

Portfolio - Eduardo Schlichting

Vue/Sass/Js/Ajax

Eduardo Bianchini Schlichting

Prof. Trevor Van Rys

Fanshawe College / Fall 2021

Development Roadmap

Pre-development notes

- Remember to always write everything before coding
- Understand the steps clearly (if possible) before coding
- Make comments on the code before coding
- Translate your comments into code ("Code is just a way to make the computer understand your comments")
- Follow a clear, simple logic, and do NOT overcomplicate
- Step by Step. DO NOT overcomplicate what's already complicated

Development map/thinking

- **Portfolio Part / Data Retrieval / Object Population with Vue**

First steps:

- CSS Styling:
 - Let the component already be stylized (hover effects, animations made with CSS (the stuff made with js, let Vue handle, when possible)).
 - Build responsiveness
- HTML:
 - Static elements can be built. Think about what doesn't need to be reused: static descriptions, icons, footer, the about / contact page. These can be already styled (from mobile up).

- **JS/Vue/Data (The complicated stuff :D)**
- First Steps:
- Select projects to be present in the portfolio (FIP Summer, Mini App, MyFavThings, the Portfolio case itself. Design: Kavorka Brand, Sportsnet, Motion Reel? (maybe create a featured section only for the motion reel as promo), Morph Commercial). These projects will change/update next semester, so don't get too attached to them.
- From there, build the database structure
 - tbl_projects for all project related stuff
 - tbl_tags for all the tags the projects may have
 - tbl_proj_tags relating projects and tags for usage
- **Observations/ideas:**
 - Analyze the database structure to match the cards/lightbox content

Cards (thumbnails)

- Title of the Project;
- Links (live/github/behance, if so),
- Tags (Scss, Vue, js, PHP, Branding, Motion, UX/UI, 3D)

2 ways: either showing project name/date/tag only on hover, then the details on the lightbox, or only thumbnail, then showing everything on the lightbox (personally, the 2nd way is my favorite but is good for the design stuff. For web may be the 1st with name/date/tags is a good idea)

1st idea:

Advantages: Show details even before opening, can make someone curious and click on the project to see more.

Disadvantages: Trigger more actions from the viewer, however, the user experience of hovering and having an interaction is a nice touch.

2nd idea:

Advantages: Aesthetically pleasant, clean, can have a cool thumbnail image in high quality.

Disadvantages: a bit abstract, no information provided makes the viewer go: ?? what is is ??

3rd idea:

Name, tag, and date

- provides basic information
- no external/GitHub links (show inside the lightbox)
- not visually polluted
- can provide basic information, matches the minimalism of the website as a whole.

Observations: As it is a web-dev directed portfolio, always have a link for live projects and Github.

Project description (lightbox)

- Title
- Tag(s)
- URL to live / github / behance
- Brief Description
- image 1
- description
- image 2
- image 3
- description
- closing image
- Back to top button?
- Previous/next project buttons
- Provide a close button (can be fixed, to make things easier to the user)

- (It will probably have more images, so think in some way to structure the database to support it: multiple columns?)

Inspiration for project gallery:

<https://guivissotto.com>

<https://berriel.me/>

- Basic Dev steps for making it happen:
 - Create a component for the thumbnail (Vue)
 - Create a component for the lightbox (Vue)
 - Trigger lightbox when the thumbnail is clicked (js, Vue (on-click, v-for) (remember that js event handlers are handled by Vue now))

2nd stage:

- Retrieve the data to populate the thumbnail
- That can impact the database structure:
 - Create a table for the project details
 - Create a table for the thumbnails
 - Create a third, relational table linking the thumbnails to the project details

OR

- Only one table, and use the project title/date/tag to populate the thumbnail, and the same data to populate the project description

3rd stage:

Make it happen! Test, use console.log, test first if the database is being fetched correctly. Then the lightbox trigger, then populate.

These are the stages for the portfolio gallery functionality.

The other pages are more static.

Contact form is going to happen with PHP in Pan's class.

- Maybe include the form at the bottom of every page and also have a dedicated page?

Make things simple!