

# Project Report: DeepDrive – A Cloud-Based File Storage System

## 1. Project Overview

DeepDrive is a cloud-based file storage application similar to Google Drive. It allows users to sign up, log in securely, and upload, download, rename, and delete their files. The app is built using **Firebase (Authentication, Cloud Storage, Hosting, and Realtime Database)**, and **JavaScript with HTML/CSS** on the frontend.

---

## 2. Cloud Features Used

### Firebase Authentication

- Secure sign-up and login using **email and password**.
- Session handling with `onAuthStateChanged()` to restrict unauthenticated access.
- UI updates dynamically based on user login status.

### Firebase Cloud Storage

- Users can upload files directly to `uploads/` folder in Cloud Storage.
- Files can be downloaded, renamed, or deleted.
- Upload progress is shown using a dynamic progress bar.

### Firebase Hosting

- The complete frontend is deployed and hosted via **Firebase Hosting**.
- Offers fast global CDN and HTTPS support.

### Firebase Realtime Database (*Added for logging actions*)

- Every time a file is uploaded or deleted, the action is logged in the **Realtime Database** with:
  - Timestamp

- File name
- Action type (upload/delete)
- User email

Example entry in Realtime DB:

```
json
CopyEdit
{
  "logs": {
    "-Nxyz123": {
      "action": "upload",
      "file": "resume.pdf",
      "timestamp": "2025-04-30T15:43:12Z",
      "user": "daksh@example.com"
    }
  }
}
```

#### Email Notifications (Optional via Integration)

- Using a **Firebase Cloud Function + SendGrid/SMTP**, you can trigger an **email notification** whenever:
  - A file is uploaded
  - A file is deleted

Example:

Subject: "📁 New Upload - resume.pdf"

You just uploaded a file at 3:43 PM. View it on DeepDrive.

---

### 3. Tech Stack

Layer	Tools Used
Frontend	HTML, CSS, JavaScript, Toastify.js

Backend (Cloud)    Firebase Authentication, Storage, Realtime Database, Hosting

Notifications        Firebase Functions + SendGrid (Optional)

---

#### **4. Security & Access Control**

- Users must be logged in to access file upload/download features.
  - All file operations are scoped to authenticated users.
  - Future improvements may include user-specific storage folders.
- 

#### **5. Future Improvements**

- Show file size and type
- Support for folders and drag-n-drop
- Version control for uploaded files
- User profile page with usage analytics

#### **6. ScreenShots**





