



# FILE SHARING PLATFORM

**Ritik Tripathi**

MCA Student

**Chandan Kumar**

MCA Student

**Mahesh**

MCA Student

**Anand Kumar Thakur**

MCA Student

**Akshay Sharma**

MCA Student

School of Computer Applications

Lovely Professional University, Phagwara, India

**Abstract:** This research paper investigates the development of a secure and cost-effective file sharing platform aimed at preventing data breaches and reducing financial burdens for users. The primary objective is to enhance data security measures to protect user information from unauthorized access, leaks, or breaches. Current file sharing platforms often pose risks to user data and incur significant costs for individuals or organizations in terms of data breaches and storage expenses. Our research focuses on implementing robust security protocols and efficient data management strategies to mitigate these risks and reduce financial strain on users. By addressing these challenges, we aim to provide a safer, more affordable, and user-friendly file sharing solution for a wide range of users.

**IndexTerms - Secure data transfer, Cost efficiency, Security protocols.**

## 1. INTRODUCTION

The words "uploading" and "downloading" frequently suggest that the material being given or received will be kept around for a long time, if not permanently. By contrast, the term "downloading" is used to describe a process that implies that the data is only usable after it has been received in its entirety. The related concept of "streaming" refers to the receiving of data that is used almost immediately as it is received, while the transmission is still ongoing, and may not be stored for a long time. More and more websites that provide streaming or in-browser video, like YouTube, and restrict users' ability to store these materials to their computers once they are received, state that downloading is prohibited. Download in this sense indicates "receive and save" rather than just "receive". [1] It's crucial to understand that downloading and "transferring" are not the same things. For example, downloading is different from "transferring" since downloading involves getting data from the Internet while transmitting and receiving data between two storage devices.

Different forms of electronic data transfers are referred to by the terms "downloading" and "uploading." The direction in which the files are being moved is what separates them. When files are moved from a computer or other electronic device to a central server, they are said to be uploaded, and when they are moved from a server to a smaller peripheral device—like a computer, smartphone, or other device—they are said to be downloaded. [2] When a file is downloaded from a website, for example, these two distinct sorts of transfers are frequently carried out over the Internet. Moving data from one system or device to another that is similar, such a desktop computer, is not seen as uploading or downloading.

## 2. RELATED WORK

The current data storage and retrieval systems are both costly and limited in accessibility. They primarily cater to paid users, requiring credit card information for access. Free services are often limited in duration, leading to user constraints. Additionally, these systems lack robust authentication measures, making them less secure and unreliable for widespread use. Key drawbacks include high costs, inadequate security, and overall unreliability.

### A. Existing Systems:

- 1) Google Drive
- 2) FTP (File Transfer Protocol)

### B. Disadvantages of Existing systems

- 1) In Google we are not able to store data more than 15GB.
- 2) FTP (File Transfer Protocol) if User relates to another user but user forgot to Logout then other user not able to connect.
- 3) Non developer user not able to understand how to use system.
- 4) Hidden costs and shipping charges
- 5) Bad User-Interface

### 3. PROJECT SCOPE

The project scope encompasses a broad range of users, including individuals and organizations, offering a versatile software solution for seamless data transfer. At a domestic level, users can leverage the software to share various types of content such as presentations, assignments, programs, and video clips quickly and efficiently. This functionality is not limited to students but extends to anyone requiring swift and secure data sharing capabilities.

Moreover, at an industrial level where data privacy and security are paramount, our software excels in providing robust security measures. These include encryption protocols, access controls, and data integrity checks to ensure that sensitive information remains protected during transfer. Furthermore, the software supports unlimited data transfer capacities, catering to the substantial data needs of industrial users without compromising on security or performance.

#### 3.1 USER CLASSES AND CHARACTERISTICS

The application is accessible to users across diverse settings, from organizations to individuals. It allows users to manage, upload, and download data on a network with ease. The login screen requires a specific Username and Password for access, offering a user-friendly interface suitable for users of all skill levels.

Additionally, the application introduces an offline mode, enabling users to work with their data even without network connectivity. This feature enhances flexibility and convenience, making the application versatile for various use cases.

##### 3.1.1 Operating Environment

Regarding operating systems, F-Secure works with: -All 64-bit Microsoft Windows versions -All 32-bit Microsoft Windows (95/98/2000/XP/Vista)

Every POSIX operating system, which includes all Linux, BSD, and UNIX-like variants. Furthermore, the front end and back end of F-Secure are open source, compatible with practically any operating system. It is apparent that F-Secure is not dependent on the computer system's operating system. Any web browser, including Mozilla, Internet Explorer, Netscape Navigator, and Opera, can view this web-based programme.

##### 3.1.2 Design and Implementation Constraint

The project's design and implementation constraints include the need for the primary server to have constant uptime, client systems' compatibility with web browsers such as Google Chrome, Mozilla Firefox, and Flash Player, and the need for development tasks to require proficiency with HTML, JavaScript, and PHP. It's also advised to have enough system resources to ensure that activities are completed quickly, and to protect data while it's being transmitted, secure communication protocols like HTTPS must be implemented. To guarantee the best possible speed, security, and developer collaboration, regular upgrades, maintenance tasks, and thorough documentation are also essential.

##### 3.1.3 Assumptions and Dependencies

Since the authentication code will be sent to the same cell phone number, each user must voluntarily enter their mobile number. Each new visitor must complete the registration form and create a password and user ID. The user must enter the right file password, authentication key, and Captcha code while downloading the data. We presume that the user wouldn't need any further documentation to use this software, beyond what is included in this report.

### 3.2 EXTERNAL INTERFACES REQUIREMENTS

#### 3.2.1 User Interfaces

To access the software and data, a unique username and password are required, as shown on the login screen of the initial interface. Individuals have the choice to decide whether they would like to download or upload data. The user interfaces of the application are particularly noteworthy for their exceptional functionality, ease of use, and accessibility, especially for people with less computer system experience. To further improve usability, the interface includes interactive components, clear directions, and user-friendly navigation. An intuitive user experience is enhanced by the interface's responsiveness and flexibility to various devices.

#### 3.2.2 Hardware Interfaces

Although the system doesn't directly interface with hardware, it relies on an internet connection for functionality. The installation and usage requirements are minimal, making it accessible on any computer system regardless of the operating system. The only hardware prerequisite is a compatible CPU, ensuring usability for a wide range of users.

#### 3.2.3 Software Interfaces

The system indirectly uses an online browser in addition to the internet connection. The code doesn't instruct any software, including the browser, on what to perform outside of PHP and HTML.

HTML: This language is used to create different forms and pages.

PHP: Data is carried by it.

JavaScript: This script is used for looping and logical calculations.

CSS: This is how the elements are arranged on pages and in forms.

Furthermore, the front end and back end of platform are open source, compatible with practically any operating system. It is

apparent that it is not dependent on the computer system's operating system. This web-based programme can be accessed and used.

### 3.2.4 Communication Interfaces

The system relies on an internet connection to establish communication with the database, facilitating access to software and information. However, the code does not directly control network controllers' operations. Additionally, communication interfaces include:

**MySQL database:** Multiple systems communicate through a shared database stored on the main server, leveraging the benefits of an open-source and freely available platform for efficient data management.

**Local Area Network (LAN):** Utilized for data transmission, LAN provides a reliable and secure communication channel within a specific geographical area, enhancing the system's connectivity and performance.

## 3.3 NON -FUNCTIONAL REQUIREMENTS

### 3.3.1 Performance

Bandwidth is a crucial requirement for the system's effective performance. While the minimum requirement is approximately 256 Kbps, providing a bandwidth of around 1 Mbps ensures smooth operation. Additionally, the system's performance can be optimized by implementing offline modes for certain functionalities, allowing users to work seamlessly without an internet connection. Online capabilities for data sharing further enhance the system's versatility and user experience, providing a balanced approach to performance optimization's system to perform precisely or affectively bandwidth is themajor requirement. Even if minimum requirement is around 256KBPS, system to run smoothly, must be provided bandwidth of around 1MBPS.

### 3.3.2 Safety

To keep the uploading/downloading data safe, there should notbe any voltage fluctuations of computer system. This application does not run in cases of wrong password/keyinsertion or wrong settings. In case of error, it provides an alert message to users with appropriate help messages.

### 3.3.3 Security

The private key, which is generated by the system, needs to be kept secret from everyone save the recipient. Those who are not legitimate or authenticated users shouldn't be able to access the database. It is recommended that IDs and passwords have a minimum length and only use alphanumeric characters for both key generation and password creation.

## 3.5 OTHER REQUIREMENTS

### 3.5.1 DB Requirements

Open-source MySQL is a relational database management system (RDBMS) that functions as a server and allows multiple users to access various databases. MySQL is widely used as the database component of LAMP software stacks for online applications. Because PHP is frequently used in conjunction with MySQL, its popularity for use with web applications is tightly linked to that of PHP. MySQL is utilised by numerous popular websites, such as Flickr, Facebook, Wikipedia, Google (not for searches), Nokia, and YouTube, for user data logging and data storage. One of the best databases on the market right now is MySQL.

A relational database with a tonne of sophisticated features and settings is MySQL. With time, MySQL has shown itself to be a quick, dependable, and affordable alternative to more costly databases like MS SQL Server and Oracle. The following are some benefits of database development with MySQL. MySQL may be effortlessly installed and used on various operating systems, such as Windows, Linux, OS2, and Solaris. Development businesses find it to be a favourable option due to its cross-platform operability. Moreover, the MySQL database system has APIs for integrating with C, C++, PHP, Java, Perl, Python, and Ruby, among other languages. It is simple to connect to several development platforms, allowing you to integrate programmes created on various operating systems and development environments.

Because all access passwords are kept in an encrypted manner, preventing any unauthorised access to the system, MySQL is a safe relational database. Additionally, it encrypts the transactions to prevent database transactions from being replicated or generated after they are executed by eavesdroppers and data harvesting programmes.

Features:

- MySQL has very high performance due to its unique storage engine architecture. Because it can support a huge range of embedded applications, MySQL is incredibly adaptable. Using stored procedures, views, and triggers to increase development productivity. Permits commit, rollback, and crash recovery of transactions.
- Triggers & cursor.
- Open Source
- Fast Development

## 4. SYSTEM ARCHITECTURE

### 4.1 MVC

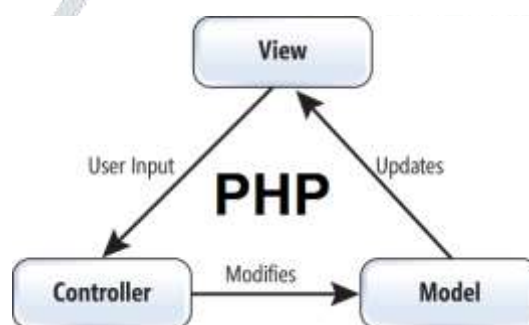
#### System Flow

As you can see above, breaking out your project into three sections—application development, database and data processing, and interface development—will inevitably make it easier for developers to grasp and increase productivity.

- Model: It manages the database and data processing portions. Model responds to events that the controller sends. Following the processing of these events, the processed data is sent either directly to the view side or to the controller, who may choose to reprocess it.

- View: View gets an interface ready for the user to see. Models or controllers instruct views on what to display to users. Additionally, view responds to user requests and notifies controller.

- Controller: Assume that the controller serves as the system's brain. That's accurate. because it prepares the model and view, among other components of the system, and processes each request. The controller's directives then tell the system what to do. Even as a junior developer, you have undoubtedly encountered the issue known as "Complexity of Project." MVC appears there. MVC aids in reducing project complexity. For example, there is a five-person squad. If they continue to work on the same project, eventually it reaches a stage where nobody can comprehend what it is going to be about. MVC divides a project into three distinct sections by default. such as Controller, View, and Model. [4]



#### Data Flow Diagram:

The data flow diagram for our file sharing platform illustrates the movement of data across various components of the system. At the core of the diagram is the user interface, where users interact with the platform to perform actions such as uploading and downloading files.

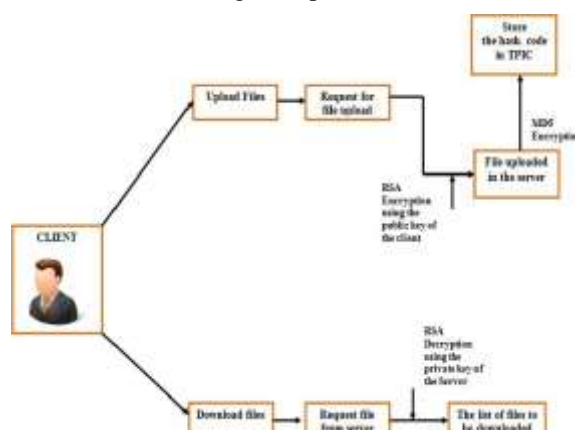
Upon initiating an upload request, the data flow begins as the file data is transmitted from the user's device to the platform's server. This data is then processed by the server, which stores the file in the appropriate location within the system's database.

Simultaneously, metadata related to the uploaded file, such as file name, size, and user details, is also captured and stored in the database. This metadata is crucial for organizing and managing files within the platform.

On the download side, when a user requests to download a file, the data flow reverses. The platform retrieves the requested file from the database and transmits it back to the user's device. This process ensures secure and efficient file sharing between users of the platform.

Additionally, the DFD may depict other components such as authentication systems for user login, data validation processes to ensure file integrity and security, and administrative functions for managing user accounts and system settings.

Overall, the data flow diagram provides a visual representation of how data moves through the file sharing platform, highlighting key interactions and processes that occur during file uploads and downloads.



## 4.2 CodeIgniter

PHP developers may create dynamic websites with CodeIgniter, an open-source framework for quick development web applications. "Its goal is to enable [developers] to develop projects much faster than...writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries."

A popular development paradigm called Model-View-Controller serves as a loose foundation for CodeIgniter. Models are optional in CodeIgniter programming; however, view and controller classes are required.

- Exceptionally Light Weight
- Fully functional database classes that are compatible with multiple systems.
- Validation of Data and Form
- Management of Sessions
- Class Email Sending. Supports numerous protocols, attachments, HTML and text emails, and more
- Uploading files for classes with search engine-friendly URLs.

## 5 TECHNICAL SPECIFICATIONS

The following technologies are employed to put the system into place:

- 1) The creation of webpages is done with HTML.
- 2) Because PHP is open source and free, it is employed as a server-side language.
- 3) JQuery and Java Script for dynamic generation and webpage validation.
- 4) The Eclipse PHP is utilised for software coding as an Integrated Development Environment (IDE) or as a Rapid Application Development Tool (RAD).
- 5) MySQL is a popular database management system due to its speed, open-source nature, and free nature.

### 5.1.1 Advantage

- Cost-effective data storage and retrieval.
- User-friendly interface for easy file management.
- Robust security measures to protect sensitive data.
- Cross-platform compatibility for seamless access.
- Scalability to accommodate growing user needs.
- Collaboration tools for enhanced teamwork.
- Offline access capabilities for flexibility.
- Integration with other applications for improved workflow.

### 5.1.2 Disadvantage

- Limited storage capacity
- Internet dependency
- Compatibility issues
- Technical support challenges
- File versioning complexities.
- User limitations

## REFERENCES

1. Enhancement of anticipative recursively adjusting mechanism for redundant parallel file transfer in data grids.
2. Albeanu, G. (2017). Blockchain technology and education. In the 12th International Conference on Virtual Learning ICVL, 271-275. Jawdhari, H.A., & Abdullah, A.A. (2021). The application of network Functions Virtualization on different networks, and its new applications in blockchain: A survey. *Webology*, 18(Special 04), 1007–1044.
3. Peter Gutmann, University of Auckland, New Zealand. The secure filesystem (sfs) for dos/windows. <http://www.cs.auckland.ac.nz/pgut001/sfs/index.html>, September 1996.
4. Alex Tormasov. The tordisk project. <http://www.cs.auckland.ac.nz/pgut001/sfs/index.html>, November 1997. Anunobi, C.V., & Ape, R. (2018). Promotional strategies for open access resources discovery and access. *Webology*, 15(1),