C4 Software Architecture Document

Portfolio

23 December 2023

OVERVIEW

Creating a portfolio website with a C4 model structure using React, Tailwind CSS, JavaScript, JSON files, and Three.js. The C4 model is about visualizing the architecture of a software system, and I am adapting that to structure my website in a better way.

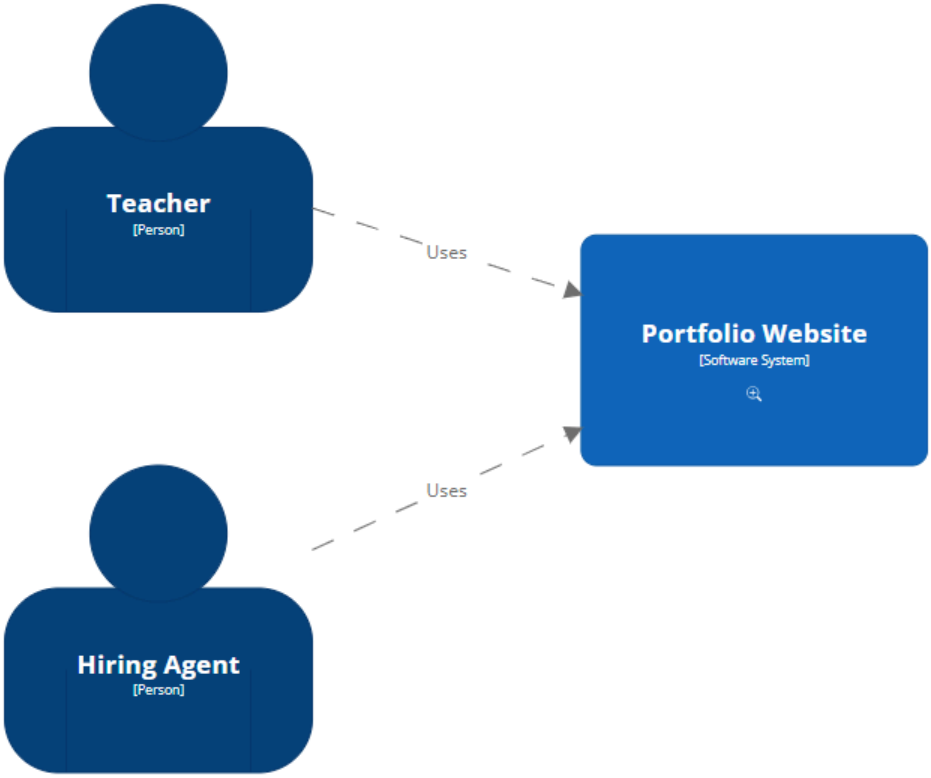
The C4 model contains of the Context, Container, Component, and Code. It is a developer-friendly, easily assimilated method of software architecture diagramming (Brown, n.d.). Effective software architecture diagrams help with architecture reviews and assessments, risk identification, threat modelling, and communication both inside and outside of software development and product teams. I decided to use it because I believe it will enhance the developing of the website portfolio, as I believe it lacks the correct structure.

The diagrams were created using the Structurizr tool as it provides a seamless way to create, visualize, and document software architectures according to Cohen (2021). I also created another diagram in the visual paradigm website because I did not like the final outcome of the coded one.

Context (Level 1)

To have a broader view and see the bigger image. Drawing and documenting a software system can be initiated with this stage.

1. **System:** My portfolio website.
2. **Actor:** Users (Teachers and hiring agents)



Containers (Level 2)

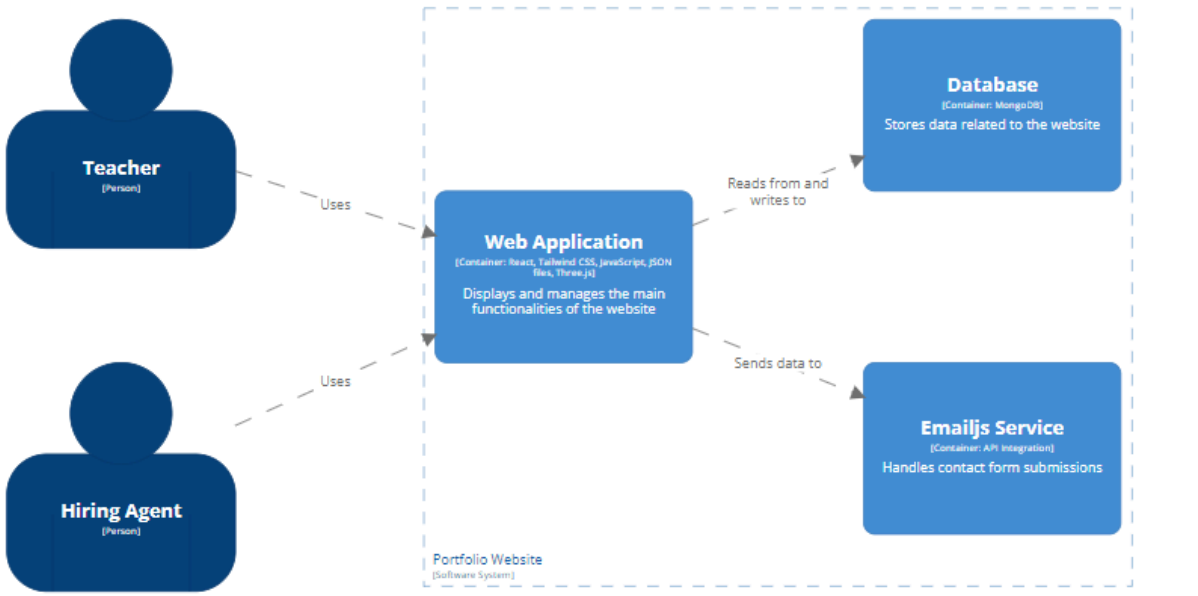
The software architecture and the division of tasks within it are shown in the Container diagram. The key technological decisions and the ways in which the containers talk to one another are also explained.

1. **Container 1: Hero section (Main Page)**
    a. **Functionality:** Displays overview of the website.
    b. **Technology:** React, Tailwind CSS, JavaScript, Three.js.
2. **Container 2: Info about Me (Main Page)**
    a. **Functionality:** Displays personal information about me.

   b. **Technology:** React, Tailwind CSS, JavaScript.
3. **Container 3: Skills (Main Page)**
   a. **Functionality:** Show technologies and tools I use in the IT field.
   b. **Technology:** React, Tailwind CSS, JavaScript, Three.js.
4. **Container 4: Contact Information**
   a. **Functionality:** Gives users a way to get in touch with me for feedback or any other messages.
   b. **Technology:**
      i. **Frontend:** React, Tailwind CSS, JavaScript, Three.js.
      ii. **Backend:** Involve in API integration (emailjs service) for handling contact form submissions.
5. **Container 5: Showcasing My Work (Sub-pages Overview Page)**
   a. **Functionality:** Showcases portfolio work during the semester.
   b. **Technology:** React, Tailwind CSS, JavaScript.
   c. **Work Details Component (linked to Sub-containers)**

      Sub-containers:

      1. Teachers Workshops
      2. Documents & Work for the Portfolio
      3. International Week with the Canadians
      4. Weyweyweb (Spain Trip)
      5. Group Project



Components (Level 3)

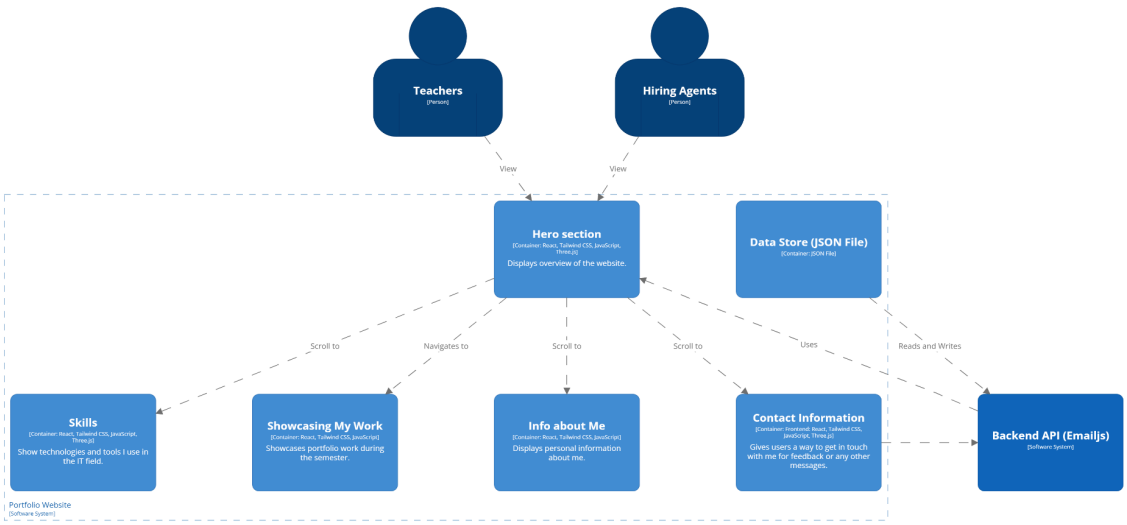
The Component Diagram illustrates the several components that make up a container, together with information on each one's functions, tasks, and technology and implementation specifics.

Components within the Main Page

1. **Header Component (Hero section):**
    a. **Functionality:** Manages the website's top navigation and user interaction elements.
    b. **Technology:** React, Tailwind CSS, JavaScript, Three.js.
    c. **Sub-components:**
        i. Navigation: Enables seamless movement between sections/pages.
            1. Reading Guide Link (Directs to a PDF file)
            2. Sub-pages Overview Link (Navigates to sub-pages section)
            3. Scroll Down (Anchors to 'About Me' section)
        ii. Scrolling element: Facilitates smooth scrolling to the 'Info about Me' section.
2. **Bio Component (Info about Me):**
    a. **Functionality:** Show information about me.
    b. **Technology:** React, Tailwind CSS, JavaScript.
    c. **Sub-components:**
        i. A brief overview about me, something like my expertise and passion.
        ii. Shows languages and tools I use in the AI tools
3. **Skills Component:**
    a. **Functionality:** Showcases the languages and tools I use in the IT field.
    b. **Technology:** React, Tailwind CSS, JavaScript.
    c. **Sub-components:**
        i. Displays languages and tools.
        ii. Presents these in an appealing 3d elements.
4. **Contact Component (Contact Information):**
    a. **Functionality:** Manages the website's top navigation and user interaction elements.\
    b. **Technology:** React, Tailwind CSS, JavaScript, Three.js.
    c. **Sub-components:**
        i. Contact Form: A form that allows users to input their details (name, email, message).

ii. Submit Button: Triggers an action to send the input information to my email.
iii. Validation: Validation for ensuring correct data entry.
iv. Interactive Elements: offers user-friendly features to enhance the user experience.
v. 3D Integration: Integrating a 3D element next to the contact form for a visually appealing experience.

5. **Work Details Component:**
    a. **Functionality:** Presents detailed information about various portfolio sections.
    b. **Technology:** React, Tailwind CSS, JavaScript.
    c. **Sub-components:**
        i. **Teachers Workshops:**
            1. **Workshops List Component:** Lists attended workshops.
            2. **Workshop Details Component:** Provides insights from each workshop.
        ii. **Documents & Work for the Portfolio:**
            1. **Document Library Component:** Organizes and displays portfolio documents based on LAs.
            2. **Portfolio Work Showcase Component:** Showcases work samples.
        iii. **International Week with the Canadians (Dutch Design Week):**
            1. **Event Overview Component:** Provides an overview of the event.
            2. **Experience Details Component:** Details of some liked arts and showcase a video.
        iv. **Weyweyweb (Spain Trip):**
            1. **Spain Trip Gallery Component:** Provides an overview of the events and attended workshops and talks.
            2. **Weyweyweb Collaboration Component:** Details of some attended workshops.
        v. **Group Project:**
            1. **Project Overview Component:** Provides an overview of the group project.
            2. **Individual Contribution Component:** Details individual contributions to the project.



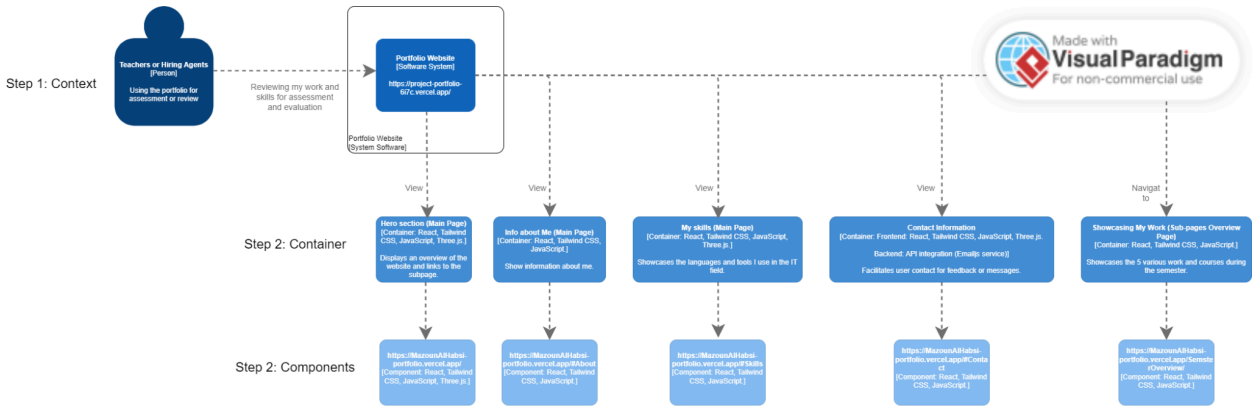
[Container] Portfolio Website

Saturday, December 23, 2023 at 7:03 PM Central European Standard Time

Code (Level 4)

This level of information is not required and can often be provided on-demand by tools like IDEs. An IDE or UML modelling tool, for example, would be ideal for automatically generating this diagram.

1. **Frontend Code:**
    a. **Technologies Used:**
        i. **React:** Makes it possible to design user interfaces using components for smooth interaction.
        ii. **Tailwind CSS:** Makes it easier to apply responsive design and style quickly.
        iii. **JavaScript:** Enables the website to be interactive, dynamic, and dynamic content fetching.
        iv. **Three.js:** Enhances the user experience by integrating 3D elements.
    b. **Functionality:**
        i. React defines the website structure and layout.
        ii. Tailwind CSS ensure a consistent and visually appealing design.
        iii. JavaScript enables interactivity, engagement, and efficient data fetching.
        iv. Three.js shows 3D visuals to enrich the experience.
2. **Data Management:**
    a. **Implementation:**
        i. Logic is implemented to retrieve and handle data from JSON files.
    b. **Purpose:**
        i. Makes it possible for dynamic material to be shown in different website areas.
        ii. Makes it easier to handle and show data effectively for a customized user experience.
3. **Backend Integration for Contact Form:**
    a. **API Integration:** Sends contact form submissions through a smooth interface with backend services using Emailjs.
    b. **Functionality:** Enables efficient contact request handling by integrating Emailjs API to transmit form data to specified email addresses.



Conclusion

This methodical strategy that makes use of the C4 model directs and provides the structure for creating the Portfolio Website. Using this model gives the development process a systematic approach that provides a well-structured architectural perspective at each level:

Level 1 (Context): Defines the parameters of the system based on the Teachers' and hiring agents' access to the Portfolio Website.

Level 2 (Containers): Makes use of React, Tailwind CSS, JavaScript, and Three.js to organize the website into Hero section, Info about Me, contact form and the subpage about the Work Details.

Level 3 (Components): Contains the all the containers components that control the content, navigation, and many subcomponents and more. React, JavaScript, and an Emailjs API are integrated by the Contact Component to enable user contact.

Level 4 (Code): This level of frontend programming is optional and makes use of Three.js, JavaScript, Tailwind CSS, and React. JSON files provide dynamic content and personalized user experiences.

This C4-driven approach guarantees a clear, organised architectural viewpoint, influencing the way the Portfolio Website is developed.

REFERENCE

- Brown, S. (n.d.). The C4 model for visualizing software architecture. Retrieved from https://c4model.com/
- Cohen, S. (2021, October 26). Understanding the C4 Model for Software Architecture Documentation. Medium. https://sheldonrcohen.medium.com/understanding-the-c4-model-for-software-architecture-documentation-e59c4edd0d56
- The created diagram in the visual paradigm website: https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&type=C4Model&gallery=/repository/c11adb2f-dc05-47e8-88cc-369892d99372.xml&name=C4%20Model%20System%20Context%20Diagram%20for%20Internet%20Banking%20System