20240806

So in getLeads.html you put all the forms using bootstrap nav, design is almost ready  
  
Now what we need to test is like that:  
1. When user submit a widget form (let's say popup) you call this method: createNewWidget('Popup', form.serializeArray(), true) - you should have 2 buttons, Publish / Save Draft (if user click publish you send true)  
2. Then to test it refresh the page and call getWidgetsList() - in the dashboard you should call this method and show user the widgets he submitted before  
3. Add in the dashboard list of widgets 2 buttons - toggle status (from published to not published and call this toggleWidgetStatus), and another button to delete this widget (deleteWidget)  
4. Now the real test, after you see widget in this list, you load this file testGetLeads.html, then in embed.js you fill onLoad() function, you get the same widgets list and you should inject them to user site (testGetLeads.html is your user site, you can't change testGetLeads.html, but you can change embed.js)

you work on testGetLeads.html (this is tester like customer website) and getLeadsEmbed.js - this is where you inject the icons and their scripts

You should also put all the site preview scripts there, same scripts should be for preview and for real site injection

you need to change only 2 files for now - getLeads.html and embedGetLeads.js

1. Need to divide the scripts into 2 files - inject widgets to page html should be in embedGetLeads.js (which will be loaded into user site), scripts that are related only to GetLeads user interface should be in getLeads.js.
2. full cycle means user customize it, then click Submit and you call createNewWidget, and you also show user widgets using getWidgetsList function, and then inject to real user site in testGetLeads.html (shouldn't change this file, only embedGetLeads.js)

20240807

when finish the cycle you can see also method in widgets.js getWidgetsReport just to prepare some chart until ready. If there is no data, let me know widgetId and i will put some data.

Please do, I will start look at it, but the full flow is top priority now, to see user create a widgets, see a list of widgets, and the widget is showing on the tester

that's the most important now.

i created getLeads.css for you so you can put style there

Sorry i wasn't here. The dashboard is the homepage of the app and it should show the list of widgets that user created.

See first page here [https://www.dropbox.com/scl/fi/h8239dhjyw6jumhfikm70/Instructions.docx?rlkey=or9zeini0x2qt3pw1oj4xm7ge&dl=0](https://www.dropbox.com/scl/fi/h8239dhjyw6jumhfikm70/Instructions.docx?rlkey=or9zeini0x2qt3pw1oj4xm7ge&dl=0" \t "https://mail.google.com/mail/u/0/?ogbl" \l "label/_blank)

But there is an issue with getWidgetList method, it return empty result, checking it now

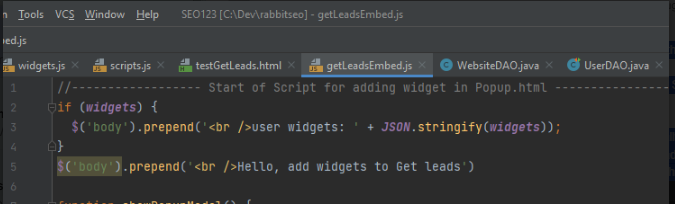
You also have a method of delete, you can create a tab before the Popup called Dashboard and there you can display all this widgets list with 2 buttons for each widget - update (that will lead to same page like in screenshot) and delete so you can delete all old items.

20240808

OK no problem, so you can remove that button and just show the widgets in the dashboard tab.

If you can just push to git, easier than send files here, they look good. On the Story widget when I can click on the on the story it doesn't open it, it should open or not yet?

In testGetLeads I know, I just open example web page because it doesn't really matter on which web page the widgets will appear on, if you want I can change to real website, but the main thing there is that getLeadsEmbed.js is being called on that page and the widgets var is filled with the widgets list from the server (see below), so you need to show the widgets there.



Since this embedded script is loaded both inside the app (to see site preview) and in the real customer site (to display the live widget) I added a var called **realSite**which is set to 'false' in the first case and to 'true' in the second case.

Then in the getLeadsEmbed.js you should do something like this:

if (***realSite***) {  
 onRealSiteLoad();  
}  
function onRealSiteLoad() {  
 ***console***.log('onRealSiteLoad start');  
 // *todo - inject widget to the page* if (***widgets***) {  
 for (let i = 0; i < ***widgets***.length; i++) {  
 $('body').prepend('<br />User widgets: ' + ***widgets***[i]);  
 }  
 }  
}

getWidgetsReportSummary() When you get to the monitoring page you can just take data from here and show in charts

20240809

It's not ready yet but it would be great if you can have a look and make sure the features match smartarget as the designer might have skipped some stuff and then we need to add them

[https://xd.adobe.com/view/c4391a93-8cbb-422a-bb2a-6d9e75ea1f54-916a/screen/713d4d56-96dc-4d29-8c94-33000686c6ce](https://xd.adobe.com/view/c4391a93-8cbb-422a-bb2a-6d9e75ea1f54-916a/screen/713d4d56-96dc-4d29-8c94-33000686c6ce" \t "https://mail.google.com/mail/u/0/_blank)

20240811

- Ok please show me also the full cycle test that you create a widget, see all widgets in dashboard and also in the getLeadsTest.html if you can send a video even better.

New projects - one is to offer user add 'Join to our newsletter form' widget in all those style, but just this input, maybe also name and phone number, but not fully customized, this is why this scripts needs to be generic, so can use in other projects.

And another project - to add widget that will embed blog in user site.

- Other projects eventually is one widgets, just user can choose how the icon will look like, how to open the dialog. But there we need to design few predefined templates to so user can choose his design (for blog and for the form).

- Yes there can be more projects, but we need to see it fits with time, try to finish everything by wednesday (I hope design will be ready tomorrow).

Can you please explain more words about the missing fields or add screenshots of those missing fields?

As for the review i asked to remove them to make it easier and also I don't think they're needed that much.

// const report = getWidgetsReport("CountryName");

// console.log("=========widget report " + JSON.stringify(report));