Local FileSystem explorer

Project by:

Abu Sayeed
Yulia Belyakova

Assembler
School of Software Engineering

Contents

Project Overview

Following projects consists in developing a basic version of an interactive File System which allows users to see, access, create, update and delete folders, navigate through them in a dynamic way and upload and preview files with different extensions.

When accessing the principal page, user is able to see all the folders and files, located in the main folder (root/). From this starting point user can see the the content of any folder in recursive way, go backwards in the root or redirect to any part of the route through the breadcrumbs seccion.

Another part of navigation is a menu on the left side, where user can see the structure of the folders (excluding files) and by clicking any of it show the content of the following folder.

Tools and technologies used in the project:

- HTML, CSS and JavaScript
- PHP using Apache Server
- Frameworks and Libraries:

    ❏ Bootstrap

    ❏ jQuery

GIT as version control tool

Project Structure

Local-FilseSystem-explorer - The root of the project

- main page - The folder that contains the php of basic elements in html to build index.php
    - *index.php*
- php - This folder contains all the php of the page that make orders in the server side. Each page is responsible for a specific functionality:
    - ***script.php** - uploading current file structure with loadFolders function, receives ajax POST request with the current root information and returns the array of objects for each folder or file in a requested root*
    - ***new.php** - receives and handles the ajax requests to create, update and delete folders*
    - ***search.php** - receives and handles the request for a search (returns folders and files matching the search input)*
    - ***upload.php** - responsible for uploading media files*
- src - This folder contains the personal _js- and *css* of the page.
    - ***style.css** - basic styling, not covered by Bootstrap*
    - ***script.js** - Javascript code using jQuery to display the information received from server in a requested way*
- *README.md* file is represent documentation.
- Case Diagram - key functionality
- Wireframes - original planned layout of the page
- **root/** folder - initial root directory of the File System

Requirements

- You **will not be able** to use **global variables** in **PHP**.
- You must use **GIT**
- You must use the **PHP v7**
- Create a **clear** and **orderly directory structure**
- Both the **code** and the **comments** must be written in **English**
- Use the **camelCase** code style to define variables and functions
- In the case of using **HTML**, **never** use **inline styles**
- In the case of using different programming languages **always define the implementation in separate terms**
- Remember that it is important to **divide the tasks into several sub-tasks** so that in this way you can associate each particular step of the construction with a **specific commit**
- You should try as much as possible that the **commits** and the **planned tasks** are the same
- **Delete files** that are not used or are not necessary to evaluate the project

Incidents and Lessons Learnt

Incidents

- Wrong request structure;
- Not being able to pass necessary information from JS to PHP
- Using PHP functions with insufficient functionality (glob())
- Not being able to create, update and delete folders inside folders (actual path not dynamic)
- Not being able to delete folder if it is not empty (recursive delete)
- Adding event listeners to the folder structure in the left side navigation menu to open the selected folder

Lessons Learnt

- Creating file system with php
- Sending data with jQuery ajax and receiving through php
- Creating file directory with php
- Understanding php recursive functions
- Creating files, removing files as well as updating files name with php
- Creating search option with php and recursive function