# Create the following application with .NET 7 API and JavaScript framework frontend

Use your preferred JS framework. Angular, React, etc. for the frontend. Implement the requested solution and showcase the following:

a. Usage of Dependency Injection

b. Generics, extension methods

c. Async programming

e. Exception handling

Beneficial things to showcase:

a. Add Swagger support

b. Add Postman support

c. Attributes creation & usage

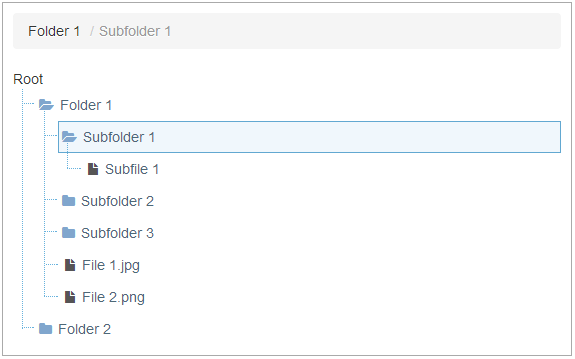
d. Middleware and filter usage

e. Unit tests

## 1. User can upload a zip file with given structure through frontend and have it validated.

Show a file tree of the uploaded zips and validation errors if required.

Example of a file tree, does not have to look like this:



**Zip file must adhere to the below structure. File (CatGame.zip) with correct structure has been included. Where RootFolder is the game name, in the attached example CatGame.**

**RootFolder**

**dlls**

\* Must contain RootFolder.dll file. Can have other files, other filetypes besides .dll are allowed

**images**

\* Must contain at least 1 image of filetype .jpg or .png. No other filetypes allowed

**languages**

\* Must contain RootFolder\_en.xml file. Folder can have only .xml files, but file name must follow the RootFolder\_xx.xml naming convention where xx is 2 letter language code.

If some of this structure is missing or is incorrect, display validation errors below the zip file tree, so the user can fix their zip.

For example: “No images in ‘images’ directory. Please add images.”. “Incorrect file type found in ‘dlls’ folder. Only .dll files are allowed.” etc.

If upload and validation was successful show a successful status.

## 2. User can save the validated zip to the machine

Add a button to save the validated zip to the machine. Store them in “zips” folder in your project directory.

## 3. Add an endpoint to delete the uploaded zips

User can delete all the uploaded zips.