filez++ (SSH File Manager)

RAHUL R - LKTE18MCA065

Department of Computer Applications
Rajiv Gandhi Institute of Technology, Kottayam

Guided By :
Prof. Jane George
Asst. Professor
Dept. of MCA

Introduction

- As remote working is going on companies have to use servers remotely.
- A Utility Tool for Browsing files from a Remote server using protocols like Secure Shell Host (SSH).
- The Software is an installable, Desktop Application developed using Electron.JS (Packaging Tool) by using Web Technologies including Node.JS, Express.JS, Angular, EJS.

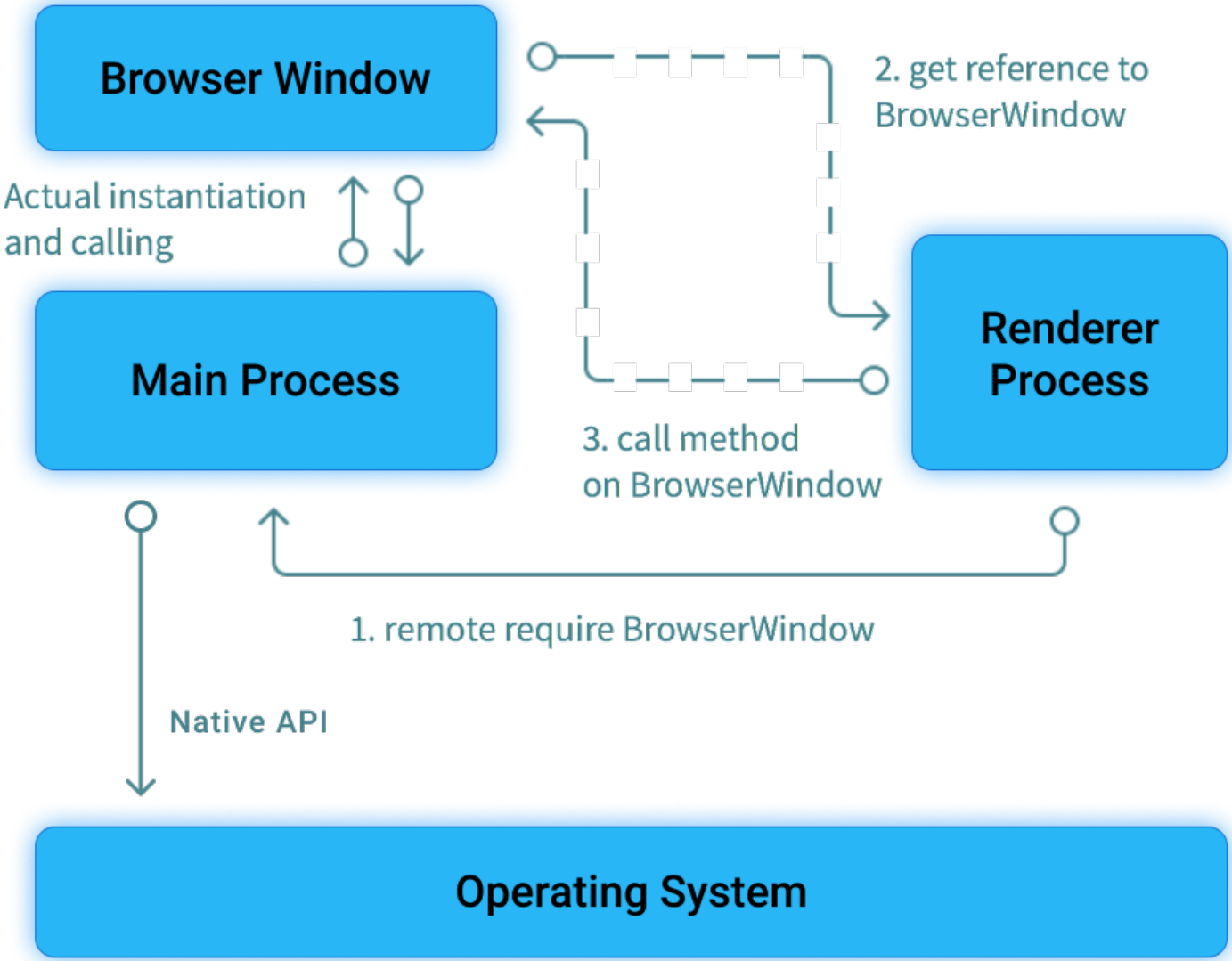
Existing System

- Cannot save connections
- Cannot connect to multiple ssh servers at once
- User Experience Issue, hard to use for naive users
- Server-side installation required
- Not Portable (User have to install the software in their PC)

Proposed System

- Add/Edit/Test/Delete Connections
- Connect to multiple SSH servers at once.
- GUI is easy to use, and highly customizable
- Most features of Native File Browsers like Windows File Explorer.
- Logs and User activity available.
- Server-side installation is not required
- Portable (Can run without installation on Users PC)
- Can run on Windows, Ubuntu, Fedora, Debian, MAC, etc.
- Logs and User activity available.
- Does not take much resources like RAM and CPU.

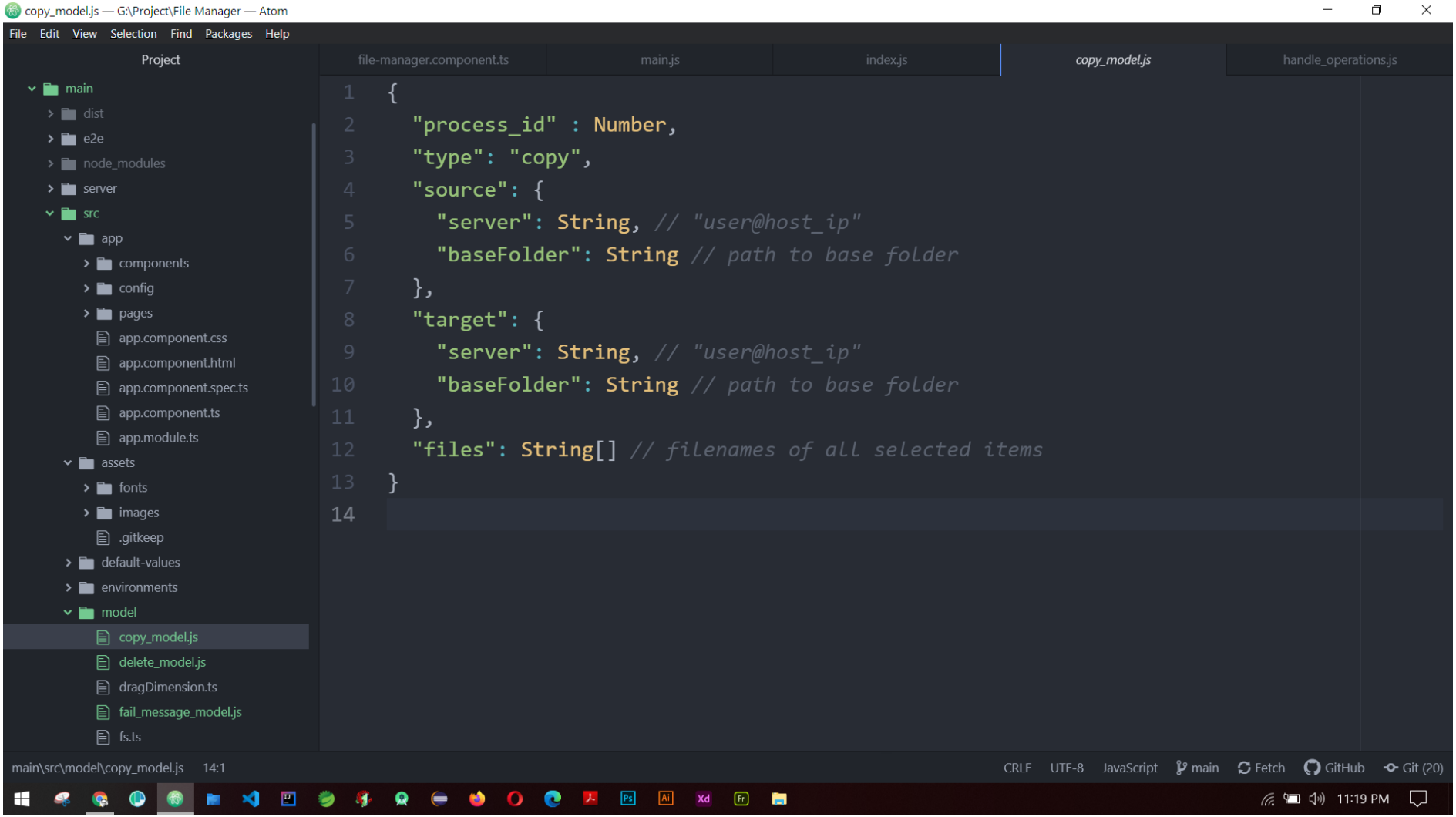
System Architecture



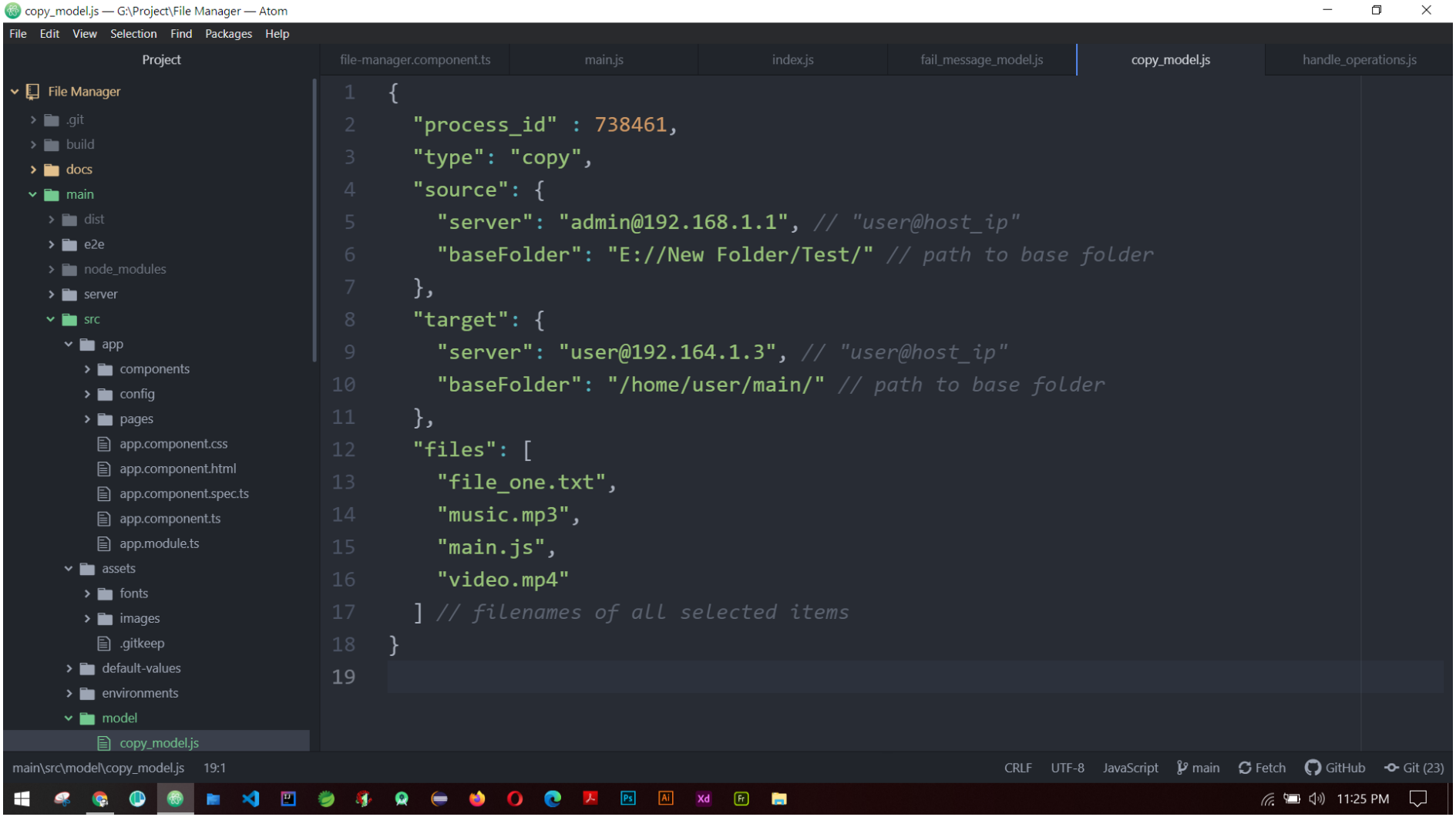
Product Backlog

Sl No.	User Stories	Priority	Estimated Time
1	User can browse local files	2	4 days
2	User can create new file	2	2 days
3	User can create new folder	2	1 day
4	User can delete files and folders	2	7 days
5	User can rename files and folders	2	5 days
6	User can open files, folders and shortcut files	2	4 days
7	User can add, edit, delete, test ssh connections	1	13 days
8	User can browse SSH files	1	2 days
9	User can create new file/folder	1	1 day
10	User can delete/rename files/folders	1	5 days
11	User can open folders	1	3 days
12	User can copy/paste files/folders	1	7 days
13	User can cut/paste files/folders	1	1 day

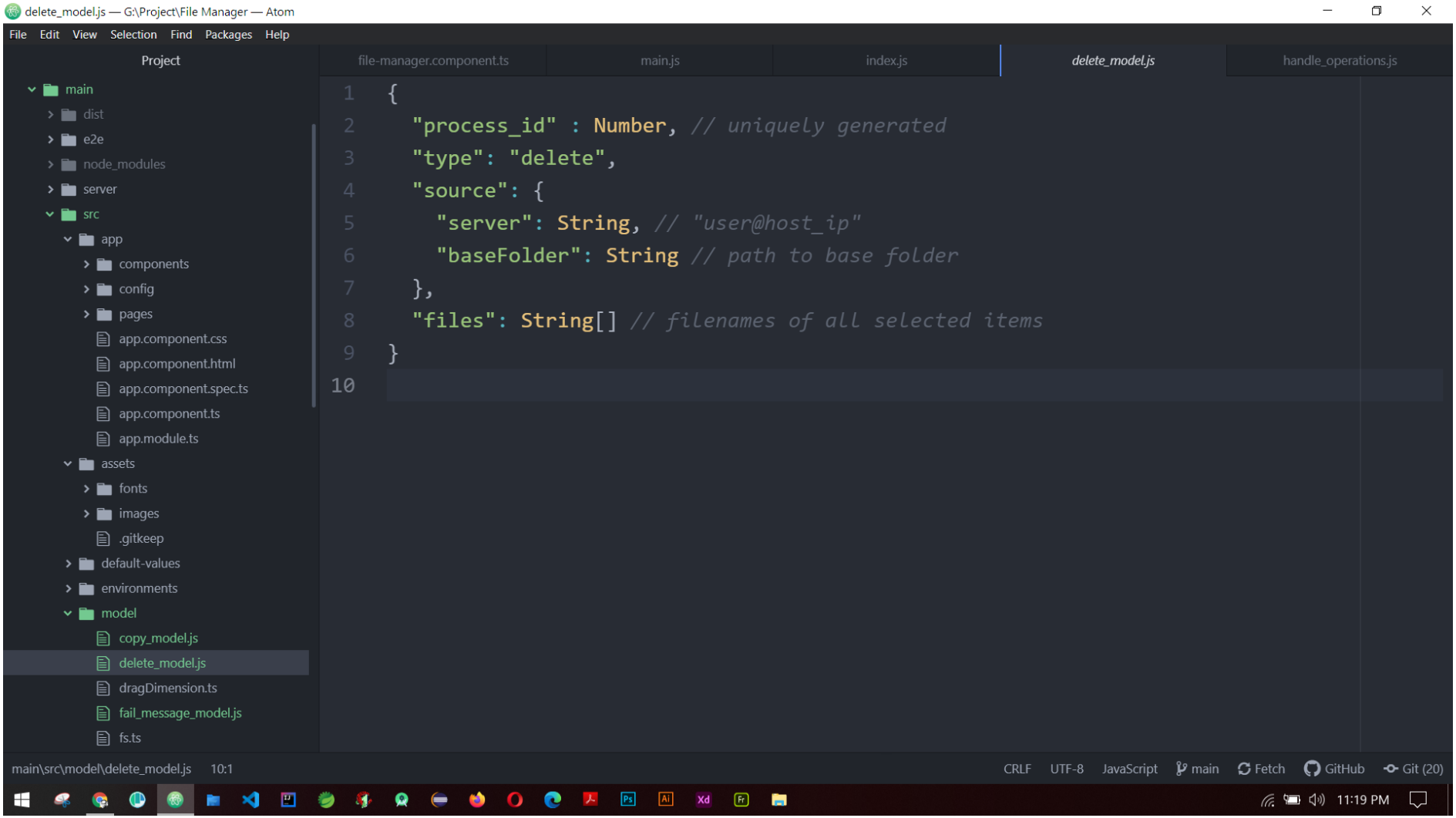
Data Model


copy_model.js — G:\Project\File Manager — Atom
File Edit View Selection Find Packages Help
Project
main
dist
e2e
node_modules
server
src
app
components
config
pages
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.module.ts
assets
fonts
images
.gitkeep
default-values
environments
model
copy_model.js
delete_model.js
dragDimension.ts
fail_message_model.js
fs.ts
file-manager.component.ts
main.js
index.js
copy_model.js
handle_operations.js
{
"process_id" : Number,
"type": "copy",
"source": {
"server": String, // "user@host_ip"
"baseFolder": String // path to base folder
},
"target": {
"server": String, // "user@host_ip"
"baseFolder": String // path to base folder
},
"files": String[] // filenames of all selected items
}
main\src\model\copy_model.js 14:1
CRLF UTF-8 JavaScript main Fetch GitHub Git (20)
11:19 PM

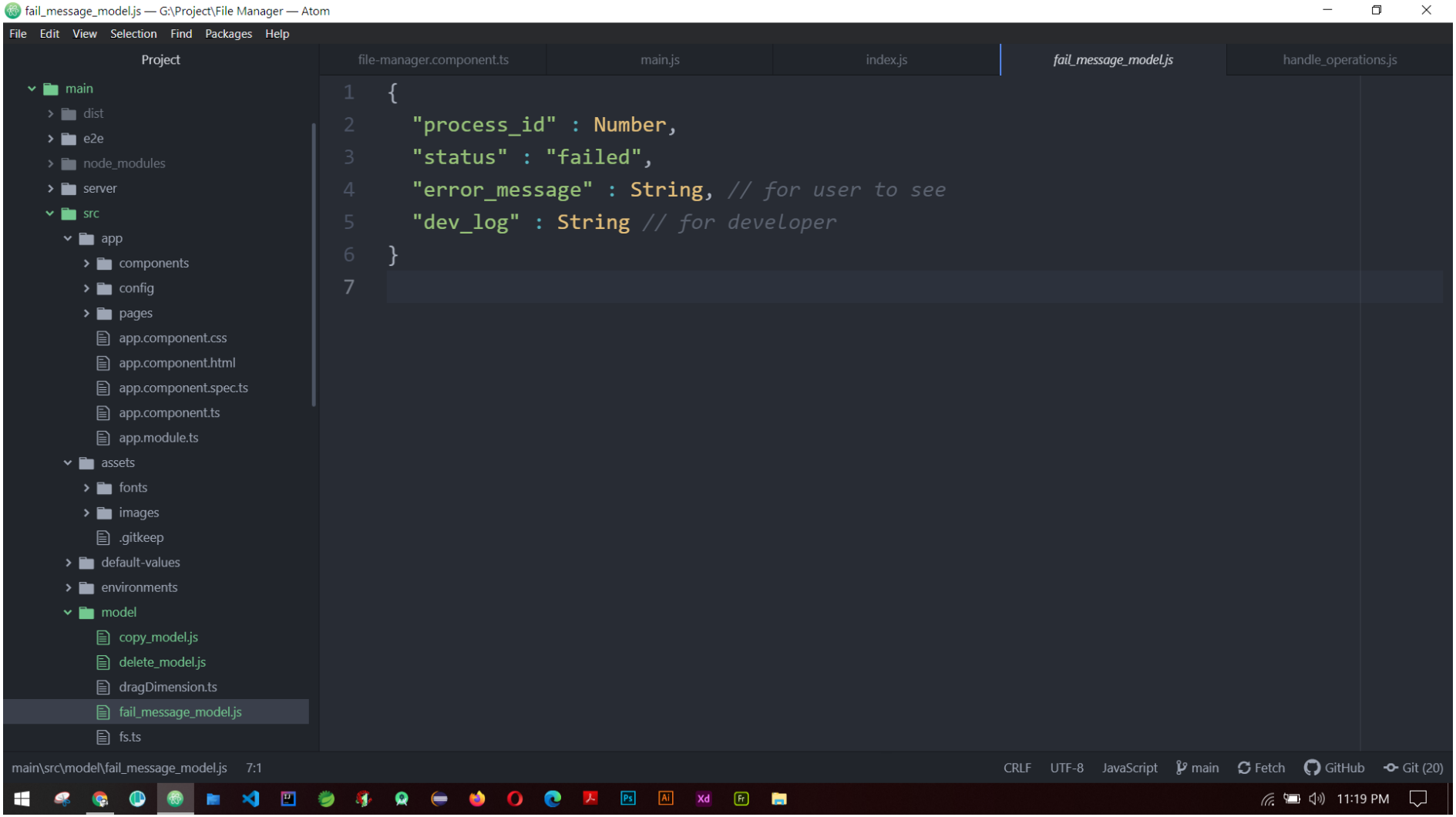
Data Model


copy_model.js — G:\Project\File Manager — Atom
File Edit View Selection Find Packages Help
Project
file-manager.component.ts
main.js
index.js
fail_message_model.js
copy_model.js
handle_operations.js
File Manager
.git
build
docs
main
dist
e2e
node_modules
server
src
app
components
config
pages
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.module.ts
assets
fonts
images
.gitkeep
default-values
environments
model
copy_model.js
{
"process_id" : 738461,
"type": "copy",
"source": {
"server": "admin@192.168.1.1", // "user@host_ip"
"baseFolder": "E://New Folder/Test/" // path to base folder
},
"target": {
"server": "user@192.164.1.3", // "user@host_ip"
"baseFolder": "/home/user/main/" // path to base folder
},
"files": [
"file_one.txt",
"music.mp3",
"main.js",
"video.mp4"
] // filenames of all selected items
}
main\src\model\copy_model.js 19:1
CRLF UTF-8 JavaScript main Fetch GitHub Git (23)
11:25 PM

Data Model


delete_model.js — G:\Project\File Manager — Atom
File Edit View Selection Find Packages Help
Project
main
dist
e2e
node_modules
server
src
app
components
config
pages
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.module.ts
assets
fonts
images
.gitkeep
default-values
environments
model
copy_model.js
delete_model.js
dragDimension.ts
fail_message_model.js
fs.ts
file-manager.component.ts
main.js
index.js
delete_model.js
handle_operations.js
{
"process_id" : Number, // uniquely generated
"type": "delete",
"source": {
"server": String, // "user@host_ip"
"baseFolder": String // path to base folder
},
"files": String[] // filenames of all selected items
}
main\src\model\delete_model.js 10:1
CRLF UTF-8 JavaScript main Fetch GitHub Git (20)
11:19 PM

Data Model


fail_message_model.js — G:\Project\File Manager — Atom
File Edit View Selection Find Packages Help
Project
main
dist
e2e
node_modules
server
src
app
components
config
pages
app.component.css
app.component.html
app.component.spec.ts
app.component.ts
app.module.ts
assets
fonts
images
.gitkeep
default-values
environments
model
copy_model.js
delete_model.js
dragDimension.ts
fail_message_model.js
fs.ts
file-manager.component.ts
main.js
index.js
fail_message_model.js
handle_operations.js
{
"process_id" : Number,
"status" : "failed",
"error_message" : String, // for user to see
"dev_log" : String // for developer
}
main\src\model\fail_message_model.js 7:1
CRLF UTF-8 JavaScript main Fetch GitHub Git (20)
11:19 PM

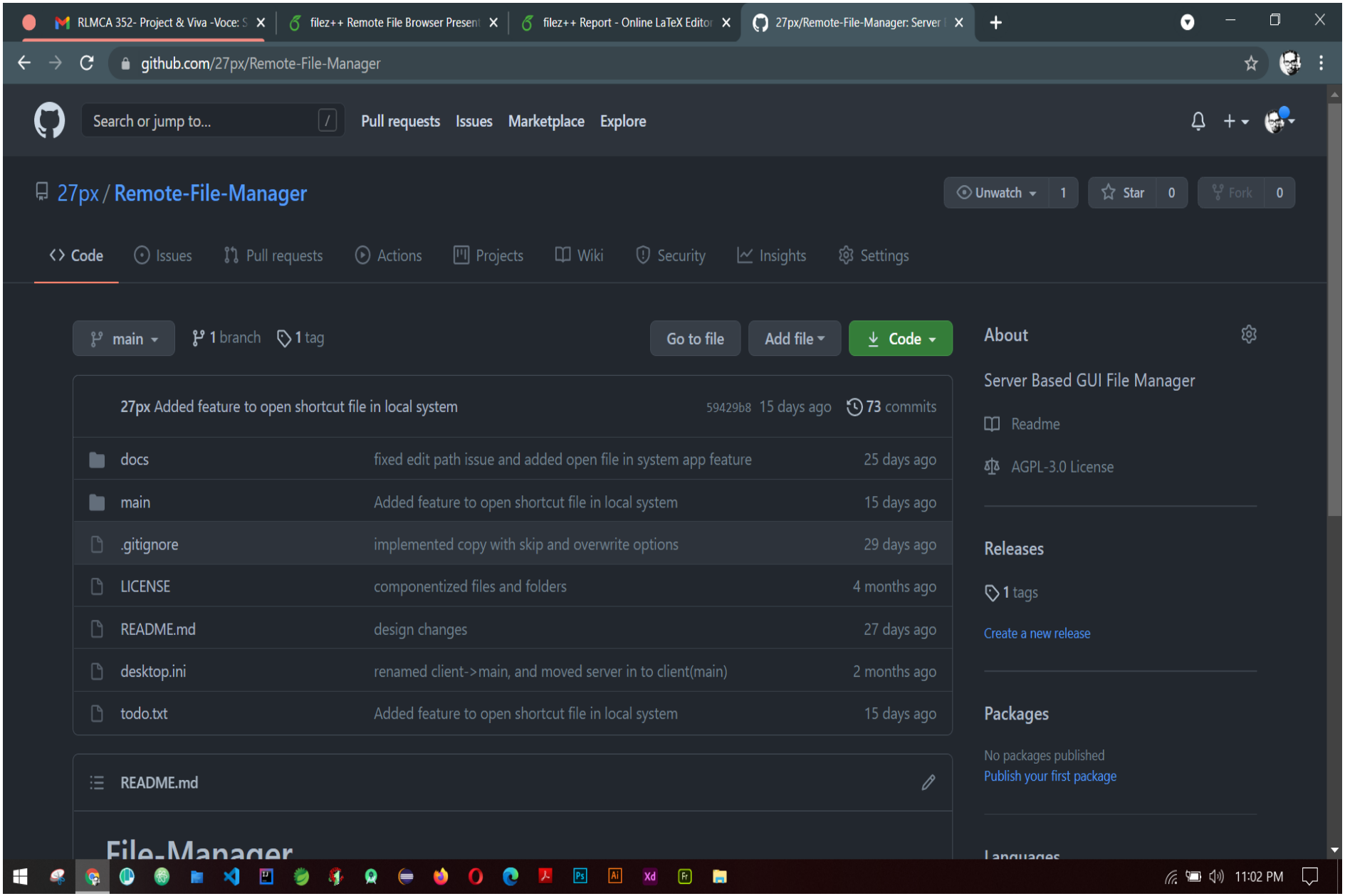
Hardware Requirements

- Storage : 350 to 400 MB (or greater) free space
- RAM : 1 GB (or more), 2GB Recommended
- Mouse : Required
- Keyboard : Not mandatory if having Touch Screen, or on screen keyboard

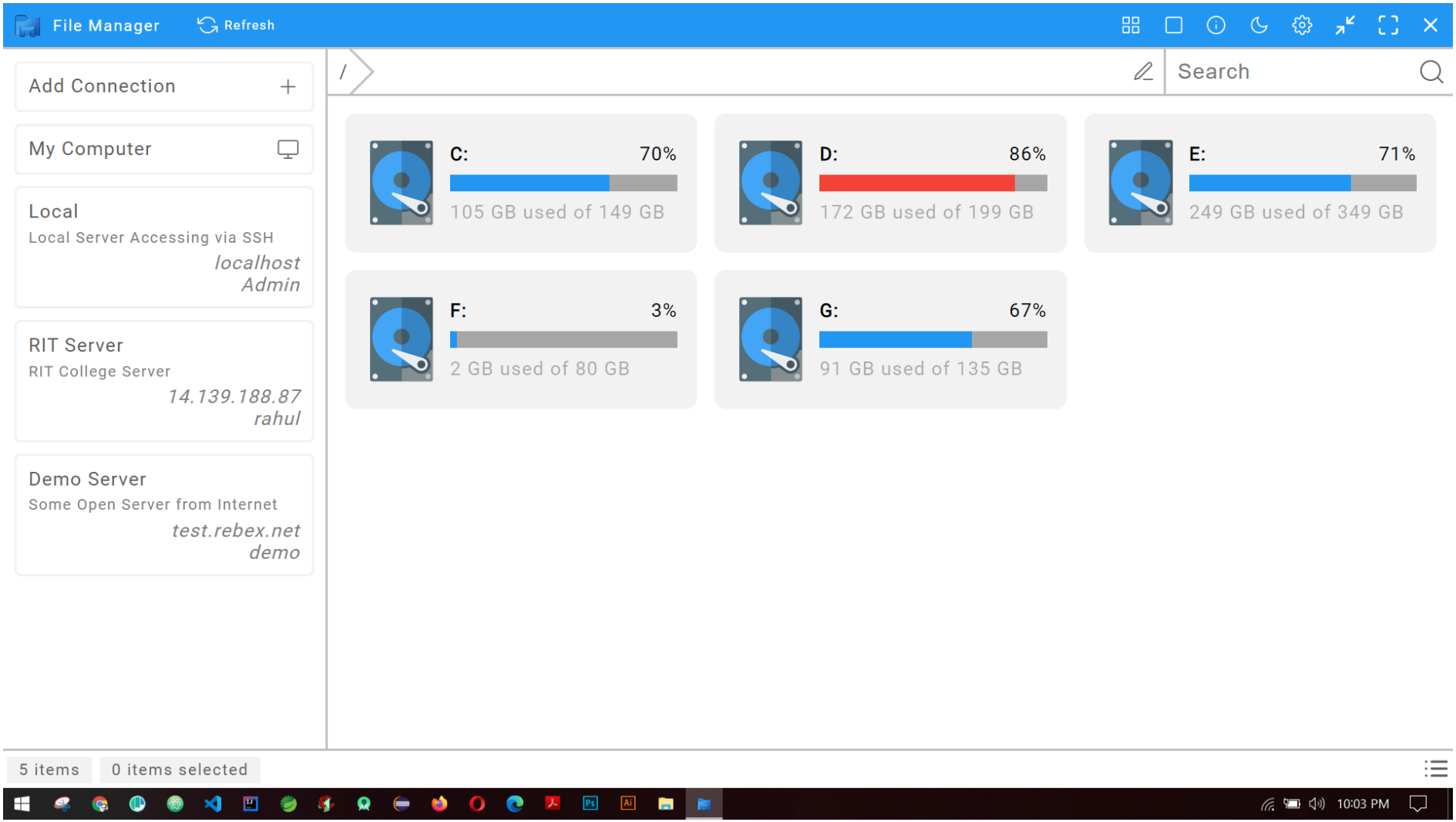
Sprint Backlog

Sl No.	Sprint	To Do	Status
1	Sprint 1	Analyze Features of Existing System	Completed
2		Develop UI Wireframe and Prototype	Completed
3		UI basic backbone development	Completed
4		Create Data Models	Completed
5	Sprint 2	SSH Connection establishing	Completed
6		SSH Connection testing	Completed
7		local file System accessing	Completed
8	Sprint 3	create files feature	Completed
9		create folders feature	Completed
10		delete files and folders feature	Completed
11		browse directoy contents feature	Completed
12	Sprint 4	GUI file management feature	Completed
13		Serach feature	Completed
14		Open files feature	Completed
15	Sprint 5	Rename feature	Completed
16		Packaging for Multiple Operating Systems	Completed
17	Sprint 6	Testing in multiple Operating Systems	Completed
18		File Copy-Paste Feature	Completed
19		File Cut-Paste Feature	Completed
20		Got To directory path feature	Completed

Version Control (Git)



Screenshot


File Manager
Refresh
Add Connection
My Computer
Local
Local Server Accessing via SSH
localhost
Admin
RIT Server
RIT College Server
14.139.188.87
rahul
Demo Server
Some Open Server from Internet
test.rebex.net
demo
/
Search
C:
70%
105 GB used of 149 GB
D:
86%
172 GB used of 199 GB
E:
71%
249 GB used of 349 GB
F:
3%
2 GB used of 80 GB
G:
67%
91 GB used of 135 GB
5 items
0 items selected
10:03 PM

Conclusion

- Developed a user friendly software inspired by FileZilla, without some of its limitations and adding some more Features.
- The Software is a Desktop application that can run on any platform and architecture.
- packaged app can be distributed and can be run without any additional dependencies or programming languages installed and is portable.
- Future modifications and scope could be adding support for System default recycle bin, including Online Storages like Google Drive, Dropbox etc.


Thank You