**PROJECT NAME: TO-DO APP**

**Submitted by:**

**Gurrala Supriya Reg.No:.20b01a0118  
 Papolu Chandrika Lakshmi Prasana Reg.No:.20B01A0138  
 polipireddy Teja Chinmayi Reg.No:.20b01a0142  
 Sunkara pavitra sri bindhu Reg.No:.20B01A0150**

**UNDER THE SUPERVISION OF**

****

**DEPARTMENT OF CIVIL ENGINEERING**

**SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN(A)**

**Vishnupur, Bhimavaram, West Godavari District , Andhra Pradesh , 534202**

***Abstract:***

The To-Do List project is an application specially built to keep track of errands or tasks that need to be done. This application will be like a task keeper where the user can enter the tasks they need to do. Once they are done with their tasks they can also remove them from the list.

***Index:***

To-do-list-app is an application that allows to manage a list of tasks to do. It performs adding, updating, deleting, and toggling the state of each task. It has a minimalistic design and basic functionality.

***Introduction:***

To-do-list-app is an application designed to help users manage their tasks effectively. It provides a user-friendly interface for adding, updating, and removing tasks, as well as marking them as completed. It has a minimalistic design and basic functionality. It is specially built to keep track of errands or tasks that need to be done. This application will be like a task keeper where the user can enter the tasks they need to do.

* App introduction:
* Open the To-do List app in your web browser.
* Use the provided input field to enter a new task.
* Click the "Add" button to add the task to your list.
* Manage tasks using the options to update, delete, or mark as completed.

***Software requirements:***

In software requirements, we use here Visual Studio Code

***Hardware requirements:***

The Hardware requirements are a summary of a computer’s internal components, including its processor, memory, windows version and other system information. The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems.

***Usage:***

1. Adding a Task:

Enter your task in the input field.

Click the "Add" button to include the task in your list.

2. Updating a Task:

Locate the task you wish to modify.

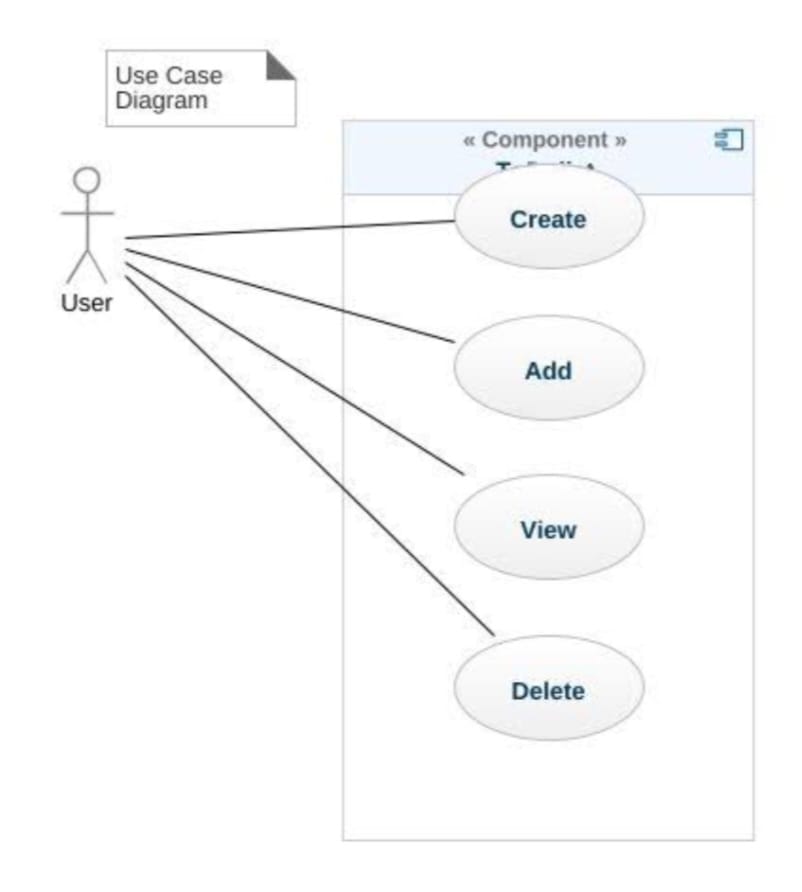
Use the provided option to edit the task content.

3. Deleting a Task:

Identify the task you want to remove.

Select the delete option to eliminate the task from your list.

4. Marking a Task as Completed:

****** Indicate that you've completed a task by using the provided option.

***Case Diagrams:***

***Features:***

**1.** **Add tasks:** Input new tasks to your list.

**2. Update tasks:** Edit the content of existing tasks.

**3. Delete tasks:** Remove tasks from the list.

**4. Mark tasks as completed:** Keep track of tasks you've finished.

**5. User-friendly interface:** Minimalistic design for easy use.

***Advantages:***

* improved organization
* task prioritization 3. reminders
* accessibility across devices.

It helps you stay focused, track progress, and manage your time effectively. Additionally, to-do apps provide collaboration features for team tasks and the ability to set deadlines for better time management.

***Disadvantages:***

To-do list apps can have some disadvantages, such as becoming overwhelming if not managed well. They might lead to over-reliance on technology, potential distractions from notifications, and a sense of guilt or stress if tasks are not completed as planned. Additionally, some people might find it difficult to adapt to a digital system if they prefer physical lists or have privacy concerns related to data stored on the app. It's important to choose an app that suits your preferences and needs while being mindful of these potential drawbacks.

***Code:***

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>To-Do App</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

<style>

body {

font-family: Arial, sans-serif;

background-color: #f8f9fa;

}

.container {

max-width: 800px;

margin: auto;

padding: 20px;

background-color: white;

border-radius: 12px;

box-shadow: 0px 6px 12px rgba(0, 0, 0, 0.1);

}

.app-title {

text-align: center;

margin-bottom: 20px;

color: #007bff;

font-size: 28px;

}

.input-group {

margin-bottom: 20px;

}

.category-select {

background-color: #f8f9fa;

border: none;

}

.add-button {

background-color: #28a745;

border: none;

}

.list-group {

list-style-type: none;

padding: 0;

}

.list-group-item {

display: flex;

justify-content: space-between;

align-items: center;

padding: 10px 20px;

margin-bottom: 8px;

border: none;

border-radius: 8px;

box-shadow: 0px 2px 4px rgba(0, 0, 0, 0.1);

}

.task-box {

display: flex;

justify-content: center;

align-items: center;

width: 40px;

height: 40px;

font-size: 18px;

font-weight: bold;

border-radius: 6px;

}

.completed {

text-decoration: line-through;

}

.work {

background-color: #ff9f43;

color: white;

}

.personal {

background-color: #007bff;

color: white;

}

.shopping {

background-color: #ffc107;

color: black;

}

.edit-delete-icons {

display: none;

}

.list-group-item:hover .edit-delete-icons {

display: block;

}

.edit-icon {

background-color: #17a2b8;

color: white;

padding: 2px 6px;

border-radius: 4px;

margin-right: 5px;

cursor: pointer;

}

.delete-icon {

background-color: #dc3545;

color: white;

padding: 2px 6px;

border-radius: 4px;

cursor: pointer;

}

.edit-task-input {

display: none;

width: 100%;

}

</style>

</head>

<body>

<div class="container">

<h1 class="app-title">Vibrant To-Do App</h1>

<div class="input-group mb-3">

<input type="text" class="form-control" id="taskInput" placeholder="Add a new task...">

<select class="form-select category-select" id="categorySelect">

<option value="work">Work</option>

<option value="personal">Personal</option>

<option value="shopping">Shopping</option>

</select>

<button class="btn btn-primary add-button" id="addTask">Add Task</button>

</div>

<div class="row">

<div class="col-md-4">

<div class="list-group work" id="workList">

<h3 class="category-label">Work</h3>

<!-- Work tasks will be added here dynamically -->

</div>

</div>

<div class="col-md-4">

<div class="list-group personal" id="personalList">

<h3 class="category-label">Personal</h3>

<!-- Personal tasks will be added here dynamically -->

</div>

</div>

<div class="col-md-4">

<div class="list-group shopping" id="shoppingList">

<h3 class="category-label">Shopping</h3>

<!-- Shopping tasks will be added here dynamically -->

</div>

</div>

</div>

</div>

<script>

document.addEventListener("DOMContentLoaded", function () {

const taskInput = document.getElementById("taskInput");

const categorySelect = document.getElementById("categorySelect");

const addTaskButton = document.getElementById("addTask");

const workList = document.getElementById("workList");

const personalList = document.getElementById("personalList");

const shoppingList = document.getElementById("shoppingList");

addTaskButton.addEventListener("click", addTask);

taskInput.addEventListener("keypress", function (event) {

if (event.key === "Enter") {

addTask();

}

});

function createTaskListItem(taskText, category) {

const listItem = document.createElement("li");

listItem.className = "list-group-item d-flex justify-content-between align-items-center";

const taskBox = document.createElement("div");

taskBox.className = "task-box";

taskBox.textContent = "✓";

listItem.appendChild(taskBox);

const taskContent = document.createElement("div");

taskContent.textContent = taskText;

listItem.appendChild(taskContent);

const editDeleteIcons = document.createElement("div");

editDeleteIcons.className = "edit-delete-icons";

editDeleteIcons.innerHTML = '<span class="edit-icon" title="Edit">&#9998;</span><span class="delete-icon" title="Delete">&#128465;</span>';

listItem.appendChild(editDeleteIcons);

const editTaskInput = document.createElement("input");

editTaskInput.className = "edit-task-input form-control";

editTaskInput.value = taskText;

listItem.appendChild(editTaskInput);

listItem.classList.add(category);

return listItem;

}

function addTask() {

const taskText = taskInput.value.trim();

if (taskText !== "") {

const selectedCategory = categorySelect.value;

let taskList;

if (selectedCategory === "work") {

taskList = workList;

} else if (selectedCategory === "personal") {

taskList = personalList;

} else if (selectedCategory === "shopping") {

taskList = shoppingList;

}

const listItem = createTaskListItem(taskText, selectedCategory);

taskList.appendChild(listItem);

taskInput.value = "";

}

}

workList.addEventListener("click", handleTaskAction);

personalList.addEventListener("click", handleTaskAction);

shoppingList.addEventListener("click", handleTaskAction);

function handleTaskAction(event) {

if (event.target.classList.contains("task-box")) {

const listItem = event.target.closest(".list-group-item");

listItem.classList.toggle("completed");

} else if (event.target.classList.contains("edit-icon")) {

const listItem = event.target.closest(".list-group-item");

const taskContent = listItem.querySelector("div:not(.task-box)");

const editTaskInput = listItem.querySelector(".edit-task-input");

taskContent.style.display = "none";

editTaskInput.style.display = "block";

editTaskInput.focus();

editTaskInput.addEventListener("keypress", function (event) {

if (event.key === "Enter") {

taskContent.textContent = editTaskInput.value;

taskContent.style.display = "block";

editTaskInput.style.display = "none";

}

});

} else if (event.target.classList.contains("delete-icon")) {

const listItem = event.target.closest(".list-group-item");

listItem.remove();

}

}

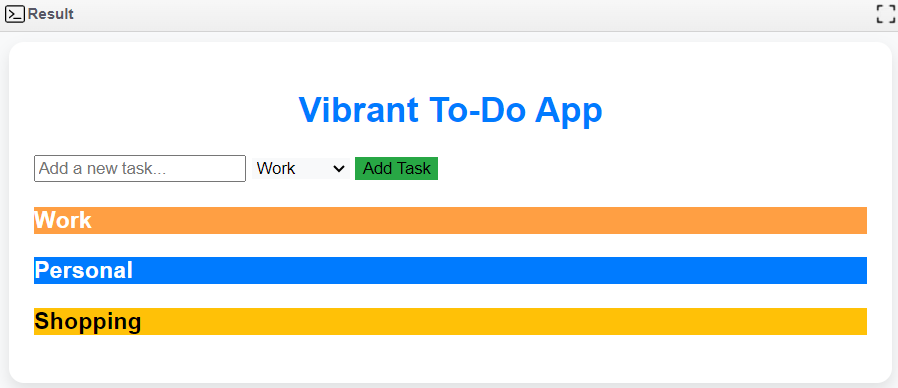
});

</script>

</body>

</html>

***Output:***



***Conclusion:***

In conclusion, a to-do list app can be a valuable tool for enhancing productivity and organization in various aspects of life. It offers benefits such as improved task management, easy access to tasks across devices, and the ability to set reminders and deadlines. However, users should be aware of potential drawbacks like digital overwhelm, distraction, and privacy concerns. The effectiveness of a to-do list app ultimately depends on the user's ability to strike a balance between technology and mindful task management to achieve their goals efficiently.