Project Name:Cloud-Based To-Do List

Team Leader:Ramya priya . J Team member:Nathiya .P Siva Sri .V This subject covers the end-to-end process of creating a cross-platform To-Do List app with cloud storage capabilities, enabling real-time task management across devices..

Process for To-Do List App description

STEP1: Requirements Gathering Understand the app’s purpose and core functionalities: add, update, and delete tasks.Decide on the cloud storage service (Firebase). Define user roles

STEP2: Planning and Design UI/UX Design: Create wireframes for screens like login, task list, and task editor.Database Structure: Define a schema for storing task details (task ID, title, description, status, timestamp).

STEP3:Environment Setup Set up the development environment using Flutter (or other tech stack). Configure Firebase or AWS for database and backend services.

STEP4: Frontend Development Design app screens: Task List Screen Task Editor Screen settings (optional) Implement task CRUD (Create, Read, Update, Delete) functionalities.

STEP5:5. Backend Integration Set up cloud storage: Firebase Firestore or AWS DynamoDB for real-time data storage. Implement APIs for syncing data between devices.

STEP6: Testing Perform unit testing for individual components. Conduct integration testing for data syncing. Test the app across multiple devices to ensure seamless synchronization.

STEP7: Deployment Deploy the app to Google Play Store and/or Apple App Store. Monitor user feedback for improvements.

STEP8:Post-Deployment Maintenance Address bugs reported by users. Add advanced features like reminders or category tagging based on feedback

Code:

Index.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>To-Do List App</title>

<link rel="stylesheet" href="s.css">

</head>

<body>

<div class="container">

<div class="todo-app">

<h2>To-Do List <img src="images/icon.png" alt="icon"></h2>

<div class="row">

<input type="text" id="input-box" placeholder="Add your Text">

<button onclick="addTask()">Add</button>

</div>

<ul id="list-container">

<!-- List items will be dynamically added here -->

</ul>

</div>

</div>

<script src="s.js"></script>

</body>

</html>

Style.css:

\* {

margin: 0;

padding: 0;

font-family: 'Times New Roman', Times, serif;

box-sizing: border-box;

}

.container {

width: 100%;

min-height: 100vh;

background: linear-gradient(135deg, #153677, #4e085f);

padding: 10px;

}

.todo-app {

width: 100%;

max-width: 540px;

background: #fff;

margin: 100px auto;

padding: 40px 30px;

border-radius: 10px;

}

.todo-app h2 {

color: #002765;

display: flex;

align-items: center;

margin-bottom: 20px;

}

.todo-app h2 img {

width: 30px;

margin-left: 10px;

}

.row {

display: flex;

align-items: center;

justify-content: space-between;

background: #edeef0;

border-radius: 30px;

padding-left: 20px;

margin-bottom: 25px;

}

input {

flex: 1;

border: none;

outline: none;

background: transparent;

padding: 10px;

font-size: 14px;

}

button {

border: none;

outline: none;

padding: 16px 50px;

background: #ff5945;

color: #fff;

font-size: 16px;

cursor: pointer;

border-radius: 40px;

}

ul li {

list-style: none;

font-size: 17px;

padding: 12px 8px 12px 50px;

user-select: none;

cursor: pointer;

position: relative;

}

ul li::before {

content: '';

position: absolute;

height: 28px;

width: 28px;

border-radius: 50%;

background-image: url(images/unchecked.png);

background-size: cover;

background-position: center;

top: 8px;

left: 8px;

}

ul li.checked {

color: #555;

text-decoration: line-through;

}

ul li.checked::before {

background-image: url(images/checked.png);

}

ul li span {

position: absolute;

right: 0;

top: 5px;

width: 40px;

height: 40px;

font-size: 22px;

color: #555;

line-height: 40px;

text-align: center;

border-radius: 50%;

}

ul li span:hover {

background: #edeef0;

}

Script.js:

const inputBox = document.getElementById("input-box");

const listContainer = document.getElementById("list-container");

function addTask() {

if(inputBox.value === '') {

alert("You must write something");

} else {

let li = document.createElement("li");

li.innerHTML = inputBox.value;

// Create the delete button (span)

let span = document.createElement("span");

span.innerHTML = "\u00d7";

li.appendChild(span);

listContainer.appendChild(li);

inputBox.value = ""; // Clear the input field

saveData();

}

}

listContainer.addEventListener("click", function(e) {

if(e.target.tagName === "LI") {

e.target.classList.toggle("checked");

saveData();

} else if(e.target.tagName === "SPAN") {

e.target.parentElement.remove();

saveData();

}

}, false);

function saveData() {

localStorage.setItem("data", listContainer.innerHTML);

}

function showTask() {

const savedData = localStorage.getItem("data");

if (savedData) {

listContainer.innerHTML = savedData;

}

}

showTask(); // Show tasks on page load

output:

