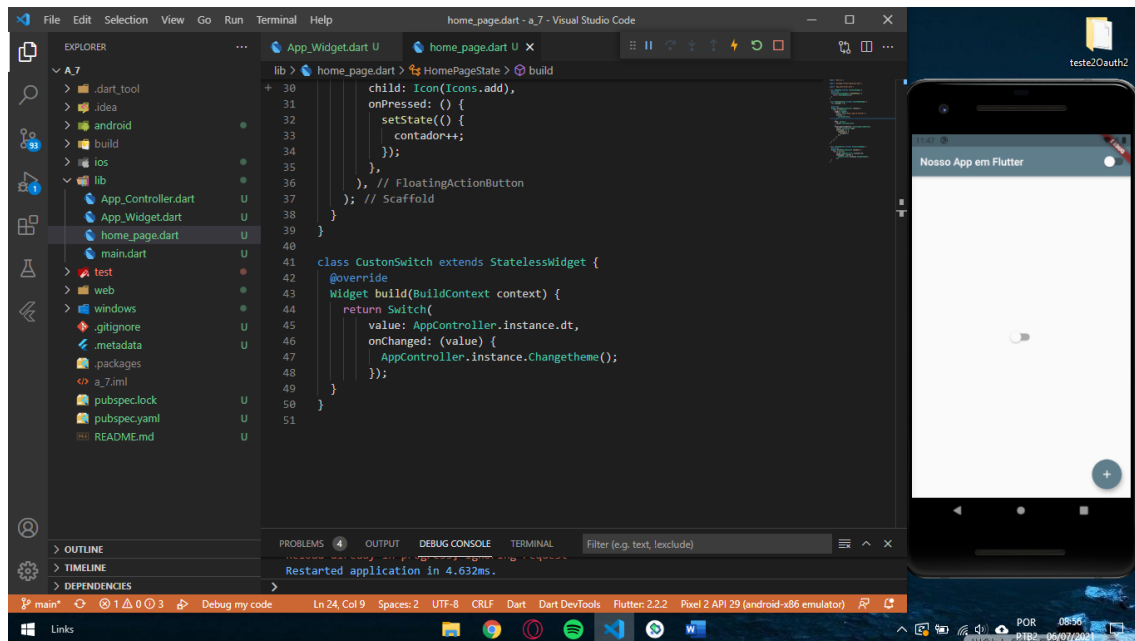
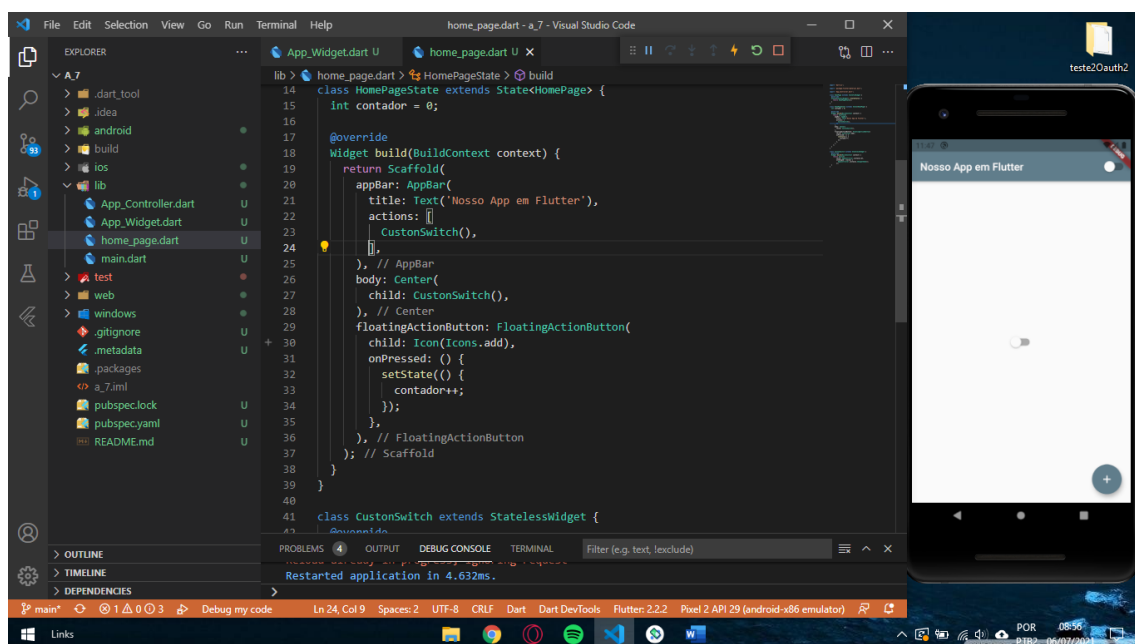


Podemos fazer uma coisa chamada componentização de widgets para poder reaproveitar o mesmo widget em diversas partes do código.



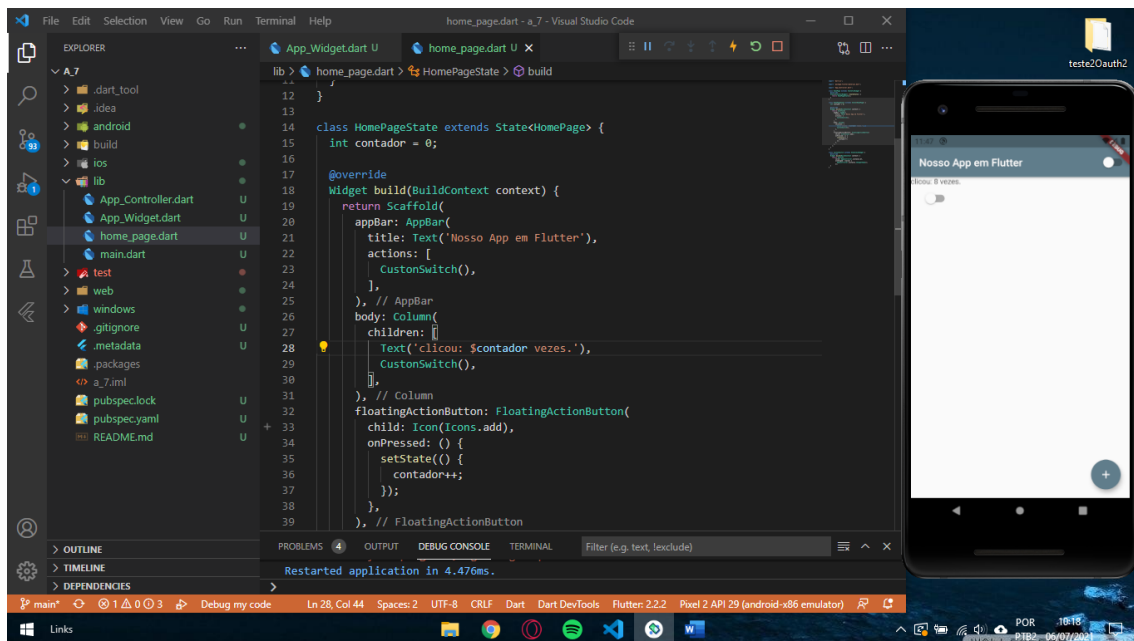
Criamos uma classe do tipo stateless mesmo pois nesse caso ao acionar o botão ele irá chamar outra classe que irá reconstruir a aplicação como um todo e não somente um estado local, criamos o nosso switch dentro dessa classe.



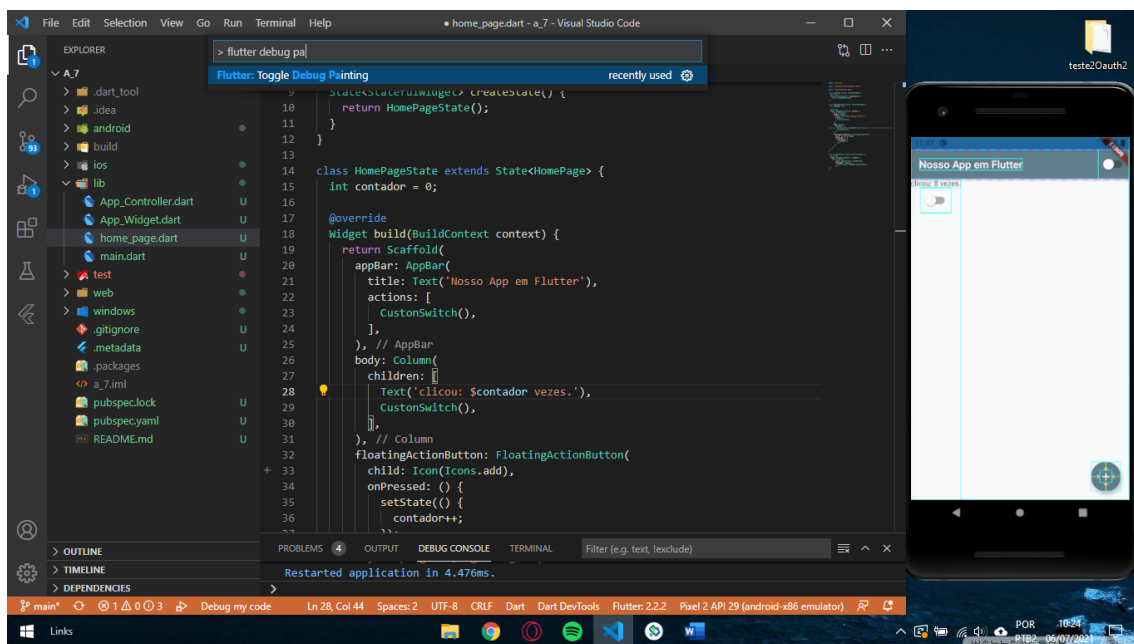
Apartir de agora usamos o `CustomSwitch` e podemos reaproveitar ele como fizemos ali, colocamos ele nos botões de ação que ficam no canto superior direito da aplicação.

Ao executar ambos tem a mesma função e funcionam juntos.

Voltando um pouco a falar de multi renders, vamos aprender também como alinhar widgets na tela.

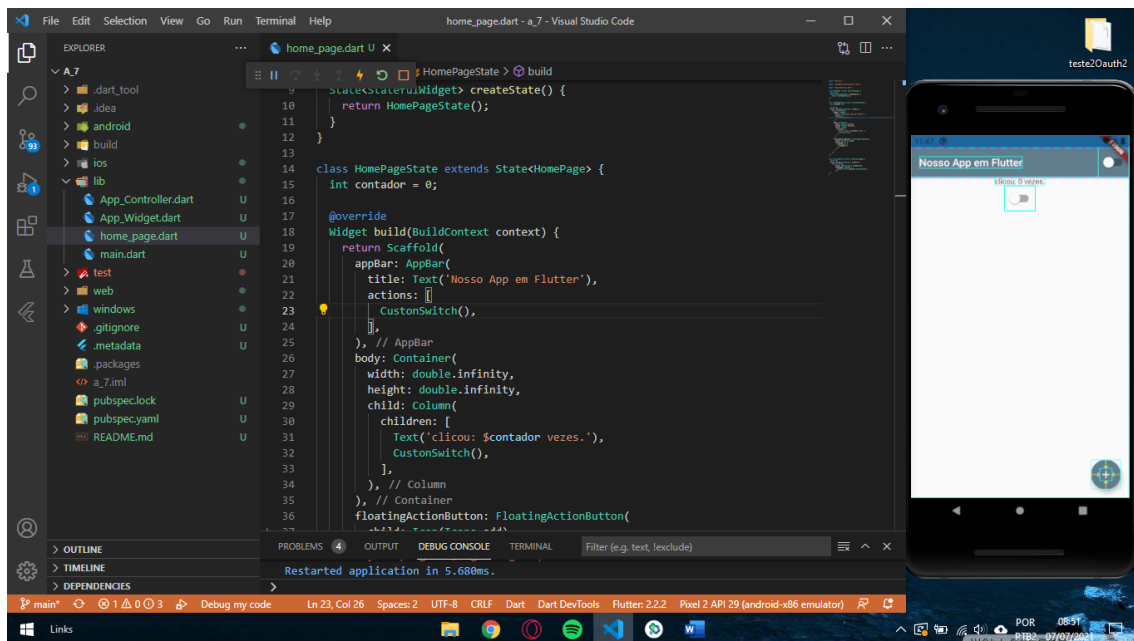


Criamos uma coluna e em colunas usamos os 'children' para colocar os itens na tela que seria como o plural de 'child' que é usado nos widgets padroes, colcoamos o nosso switch e o texto do contador dentro dessa coluna.

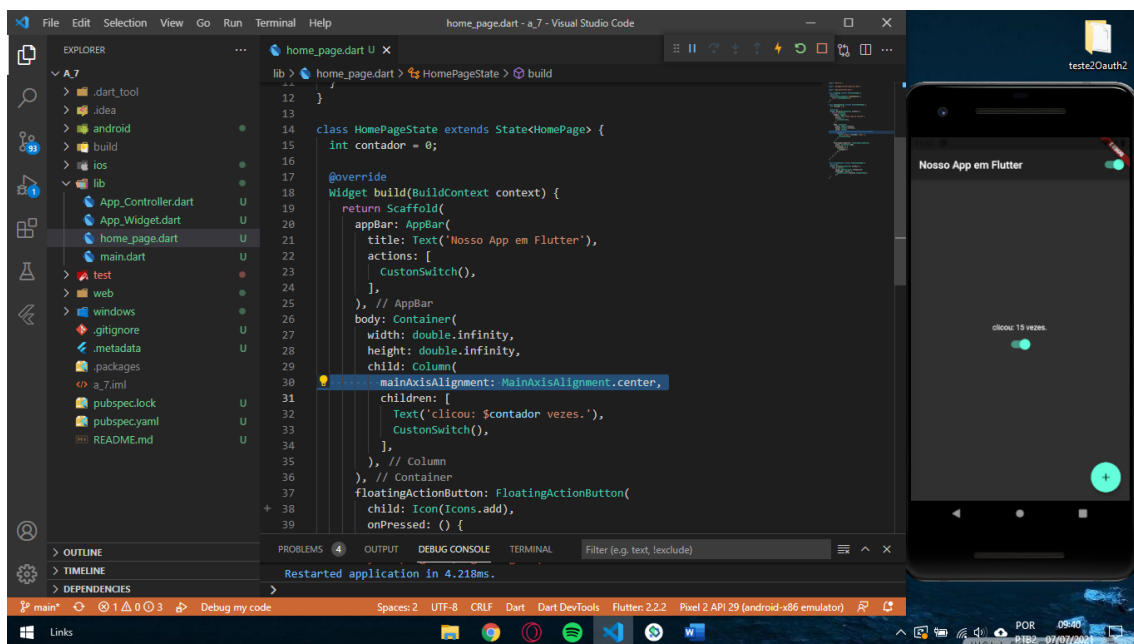


Ao utilizarmos a ferramenta 'flutter: Toggle Debug Painting' ele vai mostrar na tela onde passam as linhas de alinhamento dos widgets na tela, nos permitindo ver o alinhamento deles.

Por padrao as 'column' vem com um tamanho padrao, mas tambem por padrao as 'column' sempre vem acompanhadas de um 'container' que tem as propriedades de altura e largura, assim podemos "alterar" o tamanho da nossa coluna.

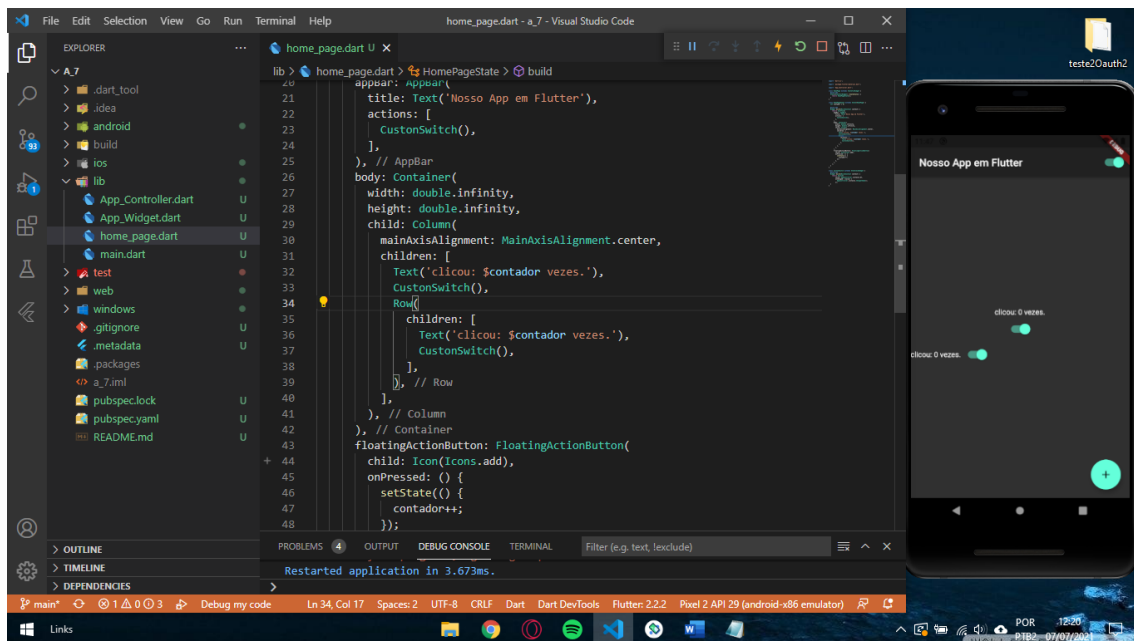


Criamos um widget container como pai da coluna, e atribuímos o seu tamanho tanto em largura 'width' quanto em altura 'height' com a constante '.infinity' que vai ate o limite da tela, podemos notar que assim que feitas as alterações as linhas de limite da coluna passaram a se localizar nas bordas da tela e o nosso conteúdo ficou centralizado no meio da coluna.

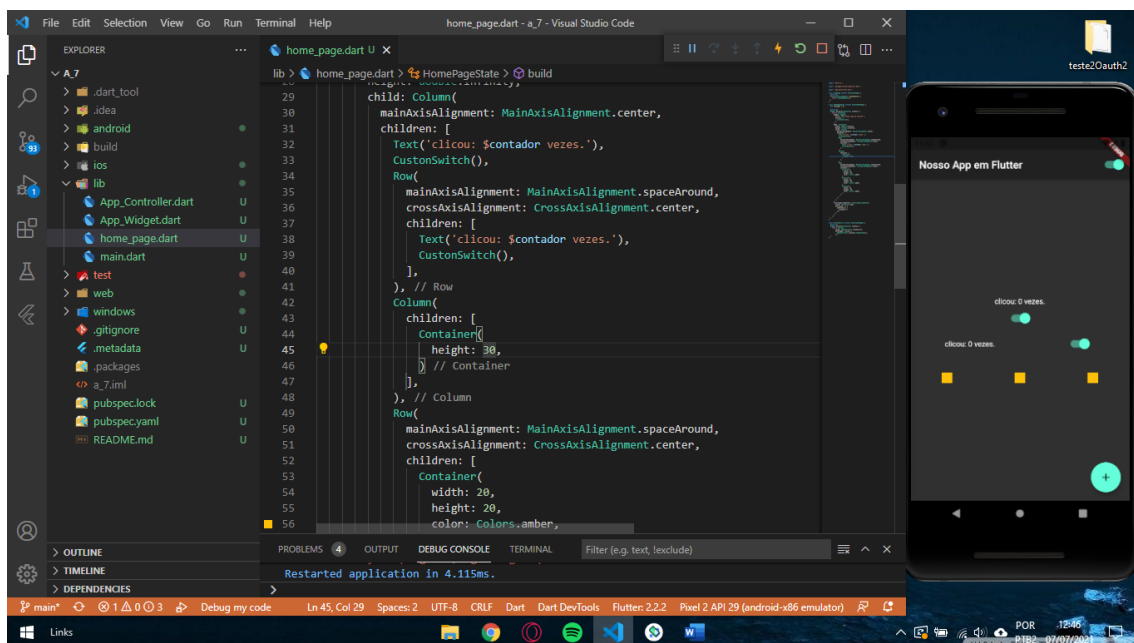


Para alinhar itens de uma maneira fácil, temos o 'mainAxisAlignment' que já tem varias constantes para alinhar automaticamente os itens, esse em especifico alinha o widget num todo, temos os widgets filhos que podemos usar o 'crossAxisAlignment' que alinha os widgets internos por assim dizer.

Também temos as linhas 'row' que funcionam da mesma maneira que as colunas porem na horizontal.



criamos uma 'row' e colocamos o nosso contador e o switch para mudar o tema, note que o alinhamento dos itens mudou, agora estão dispostos um ao lado do outro e não mais um abaixo do outro.



criamos uma linha com 3 quadrados amarelos para vermos como funciona o espaçamento, usando as mesmas propriedades de espaçamento das colunas podemos organizar os itens dentro da linha, porem para regularmos o espaço entre uma linha e outra ou podemos criar um container vazio com o tamanho desejado ou usar uma logica externa para dar esse espaço, ambas estão corretas então a mais fácil seria o container vazio mesmo.

Observando assim, e analisando tudo o que aprendemos ate agora, podemos notar que existem muitas semelhanças entre html,css e flutter nessa parte de layout, temos um title, body, temos um grid para trabalhar o layout como em alguns frameworks web composto de linhas e colunas para criar e organizar os componentes no caso do flutter os widgets.