server/index.js

```
const express = require("express");
const fs = require("fs");
const glob = require("glob");
const multer = require("multer");

const app = express();
const port = 3001;

// Configure Multer for file uploads
const upload = multer({ dest: "uploads/" });

// Set up static file serving for downloads
app.use("/downloads", express.static("downloads"));

// Delete old files on server start
const clearFiles = () => {
  const filelist = glob.sync("uploads/*");
  filelist.forEach((file) => {
    fs.unlinkSync(file);
  });
};
clearFiles();

// Route for the home page
app.get("/", (req, res) => {
  res.render("home"); // Replace with the actual home.html content
});

// Route for compressing files
app.post("/compress", upload.single("file"), (req, res) => {
  const up_file = req.file;

  if (up_file) {
    // Add the logic for compressing the file and saving it to "downloads" folder
    // Replace the "./c" command with your compression logic

    // Example: Compressing the file
    const filename = up_file.originalname;
    const ftype = "-compressed.bin";
    fs.renameSync(up_file.path, `uploads/${filename}`);
    // Add compression logic here
    fs.renameSync(`uploads/${filename}`, `downloads/${filename}${ftype}`);

    res.render("compress", { check: 1 }); // Replace with the actual compress.html content
  } else {
    res.render("compress", { check: -1 }); // Replace with the actual compress.html content
  }
});

// Route for decompressing files
app.post("/decompress", upload.single("file"), (req, res) => {
  const up_file = req.file;

  if (up_file) {
    // Add the logic for decompressing the file and saving it to "downloads" folder
    // Replace the "./d" command with your decompression logic

    // Example: Decompressing the file
    const filename = up_file.originalname;
    fs.renameSync(up_file.path, `uploads/${filename}`);
    // Add decompression logic here
    fs.renameSync(`uploads/${filename}`, `downloads/${filename}`);

    res.render("decompress", { check: 1 }); // Replace with the actual decompress.html content
  } else {
    res.render("decompress", { check: -1 }); // Replace with the actual decompress.html content
  }
});

// Route for downloading files
app.get("/download", (req, res) => {
  // Add logic to handle file download from "downloads" folder
  // For example:
  const filename = "example"; // Replace with the actual file name to download
  const ftype = "-compressed.bin"; // Replace with the actual file type

  const filePath = `downloads/${filename}${ftype}`;
  res.download(filePath);
});

// Start the server
app.listen(port, () => {
  console.log(`Server has started on port ${port}`);
});
```