

COLLEGE CODE : 9116

**COLLEGE NAME : MOTHER TERESA COLLEGE OF AND
ENGINEERING TECHNOLOGY**

**DEPARTMENT : B.TECH ARTIFICIAL INTELLIGENCE AND
DATA SCIENCE**

**STUDENT NM-ID :
D6E7D07AE191EDBBA9046DA2D2B3AD0D**

ROLL NO : 911623243014

DATE :10.10.2025

**Completed the project named as
Phase-5 TECHNOLOGY PROJECT
NAME : FILE UPLOAD MANAGER**

**SUBMITTED BY,
NAME : AR MANI
MOBILE NO: 7824898528**

GIT REPOSITORY LINK

<https://github.com/maniganesh2908-cpu/Mani.git>

CODES

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>File Upload Manager</title>

**<style> body { font-family: "Segoe UI", sans-
serif; background: linear-gradient(135deg,**

#74ebd5 0%,

#9face6 100%);

margin: 0;

padding: 0; display:

flex; justify-content:

center; align-items:

flex-start; min-height:

100vh;

}

.container {

background: #fff; margin-

```

top: 40px;    padding:
25px;    border-radius:
16px;    box-shadow: 0
6px 20px rgba(0, 0, 0,
0.15);    width: 650px;
max-width: 95%;
animation: fadeIn 0.6s
ease-in-out;
}

@keyframes fadeIn {    from {opacity: 0;
transform: translateY(-10px);}    to {opacity: 1;
transform: translateY(0);}
}

h2 { text-align: center; color: #333; }

.upload-section {    text-
align: center;    border: 2px
dashed #74a9f2;    border-
radius: 12px;    padding:

```

20px; background: #f7faff;

margin-bottom: 20px;

}

.upload-section input { display: none; }

.upload-label {

display: inline-block;

padding: 10px 20px;

background: #4cafef;

color: white; border-

radius: 8px; cursor:

pointer; margin-

right: 10px;

}

.upload-label:hover { background: #3796db; }

button { border: none; padding: 10px

20px; border-radius: 8px; color: white;

font-size: 15px;

```
        cursor: pointer;    margin-  
  
top: 15px;    transition:  
  
background 0.3s;  
  
    }  
  
    .upload-btn { background: #28a745; }  
  
    .upload-btn:hover { background: #218838; }  
  
    .progress {    margin-  
  
top: 10px;  
  
background: #eee;  
  
border-radius: 6px;  
  
overflow: hidden;  
  
height: 10px;  
  
    }  
  
    .progress-bar {  
  
width: 0%;    height:  
  
10px;    background:  
  
#4cafef;    transition:  
  
width 0.3s;
```

}

.file-list h3 { margin-top: 0; }

.file-card { display: flex; justify-content: space-between; align-items: center; background: #f9f9f9; padding: 12px 16px; border-radius: 10px; margin-bottom: 10px; box-shadow: 0 2px 5px rgba(0,0,0,0.05); }

.file-name { font-weight: 500; }

.file-actions button { margin-left: 10px; padding: 6px 12px; border-radius: 6px; font-size: 13px; }

.download-btn { background: #007bff; } .download-btn:hover { background: #0069d9; }

.delete-btn { background: #dc3545; }

.delete-btn:hover { background: #c82333; }

```
.status { margin-top: 10px; text-align: center; font-size: 14px; }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="container">
```

```
<h2> File Upload Manager</h2>
```

```
<!-- Upload Section -->
```

```
<div class="upload-section">
```

```
<input type="file" id="fileInput" />
```

```
<label class="upload-label" for="fileInput">Choose  
File</label>
```

```
<button class="upload-btn"  
id="uploadBtn">Upload</button>
```

```
<div class="progress"><div class="progress-bar"  
id="progressBar"></div></div>
```

```
<div class="status" id="statusMsg"></div>
```

```
</div>
```


<!-- File List -->

<div class="file-list">

<h3>Uploaded Files</h3>

<div id="filesContainer"></div>

</div>

</div>

**<script> const fileInput =
document.getElementById("fileInput"); const uploadBtn
= document.getElementById("uploadBtn"); const
progressBar = document.getElementById("progressBar");

 const filesContainer =
document.getElementById("filesContainer");
const statusMsg =
document.getElementById("statusMsg");
uploadBtn.addEventListener("click", () => {**

```
    if (!fileInput.files[0]) {  
  
    alert("Please select a file first!");  
  
    return;  
  
    }  
  
    const formData = new FormData();  
  
    formData.append("file", fileInput.files[0]);  
  
  
  
    const xhr = new XMLHttpRequest();  
  
    xhr.open("POST", "http://localhost:5000/upload");  
  
  
  
    xhr.upload.addEventListener("progress", (e) => {  
  
    if (e.lengthComputable) {      const percent =  
  
    (e.loaded / e.total) * 100;  
  
    progressBar.style.width = percent + "%";  
  
    }  
  
    });  
  
  
  
    xhr.onload = () => {      if (xhr.status === 200) {  
  
    statusMsg.textContent = "  Upload successful!";
```

```
progressBar.style.width = "0%";      fileInput.value =  
"";      loadFiles();  
  
    } else {  
  
        statusMsg.textContent = " Upload failed!";  
  
    }  
  
};
```

```
    xhr.onerror = () => { statusMsg.textContent = "  
Error uploading file."; };    xhr.send(formData);  
  
});
```

```
    async function loadFiles() {  
  
filesContainer.innerHTML = "";    try {    const res =  
await fetch("http://localhost:5000/files");    const data  
= await res.json();
```

```
    if (!data || data.length === 0) {  
filesContainer.innerHTML = "<p>No files uploaded  
yet.</p>";
```

```
    } else {      data.forEach((file) => {

const div = document.createElement("div");

div.classList.add("file-card");

div.innerHTML = `

    <span class="file-name">${file.fileName}</span>

    <div class="file-actions">

        <button class="download-btn"
onclick="window.open('${file.fileUrl}')">Download</button
>

        <button class="delete-btn"
onclick="deleteFile('${file.fileId}')">Delete</button>

    </div>

    `;

filesContainer.appendChild(div);

});

}

} catch (err) {      console.error("Error
loading files:", err);

}
```

```
}
```

```
    async function deleteFile(id) {        if (!confirm("Are you  
sure you want to delete this file?")) return;
```

```
        try {            const res = await  
fetch(`http://localhost:5000/files/${id}`,  
{ method: "DELETE" });
```

```
            const result = await res.json();  
alert(result.message || "Deleted");        loadFiles();  
        } catch (err) {  
            alert("Error deleting file.");  
        }  
    }  
}
```

```
loadFiles();
```

```
</script>
```

```
</body>
```

```
</html>
```

```
2.) {  
  
  "name": "file-manager",  
  
  "version": "1.0.0",  
  
  "lockfileVersion": 3,  
  
  "requires": true,  
  
  "packages": {  
  
    "": {  
  
      "name": "file-manager",  
  
      "version": "1.0.0",  
  
      "license": "ISC",    "dependencies": {  
  
        "cors": "^2.8.5",  
  
        "express": "^5.1.0",  
  
        "gridfs-stream": "^1.1.1",  
  
        "mongoose": "^8.19.0",  
  
        "multer": "^2.0.2"  
  
      }  
  
    },  
  
    "node_modules/@mongodb-js/saslprep": {
```

```
"version": "1.3.1",  
  
"resolved":  
"https://registry.npmjs.org/@mongodbjs/saslprep/-  
/saslprep-1.3.1.tgz",  
  
"integrity": "sha512-  
6nZrq5kfAz0POWyhljnbWQQJQ5uT8oE2ddX303q1uY0t  
WsivWKgBDXBBvuFPwOqRRalXJuVO9EjOdVtuhLX0zg  
==",  
  
"license": "MIT",  
  
"dependencies": {  
  
  "sparse-bitfield": "^3.0.3"  
  
}  
  
},  
  
"node_modules/@types/webidl-conversions": {  
  
  "version": "7.0.3",  
  
  "resolved":  
"https://registry.npmjs.org/@types/webidlconversions/-  
/webidl-conversions-7.0.3.tgz",  
  
  "integrity": "sha512-  
CiJJvcRtIgzaDHCYXw7dqEnMNRjhGZlYK05Mj9OyktqV  
8uVT8fD2BFOB7S1uwBE3Kj2Z+4UyPmFw/Ixgw/LAlA==
```

```
",  
  
  "license": "MIT"  
  
},  
  
"node_modules/@types/whatwg-url": {  
  
  "version": "11.0.5",  
  
  "resolved":  
"https://registry.npmjs.org/@types/whatwgurl/-/whatwg-url-  
11.0.5.tgz",  
  
  "integrity": "sha512-  
coYR071JRaHa+xEvvYqvnIHqVqaYrLPbsufM9BF63Hk  
wI5Lgmy2QR8Q5K/IYDY05AK82wOvSOS0UsLTpTG7uQ  
==",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "@types/webidl-conversions": "*"   
  
  }  
  
},  
  
"node_modules/accepts": {  
  
  "version": "2.0.0",  
  
  "resolved": "https://registry.npmjs.org/accepts/-
```


/accepts-2.0.0.tgz",

"integrity": "sha512-5cvg6CtKwfgdmVqY1WliXKc3Q1bkRqGLi+2W/6ao+6Y7gu/RCwRuAhGEzh5B4KlszSuTLgZYuqFqo5bImjNKng==",

"license": "MIT",

"dependencies": {

"mime-types": "^3.0.0",

"negotiator": "^1.0.0"

},

"engines": {

"node": ">= 0.6"

}

},

"node_modules/append-field": {

"version": "1.0.0",

"resolved": "https://registry.npmjs.org/append-field/-/append-field-1.0.0.tgz",

"integrity": "sha512-klpgFSWLW1ZEs8svjfb7g4qWY0YS5iml82dTg+QahUvJ8YqAY0P10Uk8tTyh9ZGuYEZEMaeJYCF5BFuX552hsw==

```
",  
  
  "license": "MIT"  
  
},  
  
"node_modules/body-parser": {  
  
  "version": "2.2.0",  
  
  "resolved": "https://registry.npmjs.org/body-parser/-  
/body-parser-2.2.0.tgz",  
  
  "integrity": "sha512-02qvAaxv8tp7fBa/mw1ga98OGm+eCbqzJOKoRt70sLmfEE  
i+jyBYVTDGfCL/k06/4EMk/z01gCe7HoCH/f2LTg==",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "bytes": "^3.1.2",  
  
    "content-type": "^1.0.5",  
  
    "debug": "^4.4.0",  
  
    "http-errors": "^2.0.0",  
  
    "iconv-lite": "^0.6.3",  
  
    "on-finished": "^2.4.1",  
  
    "qs": "^6.14.0",
```

```
"raw-body": "^3.0.0",  
"type-is": "^2.0.0"  
},  
"engines": {  
  "node": ">=18"  
}  
},  
"node_modules/bson": {  
  "version": "6.10.4",  
  "resolved": "https://registry.npmjs.org/bson/-/bson-  
6.10.4.tgz",  
  "integrity": "sha512-  
WIsKqkSC0ABoBJuT1LEX+2HEvNmNKKgnTAyd0fL8qz  
K4SH2i9NXg+t08YtdZp/V9IZ33cxe3iV4yM0qg8lMQng=="  
,  
  "license": "Apache-2.0",  
  "engines": {  
    "node": ">=16.20.1"  
  }  
},
```

```
"node_modules/buffer-from": {  
  "version": "1.1.2",  
  "resolved": "https://registry.npmjs.org/buffer-from/-  
/buffer-from-1.1.2.tgz",  
  "integrity": "sha512-E+XQCRwSbaaiChtv6k6Dwgc+bx+Bs6vuKJHHl5kox/BaK  
bhiXzqQOwK4cO22yElGp2OCmjwtVhT3HmxgyPGnJfQ==  
",  
  "license": "MIT"  
},  
"node_modules/busboy": {  
  "version": "1.6.0",  
  "resolved": "https://registry.npmjs.org/busboy/-/busboy-  
1.6.0.tgz",  
  "integrity": "sha512-8SFQbg/0hQ9xy3UNTB0YEnsNBbWfhf7RtnzpL7TkBiTBR  
frQ9Fxcnz7VJsleJpyp6rVLvXiuORqjlHi5q+PYuA==",  
  "dependencies": {  
    "streamsearch": "^1.1.0"  
  },
```

```
"engines": {  
  "node": ">=10.16.0"  
}  
},  
  
"node_modules/bytes": {  
  "version": "3.1.2",  
  "resolved": "https://registry.npmjs.org/bytes/-/bytes-  
3.1.2.tgz",  
  "integrity": "sha512-  
/Nf7TyzTx6S3yRJOboAV7956r8cr2+Oj8AC5dt8wSP3BQA  
oeX58NoHyCU8P8zGkNXStjTSi6fzO6F0pBdcYbEg==",  
  "license": "MIT",  
  "engines": {  
    "node": ">= 0.8"  
  }  
},  
  
"node_modules/call-bind-apply-helpers": {  
  "version": "1.0.2",
```

"resolved": "https://registry.npmjs.org/call-bind-applyhelpers/-/call-bind-apply-helpers-1.0.2.tgz",

"integrity": "sha512-Sp1ablJ0ivDkSzjcaJdxEunN5/XvksFJ2sMBFfq6x0ryhQV/2b/KwFe21cMpmHtPOSij8K99/wSfoEuTObmuMQ==",

"license": "MIT",

"dependencies": {

"es-errors": "^1.3.0",

"function-bind": "^1.1.2"

},

"engines": {

"node": ">= 0.4"

}

},

"node_modules/call-bound": {

"version": "1.0.4",

"resolved": "https://registry.npmjs.org/call-bound/-/callbound-1.0.4.tgz",

"integrity": "sha512-

**+ys997U96po4Kx/ABpBCqhA9EuxJaQWDQg7295H4hBph
v3IZg0boBKuwYpt4YXp6MZ5AmZQnU/tyMTlRpaSejg==
",**

"license": "MIT",

"dependencies": {

"call-bind-apply-helpers": "^1.0.2",

"get-intrinsic": "^1.3.0"

},

"engines": {

"node": ">= 0.4"

},

"funding": {

"url": "https://github.com/sponsors/ljharb"

}

},

"node_modules/concat-stream": {

"version": "2.0.0",

**"resolved": "https://registry.npmjs.org/concat-stream/-
/concat-stream-2.0.0.tgz",**

```
"integrity": "sha512-  
MWufYdFw53ccGjCA+Ol7XJYpAlW6/prSMzuPOTRnJGc  
GzuhLn4Scrz7qf6o8bROZ514ltazcIFJZevcfbo0x7A==",  
  
"engines": [  
  
  "node >= 6.0"  
  
],  
  
"license": "MIT",  
  
"dependencies": {  
  
  "buffer-from": "^1.0.0",  
  
  "inherits": "^2.0.3",  
  
  "readable-stream": "^3.0.2",  
  
  "typedarray": "^0.0.6"  
  
  }  
  
},  
  
"node_modules/content-disposition": {  
  
  "version": "1.0.0",  
  
  "resolved":  
  "https://registry.npmjs.org/contentdisposition/-/content-  
disposition-1.0.0.tgz",
```


"integrity": "sha512-Au9nRL8VNUut/XSzbQA38+M78dzP4D+eqg3gfJHMIHHYa3bg067xj1KxMUWj+VULbiZMowKngFFbKczUrNJ1mg==",

"license": "MIT",

"dependencies": {

"safe-buffer": "5.2.1"

},

"engines": {

"node": ">= 0.6"

}

},

"node_modules/content-type": {

"version": "1.0.5",

"resolved": "https://registry.npmjs.org/content-type/-/content-type-1.0.5.tgz",

"integrity": "sha512-nTjqfcBFEipKdXCv4YDQWCfmcLZKm81ldF0pAopTvyrFGVbcR6P/VAAd5G7N+0tTr8QqiU0tFadD6FK4NtJwOA==",

```
"license": "MIT",  
  
"engines": {  
  
  "node": ">= 0.6"  
  
}  
  
},  
  
"node_modules/cookie": {  
  
  "version": "0.7.2",  
  
  "resolved": "https://registry.npmjs.org/cookie/-/cookie-  
0.7.2.tgz",  
  
  "integrity": "sha512-yki5XnKuf750l50uGTllt6kKILY4nQ1eNIQatoXEBYZ5dWg  
nKqbnqmTrBE5B4N7lrMJKQ2ytWMiTO2o0v6Ew/w==",  
  
  "license": "MIT",  
  
  "engines": {  
  
    "node": ">= 0.6"  
  
  }  
  
},  
  
"node_modules/cookie-signature": {  
  
  "version": "1.2.2",
```

**"resolved": "https://registry.npmjs.org/cookiesignature/-
/cookie-signature-1.2.2.tgz",**

**"integrity": "sha512-
D76uU73ulSXRd1UXF4KE2TMxVVwhsnCgfAyTg9k8P6K
GZjlXKrOLe4dJQKI3Bxi5wjEsZoFXJWEINWBjPZMbHg=
=",**

"license": "MIT",

"engines": {

"node": ">=6.6.0"

}

},

"node_modules/cors": {

"version": "2.8.5",

**"resolved": "https://registry.npmjs.org/cors/-/cors-
2.8.5.tgz",**

**"integrity": "sha512-
KIHbLJqu73RGr/hnbrO9uBeixNGuvSQjul/jdFvS/KFSIH1
hWVd1ng7zOHx+YrEflnLG7q4n6GHQ9cDtxv/P6g==",**

"license": "MIT",

"dependencies": {

```
"object-assign": "^4",  
  
"vary": "^1"  
  
},  
  
"engines": {  
  
  "node": ">= 0.10"  
  
}  
  
},  
  
"node_modules/debug": {  
  
  "version": "4.4.3",  
  
  "resolved": "https://registry.npmjs.org/debug/-/debug-  
4.4.3.tgz",  
  
  "integrity": "sha512-  
RGwwWnwQvkVfavKVt22FGLw+xYSdzARwm0ru6DhTV  
A3umU5hZc28V3kO4stgYryrTILpuvgI9GiiJltAjNbcqA==",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "ms": "^2.1.3"  
  
  },  
  
  "engines": {
```

```
    "node": ">=6.0"
  },
  "peerDependenciesMeta": {
    "supports-color": {
      "optional": true
    }
  }
},
"node_modules/depd": {
  "version": "2.0.0",
  "resolved": "https://registry.npmjs.org/depd/-/depd-2.0.0.tgz",
  "integrity": "sha512-g7nH6P6dyDioJogAAGprGpCtVImJhpPk/roCzdb3fIh61/s/nPsfR6onyMwkCAR/OlC3yBC0lESvUoQEAssIrw==",
  "license": "MIT",
  "engines": {
    "node": ">= 0.8"
  }
}
```

```
  },  
  
  "node_modules/dunder-proto": {  
  
    "version": "1.0.1",  
  
    "resolved": "https://registry.npmjs.org/dunder-proto/-  
/dunder-proto-1.0.1.tgz",  
  
    "integrity": "sha512-  
KIN/nDJBQRcXw0MLVhZE9iQHmG68qAVIBg9CqmUYj  
mQIhgij9U5MFvrqkUL5FbtyyzZuOeOt0zdeRe4UY7ct+A=  
=",  
  
    "license": "MIT",  
  
    "dependencies": {  
  
      "call-bind-apply-helpers": "^1.0.1",  
  
      "es-errors": "^1.3.0",  
  
      "gopd": "^1.2.0"  
  
    },  
  
    "engines": {  
  
      "node": ">= 0.4"  
  
    }  
  
  },  
}
```

```
"node_modules/ee-first": {  
  "version": "1.1.1",  
  "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-  
1.1.1.tgz",  
  "integrity": "sha512-WMwm9LhRUo+WUaRN+vRuETqG89IgZphVSNkdFgeb6  
sS/E4OrDIN7t48CAewSHXc6C8lefD8KKfr5vY61brQlow==  
",  
  "license": "MIT"  
},  
"node_modules/encodeurl": {  
  "version": "2.0.0",  
  "resolved": "https://registry.npmjs.org/encodeurl/-  
/encodeurl-2.0.0.tgz",  
  "integrity": "sha512-Q0n9HRi4m6JuGIV1eFlmvJB7ZEVxu93IrMyiMsGC0lrMJ  
MWzRgx6WGquyfQgZVb31vhGgXnfmPNNXmxnOkRBrg  
==",  
  "license": "MIT",  
  "engines": {  
    "node": ">= 0.8"
```

```
}  
  
},  
  
"node_modules/es-define-property": {  
  "version": "1.0.1",  
  
  "resolved": "https://registry.npmjs.org/es-defineproperty/-/es-define-property-1.0.1.tgz",  
  
  "integrity": "sha512-e3nRfgfUZ4rNGL232gUgX06QNyyez04KdjFrF+LTRoOXmrOgFKDg4BCdsjW8EnT69eqdYGmRpJwiPVYNrCaW3g==",  
  
  "license": "MIT",  
  
  "engines": {  
    "node": ">= 0.4"  
  }  
},  
  
"node_modules/es-errors": {  
  "version": "1.3.0",  
  
  "resolved": "https://registry.npmjs.org/es-errors/-/eserrors-1.3.0.tgz",  
  
  "integrity": "sha512-
```


**Zf5H2Kxt2xjTvbJvP2ZWLEICxA6j+hAmMzIlypy4xcBg1v
KVnx89Wy0GbS+kf5cwCVFFzdCFh2XSCFNULS6csw=="**,

"license": "MIT",

"engines": {

"node": ">= 0.4"

}

},

"node_modules/es-object-atoms": {

"version": "1.1.1",

**"resolved": "https://registry.npmjs.org/es-object-atoms/-
/es-object-atoms-1.1.1.tgz",**

"integrity": "sha512-

**FGgH2h8zKNim9Ijj7dankFPcICIK9Cp5bm+c2gQSYePhpa
G5+esrLODihIorn+Pe6FGJzWhXQotPv73jTaldXA=="**,

"license": "MIT",

"dependencies": {

"es-errors": "^1.3.0"

},

"engines": {

```
"node": ">= 0.4"

}

},

"node_modules/escape-html": {

  "version": "1.0.3",

  "resolved": "https://registry.npmjs.org/escape-html/-
/escape-html-1.0.3.tgz",

  "integrity": "sha512-
NiSupZ4OeuGwr68lGIeym/ksIZMJodUGOSZ/FSnTxcrek
bvqrgdUxlJOMpijaKZVjAJrWrGs/6Jy8OMuyj9ow==",

  "license": "MIT"

},

"node_modules/etag": {

  "version": "1.8.1",

  "resolved": "https://registry.npmjs.org/etag/-/etag-
1.8.1.tgz",

  "integrity":
"sha512aIL5Fx7mawVa300aI2BnEE4iNvo1qETxLrPI/o05L
7z6go7f
Cw1J6EQmbK4FmJ2AS7kgVF/KEZWufBfdClMcPg==",
```

```
"license": "MIT",  
  
"engines": {  
  
  "node": ">= 0.6"  
  
  }  
  
},  
  
"node_modules/express": {  
  
  "version": "5.1.0",  
  
  "resolved": "https://registry.npmjs.org/express/-  
/express-5.1.0.tgz",  
  
  "integrity": "sha512-  
DT9ck5YIRU+8GYzzU5kT3eHGA5iL+1Zd0EutOmTE9Dtk  
+Tvuzd23VBU+ec7HPNSTxXYO55gPV/hq4pSBJDjFpA==  
",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "accepts": "^2.0.0",  
  
    "body-parser": "^2.2.0",  
  
    "content-disposition": "^1.0.0",  
  
    "content-type": "^1.0.5",
```

"cookie": "^0.7.1",
"cookie-signature": "^1.2.1",
"debug": "^4.4.0",
"encodeurl": "^2.0.0",
"escape-html": "^1.0.3",
"etag": "^1.8.1",
"finalhandler": "^2.1.0",
"fresh": ^2.0.0",
"http-errors": ^2.0.0",
"merge-descriptors": ^2.0.0",
"mime-types": ^3.0.0",
"on-finished": ^2.4.1",
"once": ^1.4.0",
"parseurl": ^1.3.3",
"proxy-addr": ^2.0.7",
"qs": ^6.14.0",
"range-parser": ^1.2.1", "router": ^2.2.0",
"send": ^1.1.0",

```
"serve-static": "^2.2.0",  
  "statuses": "^2.0.1",  
  "type-is": "^2.0.1",  
  "vary": "^1.1.2"  
},  
"engines": {  
  "node": ">= 18"  
},  
"funding": {  
  "type": "opencollective",  
  "url": "https://opencollective.com/express"  
}  
},  
"node_modules/finalhandler": {  
  "version": "2.1.0",  
  "resolved": "https://registry.npmjs.org/finalhandler/-  
/finalhandler-2.1.0.tgz",  
  "integrity": "sha512-  
/t88Ty3d5JWQbWYgaOGCCYfXRwV1+be02WqYYIL6h0I
```

EiUAMPM8o8qKGO01YIkOHZka2up08wvgYD0mDiI+q3Q
==",

"license": "MIT",

"dependencies": {

"debug": "^4.4.0",

"encodeurl": "^2.0.0",

"escape-html": "^1.0.3",

"on-finished": "^2.4.1",

"parseurl": "^1.3.3",

"statuses": "^2.0.1"

},

"engines": {

"node": ">= 0.8"

}

},

"node_modules/flushwritable": {

"version": "1.0.0",

**"resolved": "https://registry.npmjs.org/flushwritable/-
/flushwritable-1.0.0.tgz",**

"integrity": "sha512-3VELfuWCLVzt5d2Gblk8qcqFro6nuwvxwMzHaENVdHI7rxcBRtMCwTk/E9FXcgh+82DspavPNDueA9+RxXJoFg==",

"license": "MIT"

},

"node_modules/forwarded": {

"version": "0.2.0",

"resolved": "https://registry.npmjs.org/forwarded/-/forwarded-0.2.0.tgz",

"integrity": "sha512-buRG0fpBtRHSTCOASe6hD258tEubFoRLb4ZNA6NxMVHNw2gOcwHo9wyablzMzOA5z9xA9L1KNjk/Nt6MT9aYow==",

"license": "MIT",

"engines": {

"node": ">= 0.6"

}

},

"node_modules/fresh": {

```
"version": "2.0.0",  
  
"resolved": "https://registry.npmjs.org/fresh/-/fresh-  
2.0.0.tgz",  
  
"integrity": "sha512-  
Rx/WycZ60HOaqLKAI6cHRKKI7zxWbJ31MhntmtwMoaT  
eF7XFH9hhBp8vITaMidfljRQ6eYWCKkaTK+ykVJHP2A  
==",  
  
"license": "MIT",  
  
"engines": {  
  
  "node": ">= 0.8"  
  
}  
  
},  
  
"node_modules/function-bind": {  
  
  "version": "1.1.2",  
  
  "resolved": "https://registry.npmjs.org/function-bind/-  
/function-bind-1.1.2.tgz",  
  
  "integrity": "sha512-  
7XHNxH7qX9xG5mIwxkhumTox/MIRNcOgDrxWsMt2pA  
r23WHP6MrRIN7FBSFpCpr+oVO0F744iUgR82nJMfG2SA  
==",  
  
  "license": "MIT",
```



```
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
  
"node_modules/get-intrinsic": {  
  "version": "1.3.0",  
  "resolved": "https://registry.npmjs.org/get-intrinsic/-  
/get-intrinsic-1.3.0.tgz",  
  "integrity": "sha512-9fSjSaos/fRIVIp+xSJIE6lfwhES7LNtKaCBIamHsjr2na1Bi  
ABJPo0mOjjz8GJDURarmCPGqaiVg5mfjb98CQ==",  
  "license": "MIT",  
  "dependencies": {  
    "call-bind-apply-helpers": "^1.0.2",  
    "es-define-property": "^1.0.1",  
    "es-errors": "^1.3.0",  
    "es-object-atoms": "^1.1.1",  
    "function-bind": "^1.1.2",  
    "get-proto": "^1.0.1",
```

```
"gopd": "^1.2.0",  
"has-symbols": "^1.1.0",  
"hasown": "^2.0.2",  
"math-intrinsics": "^1.1.0"  
},  
"engines": {  
  "node": ">= 0.4"  
},  
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
"node_modules/get-proto": {  
  "version": "1.0.1",  
  "resolved": "https://registry.npmjs.org/get-proto/-  
/getproto-1.0.1.tgz",  
  "integrity":  
"sha512-TSfBjoXBP89JvIKIefqw7U2CCebsc74kiY6awiGog
```

KtoSGb

gjYE/G/+I9sF3MWFPNc9IcoOC4ODfKHfxFmp0g==",

"license": "MIT",

"dependencies": {

"dunder-proto": "^1.0.1",

"es-object-atoms": "^1.0.0"

},

"engines": {

"node": ">= 0.4"

}

},

"node_modules/gopd": {

"version": "1.2.0",

"resolved": "https://registry.npmjs.org/gopd/-/gopd-1.2.0.tgz",

"integrity": "sha512-

ZUKRh6/kUFoAiTAtTYPZJ3hw9wNxx+BIBOijnlG9PnrJs

CcSjs1wyyD6vJpaYtgznDrKYRSqf3OO6Rfa93xsRg==",

"license": "MIT",

```
"engines": {  
  "node": ">= 0.4"  
},  
  
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
  
"node_modules/gridfs-stream": {  
  "version": "1.1.1",  
  
  "resolved": "https://registry.npmjs.org/gridfs-stream/-  
/gridfs-stream-1.1.1.tgz",  
  
  "integrity": "sha512-  
EcELdPIjC7tpZUiZA/8trfmszLbcsZlFyDQ8DhMtyJIMDm  
uLi5Vzt/056OO6FqfvY/zwiTC01eZAqwtqrhBGMQ==",  
  
  "deprecated": "This project has been sunset. Please  
migrate to an alternative.",  
  "dependencies": {  
    "flushwritable": "^1.0.0"  
  },  
  
  "engines": {
```

```
    "node": ">= 0.4.2"
  }
},
"node_modules/has-symbols": {
  "version": "1.1.0",
  "resolved": "https://registry.npmjs.org/has-symbols/-/has-symbols-1.1.0.tgz",
  "integrity": "sha512-1cDNdwJ2Jaohmb3sg4OmKaMBwuC48sYni5HUw2DvsC8LjGTLK9h+eb1X6RyuOHe4hT0ULCW68iomhjUoKUqlPQ==",
  "license": "MIT",
  "engines": {
    "node": ">= 0.4"
  },
  "funding": {
    "url": "https://github.com/sponsors/ljharb"
  }
},
```

```
"node_modules/hasown": {  
  "version": "2.0.2",  
  "resolved": "https://registry.npmjs.org/hasown/-  
hasown-2.0.2.tgz",  
  "integrity": "sha512-  
0hJU9SCPvmMzIBdZFqNPXWa6dqh7WdH0cII9y+CyS8r  
G3nL48Bclra9HmKhVVUHyPWNH5Y7xDwAB7bfgSjkUM  
Q==",  
  "license": "MIT",  
  "dependencies": {  
    "function-bind": "^1.1.2"  
  },  
  "engines": {  
    "node": ">= 0.4"  
  }  
}, },  
"node_modules/http-errors": {  
  "version": "2.0.0",  
  "resolved": "https://registry.npmjs.org/http-errors/-  
http-errors-2.0.0.tgz",
```

```
"integrity": "sha512-  
FtwrG/euBzaEjYeRqOgly7G0qviiXoJWnvEH2Z1plBdXgby  
jv34pHTSb9zoeHMyDy33+DWy5Wt9Wo+TURtOYSQ==",  
  
"license": "MIT",  
  
"dependencies": {  
  
  "depd": "2.0.0",  
  
  "inherits": "2.0.4",  
  
  "setprototypeof": "1.2.0",  
  
  "statuses": "2.0.1",  
  
  "toidentifier": "1.0.1"  
},  
  
"engines": {  
  
  "node": ">= 0.8"  
}  
},  
  
"node_modules/http-errors/node_modules/statuses": {  
  
  "version": "2.0.1",  
  
  "resolved": "https://registry.npmjs.org/statuses/-  
/statuses-2.0.1.tgz",
```

"integrity": "sha512-RwNA9Z/7PrK06rYLLzFMlaF+l73iwpzsqRIFgbMLbTcLD6cOao82TaWefPXQvB2fOC4AjuYSEndS7N/mTCbkdQ=="
,

"license": "MIT",

"engines": {

"node": ">= 0.8"

}

},

"node_modules/iconv-lite": {

"version": "0.6.3",

**"resolved": "https://registry.npmjs.org/iconv-lite/-
iconv-lite-0.6.3.tgz",**

"integrity": "sha512-4fCk79wshMdzMp2rH06qWrJE4iolqLhCUH+OiuIgU++RB0+94NIDL81atO7GX55uUKueo0txHNtvEyI6D7WdMw=="
,

"license": "MIT",

"dependencies": {

"safer-buffer": ">= 2.1.2 < 3.0.0"


```

    },
    "engines": {
      "node": ">=0.10.0"
    }
  },
  "node_modules/inherits": {
    "version": "2.0.4",
    "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.4.tgz",
    "integrity": "sha512-k/vGaX4/Yla3WzyMCvTQOXYeIhVqOKtnqBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIFeMpXKxa4dYeZIQqewQ==",
    "license": "ISC"
  },
  "node_modules/ipaddr.js": {
    "version": "1.9.1",
    "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-1.9.1.tgz",
    "integrity": "sha512-113n1sW0D8P361j662B6kQ3dYU5V70pX8+UwuV8Vlq9x75Ri8aT8t8mB0NI59NNPKZLzBQqP51E1YU4H4="
  }
}

```

**0KI/607xoxSToH7GjN1FfSbLoU0+btTicjsQSWQlh/hZykN
8KpmMf7uYwPW3R+akZ6R/w18ZlXSHBYXiYUPO3g=="**

,

"license": "MIT",

"engines": {

"node": ">= 0.10"

}

},

"node_modules/is-promise": {

"version": "4.0.0",

**"resolved": "https://registry.npmjs.org/is-promise/-
/ispromise-4.0.0.tgz",**

"integrity": "sha512-

hvp0I6korhJMnej285dSg6nu1+e6uxs7zG3BYAm5byqDsgJ

NWwxzM6z6iZiAgQR4TJ30JmBTOWqZUw3WlyH3AQ=="

",

"license": "MIT"

},

"node_modules/kareem": {

"version": "2.6.3",

**"resolved": "https://registry.npmjs.org/kareem/-
/kareem-2.6.3.tgz",**

**"integrity": "sha512-
C3iHfuGUXK2u8/ipq9LfjFfXFxAZMQJJq7vLS45r3D9Y2x
Q/m4S8zaR4zMLFWWh9AsNPXmcFfUDhTEO8UIC/V6Q==
",**

"license": "Apache-2.0",

"engines": {

"node": ">=12.0.0"

}

},

"node_modules/math-intrinsics": {

"version": "1.1.0",

**"resolved": "https://registry.npmjs.org/math-intrinsics/-
/math-intrinsics-1.1.0.tgz",**

**"integrity": "sha512-
/IXtbwEk5HTPyEwyKX6hGkYXxM9nbj64B+ilVJnC/R6B0
pH5G4V3b0pVbL7DBj4tkhBAppbQUlf6F6Xl9LHu1g==",**

"license": "MIT",

"engines": {

```
    "node": ">= 0.4"
  }
},
"node_modules/media-typer": {
  "version": "1.1.0",
  "resolved": "https://registry.npmjs.org/media-typer/-
/media-typer-1.1.0.tgz",
  "integrity": "sha512-
aisnrDP4GNe06UcKFnV5bfMNPBUw4jsLGaWwWfnH3v0
2GnBuXX2MCVn5RbrWo0j3pczUilYblq7fQ7Nw2t5XKw=
=",
  "license": "MIT",
  "engines": {
    "node": ">= 0.8"
  }
},
"node_modules/memory-pager": {
  "version": "1.5.0",
  "resolved": "https://registry.npmjs.org/memory-pager/-
/memory-pager-1.5.0.tgz",
```

"integrity": "sha512-ZS4Bp4r/Zoeq6+NLJpP+0Zzm0pR8whtGPf1XExKLJBaczGMnSi3It14OiNCStjQjM6NU1okjQGSxgEZN8eBYKg==",

"license": "MIT"

},

"node_modules/merge-descriptors": {

"version": "2.0.0",

"resolved":

"https://registry.npmjs.org/mergedescriptors/-/merge-descriptors-2.0.0.tgz",

"integrity": "sha512-Snk314V5ayFLhp3fkUREub6WtjBfPdCPY1Ln8/8munuLuiYhsABgBVWsozAG+MWMbVEvdcpci9R7ww22l9Q3g==",

"license": "MIT",

"engines": {

"node": ">=18"

},

"funding": {

"url": "https://github.com/sponsors/sindresorhus"

```
    }  
  },  
  "node_modules/mime-db": {  
    "version": "1.54.0",  
    "resolved": "https://registry.npmjs.org/mime-db/-  
/mime-db-1.54.0.tgz",  
    "integrity": "sha512-aU5EJuIN2WDemCcAp2vFBfp/m4EAhWJnUNSSw0ixs7/k  
Xbd6Pg64EmwJkNdFhB8aWt1sH2CTXrLxo/iAGV3oPQ==  
",  
    "license": "MIT",  
    "engines": {  
      "node": ">= 0.6"  
    }  
  },  
  "node_modules/mime-types": {  
    "version": "3.0.1",  
    "resolved": "https://registry.npmjs.org/mime-types/-  
/mime-types-3.0.1.tgz",  
    "integrity": "sha512-"
```

**xRc4oEhT6eaBpU1XF7AjpOFD+xQmXNB5OVKwp4tqCu
BpHLS/ZbBDrc07mYTDqVMg6PfxUjjNp85O6Cd2Z/5HW
A==",**

"license": "MIT",

"dependencies": {

"mime-db": "^1.54.0"

},

"engines": {

"node": ">= 0.6"

}

},

"node_modules/minimist": {

"version": "1.2.8",

**"resolved": "https://registry.npmjs.org/minimist/-
/minimist-1.2.8.tgz",**

**"integrity": "sha512-
2yyAR8qBkN3YuheJanUpWC5U3bb5osDywNB8RzDVIDw
DHbocAJveqqj1u8+SVD7jkWT4yvsHCpWqqWqAxb0zCA
==",**

"license": "MIT",

```
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
  
"node_modules/mkdirp": {  
  "version": "0.5.6",  
  "resolved": "https://registry.npmjs.org/mkdirp/-  
/mkdirp-0.5.6.tgz",  
  "integrity": "sha512-FP+p8RB8OWpF3YZBCrP5gtADmtXApB5AMLn+vdyA+  
PyxCjrCs00mjyUozssO33cwDeT3wNGdLxJ5M//YqtHAJw  
==",  
  "license": "MIT",  
  "dependencies": {  
    "minimist": "^1.2.6"  
  },  
  "bin": {  
    "mkdirp": "bin/cmd.js"  
  }  
}
```



```
  },  
  
  "node_modules/mongodb": {  
  
    "version": "6.20.0",  
  
    "resolved": "https://registry.npmjs.org/mongodb/-  
/mongodb-6.20.0.tgz",  
  
    "integrity": "sha512-Tl6MEIU3K4Rq3TSHd+sZQqRBoGIFsOgNrH5ltAcFBV62  
Re3Fd+FcaVf8uSEQFOJ51SDowDVttBTONMfoYWrWIQ  
==",  
  
    "license": "Apache-2.0",  
  
    "dependencies": {  
  
      "@mongodb-js/saslprep": "^1.3.0",  
  
      "bson": "^6.10.4",  
  
      "mongodb-connection-string-url": "^3.0.2"  
  
    },  
  
    "engines": {  
  
      "node": ">=16.20.1"  
  
    },  
  
    "peerDependencies": {
```

**"@aws-sdk/credential-providers": "^3.188.0",
"@mongodb-js/zstd": "^1.1.0 || ^2.0.0",**

"gcp-metadata": "^5.2.0",

"kerberos": "^2.0.1",

"mongodb-client-encryption": ">=6.0.0 <7",

"snappy": "^7.3.2",

"socks": "^2.7.1"

},

"peerDependenciesMeta": {

"@aws-sdk/credential-providers": {

"optional": true

},

"@mongodb-js/zstd": {

"optional": true

},

"gcp-metadata": {

"optional": true

},

"kerberos": {

```
"optional": true

},

"mongodb-client-encryption": {

  "optional": true

},

"snappy": {

  "optional": true

},

"socks": {

  "optional": true

}

}

},

"node_modules/mongodb-connection-string-url": {

  "version": "3.0.2",

  "resolved":

"https://registry.npmjs.org/mongodbconnection-string-url/-
/mongodb-connection-string-url-
3.0.2.tgz",
```

"integrity": "sha512-rMO7CGo/9BFwyZABcKAWL8UJwH/Kc2x0g72uhDWzG48URRax5TCIcJ7Rc3RZqffZzO/Gwff/jyKwCU9TN8gehA==",

"license": "Apache-2.0",

"dependencies": {

"@types/whatwg-url": "^11.0.2",

"whatwg-url": "^14.1.0 || ^13.0.0"

}

},

"node_modules/mongoose": {

"version": "8.19.0",

"resolved": "https://registry.npmjs.org/mongoose/-/mongoose-8.19.0.tgz",

"integrity": "sha512-Z4iRiBkC7aR7a/rxQxtUAUBasFdiXkBuv3EY4NwkRbs92xKA4pwzi1Q4D+odFBe+ChahMNAYg2JP+7tWzZM0sQ==",

"license": "MIT",

"dependencies": {

```
"bson": "^6.10.4",  
"kareem": "2.6.3",  
"mongodb": "~6.20.0",  
"mpath": "0.9.0",  
"mquery": "5.0.0",  
"ms": "2.1.3",  
"sift": "17.1.3"  
},  
"engines": {  
  "node": ">=16.20.1"  
},  
"funding": {  
  "type": "opencollective",  
  "url": "https://opencollective.com/mongoose"  
}  
},  
"node_modules/mpath": {  
  "version": "0.9.0",
```

"resolved": "https://registry.npmjs.org/mpath/-/mpath-0.9.0.tgz",

"integrity": "sha512-ikJRQ Tk8hw5DEoFVxHG1Gn9T/xcjtdnOKIU1JTmGjZZlg9LST2mBLmcX3/IClbgJydT2GOc15RnNy5mHmzfSew==",

"license": "MIT",

"engines": {

"node": ">=4.0.0"

}

},

"node_modules/mquery": {

"version": "5.0.0",

"resolved": "https://registry.npmjs.org/mquery/-/mquery-5.0.0.tgz",

"integrity": "sha512-iQMncpmEK8R8ncT8HJGsGc9Dsp8xcgYMV Sbs5jgnm1lFHTZqMJTUWTDx1LBO8+mK3tPNZWFLBghQEIOULSTHZg==",

"license": "MIT",

"dependencies": {

```
    "debug": "4.x"

  },

  "engines": {

    "node": ">=14.0.0"

  }

},

"node_modules/ms": {

  "version": "2.1.3",

  "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",

  "integrity": "sha512-6FlzubTLZG3J2a/NVCAleEhjqzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9xAOtHnSO/tXtF3WRTIA==",

  "license": "MIT"

},

"node_modules/multer": {

  "version": "2.0.2",

  "resolved": "https://registry.npmjs.org/multer/-/multer-2.0.2.tgz",
```

```
"integrity": "sha512-  
u7f2xaZ/UG8oLXHvtF/oWTRvT44p9ecwBBqTwgJVq0+4B  
W1g8OW01TyMEGWBHbyMOYVHXslaut7qEQ1meATXg w==",  
  
"license": "MIT",  
  
"dependencies": {  
  
  "append-field": "^1.0.0",  
  
  "busboy": "^1.6.0",  
  
  "concat-stream": "^2.0.0",  
  
  "mkdirp": "^0.5.6",  
  
  "object-assign": "^4.1.1",  
  
  "type-is": "^1.6.18",  
  
  "xtend": "^4.0.2"  
  
},  
  
"engines": {  
  
  "node": ">= 10.16.0"  
  
}  
  
},  
  
"node_modules/multer/node_modules/media-typer": {  
  
  "version": "0.3.0",
```


**"resolved": "https://registry.npmjs.org/media-typer/-
/media-typer-0.3.0.tgz",**

**"integrity": "sha512-
dq+qelQ9akHpcOl/gUVRTxVIOkAJ1wR3QAvb4RsVjS8oV
oFjDGTc679wJYmUmknUF5HwMLOgb5O+a3KxfWapPQ
=="**,

"license": "MIT",

"engines": {

"node": ">= 0.6"

}

},

"node_modules/multer/node_modules/mime-db": {

"version": "1.52.0",

**"resolved": "https://registry.npmjs.org/mime-db/-
/mime-db-1.52.0.tgz",**

**"integrity": "sha512-
sPU4uV7dYlvtWJxwwwHD0PuihVNiE7TyAbQ5SWxDCB9
mUYvOgroQOwYQQOKPJ8CIbE+1ETVlOoK1UC2nU3g
Yvg=="**,

"license": "MIT",

"engines": {

```
    "node": ">= 0.6"
  }
},
"node_modules/multer/node_modules/mime-types": {
  "version": "2.1.35",
  "resolved": "https://registry.npmjs.org/mime-types/-/mime-types-2.1.35.tgz",
  "integrity": "sha512-ZDY+bPm5zTTF+YpCrAU9nK0UgICYPT0QtT1NZWFv4s++TNkcgVaT0g6+4R2uI4MjQjzysHB1zxuWL50hzaeXiw==",
  "license": "MIT",
  "dependencies": {
    "mime-db": "1.52.0"
  },
  "engines": {
    "node": ">= 0.6"
  }
},
```

```
"node_modules/multer/node_modules/type-is": {  
  "version": "1.6.18",  
  
  "resolved": "https://registry.npmjs.org/type-is/-/type-is-  
1.6.18.tgz",  
  
  "integrity": "sha512-TkRKR9sUTxEH8MdfuCSP7VizJyzRNMjj2J2do2Jr3Kym5  
98JVdEksuzPQCnlFPW4ky9Q+iA+ma9BGm06XQBy8g==  
",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "media-typer": "0.3.0",  
  
    "mime-types": "~2.1.24"  
  
  },  
  
  "engines": {  
  
    "node": ">= 0.6"  
  
  }  
},  
  
"node_modules/negotiator": {  
  
  "version": "1.0.0",  
  
  "resolved": "https://registry.npmjs.org/negotiator/-
```

/negotiator-1.0.0.tgz",

"integrity": "sha512-8Ofs/AUQh8MaEcrlq5xOX0CQ9ypTF5dl78mjlMNfOK08fzpgTHQRQPBxcPIEtIw0yRpws+Zo/3r+5WRby7u3Gg==",

"license": "MIT",

"engines": {

"node": ">= 0.6"

}

},

"node_modules/object-assign": {

"version": "4.1.1",

"resolved": "https://registry.npmjs.org/object-assign/-/object-assign-4.1.1.tgz",

"integrity": "sha512-rJgTQnkUnH1sFw8yT6VSU3zD3sWmu6sZhIseY8VX+GRu3P6F7Fu+JNDoXfklElbLJSnc3FUQHVe4cU5hj+BcUg==",

"license": "MIT",

"engines": {

"node": ">=0.10.0"

```
    }  
  },  
  "node_modules/object-inspect": {  
    "version": "1.13.4",  
    "resolved": "https://registry.npmjs.org/object-inspect/-  
/object-inspect-1.13.4.tgz",  
    "integrity": "sha512-W67iLl4J2EXEGTbfeHCffrjDfitvLANg0UIX3wFUUSTx92  
KXRFegMHUVgSqE+vvhAbi4WqjGg9czysTV2Epbew==",  
    "license": "MIT",  
    "engines": {  
      "node": ">= 0.4"  
    },  
    "funding": {  
      "url": "https://github.com/sponsors/ljharb"  
    }  
  },  
  "node_modules/on-finished": {  
    "version": "2.4.1",
```

**"resolved": "https://registry.npmjs.org/on-finished/-
onfinished-2.4.1.tgz",**

**"integrity": "sha512-
oVlzkg3ENAhCk2zdv7IJwd/QUD4z2RxRwpkcGY8psCVcC
YZNq4wYnVWALHM+brtuJjePWiYF/ClmuDr8Ch5+kg==
",**

"license": "MIT",

"dependencies": {

"ee-first": "1.1.1"

},

"engines": {

"node": ">= 0.8"

}

},

"node_modules/once": {

"version": "1.4.0",

**"resolved": "https://registry.npmjs.org/once/-/once-
1.4.0.tgz",**

"integrity": "sha512-

**INaJgI+2Q5URQBkccEKHTQOPaXdUxnZZElQTZY0MF
UAuaEqe1E+Nyvvdz/aIyNi6Z9MzO5dv1H8n58/GELp3+w=
=",**

"license": "ISC",

"dependencies": {

"wrappy": "1"

}

},

"node_modules/parseurl": {

"version": "1.3.3",

**"resolved": "https://registry.npmjs.org/parseurl/-
/parseurl-1.3.3.tgz",**

**"integrity": "sha512-
CiyeOxFT/JZyN5m0z9PfXw4SCBJ6Sygz1Dpl0wqjlhDEGG
BP1GnsUVEL0p63hoG1fcj3fHynXi9NYO4nWOL+qQ=="**

"license": "MIT",

"engines": {

"node": ">= 0.8"

}

},

```
"node_modules/path-to-regexp": {  
  "version": "8.3.0",  
  "resolved": "https://registry.npmjs.org/path-to-regexp/-  
path-to-regexp-8.3.0.tgz",  
  "integrity": "sha512-  
7jdwVIRtsP8MYpdXSwOS0YdD0Du+qOoF/AEPIt88PcCF  
rZCzx41oxku1jD88hZBwbNUIEfpqvuhjFaMAqMTWnA==  
",  
  "license": "MIT",  
  "funding": {  
    "type": "opencollective",  
    "url": "https://opencollective.com/express"  
  }  
},  
"node_modules/proxy-addr": {  
  "version": "2.0.7",  
  "resolved": "https://registry.npmjs.org/proxy-addr/-  
proxy-addr-2.0.7.tgz",  
  "integrity": "sha512-  

```


**llQsMLSUDUPT44jdrU/O37qlnifitDP+ZwrmmZcoSKyLKv
tZxpyV0n2/bD/N4tBAAZ/gJEdZU7KMraoK1+XYAg==",**

"license": "MIT",

"dependencies": {

"forwarded": "0.2.0",

"ipaddr.js": "1.9.1"

},

"engines": {

"node": ">= 0.10"

}

},

"node_modules/punycode": {

"version": "2.3.1",

**"resolved": "https://registry.npmjs.org/punycode/-
/punycode-2.3.1.tgz",**

"integrity": "sha512-

**vYt7UD1U9Wg6138shLtLOvdAu+8DsC/ilFtEVHcH+wydcS
pNE20AfSOduf6MkRFahL5FY7X1oU7nKVZFtfq8Fg==",**

"license": "MIT",

```
"engines": {  
  "node": ">=6"  
}  
,  
  
"node_modules/qs": {  
  "version": "6.14.0",  
  
  "resolved": "https://registry.npmjs.org/qs/-/qs-  
6.14.0.tgz",  
  
  "integrity": "sha512-  
YWWTjgABSKcvs/nWBi9PycY/JiPJqOD4JA6o9Sej2AtvS  
GarXxKC3OQSk4pAarbdQlKAh5D4FCQkJNkW+GAn3w  
==",  
  
  "license": "BSD-3-Clause",  
  
  "dependencies": {  
    "side-channel": "^1.1.0"  
  },  
  
  "engines": {  
    "node": ">=0.6"  
  },  
},
```

```
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
  
"node_modules/range-parser": {  
  "version": "1.2.1",  
  "resolved": "https://registry.npmjs.org/range-parser/-  
/range-parser-1.2.1.tgz",  
  "integrity": "sha512-Hrgsx+orqoygnmhFbKaHE6c296J+HTAQXoxEF6gNupRO  
mmGJRozfG3ccAveqCBrwr/2yxQ5BVd/GTl5agOwSg==",  
  "license": "MIT",  
  "engines": {  
    "node": ">= 0.6"  
  }  
},  
  
"node_modules/raw-body": {  
  "version": "3.0.1",
```

**"resolved": "https://registry.npmjs.org/raw-body/-
/rawbody-3.0.1.tgz",**

**"integrity": "sha512-
9G8cA+tuMS75+6G/TzW8OtLzmBDMo8p1JRxN5AZ+LAp
8uxGA8V8GZm4GQ4/N5QNNQEnLmg6SS7wyuSmbKepiKq
A==",**

"license": "MIT",

"dependencies": {

"bytes": "3.1.2",

"http-errors": "2.0.0",

"iconv-lite": "0.7.0",

"unpipe": "1.0.0"

},

"engines": {

"node": ">= 0.10"

}

},

"node_modules/raw-body/node_modules/iconv-lite": {

"version": "0.7.0",

```
"resolved": "https://registry.npmjs.org/iconv-lite/-  
/iconv-lite-0.7.0.tgz",  
  
"integrity": "sha512-  
cf6L2Ds3h57VVmkZe+Pn+5APsT7FpqJtEhhieDCvrE2MK  
5Qk9MyffgQyuxQTm6BChfeZNtcOLHp9IcWRVcIcBQ=="  
,  
  
"license": "MIT",  
  
"dependencies": {  
  
  "safer-buffer": ">= 2.1.2 < 3.0.0"  
  
},  
  
"engines": {  
  
  "node": ">=0.10.0"  
  
},  
  
"funding": {  
  
  "type": "opencollective",  
  
  "url": "https://opencollective.com/express"  
  
}  
  
},  
  
"node_modules/readable-stream": {
```

```
"version": "3.6.2",  
  
  "resolved": "https://registry.npmjs.org/readablestream/-  
/readable-stream-3.6.2.tgz",  
  
  "integrity": "sha512-  
9u/sniCrY3D5WdsERHzHE4G2YCXqoG5FTHUiCC4SIbr6  
XcLZBY05ya9EKjYek9O5xOAwjGq+1JdGBAS7Q9ScoA=  
=",  
  
  "license": "MIT",  
  
  "dependencies": {  
  
    "inherits": "^2.0.3",  
  
    "string_decoder": "^1.1.1",  
  
    "util-deprecate": "^1.0.1"  
  
  },  
  
  "engines": {  
  
    "node": ">= 6"  
  
  }  
  
},  
  
"node_modules/router": {  
  
  "version": "2.2.0",
```

"resolved": "https://registry.npmjs.org/router/-/router-2.2.0.tgz",

"integrity": "sha512-nLTrUKm2UyiL7rlhapu/Zl45FwNgkZGaCpZbIHajDYgwIJCOzLSk+cIPAnsEqV955GjILJnKbdQC1nVPz+gAYQ==",

"license": "MIT",

"dependencies": {

"debug": "^4.4.0",

"depd": "^2.0.0",

"is-promise": "^4.0.0",

"parseurl": "^1.3.3",

"path-to-regexp": "^8.0.0"

},

"engines": {

"node": ">= 18"

}

},

"node_modules/safe-buffer": {

"version": "5.2.1",

**"resolved": "https://registry.npmjs.org/safe-buffer/-
/safe-buffer-5.2.1.tgz",**

**"integrity": "sha512-
rp3So07KcdmmKbGvgaNxQsJr7bGVSVk5S9Eq1F+ppbRo
70+YeaDxkw5Dd8NPN+GD6bjnYm2VuPuCXmpuYvmCX
Q==",**

"funding": [

{

"type": "github",

"url": "https://github.com/sponsors/feross"

},

{

"type": "patreon",

"url": "https://www.patreon.com/feross"

},

{

"type": "consulting",

"url": "https://feross.org/support"

}


```
  ],  
  
  "license": "MIT"  
  
},  
  
"node_modules/safer-buffer": {  
  
  "version": "2.1.2",  
  
  "resolved": "https://registry.npmjs.org/safer-buffer/-  
safer-buffer-2.1.2.tgz",  
  
  "integrity": "sha512-  
YZo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXX  
evb5dJI7tpyN2ADxGcQbHG7vcyRHk0cbwqcQriUtg==",  
  
  "license": "MIT"  
  
},  
  
"node_modules/send": {  
  
  "version": "1.2.0",  
  
  "resolved": "https://registry.npmjs.org/send/-/send-  
1.2.0.tgz",  
  
  "integrity": "sha512-  
uaW0WwXKpL9bIXE2o0bRhoL2EGXlrZxQ2ZQ4mgcfoBx  
dFmQold+qWsD2jLrfZ0trjKL6vOw0j//eAwcALFjKSw==",  
  
  "license": "MIT",
```

```
"dependencies": {  
  "debug": "^4.3.5",  
  "encodeurl": "^2.0.0",  
  "escape-html": "^1.0.3",  
  "etag": "^1.8.1",  
  "fresh": "^2.0.0",  
  "http-errors": "^2.0.0",  
  "mime-types": "^3.0.1",  
  "ms": "^2.1.3",  
  "on-finished": "^2.4.1",  
  "range-parser": "^1.2.1",  
  "statuses": "^2.0.1"  
},  
"engines": {  
  "node": ">= 18"  
}  
},  
"node_modules/serve-static": {
```

```
"version": "2.2.0",  
  
"resolved": "https://registry.npmjs.org/serve-static/-  
/serve-static-2.2.0.tgz",  
  
"integrity": "sha512-  
61g9pCh0Vnh7IutZjtLGGpTA355+OPn2TyDv/6ivP2h/AdA  
VX9azsoxm2/M6nZeQZNYBEwIcsne1mJd9oQItQ==",  
  
"license": "MIT",  
  
"dependencies": {  
  
  "encodeurl": "^2.0.0",  
  
  "escape-html": "^1.0.3",  
  
  "parseurl": "^1.3.3",  
  
  "send": "^1.2.0"  
  
},  
  
"engines": {  
  
  "node": ">= 18"  
  
}  
  
},  
  
"node_modules/setprototypeof": {  
  
  "version": "1.2.0",
```

**"resolved": "https://registry.npmjs.org/setprototypeof/-
/setprototypeof-1.2.0.tgz",**

**"integrity": "sha512-
E5LDX7Wrp85Kil5bhZv46j8jOeboKq5JMmYM3gVGdGH
8xFpPWXUMsNrlODCrkoxMEeNi/XZlIwuRvY4XNwYMJp
w=="**,

"license": "ISC"

},

"node_modules/side-channel": {

"version": "1.1.0",

**"resolved": "https://registry.npmjs.org/side-channel/-
/side-channel-1.1.0.tgz",**

**"integrity": "sha512-
ZX99e6tRweoUXqR+VBrslhda51Nh5MTQwou5tnUDgbty
M0dBgmhEDtWGP/xbKn6hqfPRHujUNwz5fy/wbbhnpw==
",**

"license": "MIT",

"dependencies": {

"es-errors": "^1.3.0",

"object-inspect": "^1.13.3",

"side-channel-list": "^1.0.0",

```
"side-channel-map": "^1.0.1",  
"side-channel-weakmap": "^1.0.2"  
},  
"engines": {  
  "node": ">= 0.4"  
},  
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
"node_modules/side-channel-list": {  
  "version": "1.0.0",  
  "resolved": "https://registry.npmjs.org/side-channellist/-  
side-channel-list-1.0.0.tgz",  
  "integrity": "sha512-  
FCLHtRD/gnpCiCHEiJLOWdmFP+wzCmDEkc9y7NsYxeF  
4u7Btsn1ZuwgwJGxImImHicJArLP4R0yX4c2KCrMrTA=  
=",  
  "license": "MIT",
```

```
"dependencies": {  
  "es-errors": "^1.3.0",  
  "object-inspect": "^1.13.3"  
},  
"engines": {  
  "node": ">= 0.4"  
},  
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
"node_modules/side-channel-map": {  
  "version": "1.0.1",  
  "resolved": "https://registry.npmjs.org/side-channelmap/-/side-channel-map-1.0.1.tgz",  
  "integrity": "sha512-VCjCNfgMsby3tTdo02nbjtM/ewra6jPHmpThenkTYh8pG9ucZ/1P8So4u4FGBek/BjpOVsDCMoLA/iuBKIFXRA==",  
  "license": "MIT",
```

```
"dependencies": {  
  "call-bound": "^1.0.2",  
  "es-errors": "^1.3.0",  
  "get-intrinsic": "^1.2.5",  
  "object-inspect": "^1.13.3"  
},  
"engines": {  
  "node": ">= 0.4"  
},  
"funding": {  
  "url": "https://github.com/sponsors/ljharb"  
}  
},  
"node_modules/side-channel-weakmap": {  
  "version": "1.0.2",  
  "resolved": "https://registry.npmjs.org/side-  
channelweakmap/-/side-channel-weakmap-1.0.2.tgz",  
  "integrity": "sha512-
```

**WPS/HvHQTYnHisLo9McqBHOJk2FkHO/tlpvldyrnem4ae
Qp4hai3gythswg6p01oSoTl58rcpiFAjF2br2Ak2A==",**

"license": "MIT",

"dependencies": {

"call-bound": "^1.0.2",

"es-errors": "^1.3.0",

"get-intrinsic": "^1.2.5",

"object-inspect": "^1.13.3",

"side-channel-map": "^1.0.1"

},

"engines": {

"node": ">= 0.4"

},

"funding": {

"url": "https://github.com/sponsors/ljharb"

}

},

"node_modules/sift": {

"version": "17.1.3",

"resolved": "https://registry.npmjs.org/sift/-/sift-17.1.3.tgz",

"integrity": "sha512-Rtlj66/b0ICeFzYTUNvX/EF1igRbbnGSvEyT79McoZa/DeGhMyC5pWKOEsZKnPkqtSeovd5FL/bjHWC3CIIvCQ==",

"license": "MIT"

},

"node_modules/sparse-bitfield": {

"version": "3.0.3",

"resolved": "https://registry.npmjs.org/sparse-bitfield/-/sparse-bitfield-3.0.3.tgz",

"integrity": "sha512-kvzhi7vqKTfkh0PZU+2D2Pillw2ymqJKujUcyPMd9Y75Nv4nPbGJZXNhxsGdQab2BmlDct1YnfQCguEvHr7VsQ==",

"license": "MIT",

"dependencies": {

"memory-pager": "^1.0.2"

}

},

"node_modules/statuses": {

```
"version": "2.0.2",  
  
"resolved": "https://registry.npmjs.org/statuses/-  
/statuses-2.0.2.tgz",  
  
"integrity": "sha512-  
DvEy55V3DB7uknRo+4iOGT5fP1slR8wQohVdknigZPMp  
MstaKJQWhwiYBACJE3Ul2pTnATihhBYnRhZQHGBiRw  
==",  
  
"license": "MIT",  
  
"engines": {  
  
  "node": ">= 0.8"  
  
}  
  
},  
  
"node_modules/streamsearch": {  
  
  "version": "1.1.0",  
  
  "resolved": "https://registry.npmjs.org/streamsearch/-  
/streamsearch-1.1.0.tgz",  
  
  "integrity": "sha512-  
Mcc5wHehp9aXz1ax6bZUyY5afg9u2rv5cqQI3mRrYkGC8r  
W2hM02jWuwjtL++LS5qinSyhj2QfLyNsuc+VsExg==",  
  
  "engines": {
```

```
    "node": ">=10.0.0"
  }
},
"node_modules/string_decoder": {
  "version": "1.3.0",
  "resolved": "https://registry.npmjs.org/string_decoder/-/string_decoder-1.3.0.tgz",
  "integrity": "sha512-hkRX8U1WjJFd8LsDJ2yQ/wWWxaopEsABU1XfkM8A+j0+85JAGppt16cr1Whg6KIbb4okU6Mql6BOj+uup/wKeA=="
,
  "license": "MIT",
  "dependencies": {
    "safe-buffer": "~5.2.0"
  }
},
"node_modules/toidentifier": {
  "version": "1.0.1",
  "resolved": "https://registry.npmjs.org/toidentifier/-/toidentifier-1.0.1.tgz",
```

"integrity": "sha512-o5sSPKEkg/DIQNmH43V0/uerLrpzVedkUh8tGNvaeXpfpuwjKenlSox/2O/BTlZUtEe+JG7s5YhEz608PlAHRA==",

"license": "MIT",

"engines": {

"node": ">=0.6"

}

},

"node_modules/tr46": {

"version": "5.1.1",

"resolved": "https://registry.npmjs.org/tr46/-/tr46-5.1.1.tgz",

"integrity": "sha512-hdF5ZgjTqgAntKkklYw0R03MG2x/bSzTtkxmIRw/sTNV8YXsCJ1tfLAX23lhxhHJlEf3CRCOCGGWw3vI3GaSPw==",

"license": "MIT",

"dependencies": {

"punycode": "^2.3.1"

},

```
"engines": {  
  "node": ">=18"  
}  
,  
  
"node_modules/type-is": {  
  "version": "2.0.1",  
  
  "resolved": "https://registry.npmjs.org/type-is/-/type-is-  
2.0.1.tgz",  
  
  "integrity": "sha512-OZs6gsjF4vMp32qrCbiVSkrFmXtG/AZhY3t0iAMrMBiAZ  
yV9oALtXO8hsrHbMXF9x6L3grlFuW2oAz7cav+Gw==",  
  
  "license": "MIT",  
  
  "dependencies": {  
    "content-type": "^1.0.5",  
    "media-typer": "^1.1.0",  
    "mime-types": "^3.0.0"  
  },  
  
  "engines": {  
    "node": ">= 0.6"
```

```
    }  
  
  },  
  
  "node_modules/typedarray": {  
  
    "version": "0.0.6",  
  
    "resolved": "https://registry.npmjs.org/typedarray/-  
/typedarray-0.0.6.tgz",  
  
    "integrity": "sha512-  
/aCDEGatGvZ2BIk+HmLf4ifCJFwvKFNb9/JeZPMulfgFra  
cn9QFcAf5GO8B/mweUjSoblS5In0cWhqpfs/5PQA==",  
    "license": "MIT"  
  
  },  
  
  "node_modules/unpipe": {  
  
    "version": "1.0.0",  
  
    "resolved": "https://registry.npmjs.org/unpipe/-/unpipe-  
1.0.0.tgz",  
  
    "integrity": "sha512-  
pJy2bYhSsufwWIKwPc+l3cN7+wuJlK6uz0YdJEOlQDbl6jo  
/YlPi4mb8agUkVC8BF7V8NuzeyPNqRksA3hztKQ==",  
  
    "license": "MIT",  
  
    "engines": {
```

```
"node": ">= 0.8"  
  
}  
  
},  
  
"node_modules/util-deprecate": {  
  
  "version": "1.0.2",  
  
  "resolved": "https://registry.npmjs.org/util-deprecate/-  
/util-deprecate-1.0.2.tgz",  
  
  "integrity": "sha512-  
EPD5q1uXyFxJpCrLnCc1nHnq3gOa6DZBocAlI2TaSCA7  
VCJ1UJDMagCzIkXNsUYfD1daK//LTEQ8xiIbrHtcw==",  
  
  "license": "MIT"  
  
},  
  
"node_modules/vary": {  
  
  "version": "1.1.2",  
  
  "resolved": "https://registry.npmjs.org/vary/-/vary-  
1.1.2.tgz",  
  
  "integrity": "sha512-  
BNGbWLfd0eUPabhkXUVm0j8uuvREyTh5ovRa/dyow/Bq  
AbZJyC+5fU+IzQOzmAKzYqYRAISoRhdQr3eIZ/PXqg==  
",
```

```
"license": "MIT",  
  
"engines": {  
  
  "node": ">= 0.8"  
  
}  
  
},  
  
"node_modules/webidl-conversions": {  
  
"version": "7.0.0",  
  
"resolved":  
"https://registry.npmjs.org/webidlconversions/-/webidl-  
conversions-7.0.0.tgz",  
  
"integrity": "sha512-  
VwddBukDzu71offAQR975unBIGqfKZpM+8ZX6ySk8nYh  
Voo5CYaZyzt3YBvYtRtO+aoGlqxPg/B87NGVZ/fu6g==",  
  
"license": "BSD-2-Clause",  
  
"engines": {  
  
  "node": ">=12"  
  
}  
  
},  
  
"node_modules/whatwg-url": {
```



```
"version": "14.2.0",  
  
"resolved": "https://registry.npmjs.org/whatwg-url/-  
/whatwg-url-14.2.0.tgz",  
  
"integrity": "sha512-  
De72GdQZzNTUBBChsXueQUnPKDkg/5A5zp7pFDuQAJ5  
UFoENpiACU0wlCvzpAGnTkj++ihpKwKyYewn/XNUbKw  
==",  
  
"license": "MIT",  
  
"dependencies": {  
  
  "tr46": "^5.1.0",  
  
  "webidl-conversions": "^7.0.0"  
  
},  
  
"engines": {  
  
  "node": ">=18"  
  
}  
  
},  
  
"node_modules/wrappy": {  
  
  "version": "1.0.2",  
  
  "resolved": "https://registry.npmjs.org/wrappy/-  
/wrappy-1.0.2.tgz",
```

**"integrity": "sha512-
l4Sp/DRseor9wL6EvV2+TuQn63dMkPjZ/sp9XkghTEbV9K
IPS1xUsZ3u7/IQO4wxteFB4bgpQPRcR3QCvezPcQ==",**

"license": "ISC"

},

"node_modules/xtend": {

"version": "4.0.2",

**"resolved": "https://registry.npmjs.org/xtend/-/xtend-
4.0.2.tgz",**

**"integrity": "sha512-
LKYU1iAXJXUgAXn9URjiu+MWhyUXHsvfp7mcuYm9dS
UKK0/CjtrUwFAxD82/mCWbtLsGjFIad0wIsod4zrTAEQ=
=",**

"license": "MIT",

"engines": {

"node": ">=0.4"

}

}

}

}

```
3.) const express = require("express");

const mongoose = require("mongoose");

const multer = require("multer"); const

cors = require("cors");


const app = express(); const

PORT = 5000;


// Middleware app.use(cors());

app.use(express.json());


// MongoDB connection

mongoose.connect("mongodb://localhost:27017/fileDB",

{ useNewUrlParser: true, useUnifiedTopology: true,

});

const conn = mongoose.connection;
```

```
let gfs;
```

```
conn.once("open", () => { console.log("MongoDB connected");
```

```
  gfs = new mongoose.mongo.GridFSBucket(conn.db, {  
  bucketName: "uploads" });  
});
```

```
// Multer storage (memory storage) const
```

```
storage = multer.memoryStorage(); const
```

```
upload = multer({ storage });
```

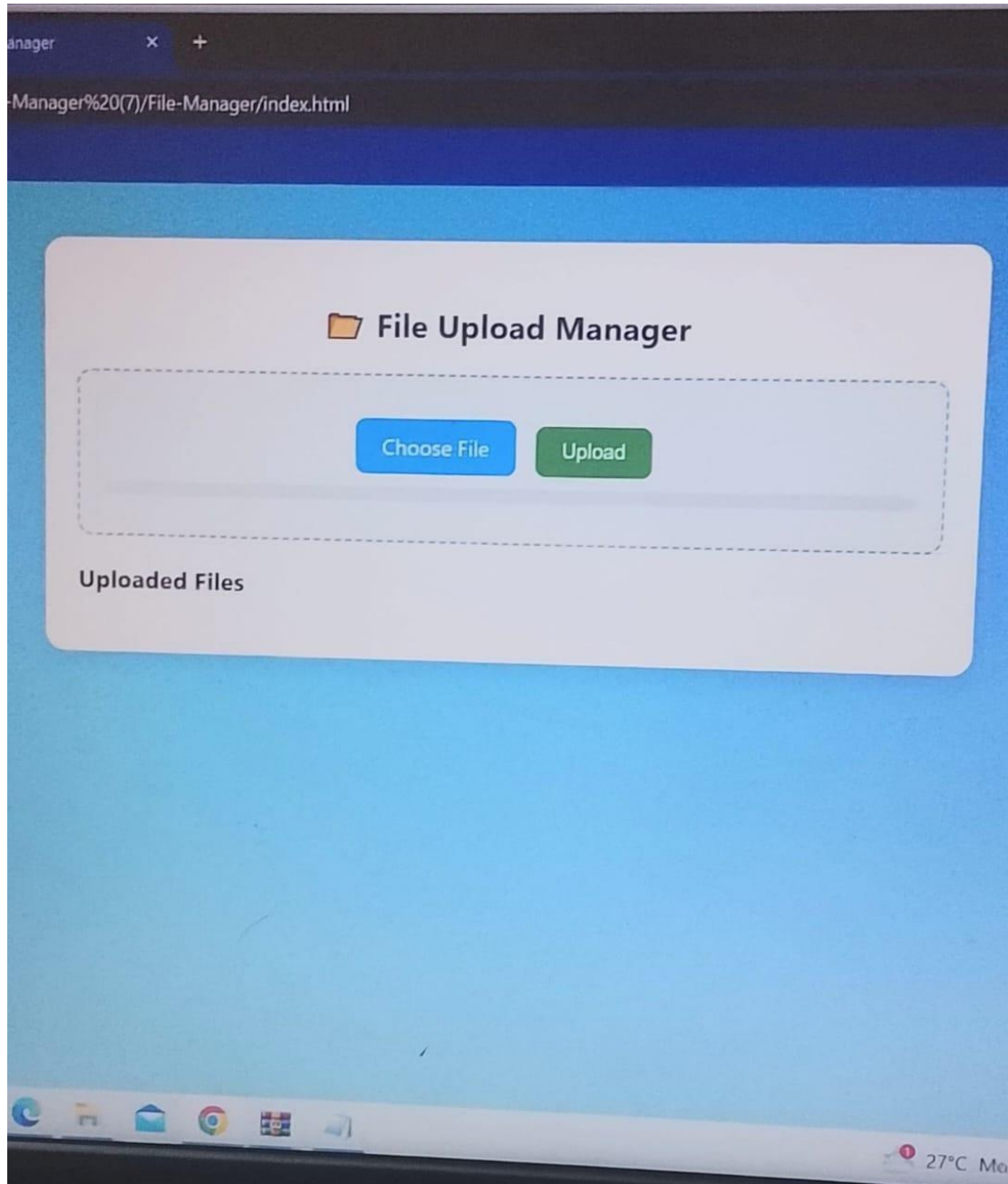
```
// Upload File app.post("/upload", upload.single("file"),
```

```
(req, res) => { if (!req.file) return res.status(400).json({  
error: "No file uploaded" });
```

```
  console.log(" Received file:", req.file.originalname);
```

```
const writeStream =  
gfs.openUploadStream(req.file.originalname, {  
contentType: req.file.mimetype,  
});  
  
writeStream.end(req.file.buffer); writeStream.on("finish",  
(file) => { console.log(" Saved to GridFS:",  
file.filename); res.json({ fileId: file._id, fileName:  
file.filename, fileUrl:  
`http://localhost:${PORT}/files/${file._id}`,  
});  
});  
  
writeStream.on("error", (err) => {  
console.error(" Error saving file:", err);  
res.status(500).json({ error: "Error saving file" });  
});  
});
```

OUTPUT:



Final demo walkthrough

1.1 Overview

1. Project Overview

To design and implement a system that allows users to upload, view, manage, and download files securely and efficiently.

1.2 Key Features:

File upload and preview

File categorization or tagging

User authentication (optional)

File download and delete options

Storage management (local or cloud)

1.3 Project Setup

Briefly show the folder structure:

/frontend – React / HTML interface

/backend – Node.js / Python API for file handling

/uploads – Directory or cloud bucket for files

Explain how the project runs (e.g., npm start or python app.py)

Mention dependencies (like multer in Node.js or Flask-Uploads in Python).

1.4 Core Features Demonstration

a. Home / Dashboard Page

Show the dashboard with file statistics (total uploads, file types, etc.)

If you have a login page, show user login/signup process.

b. File Upload Functionality

Click on “Upload File” button.

Select a file (PDF, image, or text).

Demonstrate progress bar (if implemented).

Confirm upload success message.

**Show the file entry appearing in the list with details:
filename, type, date, size.**

c. File Management

Show how files can be:

**Viewed / Previewed: Opens the file in a viewer or a
new tab.**

Downloaded: Click download icon to save locally.

Deleted: Click delete icon and confirm removal.

**Mention how the UI updates dynamically (without
reload).**

d. Categorization or Search

Demonstrate filtering or searching files by name/type/date (if implemented).

Example: “Show me all image files uploaded this week.”

e. Storage / Backend Logic

Show how uploaded files are stored:

Local folder path or

Cloud (AWS S3 / Firebase / Google Drive integration).

Explain how metadata is stored in database (filename, file path, user ID, timestamp).

1.5 Additional Features (Enhancements)

File size validation

File type restriction

Drag-and-drop upload area

Responsive design (mobile-friendly)

Error handling (invalid file, duplicate file name, etc.)

2. project report

2.1 introduction

The File Upload Manager project aims to address these issues by providing a streamlined and user-friendly platform where users can securely upload, organize, and manage their digital files. Designed as a web-based application, it supports multiple file formats, offers essential file operations (such as rename, delete, and download), and integrates access control mechanisms to ensure data protection.

This project demonstrates the effective use of web technologies to build a scalable and efficient file management system. It emphasizes simplicity, security, and performance, making it suitable for individuals, small teams, or educational institutions seeking an easy-to-use file handling solution.

2.2 objectives

To design a responsive and secure platform for uploading files.

To support multiple file formats (e.g., .jpg, .png, .pdf, .docx, etc.).

To manage files efficiently through search, filter, and delete options.

To ensure secure file storage with authentication and access control.

To provide cloud integration for scalability (e.g., AWS S3 or Firebase).

2.3 Tools and Technologies Used

Frontend: HTML, CSS, JavaScript, React.js (or Angular/Vue)

Backend: Node.js / Express.js (or Python Flask/Django)

Database: MongoDB / MySQL / Firebase

**Cloud Storage: AWS S3 / Google Drive API / Firebase
Storage**

Version Control: GitHub

Deployment: Netlify / Vercel / Render

2.4 System Architecture

The system follows a client-server model:

Client Side: Handles user interaction and file upload requests.

Server Side: Receives files, validates type and size, stores metadata in a database, and uploads the actual file to a cloud or local storage.

Database: Stores file metadata, user credentials, and activity logs.

2.5 Key Features

User Authentication: Login and registration to ensure secure access.

File Upload: Upload single or multiple files simultaneously.

File Preview: Preview images or documents before uploading.

File Management: Rename, delete, and download uploaded files.

Search and Filter: Easily find files based on type or name.

Storage Management: Displays used and available storage.

2.6 Testing and Validation

The system has been tested using various file types and sizes to ensure stability and accuracy.

Functional Testing: Verified upload, delete, and download operations.

Security Testing: Ensured files are accessed only by authorized users.

Performance Testing: Measured upload speed and response time.

2.7 Results and Discussion

The File Upload Manager successfully handles multiple concurrent uploads and provides a smooth user experience. It ensures secure file management and demonstrates scalability through cloud integration.

The project meets the intended objectives and can be extended with additional features like sharing links, file compression, or AI-based file categorization.

2.8 Future Enhancements

Integration with email or link-sharing options.

Implementation of file version control.

Real-time progress bar with upload analytics.

Encrypted file storage for enhanced security.

Mobile application version for easy access.

3. Screenshots Section

Include the following screenshots (you can take them from your working application):

Screenshot Title	Description
-------------------------	--------------------

Home Page / Dashboard	Displays the welcome page and navigation options for uploading or managing files.
------------------------------	--

Upload File Page	Shows the upload form where users can select files, view progress, and submit.
-------------------------	---

File List / Manager View	Displays a table or list of uploaded files with metadata (filename, size, date, type).
---------------------------------	---

View / Download File	Shows how a file can be previewed or downloaded from the system.
-----------------------------	---

Error / Validation Message	Screenshot of an error message for invalid file type or exceeded file size.
-----------------------------------	--

API Testing (Postman)	Screenshot of a successful file upload or retrieval API call using Postman or Swagger UI.
------------------------------	--

Tip:

If you're submitting this as a project, label screenshots like:

Fig 1. Home Page → Fig 2. Upload Form → Fig 3. API

Response for clarity.

2. API Documentation

Base URL

`http://localhost:5000/api/files`

(or your deployed URL, e.g.,

`https://fileuploadmanager.vercel.app/api/files`

)

Endpoints

1. Upload File

Endpoint:

POST /upload

Description:

Uploads a new file to the server.

Request:

Headers: Content-Type: multipart/form-data

Body:

Field	Type	Description	file	File
-------	------	-------------	------	------

The file to be uploaded				
-------------------------	--	--	--	--

Response:

```
{  
  "message": "File uploaded successfully",  
  "fileName": "document.pdf",
```

"fileUrl": "http://localhost:5000/uploads/document.pdf" }

2. Get All Files

Endpoint:

GET /list

Description:

Fetches the list of all uploaded files.

Response:

```
[  
  {  
    "id": "1",  
    "fileName": "report.pdf",
```

```
"fileSize": "1.2MB",  "uploadedAt": "2025-10-09T09:20:00Z"
```

```
}
```

```
]
```

3. Download File

Endpoint:

GET /download/:filename

Description:

Allows users to download a specific file by its name.

Response:

Returns the requested file as a binary stream.

4. Delete File

Endpoint:

DELETE /delete/:filename

Description:

Deletes the selected file from the server.

Response:

```
{  
  "message": "File deleted successfully"  
}
```

5. File Metadata

Endpoint:

GET /metadata/:filename

Description:

Fetches metadata like upload date, size, and type for a specific file.

Response:

```
{  
  "fileName": "project.zip",  
  "size": "2.3MB",  
  "type": "application/zip",  
  "uploadedAt": "2025-10-09T12:45:00Z"  
}
```

Optional Advanced Features

Authentication: Token-based file upload using JWT.

Cloud Storage: Integration with AWS S3 / Google Cloud Storage.

Progress Bar: Real-time upload progress in the frontend using Axios or Fetch events.

Versioning: Maintain different versions of the same file.

Would you like me to create a formatted project report section that includes these screenshots placeholders and API documentation (ready for submission in a .docx or .pdf)? I can generate that for you.

You said:

Given in the point wise

ChatGPT said:

Sure Here's your File Upload Manager – Screenshot and API Documentation given in clear point-wise format Screenshots Section

Home Page / Dashboard

Displays welcome screen and navigation options (Upload, Manage Files, etc.).

Upload File Page

Contains a form to select and upload files.

Shows file progress and success message.

File List Page

Lists all uploaded files with details such as file name, size, and upload date.

View / Download Page

Allows the user to preview or download files.

Error / Validation Message

Displays alerts for unsupported file types or large file sizes.

API Test Screenshot (Postman / Swagger)

Shows API testing with successful upload or list retrieval response.

API Documentation 1.

Base URL

<http://localhost:5000/api/files>

2. Upload File API

Method: POST

Endpoint: /upload

Description: Used to upload files to the server.

Request Header:

Content-Type: multipart/form-data

Request Body:

file: File (the file to be uploaded)

Response Example:

```
{  
  "message": "File uploaded successfully",  
  "fileName": "document.pdf",  
  "fileUrl": "http://localhost:5000/uploads/document.pdf"  
}
```

3. Get All Files API

Method: GET

Endpoint: /list

Description: Returns all uploaded file details.

Response Example:

```
[  
  {  
    "id": "1",  
    "fileName": "report.pdf",  
    "fileSize": "1.2MB",  
    "uploadedAt": "2025-10-09T09:20:00Z"  
  }  
]
```

4. Download File API

Method: GET

Endpoint: /download/:filename

Description: Used to download a specific file from the server.

5. Delete File API

Method: DELETE

Endpoint: /delete/:filename

Description: Deletes a file from the server.

Response Example:

```
{
```

```
  "message": "File deleted successfully"
```

```
}
```

6. Get File Metadata API

Method: GET

Endpoint: /metadata/:filename

Description: Fetches metadata (name, size, type, date) of a particular file.

Response Example:

```
{  
  "fileName": "project.zip",  
  "size": "2.3MB",  
  "type": "application/zip",  
  "uploadedAt": "2025-10-09T12:45:00Z"  
}
```

Optional Advanced Features

JWT Authentication for secure file uploads.

Cloud storage integration (AWS S3 / Google Cloud).

Real-time progress bar during upload.

File version tracking.

Role-based access control

4. Challenges and Solutions for File Upload Manager

1. File Size Limitation

Challenge: Uploading large files can cause slow performance, timeout errors, or exceed server limits.

Solution:

Implement chunked uploads (split files into smaller parts).

Configure server settings (e.g., `max_file_size` in Node.js, PHP, etc.).

Use cloud storage services (AWS S3, Google Cloud Storage) for handling large files efficiently.

2. File Format Restrictions

Challenge: Users may upload unsupported or unsafe file types (e.g., .exe, .bat).

Solution:

Validate file MIME types and extensions before upload.

Allow only specific file formats (e.g., .jpg, .pdf, .png).

Perform server-side validation for extra security.

3. Security and Malware Risks

Challenge: Uploaded files may contain viruses or malicious scripts that threaten system security.

Solution:

Scan files with antivirus libraries/APIs.

Store uploads outside the web root to prevent direct access.

Rename files or generate unique IDs to avoid path traversal attacks.

4. Storage Management

Challenge: Managing limited storage space and organizing uploaded files efficiently.

Solution:

Use cloud-based storage solutions.

Set up automatic cleanup of old or unused files.

Maintain folder structure or database mapping for better file management.

5. Upload Speed and Network Reliability

Challenge: Slow or unstable internet connections can interrupt uploads.

Solution:

Use resumable uploads so users can continue from where they left off.

Implement progress indicators to enhance user experience.

Compress files before uploading.

6. Duplicate File Handling

Challenge: Users may upload the same file multiple times, wasting space.

Solution:

Check for duplicate file names or hashes before saving.

Provide an option to replace or skip duplicate files.

7. User Experience (UX)

Challenge: Confusing or slow upload interfaces frustrate users.

Solution:

Show upload progress bars, file previews, and status messages.

Allow drag-and-drop uploads for ease of use.

Provide error notifications with clear messages.

8. Scalability

Challenge: Handling many simultaneous uploads from multiple users.

Solution:

Use load balancers and asynchronous processing.

Implement microservices architecture for scalability.

Optimize database and storage handling for concurrent uploads.

9. Data Integrity

Challenge: Uploaded files may get corrupted during transfer.

Solution:

Use checksums (MD5, SHA) to verify integrity.

Retry mechanism for failed uploads.

10. Authentication and Authorization

Challenge: Unauthorized users might access or upload files.

Solution:

Implement login-based access control (JWT, OAuth).

Assign user-specific folders or access permissions.

5. gitub readme & setup guide

Setup Guide

1. Prerequisites

Make sure you have installed:

Node.js

(v14+)

npm

Git

MongoDB

or local database (optional)

2. Clone the Repository git clone

**[https://github.com/your-username/file-
uploadmanager.git](https://github.com/your-username/file-uploadmanager.git) cd file-upload-manager**

3. Backend Setup cd backend

npm install

Create a .env file in the backend folder:

PORT=5000

MONGO_URI=your_mongo_connection_string

UPLOAD_PATH=uploads/

Start the backend server:

npm start

Server will run at: <http://localhost:5000>

4. Frontend Setup cd ../frontend

npm install

Start the frontend:

npm start

Frontend will run at: <http://localhost:3000>

5. Upload & Manage Files

Open your browser and visit: <http://localhost:3000>

Use the upload interface to select and upload files.

View uploaded files in the list and download/delete as needed.

API Endpoints

Method	Endpoint	Description
POST	/api/upload	Upload a file
GET	/api/files	Fetch all uploaded files
GET	/api/files/:id	Download specific file
DELETE	/api/files/:id	Delete a file

Security & Validation

File type and size validation before upload

Sanitized file names to prevent malicious paths

Secure file storage using backend middleware

Deployment

You can deploy using:

Frontend: Netlify / Vercel

Backend: Render / Railway / AWS EC2

Database: MongoDB Atlas

Screenshots

File Upload Interface

Upload button and progress bar

File List Page

Displays uploaded files with download/delete options

Backend API Logs

Shows file upload success messages in the console

Future Enhancements

Add user authentication

Enable cloud storage integration (AWS S3, Firebase)

Implement drag-and-drop upload

Add file sharing via links

Author

Your Name

your.email@example.com

6. final decision

- git hub repository link:[insert link]

-deployed application link:[insert link]

Conclusion: The File Upload Manager project successfully demonstrates a robust, efficient, and user-friendly system for uploading, managing, and organizing files. It ensures secure file handling, supports multiple file formats, and provides a responsive interface for end-users. Through this project, key functionalities such as file validation, storage management,

and error handling have been effectively implemented, highlighting the practicality and scalability of the system. Overall, the project meets its objectives of simplifying file management and enhancing productivity.

Further Scope

The File Upload Manager can be enhanced and expanded in the following ways:

Cloud Integration: Support uploading files directly to cloud storage platforms such as AWS S3, Google Drive, or Azure Blob Storage for scalability.

Advanced Security: Implement encryption for uploaded files, multi-factor authentication, and virus scanning to enhance security.

File Versioning: Add version control for files to track changes and maintain previous versions.

Search & Categorization: Introduce intelligent search, tagging, and categorization for faster file retrieval.

Collaboration Features: Allow multiple users to access, share, and collaborate on files in real-time.

Mobile App Integration: Create a mobile-friendly version or app for on-the-go file management.

Analytics & Reporting: Provide insights into file usage patterns, storage consumption, and user activity.