Online Portfolio

Guillaume Pelletier-Auger

September 29th 2016

Abstract

This online Art Portfolio is a static site generated with Node.js.

1 Static Site Generation with Node.js

I'm making an online Portfolio for my visual artworks and I plan to update it fairly frequently. The update process should be as simple as possible. The goal is to generate the content (the html files) using a JSON file or a JavaScript object literal. I will be using the *fs* module of Node.js to create or overwrite the html files.

This is what I have so far :

```javascript
var fs = require("fs");

var header = `<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Guillaume Pelletier-Auger</title>
</head>
`;

var body = `
<body><h1>I'm writing my own HTML files with Node.js. It's pretty neat.</h1>
It's actually like... awesome! I can now think of a website as a data structure.
</body>
</html>
`;

fs.writeFile('index2.html', header + body, function(err) {
    if (err) {
        return console.error(err);
    } else {
        console.log("Data written successfully!");
    }
});
```

I previously thought I would be using a Static Site Engine like Harp.js, but I think it might be a much better idea to just do everything myself using Node.js. The less dependencies, the better. Also, the Harp.js documentation is pretty bad.

2 A Website as a data structure

I have to think of my portfolio as a data structure.

This is the JSON file I have so far :

```
{
    header: "<head></head><body><h1></h1>",
    items: {
        oscillators: {
            title_fr: "Les Oscillateurs",
            title_en: "Oscillators",
            content_fr: `

            `,
            content_en: `
            `,
        },
        joy_and_confusion: {

        },
        dunes: {
        }
    }
}
```

I must have a function that looks like this :

```
function buildWorks() {
    var content = ``;
    for (var i = 0; i < portfolio.items.length; i++)  {
        var title = portfolio.items[i].title;
        content = content + `<img src="${title}.jpg"></a>`;
    }
}
```