

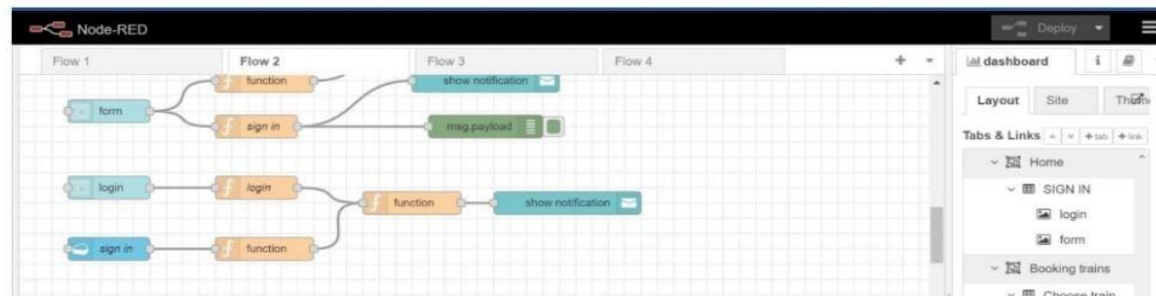
# SPRINT - 1

|              |                                       |
|--------------|---------------------------------------|
| Team ID      | PNT2022TMID37954                      |
| Project Name | Project –Smart Solutions for Railways |

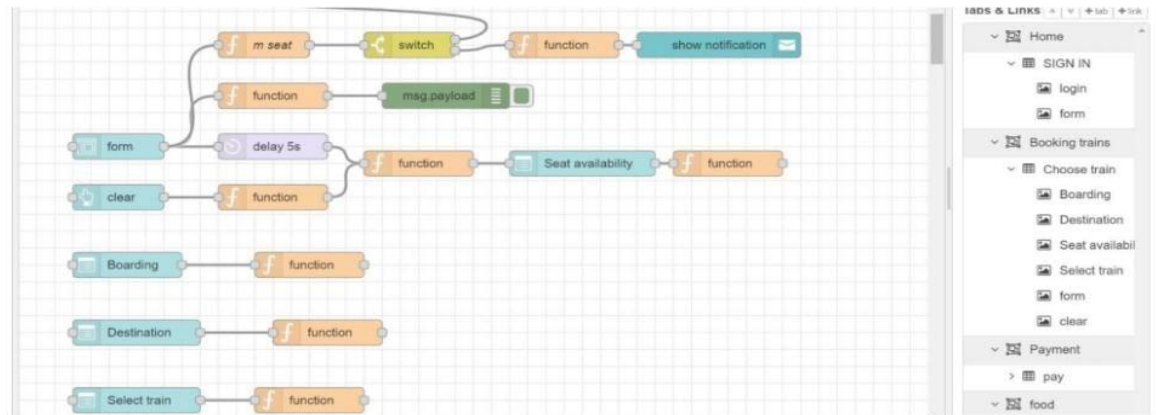
## PROCEDURE:

- Step1: Develop node red application
- Step2: Install the required nodes from manage palette option
- Step3: Connect the node flow
- Step4: Deploy the flow

## WEB APPLICATION :



## NODES TO BOOK TRAIN:



**FUNCTION NODE COMMAND TO INDICATE THE AVAILABLE SEATS:**

```
var a=global.get('a') var
s= []
for(let i=0;i<a.length==0;i++){
s.push(a[i]) }
if(s.length==0){
    msg.options=[{"No seats available":0}]
}
else{
msg.options= s
}
msg.payload= s
return msg;
```

**FUNCTION NODE COMMAND TO CHOOSE THE AVAILABLE SEATS:**

```
var s=global.get('s') var
a=global.get('a') function
reg(x){  for(let
i=0;i<a.length;i++){
if(a[i]==x){
    a.splice(i,1)
}
}
}
if(s==1){
global.set('s1',s)
reg(s)
}
else if(s==2){
global.set('s2',s)
reg(s)
}
else if(s==3){
global.set('s3',s)
reg(s)
}
else if(s==4){
global.set('s4',s)
reg(s)
}
else if(s==4){
global.set('s4',s)
reg(s)
}
}
return msg;
```

## FUNCTION NODE COMMAND TO STORE DATA IN DATABASE:

```
var m=global.get('m')
var d=new Date();
var utc=d.getTime()+(d.getTimezoneOffset()*60000); var
offset=5.5;
newDate=new Date(utc+(3600000*offset));
var n=newDate.toISOString() var
date=n.slice(0,10) var time=n.slice(11,19)
var d1=date+' '+time msg.payload={
  "_id":d1,
  "Name":m.Name,
  "Age":m.Age,
  "Mobile":m.Num,
  "boarding":global.get('b'),
  "destination":global.get('d'),
  "Seat":global.get('s')
}
return msg;
```

## FORM DETAILS:

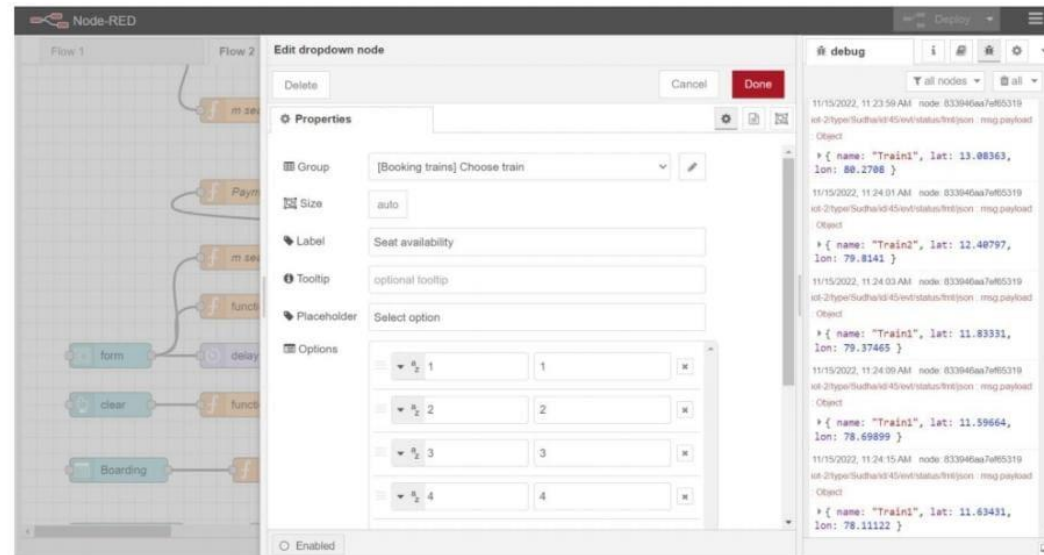
The screenshot shows the 'Edit form node' interface in Node-RED. The 'Properties' section on the left includes a 'Group' dropdown set to '[Booking trains] Choose train', a 'Size' dropdown set to 'auto', and a 'Label' text field containing 'optional label'. Below these are 'Form elements' which are listed in a table:

| Label      | Name | Type   | Required                            | UI Rows | Remove |
|------------|------|--------|-------------------------------------|---------|--------|
| Name       | name | Text   | <input checked="" type="checkbox"/> |         |        |
| Age        | age  | Number | <input checked="" type="checkbox"/> |         |        |
| Mobile num | num  | Number | <input checked="" type="checkbox"/> |         |        |

At the bottom left, there is a checkbox labeled 'Enabled' which is currently unchecked. On the right side, the 'debug' console shows a series of log messages. Each message is an object with 'name' and 'lat' properties, representing train data. The messages are as follows:

- 11/15/2022, 11:23:25 AM node: 833940ea7ef05319  
msg: { name: "Train1", lat: 13.08363, lon: 89.2708 }
- 11/15/2022, 11:23:27 AM node: 833940ea7ef05319  
msg: { name: "Train2", lat: 12.48797, lon: 79.8141 }
- 11/15/2022, 11:23:29 AM node: 833940ea7ef05319  
msg: { name: "Train1", lat: 11.83331, lon: 79.37465 }
- 11/15/2022, 11:23:35 AM node: 833940ea7ef05319  
msg: { name: "Train1", lat: 11.59664, lon: 78.69899 }
- 11/15/2022, 11:23:41 AM node: 833940ea7ef05319  
msg: { name: "Train1", lat: 11.63431, lon: 78.11122 }

## SEAT DROPDOWN BOX:



## WEB UI OUTPUT:

[Home](#)

### SIGN IN


sudha

55

55

### login

## WEB UI FOR TRAIN BOOKING:

 Booking trains

Choose train

Boarding

Destination

Seat availability

Select train

Name \*

Age \*

Mobile num \*

SUBMIT

CANCEL

CLEAR

