

Final task Report

1. **Project Name** - Create an Expenditure Calculator using MERN stack.

2. **Group Information** –

Name	Branch	GR no.
Ketan Gangwal	AI & DS	12010720
Nayan Bari	ENTC	12010451
Pavan Sanjay	Computer Science	12010017

3. **Project Github Link** – https://github.com/RoboSpark-2021/robospark-2021-Expenditure_Calculator.git

4. **Project Workflow** –

- Finalizing Design of Frontend using Figma
- Front Development using React js
- Testing Frontend
- Creating a Server
- Creatin
- Connecting React js with MongoDB Atlas (Cloud)
- Creating API or routes for storing and fetching the data
- Connecting the Frontend Form with Backend using API

- Showing Expenditure History in History Tab via calling the API
- Testing Whole Project (Frontend + Backend)

5. **Problems Faced –**

- Problem facing while appending the Data in react state (frontend)
- Connecting Server to the MongoDB Cluster
- Repetition of Today's Expenditure Data while showing it in the Frontend

6. **Alternative Solutions found for problems mentioned above –**

- Instead of Storing the Data in React state, we directly showed the Data using API Call
- Instead of using Mongoose Package, we used MongoDB Package
-

7. **Code Snippet ss –**

Frontend –

A. Form.js - -

```

JS Form.js U X
src > components > JS Form.js > [E] Form > [E] addItem
  1  '../styles/form.css'
  2  { useState, useEffect } from 'react'
  3
  4  Form = () => {
  5    nst [number, setNumber] = useState(0)
  6    nst [expenses, setExpenses] = useState({});
  7    nst [item, setItem] = useState(['']);
  8
  9    nst addItem = () => {
10      let value = number
11      const form = document.querySelector('.form');
12      const name = form.querySelector('.form .form-name #item').value;
13      const price = form.querySelector('.form .form-name #price').value;
14      const amount = form.querySelector('.form .form-name #amount').value;
15      const status = new Date();
16      const time = `${status.getHours()}:${status.getMinutes()}` ;
17      const date = `${status.getDate()}-${status.getMonth() + 1}-${status.getFullYear()}` ;
18      console.log(number)
19      if(status.getHours() == 24){
20        setNumber(0)
21      }
22      setItem([name, price, amount, time, date])
23      setExpenses([...expenses, [status.getFullYear()]: { [status.getMonth() + 1]: { [status.getDate()]: { [value]: { item: name, price: price,
24        amount: amount, time: time, date: date }}}}]])
25      setNumber(++value)
26
27      nsole.log(item);
28      nsole.log(expenses);
29      turn (
30        <div className="addItem">
31          <h4>add item</h4>
32          <div className="form">
33            <div className="form-name">
34              <input type="text" name="item" id="item" autoComplete="off" required />
35              <label className="label-name">
36                <span className="content-name">
37                  <h5>Item Name</h5>

```

```

37          <h5>Item Name</h5>
38        </span>
39      </label>
40    </div>
41    <div className="form-name">
42      <input type="number" name="price" id="price" autoComplete="off" required />
43      <label className="label-name">
44        <span className="content-name">
45          <h5>Price</h5>
46        </span>
47      </label>
48    </div>
49    <div className="form-name">
50      <input type="number" name="amount" id="amount" autoComplete="off" required />
51      <label className="label-name">
52        <span className="content-name">
53          <h5>Amount</h5>
54        </span>
55      </label>
56    </div>
57  </div>
58  <div className="submitButton">
59    <input className="btn" type="submit" onClick = {addItem} value="Add" />
60  </div>
61 </div>
62
63
64
65  default Form

```

B. History.js –

```
JS History.js U X
src > components > JS History.js > [X] History
1  import '../styles/search.css'
2  import '../styles/table.css'
3  import HistoryTable from './HistoryTable.js'
4
5  const History = () => {
6      return (
7          <div >
8              <section className="sect2">
9                  <form className="infi">
10                     <input type="text" placeholder="search" id="search"/>
11                     <select id="lm">
12                         <option value = "Month"> Month
13                     </option>
14                     <option value = "January"> January
15                     </option>
16                     <option value = "Febuary"> Febuary
17                     </option>
18                     <option value = "March">March
19                     </option>
20                     <option value = "April"> April
21                     </option>
22                     <option value = "May"> May
23                     </option>
24                     <option value = "June"> June
25                     </option>
26                     <option value = "July"> July
27                     </option>
28                     <option value = "August"> August
29                     </option>
30                     <option value = "September"> September
31                     </option>
32                     <option value = "October"> October
33                     </option>
34                     <option value = "November"> November
35                     </option>
36                     <option value = "December"> December
37                     </option>
38                 </select>
39                 </form>
40                 </section>
41                 <HistoryTable />
42             </div>
43         )
44     }
45
46     export default History
```

C. HistoryRow.js –

JS HistoryRow.js U X

src > components > JS HistoryRow.js > ...

```
1  import '../styles/table.css'
2
3  const HistoryRow = () => {
4    return (
5      <div className="row">
6        <div className="number">
7          <h4>1</h4>
8        </div>
9        <div className="name">
10         <h4>Note</h4>
11       </div>
12       <div className="price">
13         <h4>180</h4>
14       </div>
15       <div className="amount">
16         <h4>2</h4>
17       </div>
18       <div className="total">
19         <h4>360</h4>
20       </div>
21       <div className="actions">
22         <h4>
23
24           12-01-2021
25         </h4>
26       </div>
27     </div>
28   )
29 }
30
31 export default HistoryRow
```

D. HistoryTable.js –

```
JS HistoryTable.js U X
src > components > JS HistoryTable.js > ...
1 |import '../styles/table.css'
2 |import HistoryRow from './HistoryRow.js'
3
4 const HistoryTable = () => {
5   return (
6     <div className='table'>
7       <div className="row">
8         <div className="number">
9           <h4>Sr no</h4>
10        </div>
11        <div className="name">
12          <h4>Name</h4>
13        </div>
14        <div className="price">
15          <h4>Price</h4>
16        </div>
17        <div className="amount">
18          <h4>Quantity</h4>
19        </div>
20        <div className="total">
21          <h4>Total</h4>
22        </div>
23        <div className="actions">
24          <h4>Date</h4>
25        </div>
26      </div>
27      <HistoryRow />
28    </div>
29  )
30 }
31
32 export default HistoryTable
```

E. Home.js –

```
JS Home.js U X
src > components > JS Home.js > ...
1  import Table from "../Table"
2  import TopSection from "../TopSection"
3
4  const Home = () => {
5    return (
6      <div>
7        <TopSection />
8        <Table />
9      </div>
10    )
11  }
12
13  export default Home
```

F. Navigation.js –

```
JS Navigation.js U X
src > components > JS Navigation.js > ...
1  import {Link} from 'react-router-dom'
2  import NavigationLogo from '../assets/navigation_logo.png'
3  import '../styles/utilities.css'
4  import '../styles/navigation.css'
5
6  const Navigation = () => {
7    return (
8      <nav className="navigation">
9        <div className="logo">
10          <img src={NavigationLogo} alt="Logo" />
11        </div>
12        <div className="navlinks">
13          <Link style={{textDecoration: 'none'}} to='/'>
14            <h4 >Dashboard</h4>
15          </Link>
16          <Link style={{textDecoration: 'none'}} to='/history'>
17            <h4 >History</h4>
18          </Link>
19        </div>
20      </nav>
21    )
22  }
23
24
25  export default Navigation
```

G. Row.js –

```

JS Row.js U X
src > components > JS Row.js > ...
1  import './styles/table.css'                // Imported CSS file of the Table
2  const Row = (data) => {
3      const deleteRow = (e) => {              // To delete the Row
4          const row = e.nativeEvent.path[4]
5          row.style.display = 'none'
6      }
7      return (
8          <div className="row">
9              <div className="number">
10                 <h4>1</h4>                    {/* Input of the Serial no. */}
11             </div>
12             <div className="name">
13                 <h4>Note</h4>                {/* Input of the Item Name */}
14             </div>
15             <div className="price">
16                 <h4>180</h4>                {/* Input of the Price */}
17             </div>
18             <div className="amount">
19                 <h4>2</h4>                  {/* Input of the Quantity */}
20             </div>
21             <div className="total">
22                 <h4>360</h4>                {/* // Output(Total Cost) - Price*Quantity */}
23             </div>
24             <div className="actions">
25                 <div className="icon"onClick={deleteRow}>
26                     {/* // Adding Image of the delete Button */}
27                     <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path d="M3 6v18h18v-18h-18zm5 14c0 .552-.4
28                     </div>
29                     <div className="icon">
30                         {/* // Adding Image of the Edit Button */}
31                         <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path d="M7.127 22.564l-7.126 1.436 1.438-7
32                         </div>
33                 </div>
34             </div>
35         )
36     }

```

```
38 export default Row
```

H. Table.js –


```
JS Table.js U X
src > components > JS Table.js > [0] Table
1 import '../styles/table.css' // Imported CSS file of the Table
2 import Row from './Row' // Importing the Row Component
3
4 const Table = () => {
5
6     return (
7         <div className='table'>
8             <div className="row">
9                 <div className="number">
10                     <h4>Sr no</h4>
11                 </div>
12                 <div className="name">
13                     <h4>Name</h4>
14                 </div>
15                 <div className="price">
16                     <h4>Price</h4>
17                 </div>
18                 <div className="amount">
19                     <h4>Quantity</h4>
20                 </div>
21                 <div className="total">
22                     <h4>Total</h4>
23                 </div>
24                 <div className="actions">
25                     <h4>Actions</h4>
26                 </div>
27             </div>
28             <Row/> {/**/ Adding Row Component */}
29             <Row />
30             <Row />
31         </div>
32     )
33 }
34
35 export default Table
```

I. TopSection.js –

```
JS TopSection.js U X
src > components > JS TopSection.js > ...
1 import '../styles/utilities.css'
2 import '../styles/topSection.css'
3 import { useState } from 'react'
4 import Form from './Form'
5 const TopSection = () => {
6     const [todaysTotal, setTodaysTotal] = useState('1460.00')
7     return (
8         <div className="top-section">
9             <div className='card'>
10                 <h4>today's total</h4>
11                 <h2>Rs. {todaysTotal}</h2>
12             </div>
13             <Form />
14         </div>
15     )
16 }
17
18 export default TopSection
```

J. Form.css –

form.css U X

src > styles > # form.css > ...

```
1  .additem{
2      height: 100%;
3      width: 60%;
4      display: flex;
5      flex-direction: column;
6      justify-content: space-between;
7  }
8
9  .additem h4{
10     padding: 0 0 2rem 0;
11 }
12
13 .form {
14     position: relative;
15     height: 100%;
16     width: 100%;
17     display: flex;
18     align-items: center;
19     justify-content: space-between;
20 }
21
22 .form .form-name {
23     position: relative;
24     width: 100%;
25     margin-right: 1rem;
26 }
27
28 .form .form-name input {
29     font-family: "poppins", sans-serif;
30     width: 100%;
31     height: 35px;
32     color: #252525;
33     border: none;
34     outline: none;
35     padding-top: 5px;
36 }
```

```
37
38 .form .form-name input::-webkit-outer-spin-button,
39 .say-hello .contact-form .form .form-name input::-webkit-inner-spin-button {
40     -webkit-appearance: none;
41     margin: 0;
42 }
43
44 .form .form-name label {
45     position: absolute;
46     font-family: "poppins", sans-serif;
47     bottom: 0;
48     left: 0;
49     width: 100%;
50     height: 35px;
51     pointer-events: none;
52     border-bottom: 1px solid #AEA656;
53 }
```

```

54 .form .form-name input:focus + .label-name .content-name,
55 .form .form-name input:valid + .label-name .content-name {
56     -webkit-transform: translateY(-50%);
57     transform: translateY(-50%);
58     color: #AEA656;
59 }
60
61 .form .form-name input:focus + .label-name::after,
62 .form .form-name input:valid + .label-name::after {
63     -webkit-transform: translateX(0%);
64     transform: translateX(0%);
65 }
66
67 .form .content-name {
68     position: absolute;
69     font-size: 1rem;
70     font-weight: 400;
71     letter-spacing: 1.5px;
72     color: #A7A7A7;
73     bottom: 5px;
74     left: 0px;
75     -webkit-transition: all 0.5s ease;
76     transition: all 0.5s ease;
77 }
78
79 .submitButton{
80     margin: 0 auto;
81     text-align: center;
82     padding: 3rem 0 0 0;
83 }
84
85 .submitButton .btn{
86     display: flex;
87     align-items: center;
88     justify-content: center;
89     width: 150px;
90     height: 60px;

```

```

91     border-radius: 30px;
92     background-color: #AEA656;
93     color: #fff;
94     outline: none;
95     border: none;
96     font-size: clamp(12px, 0.9vw, 24px);
97     font-weight: 400;
98 }

```

K. navigation.css –

```
# navigation.css U X
src > styles > # navigation.css > .navigation
1  .navigation {
2      z-index: 1;
3      width: 100%;
4      display: -webkit-box;
5      display: -ms-flexbox;
6      display: flex;
7      -webkit-box-align: center;
8      -ms-flex-align: center;
9      align-items: center;
10     -webkit-box-pack: justify;
11     -ms-flex-pack: justify;
12     justify-content: space-between;
13     padding: 3rem 5rem;
14 }
15
16 .navigation .navlinks {
17     width: 30%;
18     display: -webkit-box;
19     display: -ms-flexbox;
20     display: flex;
21     -webkit-box-align: center;
22     -ms-flex-align: center;
23     align-items: center;
24     -ms-flex-pack: distribute;
25     justify-content: space-around;
26 }
27
28 .navigation .navlinks h4 {
29     color: #000;
30     font-size: clamp(16px, 1.25vw, 36px);
31     font-weight: 400;
32     text-decoration: none;
33 }
34
35 .navigation .navlinks h4:hover {
36     color: #636363;
37 }
38
39 @media screen and (max-width: 960px) {
40     .navigation .navlinks {
41         width: 50%;
42     }
43 }
```

L. search.css –

```
# search.css U X
src > styles > # search.css > #lm
1  #lm{
2      margin-left: 20%;
3      border-top: none;
4      border-left: none;
5      border-right:none;
6      border-color: #AEA656;
7      font-size: clamp(12px, 0.9vw, 24px);
8      font-weight: 400;
9      outline: none;
10
11 }
12 #search{
13     margin-left: 8%;
14     width:40%;
15     border-top: none;
16     border-left: none;
17     border-right:none;
18     border-color: #AEA656;
19     font-size: clamp(12px, 0.9vw, 24px);
20     font-weight: 400;
21     outline: none;
22 }
```

M. table.css –

```
# table.css U X
src > styles > # table.css > .table
1  .table{
2      padding: 5rem 5rem 0 5rem;
3      width: 100%;
4      display: flex;
5      flex-direction: column;
6  }
7
8  .table .row{
9      width: 100%;
10     /* padding: 1rem 0.5rem; */
11     /* border: 1px solid #000; */
12     display: flex;
13 }
14
15 .table .row .number{
16     width: 5%;
17     padding: 1rem 0.5rem;
18     border: 1px solid #000;
19     display: flex;
20     align-items: center;
21     justify-content: space-around;
22 }
23 .table .row .name{
24     width: 40%;
25     padding: 1rem 0.5rem;
26     border: 1px solid #000;
27 }
28 .table .row .price{
29     width: 10%;
30     padding: 1rem 0.5rem;
31     border: 1px solid #000;
32     display: flex;
33     align-items: center;
34     justify-content: space-around;
35 }
36 }
```

```

37 .table .row .amount{
38     width: 7%;
39     padding: 1rem 0.5rem;
40     border: 1px solid #000;
41     display: flex;
42     align-items: center;
43     justify-content: space-around;
44 }
45 }
46 .table .row .total{
47     width: 10%;
48     padding: 1rem 0.5rem;
49     border: 1px solid #000;
50     display: flex;
51     align-items: center;
52     justify-content: space-around;
53 }
54 }
55 .table .row .actions{
56     width: 10%;
57     padding: 1rem 0.5rem;
58     border: 1px solid #000;
59     display: flex;
60     align-items: center;
61     justify-content: space-around;
62 }

```

N. topSection.css –

```

# topSection.css U X
src > styles > # topSection.css > .top-section
1  |.top-section{
2      width: 100%;
3      padding: 3rem 5rem;
4      display: flex;
5      align-items: center;
6      justify-content: space-between;
7  }
8
9  .top-section .card{
10     min-width: 460px;
11     padding: 2rem;
12     background-color: #fff;
13     box-shadow: 8px 16px 32px rgba(188, 221, 160, .25);
14     border-radius: 20px;
15 }
16
17 .top-section .card h2{
18     padding: 1rem 0;
19     color: #187701;
20 }

```

O. Utilities.css –

```
# utilities.css U X
src > styles > # utilities.css > ...
1  @import url("https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display=swap");
2  * {
3      margin: 0;
4      padding: 0;
5      -webkit-box-sizing: border-box;
6      box-sizing: border-box;
7      font-family: "poppins", sans-serif;
8  }
9
10 h1 {
11     font-size: clamp(48px, 3.3vw, 72px);
12     font-weight: 700;
13 }
14
15 h2 {
16     font-size: clamp(36px, 2.5vw, 64px);
17     font-weight: 600;
18 }
19
20 h3 {
21     font-size: clamp(24px, 1.9vw, 48px);
22     font-weight: 500;
23 }
24
25 h4 {
26     font-size: clamp(16px, 1.25vw, 36px);
27     font-weight: 400;
28 }
29
30 h5 {
31     font-size: clamp(12px, 0.9vw, 24px);
32     font-weight: 400;
33 }
34
35 p {
36     font-size: clamp(10px, 0.7vw, 16px);
37     font-weight: 400;
```


8. Output ss (Frontend)–

expense

DashboardHistory

today's total

Rs. 1460.00







add item

Item Name

Price

Amount

Add

Sr no	Name	Price	Quantity	Total	Actions
1	Note	180	2	360	 
1	Note	180	2	360	 
1	Note	180	2	360	 

expense

DashboardHistory

search

Month

Month

January

February

March

April

May

June

July

August

September

October

November

December

Sr no	Name	Price	Quantity	Total	Month
1	Note	180	2	360	January 2021

Backend –

A. Server.js –

```
server > JS server.js > ...
1  const express = require("express");
2  const app = express();
3  const cors = require("cors");
4  require("dotenv").config({ path: "./config.env" });
5  const port = process.env.PORT || 5000;
6  app.use(cors());
7  app.use(express.json());
8  app.use(require("./routes/history"));
9  // get driver connection
10 const dbo = require("./db/conn");
11
12 app.listen(port, () => {
13   // perform a database connection when server starts
14   dbo.connectToServer(function (err) {
15     if (err) console.error(err);
16   });
17   console.log(`Server is running on port: ${port}`);
18 });
```

B. Conn.js –

```
server > db > JS conn.js > ...
1  const { MongoClient } = require("mongodb");
2  const Db = process.env.ATLAS_URI;
3  const client = new MongoClient(Db, {
4    useNewUrlParser: true,
5    useUnifiedTopology: true,
6  });
7
8  var _db;
9
10 module.exports = {
11   connectToServer: function (callback) {
12     client.connect(function (err, db) {
13       // Verify we got a good "db" object
14       if (db)
15       {
16         _db = db.db("trf");
17         console.log("Successfully connected to MongoDB.");
18       }
19       return callback(err);
20     });
21   },
22
23   getDb: function () {
24     return _db;
25   },
26 };
```

C. History.js –

```
server > routes > JS history.js > get() callback
1  const express = require("express");
2
3  // recordRoutes is an instance of the express router.
4  // We use it to define our routes.
5  // The router will be added as a middleware and will take control of requests starting with path /record.
6  const recordRoutes = express.Router();
7
8  // This will help us connect to the database
9  const dbo = require("../db/conn");
10
11 // This help convert the id from string to ObjectId for the _id.
12 const ObjectId = require("mongodb").ObjectId;
13
14
15 // This section will help you get a list of all the records.
16 recordRoutes.route("/history").get(function (req, res) {
17   let db_connect = dbo.getDb("trf");
18   db_connect
19     .collection("trftask")
20     .find({})
21     .toArray(function (err, result) {
22       if (err) throw err;
23       res.json(result);
24     });
25 });
26
27 // This section will help you get a single record by id
28 recordRoutes.route("/history/:id").get(function (req, res) {
29   let db_connect = dbo.getDb();
30   let myquery = { _id: ObjectId( req.params.id )};
31   db_connect
32     .collection("trftask")
33     .findOne(myquery, function (err, result) {
34       if (err) throw err;
35       res.json(result);
36     });
37 });
```

```
server > routes > JS history.js > get() callback
39 // This section will help you create a new record.
40 recordRoutes.route("/history/add").post(function (req, response) {
41   let db_connect = dbo.getDb();
42   let myobj = {
43     name: req.body.name,
44     price: req.body.price,
45     quantity: req.body.quantity,
46     time: req.body.time,
47     date: req.body.date,
48   };
49   db_connect.collection("trftask").insertOne(myobj, function (err, res) {
50     if (err) throw err;
51     response.json(res);
52   });
53 });
54
55 // This section will help you update a record by id.
56 recordRoutes.route("/update/:id").post(function (req, response) {
57   let db_connect = dbo.getDb();
58   let myquery = { _id: ObjectId( req.params.id )};
59   let newvalues = {
60     $set: {
61       name: req.body.name,
62       price: req.body.price,
63       quantity: req.body.quantity,
64       time: req.body.time,
65       date: req.body.date,
66     },
67   };
68   db_connect
69     .collection("trftask")
70     .updateOne(myquery, newvalues, function (err, res) {
71       if (err) throw err;
72       console.log("1 document updated");
73       response.json(res);
74     });
75 });
```

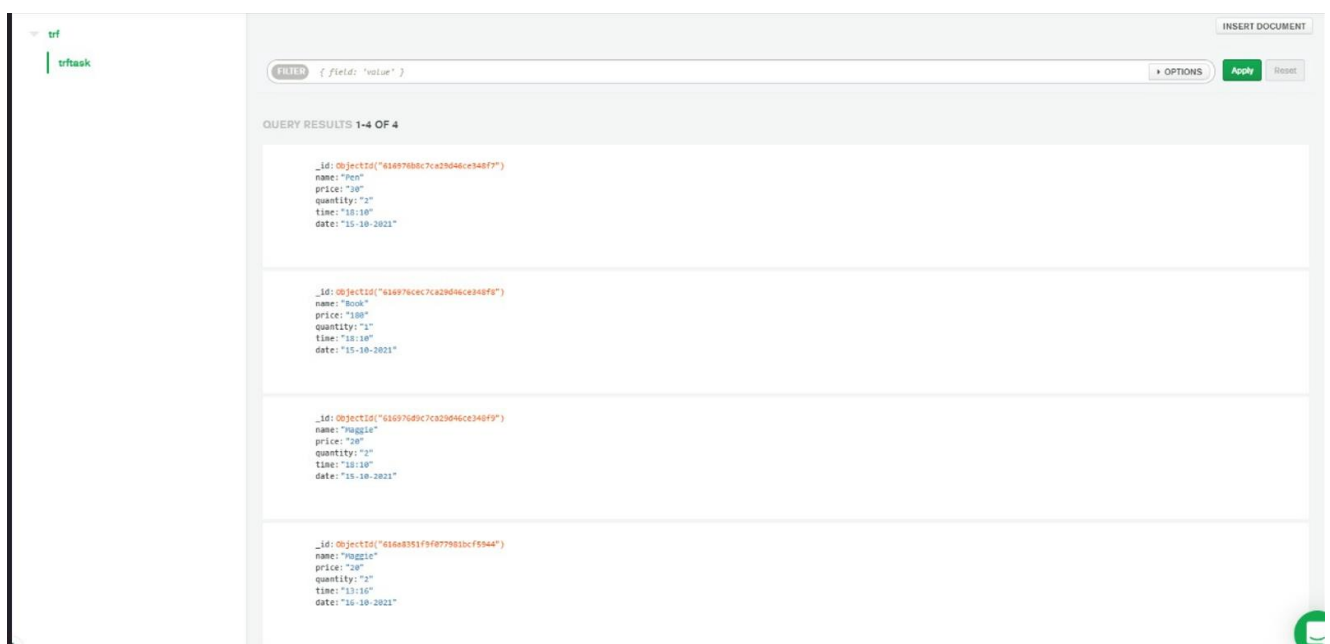
```

75 });
76
77 // This section will help you delete a record
78 recordRoutes.route("/:id").delete((req, response) => {
79   let db_connect = dbo.getDb();
80   let myquery = { _id: ObjectId( req.params.id )};
81   db_connect.collection("trftask").deleteOne(myquery, function (err, obj) {
82     if (err) throw err;
83     console.log("1 document deleted");
84     response.status(obj);
85   });
86 });
87
88 module.exports = recordRoutes;

```

Output ss – (Backend) –

A. MongoDB – Data Collection -



The screenshot shows the MongoDB Compass interface. On the left, the database 'trf' is selected, and the collection 'trftask' is highlighted. The main area displays 'QUERY RESULTS 1-4 OF 4'. Each result is a JSON document with the following structure:

- `_id`: ObjectId (e.g., "6169768bc7ca29d46ce348f7")
- `name`: String (e.g., "Pen")
- `price`: Number (e.g., "30")
- `quantity`: Number (e.g., "2")
- `time`: String (e.g., "10:10")
- `date`: String (e.g., "15-10-2021")

The four documents shown are for items named 'Pen', 'Book', 'Bagg14', and 'Bagg1e'. The interface includes a filter bar at the top with the text '{ field: 'value' }' and buttons for 'OPTIONS', 'Apply', and 'Reset'. An 'INSERT DOCUMENT' button is also visible in the top right corner.

History –

expense

Dashboard

History

search

Sr no	Name	Price	Quantity	Total	Date
1	Pen	30	2	60	15-10-2021
1	Book	180	1	180	15-10-2021
1	Maggie	20	2	40	15-10-2021
1	Maggie	20	2	40	16-10-2021

- Month
- Month
 - January
 - February
 - March
 - April
 - May
 - June
 - July
 - August
 - September
 - October
 - November
 - December

Entering the Data –











Rs. 1460.00

Item Name

Price

Amount

Add

Sr no	Name	Price	Quantity	Total	Actions
1	Pen	10	5	50	 
1	Maggi	20	5	100	 
1	Pen	10	5	50	 
1	Maggi	20	5	100	 
1	Pen	10	5	50	 
1	Maggi	20	5	100	