Boas Práticas com as classes utilitárias do WPF

Com seu projeto HTML definido e o WPF3 importado corretamente, começar a recortar a sua página web é de longe a etapa mais fácil de todo o processo.

Apesar de não existir uma lógica definida ou uma forma obrigatória de organizar as suas classes dentro do atributo *class=””* na sua tag HTML, existe uma ordem que pessoalmente recomendamos para que você siga e se acostume, para garantir que seu código permaneça legível e organizado, mantendo um único padrão constante entre as páginas.

É importante lembrar que o WPF3 foi desenvolvido tendo em mente um escopo de projeto que é baseado em **componentes** e, por mais que ele seja ótimo para o recorte e desenvolvimento web tradicional, é através de projetos que utilizam a estruturação MVC que o WPF realmente brilha.

Sugerimos sempre você separe suas classes na seguinte ordem:

<div class=”*custom* func() *pos display conf. display tam orient extra*”>

Vamos entrar em mais detalhes sobre como essa organização funciona:

- **Custom:** Aqui vão todas as propriedades e classes criadas por você que não fazem parte do escopo do WPF. A ideia é que seja fácil visualizar o que é feito especificamente por você para incrementar os estilos que devem ser aplicados somente à uma ocasião específica;
- **Funções:** Quaisquer **funções de classe** são atribuídas logo depois. Essas classes especiais costumam configurar cores de todos os tipos (plano de fundo, borda, texto, entre outras) ou até mesmo definem estruturações visuais básicas (como a família ou peso de uma fonte que uma porção de texto terá);
- **Posicionamentos:** Nessa parte é onde inserimos as primeiras classes específicas do WPF. Aqui vamos tratar de posicionamentos ou tratamentos de posição, como todas as classes que envolvem as propriedades *position, box-sizing* entre outros similares;
- **Display:** Agora vamos especificar o tipo de exibição que seu elemento terá: *grid, table, block, inline-block, contents, flex* entre outros;
- **Conf. do Display:** E depois damos os detalhes e configurações disponíveis dessa exibição específica, como é o caso de *align-items, justify-block, vertical-align* entre outros;
- **Tamanho e Espaçamento:** Definimos então o tamanho do nosso elemento com *width* e *height*, além dos espaçamentos disponíveis, como *margin* e *padding*;

- **Orientações:** Por último, caso sua tag HTML necessite, aqui entra todos os tipos de orientações, como orientação de texto em *text-align* ou denominar um espaço específico do elemento para não renderizar mais linhas de texto do que o instruído, como *clamp* etc;
- **Extra:** Se sua tag HTML necessita de outras classes que fazem parte da estrutura do WPF, mas não entram em nenhuma dessas categorias, como *transition, transform, filter,* entre outras, você pode livremente alocá-las em qualquer ordem após ter organizado tudo seguindo este roteiro. As classes extras também abordam todas as classes que você precisa inserir para substituir um valor padrão que você atribuiu anteriormente. Nesse caso, uma classe de qualquer uma das outras categorias usando a instrução "!" (como por exemplo, uma que possuí valor padrão w-50 mas que precisa receber um valor novo de w-100 ou w-25 receberia o valor como !w-100 ou !w-25)

Lembrando que não é obrigatório utilizar essa organização específica. Ela se trata apenas de uma sugestão. Sinta-se encorajado a encontrar a melhor forma que funciona para você e sua equipe.

Revisão #3
Criado 5 novembro 2024 13:16:51 por Lyautey Maluf Neto
Atualizado 11 julho 2025 20:41:55 por Lyautey Maluf Neto