VUE JS

Overview

The goal is to understand the framework for rendering view to build user interfaces and single-page applications.

Table of contents

Phase I - Project Initiation

Project requirements

- Requests for booksellers with npm we can also with cdn
    - Axios library.
    - Checking the new directory called node_modules

Project specifications

- The project must be developed in Vuejs or Javascript, Html,SCSS.
- Create a Git repository
- Don't raise dependence
- The directory structure of the project must be well defined and organized.
- The code must be documented correctly using the English language.
- Your code must use a *camelCase style.*
- If you use HTML do not use inline styles
- The project must not contain unused files.
- The project must be developed using *git*, using explicit and concise deconfirmation messages.
- Delete files that are not necessary to evaluate the project
- The project must contain a *README* file written in *Markdown* that shows a brief description and the steps for runél.

Phase II - Project Planning

Reasoning

This pill will develop a spa showing gallery of images, which will load groups of 5 images as the user scrolls vertically.

Take into account
- Position of the scroll of the user who is visiting the page
- Consume the API, get blocks of images in the request.
- Control the number of requests that are made
- Make a component so as not to repeat the code with the data data
- Putting the methods of a component into practice

Organize the code

In this small project we have focused on testing with the api in VUEJS,so we have taken into account the organization of ourcode. It is very important that you organize properly. Create the following directories:

- api (will be responsible for containing the source code of your app)
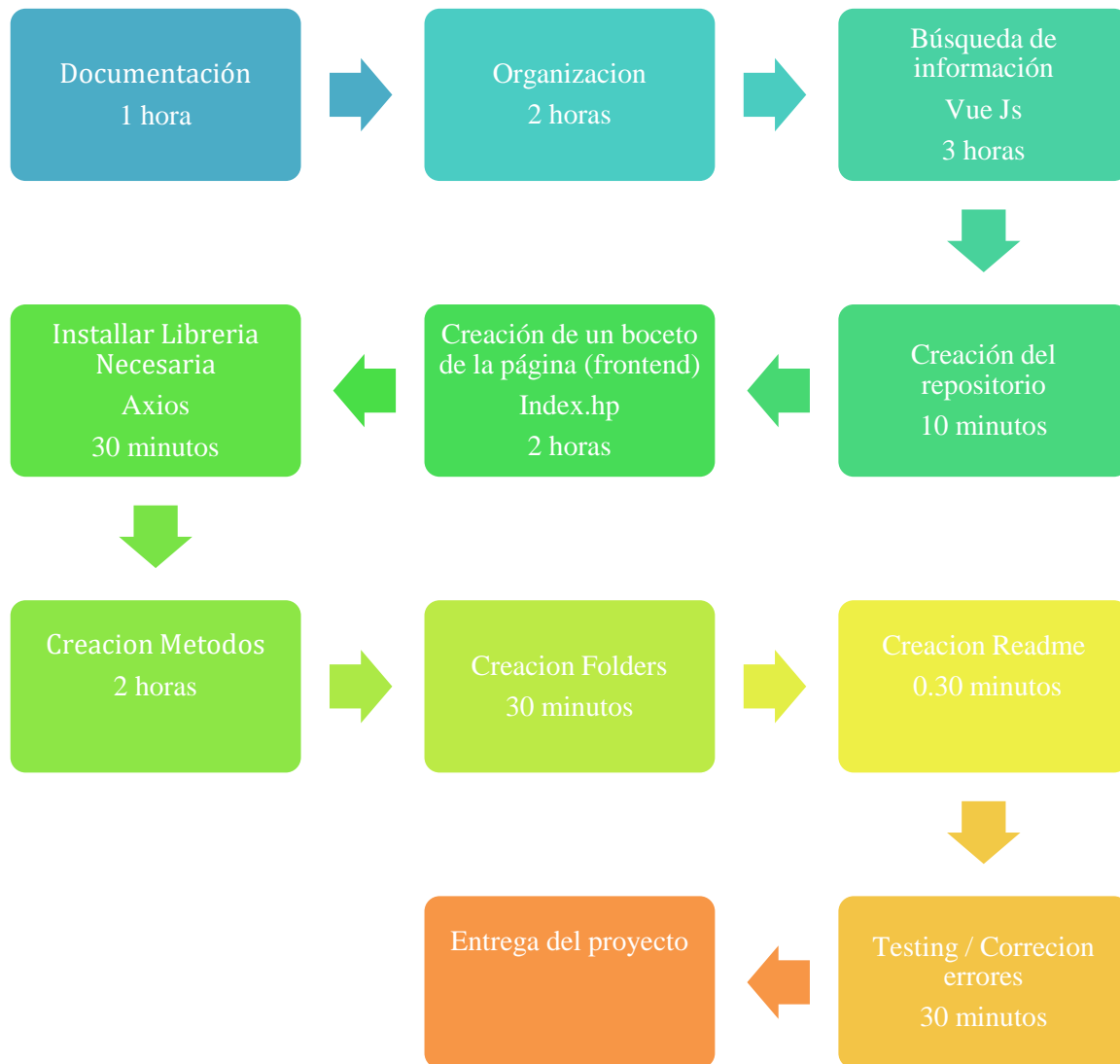- dist (will be responsible for containing the output data)
    - bundle.js/ (this directory is created to maintain the same structureas the original api to facilitate the location of the application outputs of each of the files)
    - index.html to collect the input data from bundle.js
    - assets/sccs is the output of css that we had integrated into scs

Task planning

LISTA DE TAREAS A REALIZAR

Task	Priority	Hours	Difficulty	ID
Documentation	High	1,00	High	1
Organization	High	2,00	High	2
Pre-search for information	Normal	3,00	Normal	3
Repository creation	Low	0,15	Low	4
Indexstructure .html	Low	0.30	Normal	5
InstallAr Library Required as axios	Normal	0.10	Low	6
Metodos Creation	Normal	0.30	Normal	7
CREATION README	Low	0,30	Low	12
Testing / Correction Errors	High	0,30	Normal	13
Project delivery	High	0.20	High	14

Project Calendar Tracking

```
┌─────────────────┐      ┌─────────────────┐      ┌─────────────────┐
│  Documentación  │  →   │   Organizacion  │  →   │   Búsqueda de   │
│     1 hora      │      │    2 horas      │      │   información   │
│                 │      │                 │      │     Vue Js      │
│                 │      │                 │      │     3 horas     │
└─────────────────┘      └─────────────────┘      └─────────────────┘
                                                           ↓
┌─────────────────┐      ┌─────────────────┐      ┌─────────────────┐
│ Installar Libreria │ ← │ Creación de un boceto │ ← │  Creación del  │
│    Necesaria    │      │ de la página (frontend) │   │  repositorio   │
│     Axios       │      │    Index.hp     │      │   10 minutos    │
│   30 minutos    │      │    2 horas      │      │                 │
└─────────────────┘      └─────────────────┘      └─────────────────┘
         ↓
┌─────────────────┐      ┌─────────────────┐      ┌─────────────────┐
│ Creacion Metodos│  →   │ Creacion Folders│  →   │ Creacion Readme │
│    2 horas      │      │   30 minutos    │      │   0.30 minutos  │
└─────────────────┘      └─────────────────┘      └─────────────────┘
                                                           ↓
┌─────────────────┐      ┌─────────────────┐
│ Entrega del proyecto │ ← │ Testing / Correcion │
│                 │      │     errores     │
│                 │      │   30 minutos    │
└─────────────────┘      └─────────────────┘
```

GIT WORKFLOW documentation

- Creating Git Hub **https://github.com/robertfox11/PillsVuejs.git**
- We make commits of the structure of the main page.
- Chance of it occurring 80%
- Project impact 60%
- Possible alternative (mitigation) Ask colleagues for help
- Chance of it occurring 30%
- Project impact 60%
- Possible alternative (mitigation) Ask colleagues for help
- Not easily finding information related to the project
- Chance of it occurring 30%
- Project impact 60%
- Alternative alternative (mitigation)
- Ask colleagues for help

From the realization of the structure, work continued only on the "master" branch, through the Workflow "Gitflow".

But information --> https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow

Tools

Different tools were used in the development of the project. They are as follows:

- *git: A powerful version control system* that helps track changes in the work tree.
- *Visual Studio Code: A code editor* optimized for creating and debugging modern web applications.
- *Vuejs, JavaScript, Html, SCSS*
- *Composer and PHP UNIT* Library
- *Google Chrome Developer Tools:* Used to debug JavaScript code and to test design settings.
- *Google Docs:* Used to write project documentation.
- *W3C Validator*– Used to validate HTML and CSS code.
- *ESLint*– Used to validate JavaScript code.
- ***nano: A basic text editor that uses the*** command-line interface.
- ***curl: A*** command-line tool used to transfer data using various network protocols.
- ***Google Docs:*** Used to write project documentation.

Phase III - Project execution

Incidents

None, luckily!

Lessons

All tasks were completed without having to face any major obstacles.

Phase IV - Project closure

General comments

The pill was successfully completed in the time interval that was predicted in task *planning.*