\*\*ETAPAS NECESSÁRIAS PARA UM SITE REAL, MAS PROVAVELMENTE SEM MUITA UTILIDADE PARA UM PORTFÓLIO\*\*

\* Clicar página de cadastro e de login do administrador

\*Criar área administrativa

\*\*\*\*\*\*\*\*\*\*\*

\*\*COMPONENTES

\*Criar o carrinho de compras

\*Criar página de favoritos

\*Criar página de contato

\*Clicar página de cadastro e de login do usuário

\*Criar página de checkout

\*Criar filtro personalizado

\*\*FEATURES

\*Inserir links na sessão ‘shop by room’ para que o usuário seja direcionado para a página de produtos já com a categoria do ambiente selecionada.

\*Transformar ‘bestDeal’ em slider

\*Criar possibilidade de busca de produtos pela lupa

\*criar checkout

\*Inserir animação nos slides

\*Inserir links e descrições nos slides da página principal para que o usuário seja direcionado para a página do produto

\*Criar função para que a escolha de cores dos produtos funcione e seja congruente com as cores disponíveis. 🡨 🡨

\*Criar o menu lateral que será exibido ao se clicar no menu hamburguer no canto superior esquerdo

\*Inserir links na sessão de descontos para direcionar o usuário para os produtos ou lista de produtos.

\*Inserir função para que seja possível salvar os produtos nos carrinhos ao se clicar no botão de mala que está em cada produto - DB

\*Criar função para ‘favoritar’ produtos - DB

\*Criar possibilidade aumentar quantidade de produtos (Quando o produto for salvo no carrinho) - DB

\*Inserir possibilidade de compartilhar (oportunidade para utilizar as vinculações com redes sociais)

\*Criar ranger slider para filtrar preços no filtro

\*Criar as funcionalidades do componente ‘filter’.

\*Criar as funcionalidades do componente ‘Cart’.

\*Inserir possibilidade de cálculo do frete

\*Criar backend para favoritos - DB

\*Configurar botões de sign up e login para aparecerem somente quando o user não estiver logado e os demais botões somente quando estiver - DB

\*Criar possibilidade para realizar cadastro e login com gmail e redes sociais (oportunidade para aprender como fazer isso) – DB

\*Botão para remover produto do carrinho - DB

\*\*MELHORIAS TÉCNICAS E ESTÉTICAS

\*Inserir a possibilidade de mais de 3 imagens no slide em ‘productsCard’, ainda que o máximo a ser exibido na barra abaixo do slider seja 3. Nesse caso as imagens não ficarão fixas, mas serão geradas por um loop.

\*Configurar responsividade do site

\*Transformar ‘tag’ em array, para que seja possível haver mais de uma tag

\*Melhorar slider do filtro de preços, tanto esteticamente quanto funcionalmente.

\*Entender o porquê do filtro estar ultrapassando os limites da div container e o porquê da galeria de produtos não estar alinhada com o cabeçalho de navegação.

\*Melhorar transição para abrir o filtro

\*Passar classes para módulos e ver o que pode ser reaproveitado

\*Inserir título na página de compra por ambiente, a que o usuário é direcionado ao clicar em ‘shop by room’

\*Corrigir detalhes existentes no layout geral em todos os componentes

\*Fazer com que a busca possa ser realizada com o enter

\*Transformar seleção de cores em um componente

\*\*TALVEZ.:

\*Talvez automatizar o slider de ‘productCard’

\*Talvez encontrar uma maneira do slider de ‘bestDeal’ girar 360º para que não sobre espaço no fim.

\*Talvez mudar topografia para FUTURA PT (ou algo semelhante), que é o indicado pelo design.

\*\*BUGS.:

\*Corrigir botão ‘show menos’ em ‘products’, está ficando habilitado quando há poucos produtos e zerando a lista de produtos se for pressionado quando há somente uma fileira.

\*Corrigir destaques nos itens da navegação

\*Descobrir como encerrar a atividade do modal após fechá-lo. Depois que ele é aberto continua sendo executado, ainda que se feche. Aparentemente no caso de componentes de classe é possível utilizar um lifecycle para realizar isso. Descobrir como fazer com hooks.

\*Resolver ponto nos slides da página principal onde há uma faixa no meio que impede o clique em uma parte dos slides, já que a div com o texto, barra de progresso e ícones de redes sociais está na frente.

- Vou decidir entre três possibilidades: deixar somente o texto do slide (onde fica o preço) clicável ou cobrir tudo com uma div invisível que ficaria por cima do texto e receberia o click. A terceira alternativa seria colocar os slides à frente da div e aplicar o ‘z-index’ diretamente nos elementos da div com o texto, assim as partes da div que não interessam ficariam por trás do slider e somente o necessário a sua frente.

\*Inserir ‘changeQtdeCallBack’ no componente de quantidade do modal

\*Soma dos valores dos produtos no carrinho não funcionando bem. Aparentemente o valor é resetado para o inicial ao aumentar a quantidade de cada produto.

\*O slider do filtro de preços às vezes falha

\*\*OBS.:

\*Verificar onde realmente é necessário importar ‘productsData’:

- Em ‘MainPageProducts’ aparentemente não é, já que pode ser importado diretamente em ‘Products’, o mesmo ocorre para ‘ShopProduct’, no entanto este último passa como props para ‘BestDeal’ além de somente para ‘products’, o que ocasiona uma importação a mais.

- O motivo de eu ter resolvido importar diretamente em ‘Products’ por ora é que ‘ShoByRoom’ não possuí o componente não o possui como child, o que impossibilita redirecionar o usuário de tal sessão diretamente para ‘Products’, já que a props referente à lista de produtos não existe. Importando diretamente em ‘Products’ o problema é eliminado.

- É preciso verificar esse detalhe para deixar o código o mais enxuto possível.

- Acredito que não será realizado diretamente em ‘Products’. Após o início da criação da página de favoritos me parece que realizar o fetch diretamente em ‘Products’ deixa a situação mais rígida e a lógica em ‘Products’ mais complexa.

\*Automatizar a população do DB (Não me lembro o nome, mas é o que fiz no app crud easy do angular)

\*Decidir o método de fetch a ser usado para exibir os produtos no carrinho. Por ora estou salvando o objeto com os produtos na tabela e realizando o fetch desta em ‘Cart’, mas também seria possível salvar somente o id, realizar o fetch de todos os produtos e extrair o necessário com um filter baseado no id, da maneira que fiz em ‘Wishlist’.