiFi.localIP()); //IP address assigned to your ESP
}

//=====
//                               Main Program Loop
//=====
void loop() {
  HTTPClient http; //Declare object of class HTTPClient

  String ADCData, station, postData, data;
  if(Serial.available())
  {
    count = 0;
    while(Serial.available())
    {
      input[count] = Serial.read();
      data = data + input[count];
      count++;
      delay(5);
    }
  }
  Serial.println(data);
  ADCData = data;
```

CODE SNIPPET

```
station = "A";

//Post Data
postData = "status=" + ADCData + "&station=" + station ;
http.begin("http://192.168.0.102/tarp-nt/postdemo.php");           //Specify request destination
http.addHeader("Content-Type", "application/x-www-form-urlencoded"); //Specify content-type header

int httpCode = http.POST(postData); //Send the request
String payload = http.getString(); //Get the response payload

Serial.println(httpCode); //Print HTTP return code
Serial.println(payload); //Print request response payload

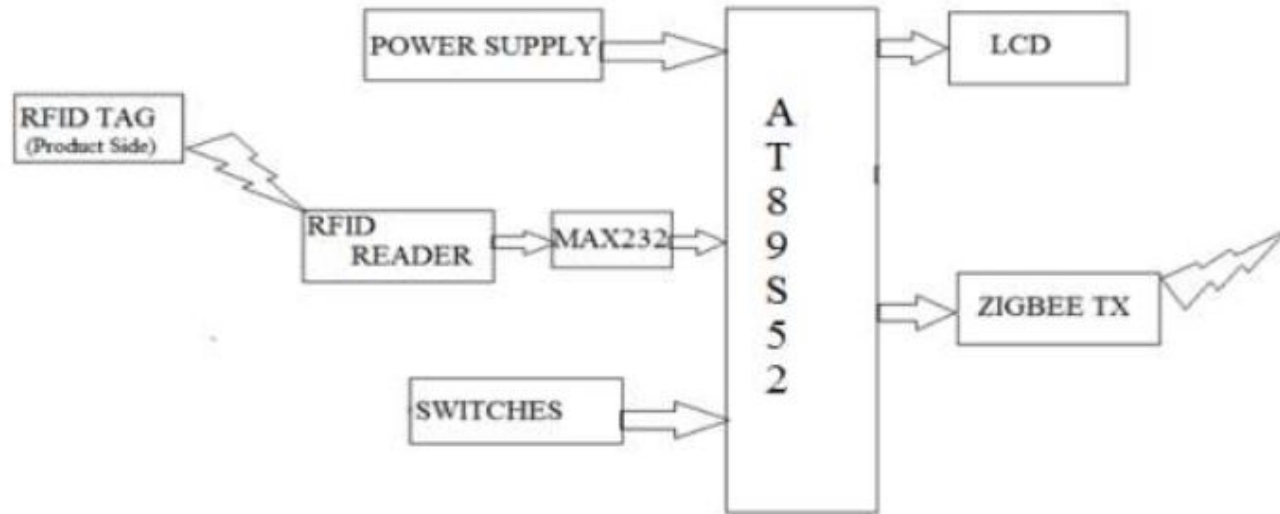
http.end(); //Close connection

delay(1000); //Post Data at every 5 seconds
}
//=====
```

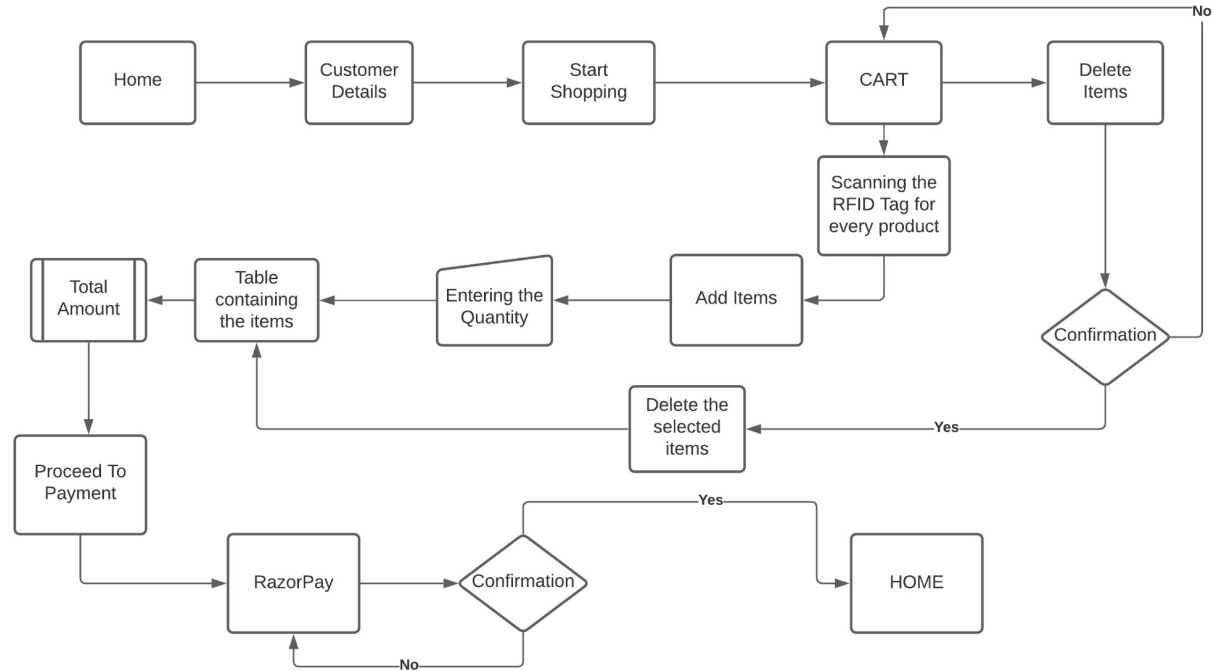
METHODOLOGY

- Our plan is to initially begin with the hardware portion of the project first , complete it and then move to the software portion. The circuit is designed in such a way that :
- Initially when the customer places the product equipped with RFID tag in the trolley it is read by the EM18 RFID reader.
- This sends information to controlling unit (NodeMCU) which checks for the product details in its database.
- Transfer the required details to a new customer table.
- The database of the details of the products available in the supermarket are fed to the website by the admin (shopkeeper)
- The prices of each product and final bill amount is displayed on the website.
- The bill calculation is also done using the backend scripts

BLOCK DIAGRAM



FLOW CHART



RESULTS: Back-end

The screenshot shows the phpMyAdmin web interface. The browser address bar indicates the URL: `localhost/phpmyadmin/index.php?route=/sql&server=1&db=smartrolley&table=default_products&pos=0`. The interface displays the results of a SQL query executed on the `default_products` table in the `smartrolley` database.

The query executed is: `SELECT * FROM `default_products``. The results show 4 rows (rows 0-3) with the following columns: `RFID`, `NAME`, `TYPE`, and `PRICE`.

RFID	NAME	TYPE	PRICE
44	GLOVES	LEATHER	2500
55	BINGO	FOOD	20
24	MUG	UTILITY	350
33	GRAPES	FRUITS	230

Below the table, there are options to perform actions on the selected rows (all rows are selected). The actions include: Check all, With selected, Edit, Copy, Delete, and Export. The interface also includes a section for "Query results operations" with links for Print, Copy to clipboard, Export, Display chart, and Create view. At the bottom, there is a section for "Bookmark this SQL query" with a label field and a checkbox to let every user access this bookmark.

RESULTS: Back-end

The screenshot shows the phpMyAdmin interface in a web browser. The address bar indicates the URL: `localhost/phpmyadmin/index.php?route=/sql&server=1&db=smartrolley&table=customer_products&pos=0`. The interface includes a sidebar with a database tree on the left and a main content area on the right.

Database Tree (Left Sidebar):

- Recent
- Favorites
- New
- information_schema
- mysql
- performance_schema
- phpmyadmin
- smartrolley
 - New
 - customer_information1
 - customer_products
 - default_products
 - test

Main Content Area:

- Server: 127.0.0.1 » Database: smartrolley » Table: customer_products
- Navigation tabs: Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Message: **Showing rows 0 - 1 (2 total. Query took 0.0006 seconds.)**
- SQL Query: `SELECT * FROM `customer_products``
- Buttons: Profiling, Edit inline, Edit, Explain SQL, Create PHP code, Refresh.
- Controls: Show all, Number of rows: 25, Filter rows: Search this table, Sort by key: None.
- Table Data:

				RFID	NAME	TYPE	QUANTITY	PRICE	FINAL_AMOUNT
<input type="checkbox"/>	Edit	Copy	Delete	33	GRAPES	FRUITS	5	230	0
<input type="checkbox"/>	Edit	Copy	Delete	55	BINGO	FOOD	12	20	0

Below the table, there are buttons for "Check all", "With selected:", "Edit", "Copy", "Delete", and "Export".

Query results operations:

- Print
- Copy to clipboard
- Export
- Display chart
- Create view

Bookmark this SQL query:

Label: ☐ Let every user access this bookmark

Bookmark this SQL query

RESULTS: Back-end

The screenshot shows the phpMyAdmin web interface in a browser. The address bar indicates the URL: localhost/phpmyadmin/index.php?route=/sql&server=1&db=smartrolley&table=customer_information1&pos=0. The interface displays the 'customer_information1' table with 17 rows. A message at the top states: 'Showing rows 0 - 16 (17 total. Query took 0.0008 seconds.)'. Below this, the SQL query 'SELECT * FROM `customer_information1`' is shown. The table has three columns: NAME, EMAILID, and PHONENO. The data includes multiple entries for 'Vighnesh' with the same email and phone number, followed by 'VIGHNESH M', 'Vel', 'NT', and 'gg'. At the bottom, there are options to 'Bookmark this SQL query' and a checkbox to 'Let every user access this bookmark'.

NAME	EMAILID	PHONENO
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
VIGHNESH M	vighnesh1209@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
Vighnesh	vighneshminda@gmail.com	2147483647
dddd	ttyyy	0
Vel	av@gmail.com	986789234
NT	Nt_Trisha@gmail.com	999999999
gg	sff	466

RESULTS: Front-end



RESULTS: Front-end

[RW](#)[About](#)[Support](#)

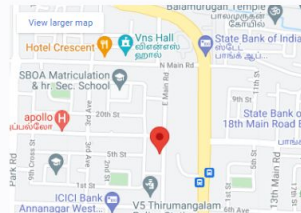
Hey. Welcome to our Shop.

We are a startup that designs Smart Trolley Systems that makes that makes life simple for the customers.

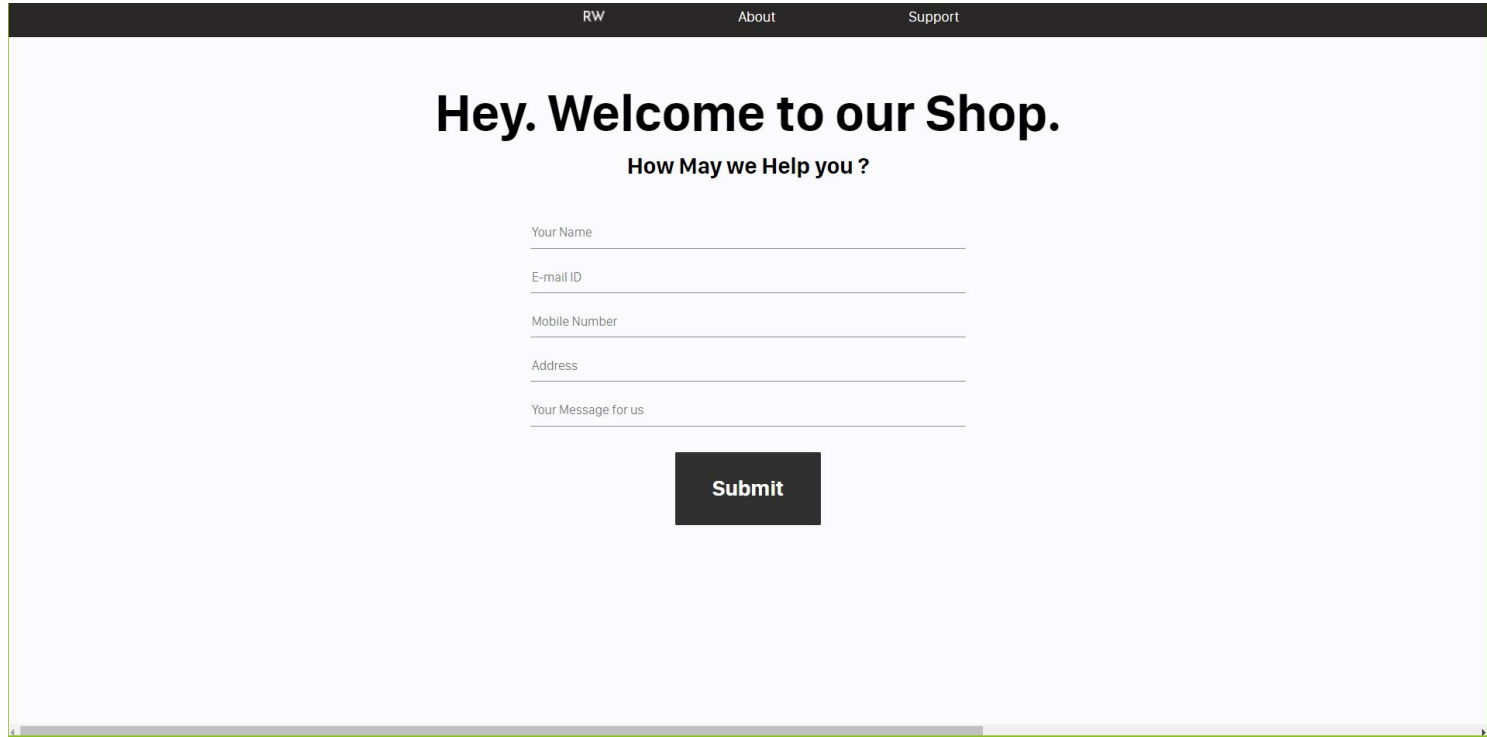
Have a nice day :)

[Click Here](#)

Our Location:
No 12, East Main Road
Anna Nagar West Extension
Chennai - 600127.



RESULTS: Front-end

A screenshot of a web form for a shop. The form has a dark header with navigation links 'RW', 'About', and 'Support'. The main content area is light gray and contains a welcome message, a question, and a form with five input fields and a submit button. The form is outlined with a green border.

RW About Support

Hey. Welcome to our Shop.

How May we Help you ?

Your Name

E-mail ID

Mobile Number

Address

Your Message for us

Submit

RESULTS: Front-end

RW

About

Support

Hey. Welcome to our Shop.

We are a Department shop based in Chennai. Press Below to Start Shopping with us. Happy Shopping!

Your Name

Mobile Number

E-mail ID

Continue 

RESULTS: Front-end

RW

About

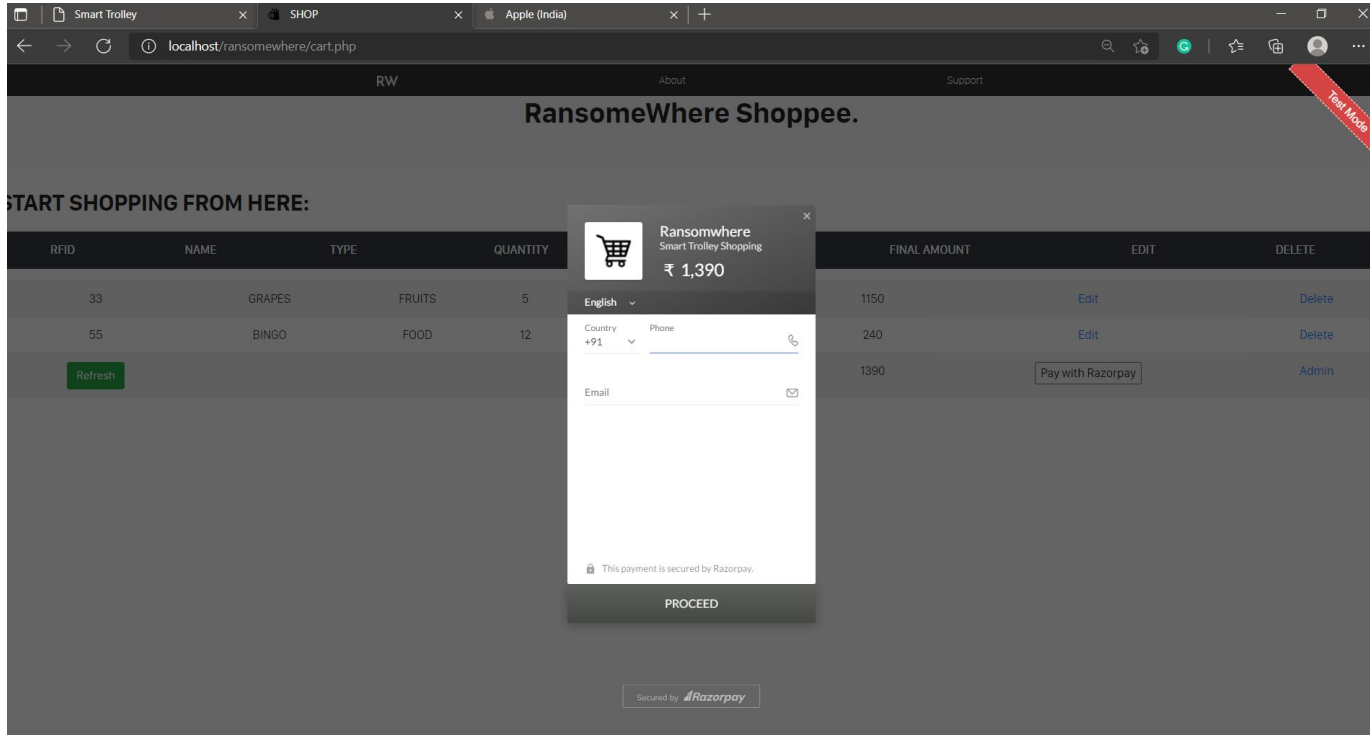
Support

RansomeWhere Shoppee.

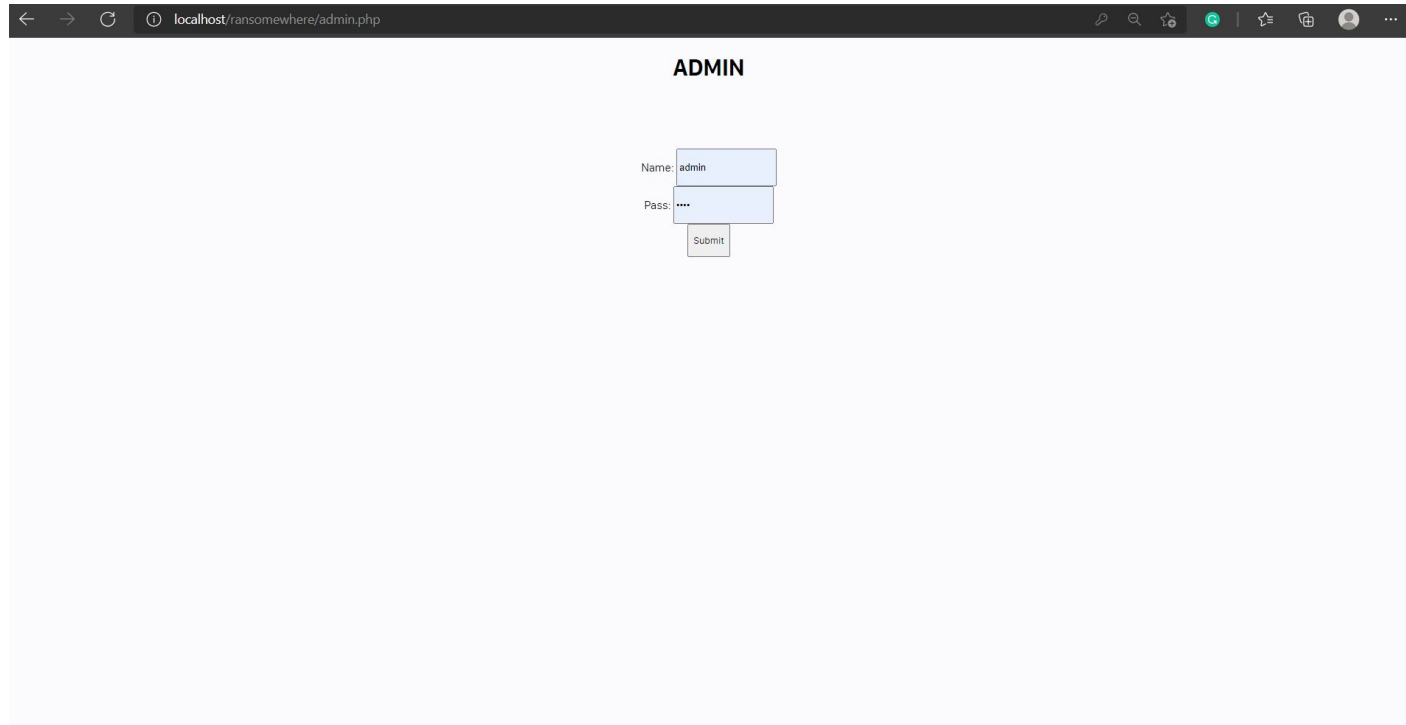
START SHOPPING FROM HERE:

RFID	NAME	TYPE	QUANTITY	PRICE	FINAL AMOUNT	EDIT	DELETE
33	GRAPES	FRUITS	5	230	1150	Edit	Delete
55	BINGO	FOOD	12	20	240	Edit	Delete
44	GLOVES	LEATHER	0	2500	0	Edit	Delete
<div>Refresh</div>				TOTAL AMOUNT	1390	<div>Pay with Razorpay</div>	Admin

RESULTS: Front-end



RESULTS: Front-end



A screenshot of a web browser window displaying a login form. The browser's address bar shows the URL `localhost/ransomewhere/admin.php`. The page has a light gray background. At the top center, the word **ADMIN** is displayed in a bold, black, sans-serif font. Below this, the login form consists of two labels, 'Name:' and 'Pass:', each followed by a light blue rectangular input field. The 'Name' field contains the text 'admin', and the 'Pass' field contains four black dots. Below these fields is a small, light gray rectangular button with the word 'Submit' in black text.

localhost/ransomewhere/admin.php

ADMIN

Name: admin

Pass:

Submit

TIMELINE

- **Week (1 - 3) :** Project title selection , Literature Survey and Field Survey
- **Week (4 - 6) :** Building up the Hardware of the Project
- **Week (7 - 9) :** Designing the Website and Developing the Software
- **Week (10 - 12) :** Testing, Summary, Results and Report

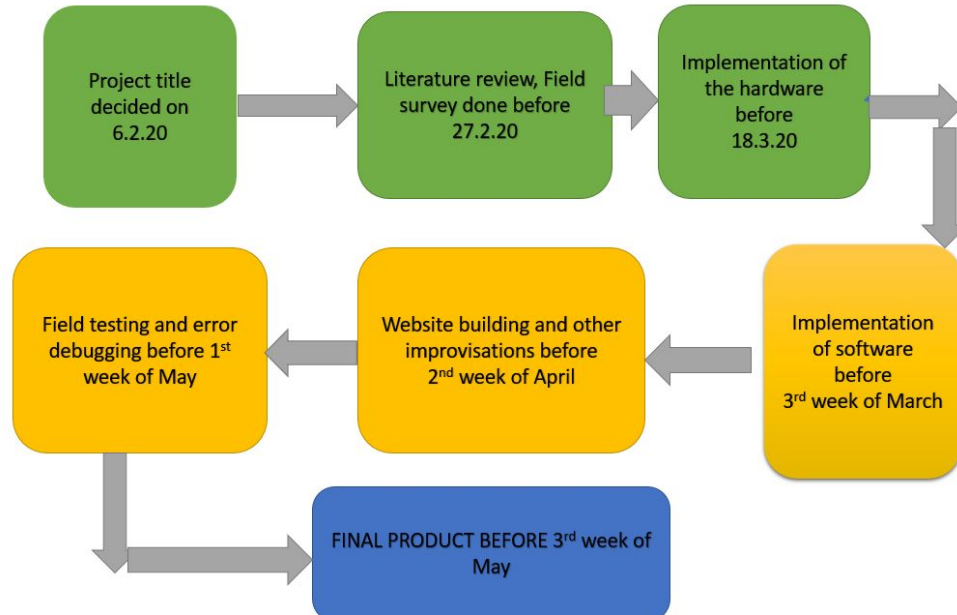
FUTURE SCOPE

- We hope to save people's time by reducing the crowd at the particular shop. All the product details are stored in a cloud so the people can see the details of the products from anywhere.
- We also hope to be eco-friendly by saving paper as the soft copy of the customer's bills will be sent to their respective mobile phone then and there. We hope to employ more engineers to program and make our device, thus increasing employment opportunities.
- We finally aim to move one step closer to a Digital India. If multiple supermarkets implement our technology, we will also be able to tell our customers the nearest venue from their current location where a particular product is available, if in case it isn't available in one shop

WORK DONE

- Designing a user friendly UI from scratch for the Smart Trolley system.
- Building up of RFID Rx/Tx with Arduino and interfacing with User Interface.
- Styling the website UI with CSS.
- Creating the database for tallying the user products using MySQL
- PHP is used to fetch data from the database to the website.
- Integrating the database (backend) with the Website UI (frontend).
- Integrating the RazorPay API for checkout and payment.

TIMELINE FLOW



CONCLUSION

The Smart Trolley was designed to function as a mobile self-checkout system providing users the flexibility to make transactions from it within the retail store. It is designed to be highly efficient and fully synchronised with the retailer's current system. It is more reliable and provides ease for shopping. Here we conclude that the proposed system is time saving i.e. faster check out process. This system reduces the number of salesmen at the counter.

FINAL REPORT LINK

**[Smart Trolley System -
RansomWhere](#)**



PRESENTATION AND COMPLETE DEMO LINKS

[Presentation Link](#)



[Complete Demo Link](#)



An aerial photograph of the New York City skyline at dusk. The sky is a mix of dark blue and orange, with scattered clouds. The city is densely packed with skyscrapers, many of which are illuminated with their lights. The Empire State Building is prominent in the center, with its top lit in red and green. The Hudson River is visible on the right side of the image. The text "Thank You :)" is overlaid in the center in a large, white, sans-serif font.

Thank You :)

ROLES

Project Manager - Vighnesh M, Sushil Kumar

Design Leads - R Vignesh, Abinesh Vel

Liaison - Andrew John

Financial Officer - Pramod K

Project Archivist - N T Srihari, Sushil Kumar