



THE YENEPOYA INSTITUTE OF ARTS SCIENCE COMMERCE AND MANAGEMENT

(a constituent unit of Yenepoya Deemed to be University)

# PROJECT SYNOPSIS

**Secure File Storage with IBM Cloud Object Storage**

MASTER OF COMPUTER SCIENCE AND APPLICATIONS

SUBMITTED BY : GUIDED BY:

MR. SHASHANK

Adeebah Gulzar 24MCA001

Asma Suhana 24MCA003

Ankith KV 24MCA002



**TITLE PAGE**

1. Name of the student: Adeebah Gulzar
2. Class Roll No. 24MCA001
3. Campus ID: 38638
4. Present official Address: YIASCM Blamatta, Mangalore 575002
5. Email: [38638@yenepoya.edu.in](mailto:38638@yenepoya.edu.in)
6. Phone No. +91 7795748534
7. Branch: Computer Science
8. Batch: 2024-2026
9. Proposed Topic: Secure File Storage with IBM Cloud Object Storage



# TABLE OF CONTENTS

Cover page - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 1

Title - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 2

Content - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 3

* 1. Introduction - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 4
  2. **Key Features**
  3. **Technology Stack**
  4. **Specialized Field: Cloud Computing and Virtualization**

1. Methodology- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - 5
   1. **Requirement Analysis and Tool Selection**
   2. **System Architecture and Design**
   3. **Frontend Development**
   4. **Backend Integration**
   5. **Final Testing and Documentation**
2. Facilities required for proposed work - - - - - - - - - - - - - - - - - - 6
   1. **Development Environment**
   2. **Deployment Tools**
   3. **Testing Tools**
   4. **Reporting Tools**



1. **Introduction**

The Secure File Storage System is a cloud-based project designed to store and retrieve data securely using IBM Cloud Object Storage. It provides users with the ability to upload and download files through a web interface or application, with robust access control mechanisms provided by IAM policies.  
  
Using IBM’s S3-compatible interface and cloud IAM capabilities, the system ensures encrypted data storage, secure access, and simplified integration with web applications. It offers a scalable, highly available, and cost-effective solution for real-time file storage needs in academic or organizational environments.

* 1. **Key Features**
* Secure File Upload and Download – Users can upload files to cloud storage and retrieve them securely using generated URLs or a web interface.
* Access Control using IAM – Ensures only authorized backend systems can access storage resources.
* Data Encryption – Files are encrypted at rest and during transmission.
* API Integration – Enables backend connectivity using Node.js/Python SDKs.
* S3-Compatible SDK Support – Seamless integration using AWS SDK for JavaScript/Python.
* Scalable Storage Infrastructure – Uses IBM Cloud’s highly available Object Storage.
  1. **Technology Stack**

**Frontend:**

* + HTML/CSS (Basic UI for upload/download)
  + Optionally: React JS

**Backend:**

* Node.js (Express framework)
* AWS SDK for JavaScript (for IBM COS integration)

**Cloud Services:**

* IBM Cloud Object Storage
* IBM IAM (for API key and HMAC-based access)  
  1. **Specialized Field: Cybersecurity and Ethical Hacking**

This project falls under the domain of **Cloud Computing** and **Virtualization domain**. It demonstrates secure data handling in the cloud, emphasizing access control, encrypted storage, and the use of virtualization friendly infrastructure offered by IBM Cloud.



1. **Methodology**
   1. **Requirement Analysis & Tool Selection**

* Define use cases: secure file upload/download, admin-only access control.
* Select technologies: IBM Cloud Object Storage, Node.js, AWS SDK.
* Ensure secure communication and credential protection.
  1. **System Architecture and Design**
* Design an architecture with three layers: frontend → backend → cloud storage.
* Backend uses AWS SDK configured with HMAC credentials.
* Object Storage bucket is created with restricted permissions via IAM policies.  
  1. **Frontend Development**
* Create an HTML or React-based interface.
* Use forms to allow file selection and upload.
* Fetch file URLs or trigger download links dynamically.
  1. **Backend Integration**
* Use Node.js and Express to handle /upload and /download/:filename routes.
* Connect to IBM Cloud using AWS SDK.
* Handle file streaming, temporary storage, and error management.
  1. **Final Testing and Documentation**
* Test with different file types and sizes.
* Verify access control: only backend can access the bucket.
* Document architecture, endpoints, and credentials setup.



**Facilities required for proposed work**

The developing of this IDS project requires all of these software.

* 1. **Development Environment**
* IBM Cloud Account
* Node.js & NPM
* Visual Studio Code
* Git for version control
  1. **Deployment Tools**
* IBM Cloud Console
* IBM CLI (optional)
* Postman (for API testing)  
  1. **Testing Tools**
* Postman
* Web Browser (for frontend)
* Console logs and error handlers
  1. **Reporting Tools**
* MS Word / Google Docs – Project documentation
* ReportLab (optional) – For auto-generated PDF logs from backend
* Lucidchart / Draw.io – For system architecture diagrams