

**MINI PROJECT  
(2020-2021)**

**Real-Time file sharing App**

**MID-TERM REPORT**



**Institute of Engineering & Technology**

**Submitted by :**

**Harsh Saxena**

**(181500248)**

**Rishika Singh**

**(181500570)**

**Ritik Kumar**

**(181500576)**

**Ritik Goel**

**(181500574)**

**Jayant Singh**

**(181500296)**

***Supervised By: -***

**Harvinder Kaur Mam**

**Technical Trainer**

**Department of Computer Engineering & Applications**

## Contents

---

<b>Abstract</b>	<b>3</b>
<b>1. Introduction</b>	<b>4</b>
1.1 General Introduction to the topic	4
1.2 Area of Computer Science	6
1.3 Hardware and Software Requirements	6
<b>2. Problem Definition</b>	<b>7</b>
<b>3. Objectives</b>	<b>8</b>
<b>4. Implementation Details</b>	<b>9</b>
<b>5. Progress till Date &amp; The Remaining work</b>	<b>16</b>
<b>6. Some Screenshots</b>	<b>16</b>
<b>7. References</b>	<b>24</b>

## **Abstract**

File sharing is one of the oldest applications of the internet. One way of sharing files online is for a user to upload files to a common space on the web and others users can download the files from the common web space.

The objective of this project was to design an online file sharing website where users can upload files and other users can download them. To make the website more user friendly, users were given two space-constrained visualizations of their file system to view space occupied by the files and folders, and three AJAX based file management system that works like browsing files on a desktop computer with drag and drop etc.

This report discusses the implementation details of the website, and the advantages of having different visualizations of the file system.

## **Introduction**

### **1.1 General Introduction to the topic**

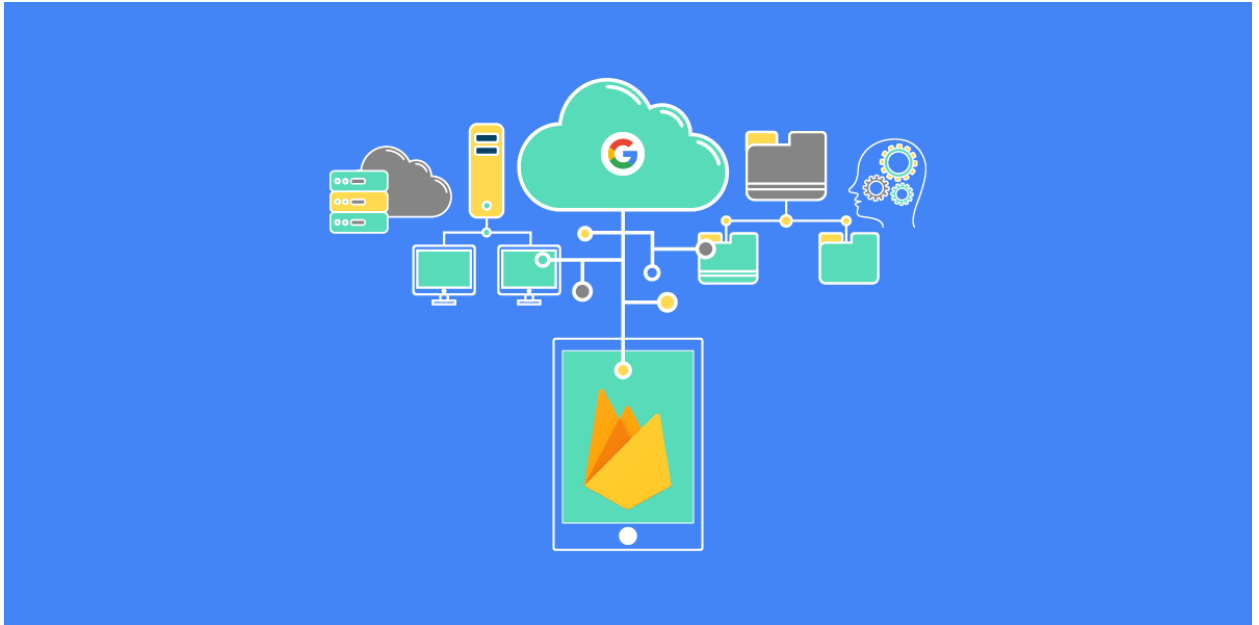
This project is a real-time file sharing application. File sharing is one of the oldest applications of the internet. One way of sharing files online is for a user to upload files to a common space on the web and other users can download the files from the common web space.

The objective of this project was to design an online file sharing website where users can upload files and other users can download them.

In this project users can upload the file by selecting the file or just doing drag and drop. After that a link will be generated that will work for 24 hours. Users can either send that link to another user or can directly mail it to the other user email address. The other user can click on the link and a download page will appear from there the other user can download the file.

### **1.2 Area of Computer Science**

In this we are using HTML, CSS and JavaScript for the frontend. HTML provides the basic structure of sites, which is enhanced and modified by other technologies like CSS and JavaScript. CSS is used to control presentation, formatting, and layout. JavaScript is used to control the behavior of different elements. On the other hand we will use Node.js for backend and MongoDB as database. Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices. MongoDB is a document-oriented NoSQL database used for high volume data storage. Instead of using tables and rows as in the traditional relational databases, MongoDB makes use of collections and documents. Documents consist of key-value pairs which are the basic unit of data in MongoDB. Collections contain sets of documents and function which is the equivalent of relational database tables. MongoDB is a database which came into light around the mid-2000s.



### **Web Technology:**

- HTML
- CSS
- JavaScript
- Node Js

## **1.3 Hardware and Software Requirements**

### **a) Hardware Requirements :**

- Latest Configuration Laptop

### **b) Software Requirements :**

- Visual Studio (Version 1.38)
- MongoDB (database)

## **Problem Definition**

Online File Sharing is practice of sharing files among different users across the internet. Common forms of file sharing are FTP (File Transfer Protocol) model and P2P(Peer-to-Peer) file sharing network. Another common form of sharing files over the internet is for a user to upload files to a website and allow other users to download them from the website. There are a lot of issues to consider when developing such a website.

Users of an online file sharing website who use features like upload, download, etc would want a website that is very interactive and fast and not annoying with a lot of post backs and flashing screens. Another issue is the visualization of their file system where usually users have a limit to upload files. The normal web based file folder view would be good, but if there are other types of visualizations it would be great.

## **Objective**

The objective of this project was to design an online file sharing website where users can upload files and can generate links using which the other user can download the file easily.

## **Implementation Details**

**Part 1:**making frontend of the website using html and css.

**Part 2 :**Making frontend interactive by adding javascript.

**Part 3 :** setting up the project . downloading all the dependencies.Making file upload api using node js

**Part 4 :** creating server , making mangoDB connections ,Making file upload api ,downloading file api and sending mail api using node js.

**Part 5 :** deploying project on heroku for free

### **Progress**

1. Part 1 is completed
2. Part 2 is completed
- 3.Part 3 setting up the project and downloading all the dependencies completed

### **Remaining work :**

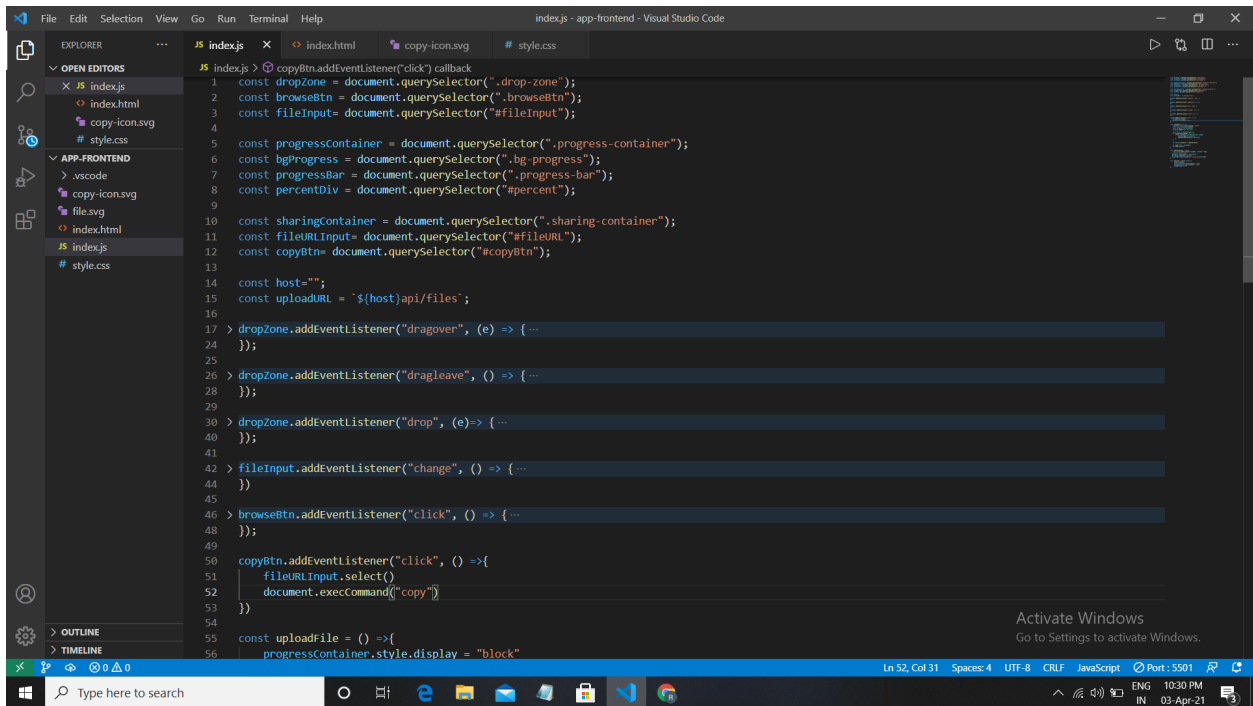
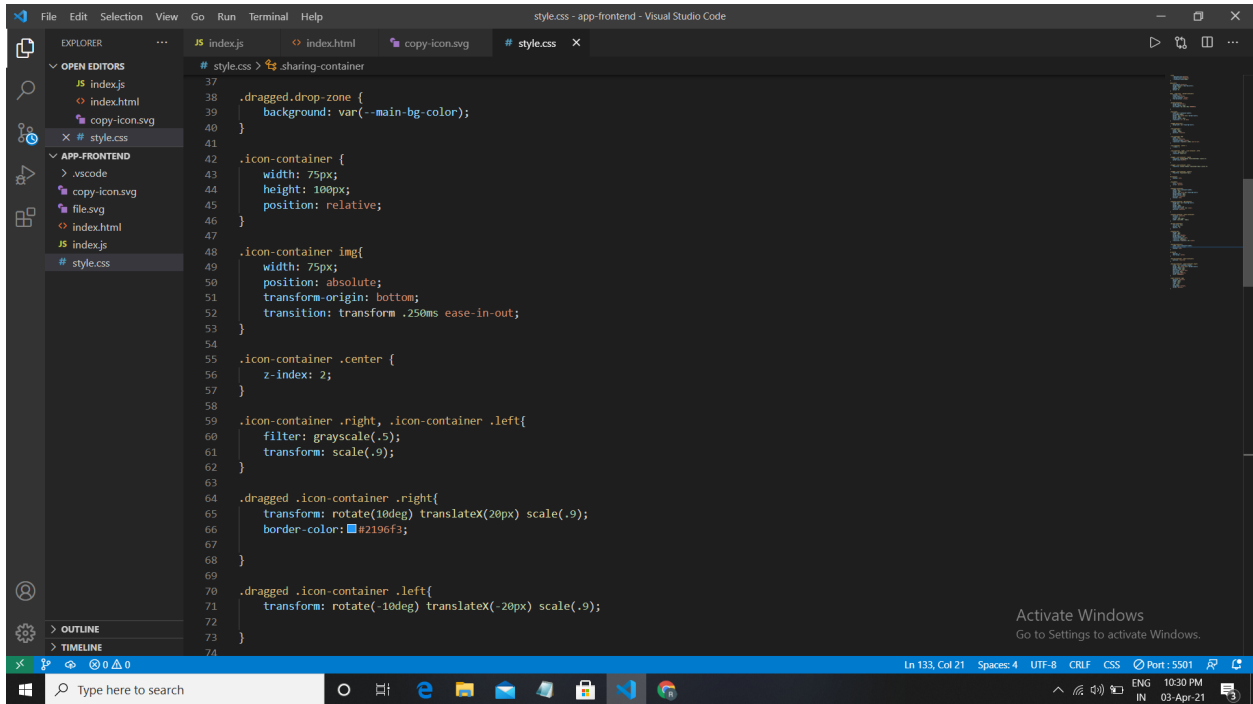
- Some task of Part 3
- Part 4
- Part 5

### **SCREENSHOTS**

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>ShareIn - Easy file sharing</title>
8   <link rel="stylesheet" href="style.css">
9 </head>
10 <body>
11   <section class="upload-container">
12     <div class="drop-zone">
13       <div class="icon-container">
14         
15         
16         
17       </div>
18       <input type="file" id="fileInput">
19       <div class="title">Drop your Files here or, <span class="browseBtn">browse</span></div>
20     </div>
21     <div class="progress-container">
22       <div class="bg-progress"></div>
23       <div class="inner-container">
24         <div class="title">Uploading...</div>
25         <div class="percent-container"><span id="percent">0</span>%</div>
26         <div class="progress-bar"></div>
27       </div>
28     </div>
29     <div class="sharing-container">
30       <p class="expire">Link expires in 24 hrs<p>
31       <div class="input-container">
32         <input type="text" id="fileURL" readonly>
33         
34       </div>
35     </div>
36   </section>
37   <script src="index.js">
38
```

```
1 :root{
2   --main-bg-color: #eff5fe;
3   --border-color: #0288d4;
4   --container-width: 500px;
5 }
6
7 body,html{
8   font-family: system-ui;
9   background: var(--main-bg-color);
10  height: 100%;
11  padding: 0;
12  margin: 0;
13 }
14
15 body, .drop-zone, .upload-container{
16  display: flex;
17  align-items: center;
18  justify-content: center;
19  flex-direction: column;
20 }
21
22 .upload-container{
23  background: #ffffff;
24  border-radius: 25px;
25  box-shadow: 0px 20px 20px #0000002e;
26 }
27
28
29 .drop-zone{
30  width: var(--container-width);
31  min-height: 200px;
32  border: 2px dashed var(--border-color);
33  margin: 30px;
34  border-radius: 10px;
35  transition: all .2s ease;
36 }
37
38 .dragged-drop-zone /
```



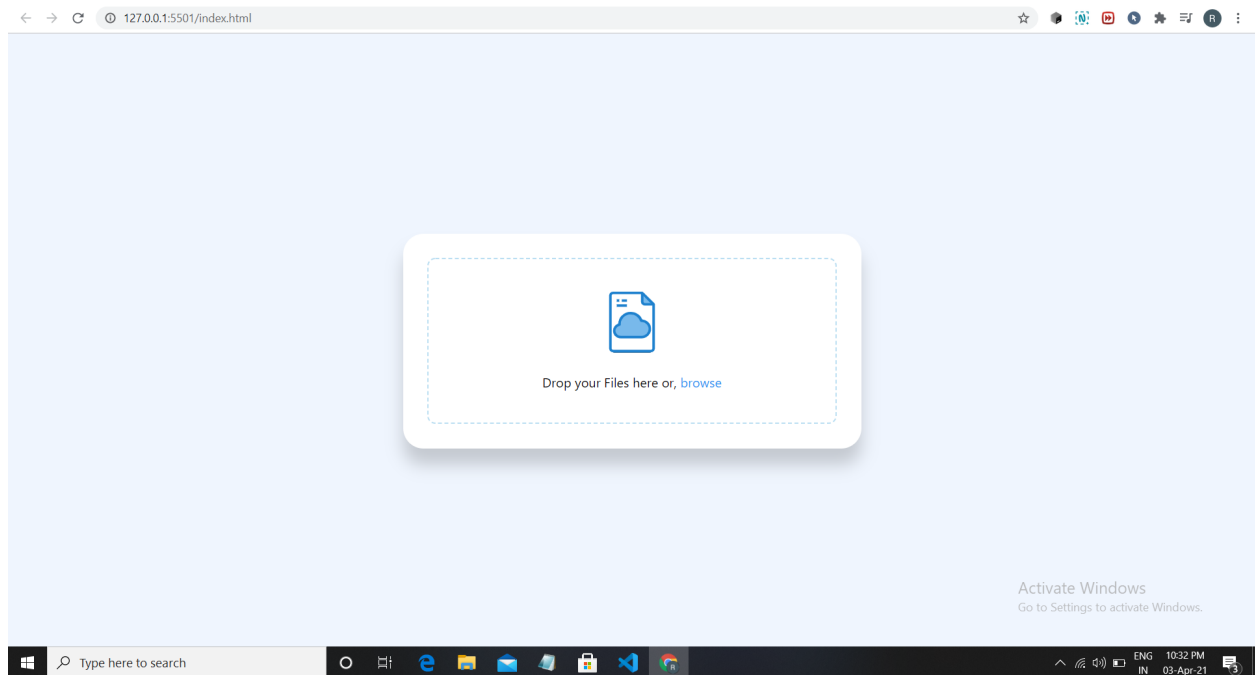


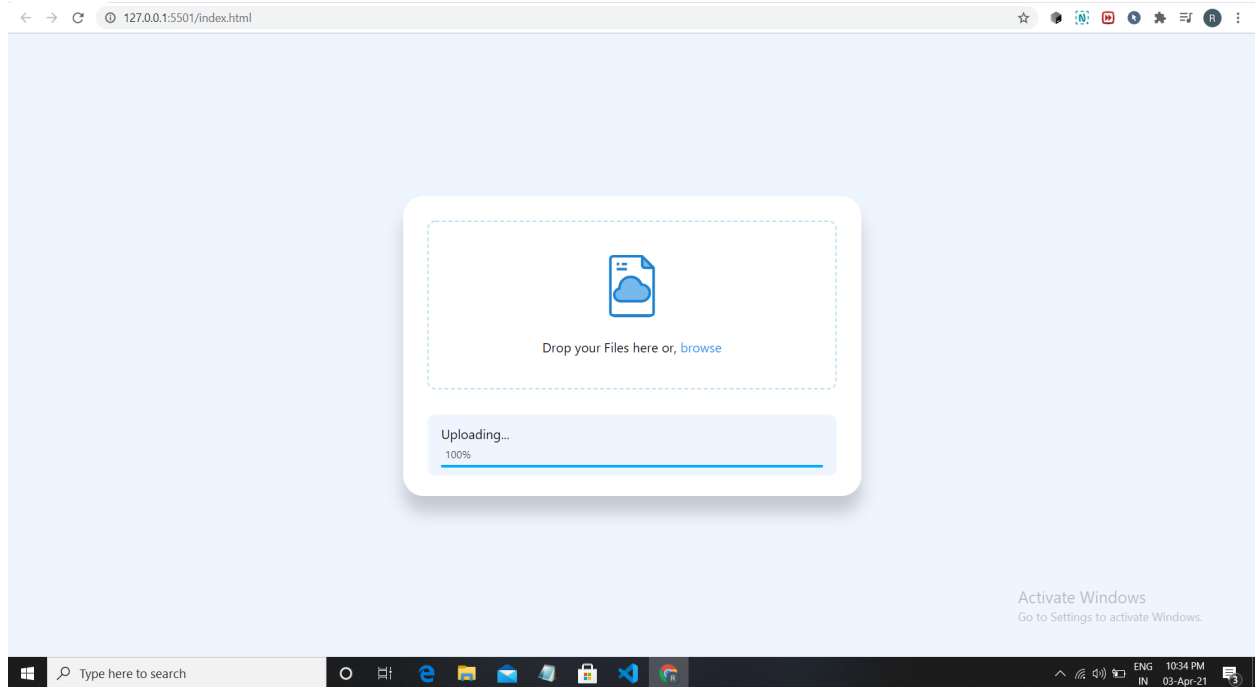
```
File Edit Selection View Go Run Terminal Help index.js - app-frontend - Visual Studio Code

EXPLORER
  OPEN EDITORS
    index.js
    index.html
    copy-icon.svg
    style.css
  APP-FRONTEND
    .vscode
    copy-icon.svg
    file.svg
    index.html
    index.js
    style.css

index.js
49  // copyBtn.addEventListener("click") callback
50  copyBtn.addEventListener("click", () =>{
51    fileURLInput.select()
52    document.execCommand("copy")
53  })
54
55  const uploadFile = () =>{
56    progressContainer.style.display = "block"
57    const file = fileInput.files[0];
58    const formData= new FormData();
59    formData.append("myfile",file);
60
61    const xhr = new XMLHttpRequest();
62    xhr.onreadystatechange = () => {
63      if(xhr.readyState === XMLHttpRequest.DONE){
64        console.log(xhr.response);
65        showLink(JSON.parse(xhr.response))
66      }
67    };
68
69    xhr.upload.onprogress = updateProgress;
70
71    xhr.open("POST", uploadURL);
72    xhr.send(formData)
73  };
74
75  const updateProgress = (e)->{
76    const percent=Math.round((e.loaded / e.total) * 100);
77    // console.log(percent);
78    bgProgress.style.width = `${percent}%`;
79    percentDiv.innerText = percent;
80    progressBar.style.transform = `scaleX(${percent/100})`;
81  };
82
83  const showLink = ({file: url})->{
84    console.log(url);
85    progressContainer.style.display = "none"
86    sharingContainer.style.display = "block"
  }

```





## References

- [www.google.com](http://www.google.com)
- <https://www.youtube.com/>