**Building a Side Modal Portal with ChatGPT**

Student Details

Institution Details

Professor Details

Course Details

Due Date

**Abstract**

This research paper examines the development of a side modal portal through leveraging the conversational abilities of ChatGPT. Side modal portals provide a dynamic means of presenting website content through modular windows accessed from the sidebar. By integrating ChatGPT, interactivity within the portal can be augmented. ChatGPT utilizes natural language processing to facilitate human-like dialogue, enabling the delivery of personalized responses and recommendations. This research aims to investigate how ChatGPT may be incorporated to enhance the functionality and user experience of a side modal portal, offering users a more engaging interface through which they can obtain customized assistance.

**Key words:** Modal, Portal, ChatGPT, JavaScript

**Introduction**

Web development increasingly focuses on immersive user experiences. Side modal portals combine modals and portals to access varied content seamlessly. This research integrates ChatGPT to augment interactivity and personalization within a side modal portal. For a school project site redesign, I wanted to develop a side modal portal to feature program details and an events calendar. ChatGPT's conversational AI presented an opportunity to source coding guidance.

**Querying ChatGPT**

The objective was constructing a side modal accessible through the navigation bar, containing specific information sections within separate modal windows. I asked the chatbot how to build a portal utilizing an individual button, with HTML and CSS for structure and JavaScript for functionality.

**ChatGPT's Response**

Sample code was provided demonstrating nested modals containing unique IDs triggered by separate links, along with CSS styles and JavaScript functions.

**HTML:**

<div id="sideDrawerModal" class="modal">

        <div class="modal-content">

            <span class="close" id="closeModalBtn">&times;</span>

            <h2>Documents</h2>

            <ul>

                <li><a href="/Resources/Documents/866714142 Language and Power.docx">Research</a></li>

                <li><a href="/Resources/Documents/dig.pptx">Presentation</a></li>

            </ul>

        </div>

    </div>

**CSS:**

\* Add your CSS styles here \*/

.modal {

display: none;

position: fixed;

z-index: 1;

top: 0;

left: 0;

width: 250px; /\* Set the width of the side drawer \*/

height: 100%; /\* Set the height to cover the entire viewport \*/

background-color: #ffffff;

overflow-x: hidden;

overflow-y: hidden;

transition: 0.5s; /\* Add a smooth transition effect \*/

box-shadow: 0px 0px 5px 0px rgba(0,0,0,0.5); /\* Add shadow effect \*/

}

.modal-content {

padding: 10px;

width: 100%;

height: 100%;

overflow-y: auto; /\* Add vertical scroll if content overflows \*/

display: flex;

flex-direction: column;

justify-content: center; /\* Align items vertically center \*/

align-items: center; /\* Align items horizontally center \*/

opacity: 1; /\* Hide initially \*/

transition: opacity 0.5s ease; /\* Add opacity transition \*/

}

**Javascript:**

**/**/ Get the modal and the buttons

const modal = document.getElementById('sideDrawerModal');

const openModalBtn = document.getElementById('openModalBtn');

const closeModalBtn = document.getElementById('closeModalBtn');

const mainContent = document.querySelector('.main-content');

openModalBtn.addEventListener('click', () => {

modal.style.display = 'block';

mainContent.style.marginLeft = '250px';

});

closeModalBtn.addEventListener('click', () => {

modal.style.display = 'none';

mainContent.style.marginLeft = '0';

});

window.addEventListener('click', (event) => {

if (event.target == modal) {

modal.style.display = 'none';

mainContent.style.marginLeft = '0';

}

});

**Practical Implementation**

The code example was modified to suit the layout of the developer's website. Each modal window focused on containing either details of assignments for research or a PowerPoint presentation that could be downloaded. The JavaScript events were customized so that the specific modal associated with its triggering button would open. CSS adjustments were made to position and format the modal containers appropriately. By conversing with ChatGPT, rapid sample code addressing the complex modal portal concept was obtained in a timely manner without requiring extensive research. One potential advantage that ChatGPT offers is accelerating the coding process (Wahyu Rahmaniar, 2023). The natural language interface streamlined obtaining the necessary design assistance. The modal windows simply contained two links for downloading research and presentation documents as PDF files. ChatGPT's prompt response provided a helpful starting point that could then be built upon further to fit the requirements of the project website.

**Results and Conclusion**

Iterating upon ChatGPT's sample code allowed me to successfully develop the customized side modal portal. Each modal window focuses on a specific program attribute or calendar month. ChatGPT's conversational responses provided the foundation to construct the portal's underlying structure and interactive elements. Further customization involved styling with CSS and customizing JavaScript events to open particular modals. While full projects require in-depth resources, ChatGPT efficiently addressed my questions and needs for this fundamental coding task. ChatGPT stands out in code generation (Nazir & Wang, 2023). The portal now seamlessly presents relevant details through an intuitive interface. As a learning experience, leveraging ChatGPT's assistance validated the potential of AI tools for expediting web development workflows.

**References**

Nazir, A., & Wang, Z. (2023). A Comprehensive Survey of ChatGPT: Advancements, Applications, Prospects, and Challenges. *Meta-Radiology*, *1*(2), 100022–100022. https://doi.org/10.1016/j.metrad.2023.100022

Wahyu Rahmaniar. (2023). *ChatGPT for Software Development: Opportunities and Challenges*. https://doi.org/10.36227/techrxiv.23993583.v1