Table of Contents (TOC)

Acknowledgement

We express our sincere gratitude to our faculty and institution for giving us the opportunity to work on this project titled **“Cloud-Based To-Do List Using Firebase Realtime Database.”**
The technical guidance and academic environment provided by the department played a crucial role in the successful completion of this cloud computing project.

Working on this project enhanced our understanding of cloud computing concepts, real-time database syncing, and client-side JavaScript development.

Abstract

This project presents a **Cloud-Based To-Do List Application** built using **Firebase Realtime Database** and **JavaScript**. The application enables users to create, update, and delete their tasks, with all data stored online in a cloud-hosted database.

Since the database is real-time, any changes made by the user are instantly reflected on all connected devices without requiring manual refresh. This project demonstrates fundamental cloud concepts such as serverless architecture, NoSQL cloud databases, real-time synchronization, and CRUD operations.

Introduction

The purpose of this project is to design and develop a simple yet powerful cloud-based task management application using Firebase.

As cloud computing has become an integral part of modern applications, this project aims to implement a small but fully functional real-time cloud app.

The application uses:

- **HTML + CSS** for frontend design
- **JavaScript (ES6)** for logic
- **Firebase Realtime Database** for backend storage

This To-Do List application stores user tasks in a cloud-hosted JSON structure, ensuring persistent availability across multiple devices. Each change in task data triggers automatic UI updates due to Firebase's built-in event listeners.

This project provides a practical demonstration of cloud technologies and is ideal for understanding how modern serverless web applications function.

.

Cloud Computing Background

Cloud computing allows data, applications, and services to be stored and processed on remote servers rather than on local devices.
Key cloud concepts used in this project:

1. Cloud Storage

Data is saved on internet-hosted servers rather than stored locally.

2. Serverless Architecture

Firebase handles the backend internally — no need for servers.

3. Scalability

Firebase automatically scales depending on usage.

4. Real-Time Data Sync

Changes made by one client instantly reflect on all clients connected.

5. NoSQL JSON Database

Data stored in Firebase is structured as key-value JSON pairs.

This project integrates all these cloud computing concepts into one functional application.

Firebase Realtime Database Background

Firebase Realtime Database is a NoSQL database that stores data in JSON format.

Key features:

- Cloud-hosted
- Real-time synchronization
- Automatic scalability
- Offline caching
- Event-based updates
- Key-based data structure

Why Firebase for this project?

1. No server setup required
2. Simple SDK-based integration
3. Real-time updates are built-in
4. Easy to store structured JSON data
5. Perfect for small and medium cloud projects

Objectives

Technical Objectives

- To integrate Firebase SDK into a web application
- To perform CRUD operations on a cloud database
- To understand JSON-based data modeling
- To use event listeners for real-time data sync

Functional Objectives

- Add tasks to the cloud
- Mark tasks as completed
- Delete tasks from cloud
- Display tasks dynamically on screen
- Sync tasks across all devices

System Requirements

Hardware Requirements

- PC/Laptop
- Minimum 2 GB RAM
- Stable Internet Connection

Software Requirements

- Web Browser (Chrome recommended)
- Firebase Console
- Code Editor (VS Code)

Cloud Requirements

- Firebase Project
- Realtime Database enabled
- API configuration

Implementation Details

1. Adding a Task

- User enters text
- JavaScript calls push () to generate a new key
- Firebase stores task with fields:
    - text
    - completed
    - createdAt

2. Reading Tasks

- onValue() is used
- Firebase sends entire database snapshot
- JS loops through tasks and displays them
- Syncs instantly if database changes

3. Updating a Task

- Checking a box triggers update ()

- Only “completed” field is changed

4. Deleting a Task

- "Delete" button triggers remove ()

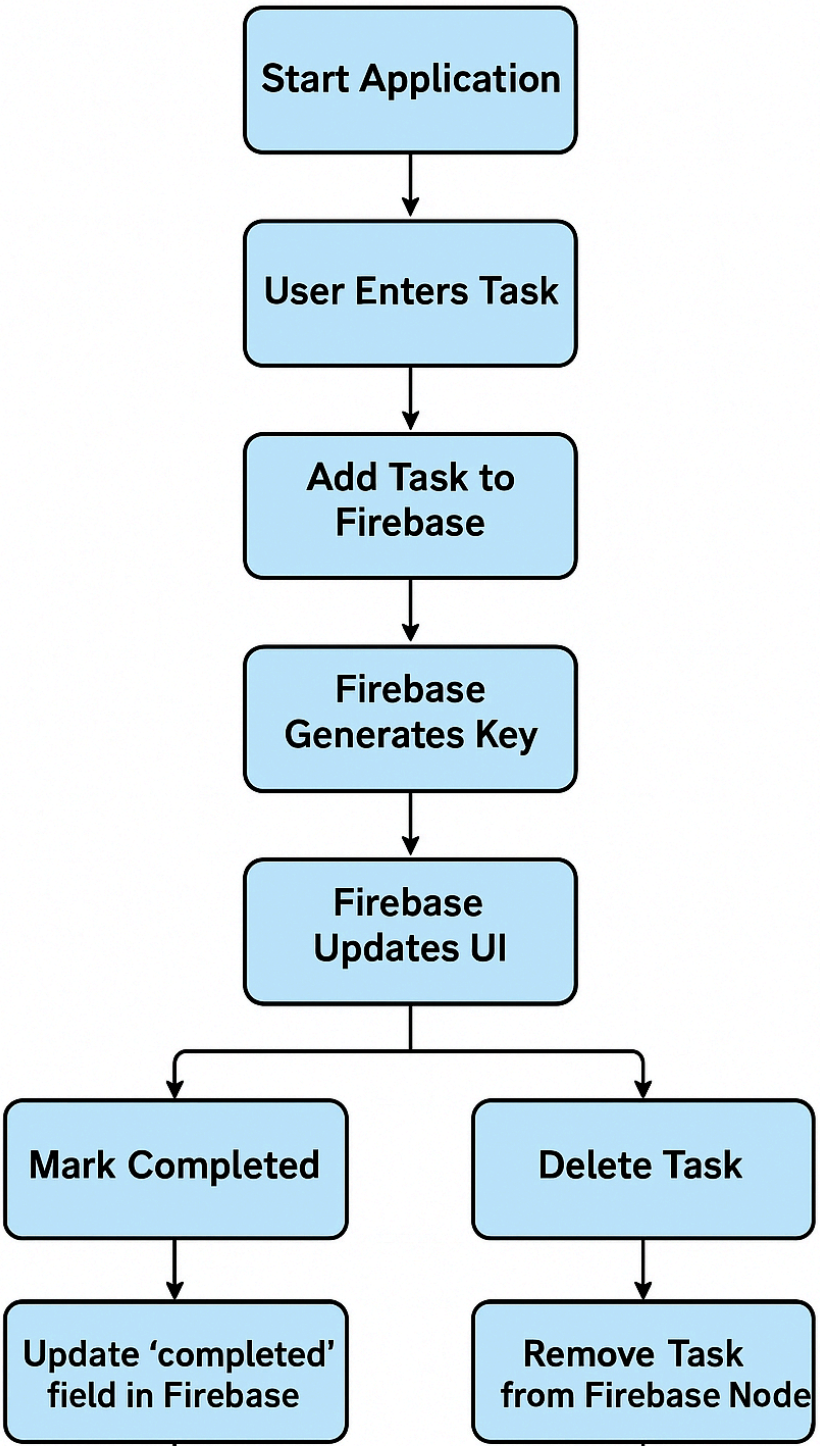
- Task removed from cloud and UI

5. JSON Database Structure

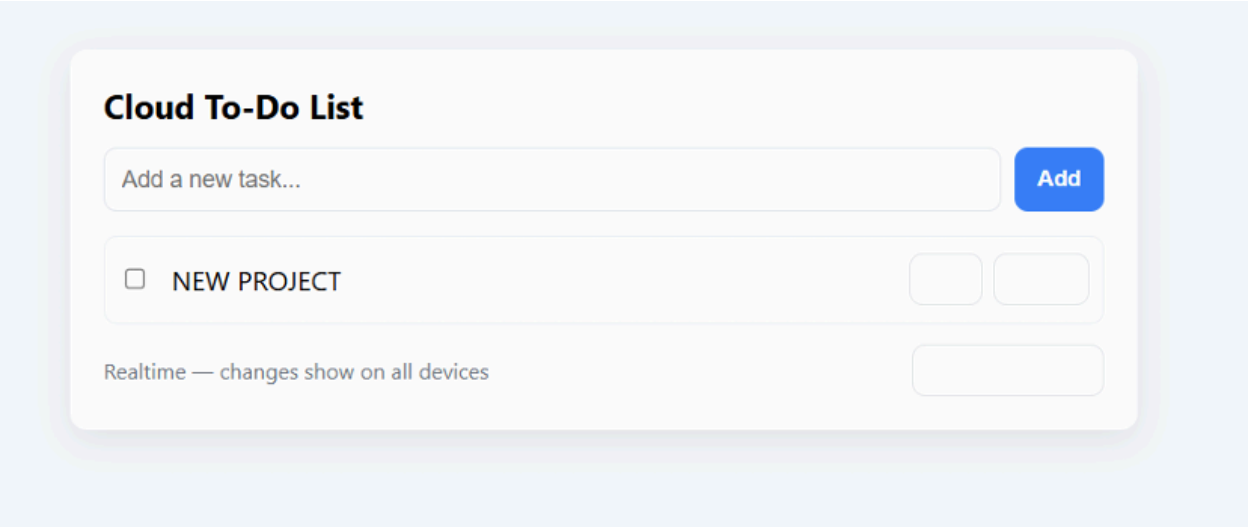
tasks

```
├── -Nabc123xyz
│    ├── text: "Buy Groceries"
│    ├── completed: false
│    └── createdAt: 1710020112
└── -Nddf564pqr
     ├── text: "Study Cloud Computing"
     ├── completed: true
     └── createdAt: 1710020219
```

FLOWCHART


Start Application
User Enters Task
Add Task to Firebase
Firebase Generates Key
Firebase Updates UI
Mark Completed
Delete Task
Update 'completed' field in Firebase
Remove Task from Firebase Node

Display Output :



Conclusion

The “Cloud-Based To-Do List Application” successfully demonstrates the practical use of cloud services in modern web development. By using Firebase Realtime Database, this project achieves:

- Real-time synchronization
- Serverless backend
- Persistent cloud data storage

- User-friendly interface
- Simple and scalable architecture

This project provides a solid foundation for learning cloud technologies and understanding how modern web applications handle data online.