

PROJECT REPORT

To-Do List



Developed by

Name : MD. RIZWAN FATMI

Address : Patna,Bihar-800 006

Mobile : + 91 8210006010

ACKNOWLEDGEMENT

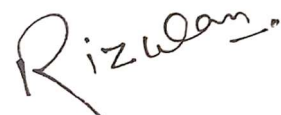
My acknowledgment extends to all those who teach me to code and prepare projects. I am thankful to all of my teachers who give me valuable and very important guidance and information to make this Project Report Successful.

I am grateful to other members of my study center and regional center, Patna for imparting to me the required & necessary training, information, guidelines, sympathy, and criticism which upgraded my programming skills and troubleshooting while doing this project very much.

Lastly, I would like to thank my family, my friends, and my respected seniors for their moral support and other precious co-operations.

Place: Patna.

Date: 24/10/2022.

A handwritten signature in black ink, appearing to read 'Rizwan', with a horizontal line extending from the end.

Signature

PROJECT SUMMARY

SUBJECT / TITLE : “TO-DO LIST”

PROJECT CATEGORIES : WEB BASED PROJECT.

PREPARED BY : MD.RIZWAN FATMI.

ADDRESS : PATNA, BIHAR

MOBILE NO. : + 91 8210006010.

EMAIL : rizwanfatmi@gmail.com.

DATE : 24/10/2022.

TABLE OF CONTENTS

1	Project Title	4
2	Problem Statement.....	4
3	Project Description	4
3.1	Scope of the Work	4
3.2	Project Modules	5
3.3	Context Diagram (High Level)	5
4	Implementation Methodology.....	6
5	Technologies to be used	8
5.1	Software Platform	8
5.2	Hardware Platform	8
5.3	Tools.....	8
6	Advantages of this Project	8
7	Assumptions, if any.....	8
8	Future Scope and further enhancement of the Project.....	9
9	Conclusion	9
10	Source Code.....	10

‘Appendix’

A:Data Flow Diagram (DFD)

B:Entity Relationship Diagram (ERD)

C:Screen Shots

1 Project Title

“TO-DO LIST”

2 Problem Statement

Social media and other easily accessible online distractions make it hard for us to stay focused on our tasks and make it difficult for us to do our work efficiently. The goal of this web application is to help us become more aware of how we spend time in the process of doing tasks and how productive that time is. It can help set some constraints on social media to reduce distraction and track the time we spend working on to-do items. When we have a better sense of the estimated time we'll need to spend on our tasks, along with the validated time spent on the items for reference or personal/team reviews, we are able to manage our daily routines more efficiently.

3 Project Description

It is a web-based application specially built to keep track of tasks that need to be done. This application will be like a task keeper where the user would be able to enter the tasks that they need to do. Once they are done with their tasks they can also remove them from the list.

3.1 Scope of the Work

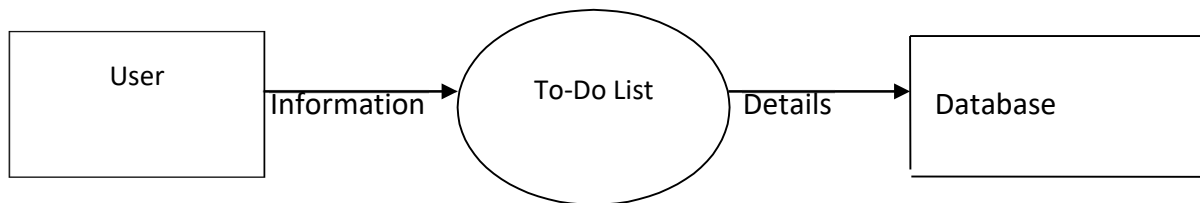
This application-based project deals with the scope of help millions of people who want to manage their tasks, specially for those who does not want to waste their time to making the notes of tasks.

3.2 Project Modules

User Module: -

In this module user can enter, update and delete the information of tasks and can also see the information of tasks in the form of table.

3.3 Context Diagram (High Level)



4 Implementation Methodology

This is a web-based application with real-time database support. This project is based on a real-world problem of every individual who need to manage the tasks. Anyone can opt this facility, We made this project dynamic so that any person who wants to adopt this project can adapt it easily.

Context Level/0 Level DFD of To-Do List:



DATABASE TREE:

ATTRIBUTE	DATATYPE	CONSTRAINTS	PURPOSE
Title	VARCHAR	Not Null	For uniquely identifying the task
Description	VARCHAR	Not Null	For saving the description of task information

5 Technologies to be used

5.1 Software Platform

a) Front-end

HTML 5, CSS3, Bootstrap5, JavaScript

b) Back-end

JavaScript, WebSQL

c) Database:

WebSQL

5.2 Hardware Platform

- **RAM:** 2GB
- **Hard Disk:** 150GB
- **OS:** Windows 10
- **Editor:** Visual Studio Code

5.3 Tools

- **LANGUAGES:** JavaScript
- **DATABASE:** WebSQL
- **IDE:** Visual Studio Code

6 Advantages of this Project

- Help user to manage their task information.
- Time saver application.
- Easy to retrieve information of the task.
- Easy to enter/update and delete task information.

7 Any Assumptions

None

8 Future Scope and further enhancement of the Project

The future scope and the further enhancement of the project have been given below:-

- Login function for user, so that other person cannot insert, delete, update or retrieve the confidential information of tasks.
- More user-friendly and easy to use.

9 Conclusion

Time and tasks are essential for anybody. Implementing this type of system will be highly useful in the current scenario. Nowadays people do not want to waste their time. This application will provide them with ease to manage their tasks information in terms of time, money, comfort, and help. This will be highly useful in this modern and digital era. It will help a lot of the uncountable user today and in the future.

10 Source Code

```
<!--TO DO LIST-->

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>
    To-Do List
  </title>
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet"
    integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWFspD3yD65VohhpUuCOmLASjC"
crossorigin="anonymous">
  <script
src="https://cdn.jsdelivrivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
    crossorigin="anonymous"></script>

</head>

<body>

  <!--BACKGROUND IMAGE SETTINGS-->
  <div style="background:url(Background.jpg); background-size: 100%; width: 100vw;
height: 100vh;">
    <div class="bg"
      style="width: 100vw; height:auto; background:url(Background.jpg); background-
size: 100%; padding-top: 20px; padding-bottom: 30px;">

      <div class="text-center" style="background-color: red; margin-bottom: 20px;">
        <h2 style="font-family:'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
color: rgb(238, 238, 238);"><b>To Do
List</b></h2>
```

```

</div>
<!--TO DO LIST ENTRY FORM DESIGN -->

<div class="container-fluid"
    style="background-color:white; height: auto; width : 700px; padding-left:
40px; padding-right: 40px;padding-top: 20px; border-radius: 23px; border-style:
solid; border-color: rgb(31, 30, 29);">
    <div class="mb-3">
        <label for="exampleFormControlInput1" class="form-label">Title</label>
        <input type="text" class="form-control" id="t1">
    </div>
    <div class="mb-3">
        <label for="exampleFormControlTextarea1" class="form-
label">Description</label>
        <textarea input type="text" class="form-control" id="ds"
rows="3"></textarea>
    </div>

    <!--BUTTONS DESIGNS-->

    <div class="row g-3">
        <div class="col-3">
            <button type="button" class="btn btn-success"
                style="width: 140px;margin-bottom: 40px; margin-top: 10px; background-
color: rgb(42, 163, 68); "
                onclick="Save();">Save</button>
        </div>
        <div class="col-3">
            <button type="button" class="btn btn-info"
                style="width: 140px;margin-bottom: 40px; margin-top: 10px;background-
color: rgb(56, 218, 247);"
                id="upd">Update</button>
        </div>

        <div class="col-3">
            <button type="button" class="btn btn-warning"
                style="width: 140px; margin-bottom: 40px; margin-top: 10px;
background-color: rgb(243, 186, 29);"
                onclick="clr();">Clear</button>
        </div>

        <div class="col-3">
            <button type="button" class="btn btn-danger" style="width: 140px;margin-
bottom: 40px; margin-top: 10px;"

```

```

        onclick="DeleteAll();">Delete All</button>
    </div>
</div>

<!--TO DO LIST TABLE DESIGN-->

<div class="container-fluid"
    style=" margin-top: 20px; background-color:white; height: auto; width :
1000px; padding-left: 72px; padding-right: 72px; padding-top: 20px ; border-radius:
23px; border-style: solid; border-color: rgb(23, 24, 24);">

    <div class="text-center">
        <h2 style="font-family:'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
color: rgb(28, 7, 104);"><b>Your
        List</b></h2>

    </div>

    <div class="table-responsive" style=" margin-top: 30px; height: 25vh;">

        <table class="table" id="tbl" class="sticky-top">

            <tr class="sticky-top" id style="background-color: rgb(255, 255, 255);
color: rgb(9, 7, 34);">
                <th scope="row" style="width: 100px;">Sr.No.</th>
                <th scope="col">Title</th>
                <th scope="col">Description</th>
                <th scope="col" style="width: 40px;">Edit</th>
                <th scope="col" style="width: 40px;">Delete</th>

            </tr>
        </div>
    </table>

</div>
</div>
</div>

```

```

</div>

</body>

<!--JAVA SCRIPT-->

<script>

    //CONNECTING TO DATABASE:

    var db = openDatabase('ToDoList', '1.0', 'Test DB', 2 * 1024 * 1024);

    db.transaction(function (tx) {
        tx.executeSql('CREATE TABLE IF NOT EXISTS ToDoList(title,des)');
    });

    // FUNCTION FOR TABLE DATA:

    function table() {

        document.getElementById("tbl").innerHTML = '<tr class="sticky-top" id
style="background-color: rgb(255, 255, 255); color: rgb(9, 7, 34);"><th><th
scope="row" style="width: 100px;">Sr.No.</th><th scope="col">Title</th><th
scope="col">Description</th><th scope="col" style="width: 40px;">Edit</th><th
scope="col" style="width: 40px;">Delete</th></tr>';

        db.transaction(function (tx) {
            tx.executeSql('SELECT * FROM ToDoList', [], function (tx, results) {
                var len = results.rows.length, i, srno;
                var str = '';
                for (i = 0; i < len; i++) {

                    srno = i + 1;
                    Title = results.rows.item(i).title;
                    Description = results.rows.item(i).des;

                    str += '<tr>';

```

```

        str += "<th>" + srno + "</th>"
        str += "<td>" + Title + "</td>";
        str += "<td>" + Description + "</td>";
        str += "<td>" + '' + "</td>";
        str += "<td>" + '' + "</td>";

        str += "</tr>";
        document.getElementById("tbl").innerHTML += str;
        str = '';
    }
    }, null);
});
}

table();

// FUNCTION FOR SAVING THE DATA INTO DATABASE:

function Save() {
    title = document.getElementById("t1").value;
    description = document.getElementById("ds").value;

    var a = 0;

    db.transaction(function (tx) {
        tx.executeSql('SELECT * FROM ToDoList', [], function (tx, results) {
            var len = results.rows.length, i, srno;
            var str = '';
            for (i = 0; i < len; i++) {

                t = results.rows.item(i).title;
                Description = results.rows.item(i).des;

                if (t == title) {
                    a = a + 1;
                }

                else {
                    a = a + 0;

```

```

    }

}

if (title == "") {

    alert("Please Add Title")

}
else if (description == "") {
    alert("Please Add Description")
}

else if (a == 1) {
    alert("This Title is Already Saved!")
}

else {
    db.transaction(function (tx) {
        tx.executeSql('CREATE TABLE IF NOT EXISTS ToDoList(title,des)');
        tx.executeSql('INSERT INTO ToDoList (title,des) VALUES(' + title +
'', '' + description + '')');
        window.alert("Record Added!!")

        clr();
        table();

    });

}
}, null);
});

}

// FUNCTION FOR DELETING ALL DATA:

function DeleteAll() {

```



```

var result = confirm("Are You Sure You Want To Delete All Data?");
if (result) {

    db.transaction(function (tx) {
        tx.executeSql('SELECT * FROM ToDoList', [], function (tx, results) {
            var len = results.rows.length, i, srno;
            var str = '';
            for (i = 0; i < len; i++) {

                srno = i + 1;
                Title = results.rows.item(i).title;

                tx.executeSql('DELETE from ToDoList WHERE title=?', [Title]);

            }
        }, null);
    });
    clr();
    table();
}

// FUNCTION FOR CLEARING THE FORM:

function clr() {

    document.getElementById("t1").value = ""
    document.getElementById("ds").value = ""

}

// FUNCTION FOR UPDATING THE DATA:

function EditData(button) {

    let num = button.id
    title1 = document.getElementById("tbl").rows[num].cells[1].innerHTML;

    db.transaction(function (tx) {

```

```

        tx.executeSql('SELECT * FROM ToDoList WHERE title=? ', [title1], function (tx,
results) {
    var len = results.rows.length, i;

    title2 = results.rows.item(i).title;
    description2 = results.rows.item(i).des;
    title = document.getElementById("t1").value;
    description = document.getElementById("ds").value;

    document.getElementById("t1").value = "" + title2 + ""
    document.getElementById("ds").value = "" + description2 + ""

    document.getElementById("upd").onclick = function () { up() };

    function up() {

        title = document.getElementById("t1").value;
        description = document.getElementById("ds").value;

        if (title == "" || description == "") {
            alert("Please Select Data!")
        }

        else {
            db.transaction(function (tx) {

                tx.executeSql('update ToDoList set title=? , des=? where title=?',
[title, description, title2]);

                });

            clr();
            table();
        }

    });
});

```

```

    });

    title = ""
    description = ""

}

// FUNCTION FOR DELETING THE DATA

function DeleteData(button) {

    var result = confirm("Are You Sure You Want To Delete This Data?");
    if (result) {
        let num = button.id
        title1 = document.getElementById("tbl").rows[num].cells[1].innerHTML;

        db.transaction(function (tx) {

            tx.executeSql('DELETE from ToDoList WHERE title=?', [title1]);

        });
        clr();
        table();
    }
}

</script>

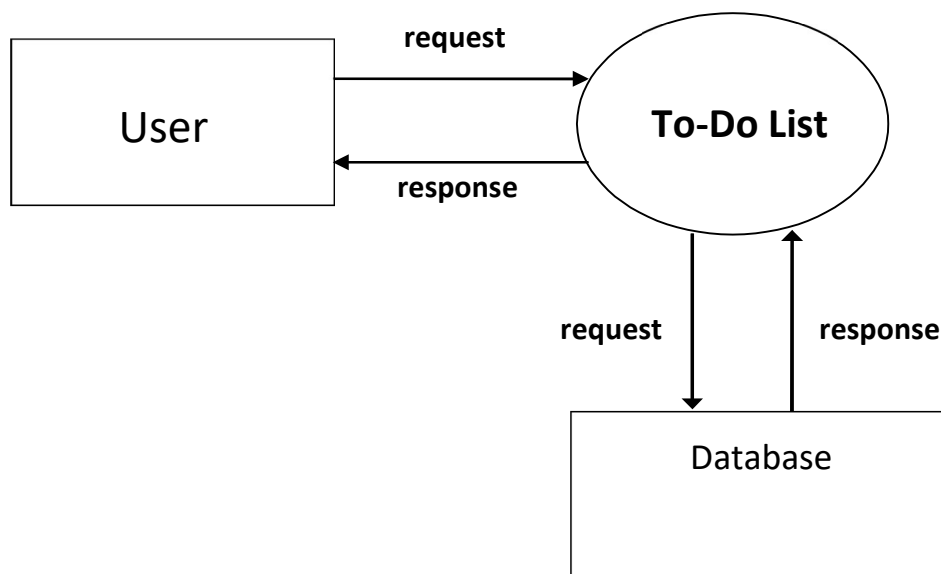
</html>

```

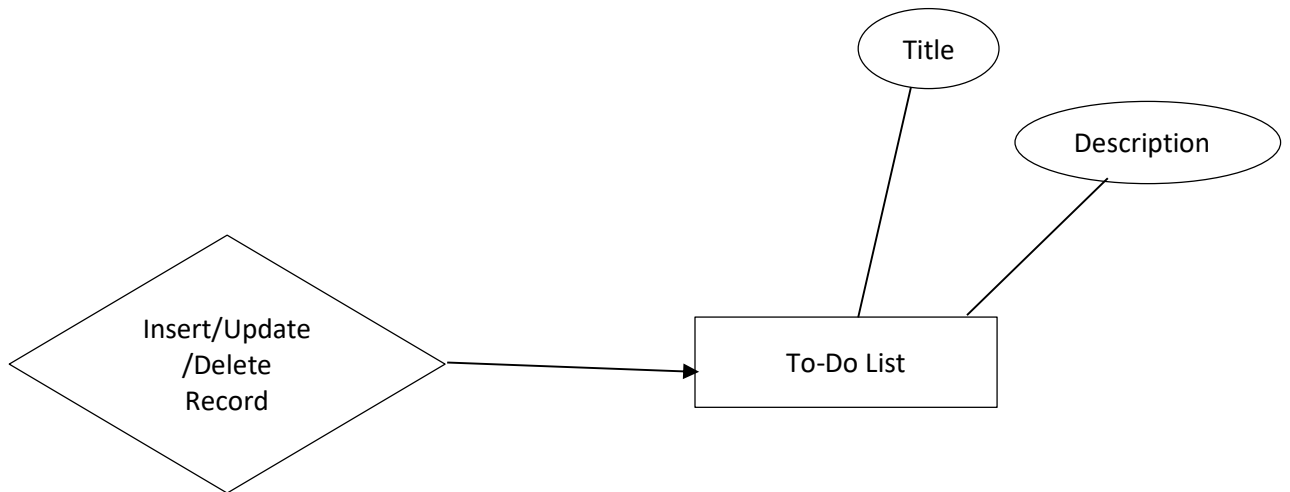
Annexure A

Data Flow Diagram (DFD)

0 – Level DFDs



Entity-Relationship Diagram (ERD)



Annexure C

Screen Shots

Home Page

