

Earth Engine User Interfaces and Apps

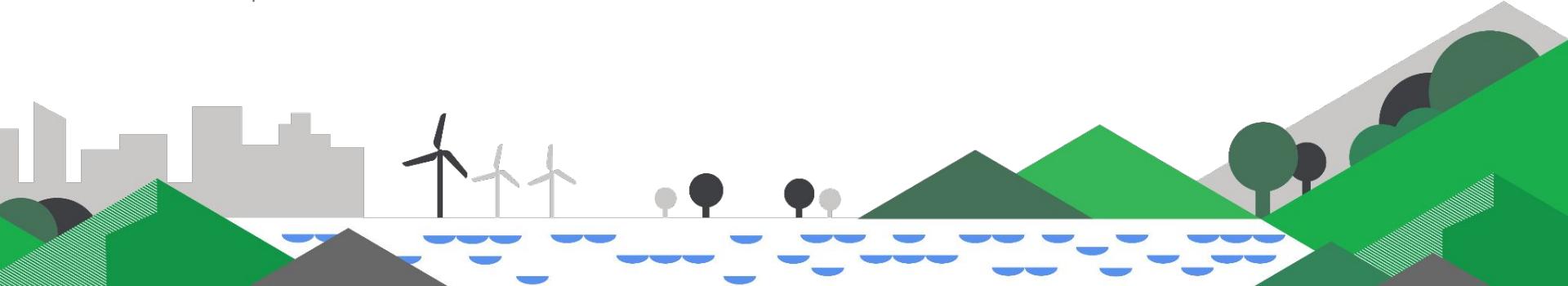
How to build an awesome UI in Earth Engine



Jorge Garcia and Sufyan Abbasi

Earth Engine Frontend Team

October 2022 | #GeoForGood22



But first...

⚡ Lightning Talk ⚡ How to Build a User Interface in 6 easy steps

Step 1: Figure out what you want your user to be able to do.

“

My deforestation detector app will allow users to click on the map to show a chart of NDVI over time and a way to set a threshold for masking pixels.

”



Step 2: Draw a sketch of how you want the app to look.



Step 3: Identify the interactions of the app.



Step 4: Structure the app code as callable functions for each interaction.

```
function makeChart(mapPoint) {  
    return ui.Chart(...)  
}
```

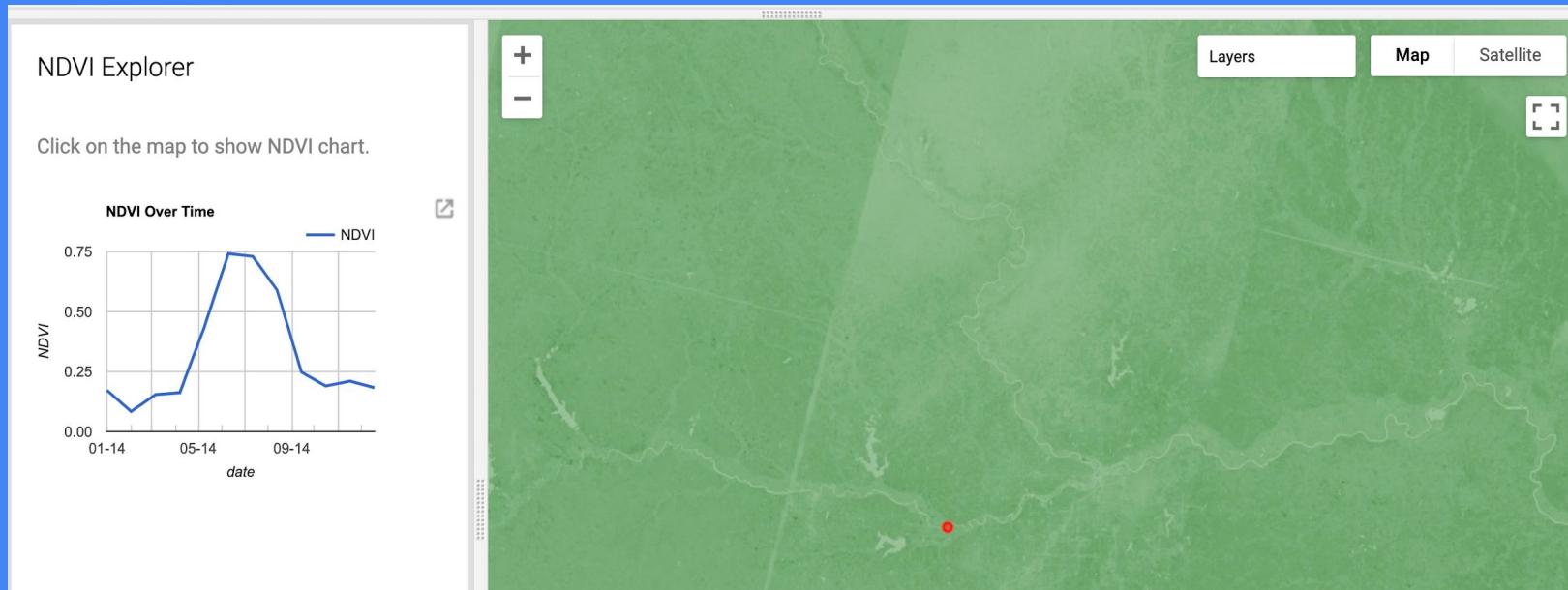


Step 5: Find the widgets, compose them together, bind the callback functions, style them, test it out, debug...

???



Step 6: Success!





Thank you!



Geo for Good Summit 2022

#GeoForGood22

Goals

01 Widgets

Learn about the various Earth Engine widgets and how they operate.

02 Earth Engine Apps

Learn how to build beautiful Earth Engine apps.

03 Code Structure

Learn techniques to structure your code to make developing UIs easier.

Logistics

Earth Engine User Interfaces and Apps

Get the example repository here:

[https://code.earthengine.google.com/?
accept_repo=users/sufy/q4g-ui-apps](https://code.earthengine.google.com/?accept_repo=users/sufy/q4g-ui-apps)

Find the slides in the resources section:

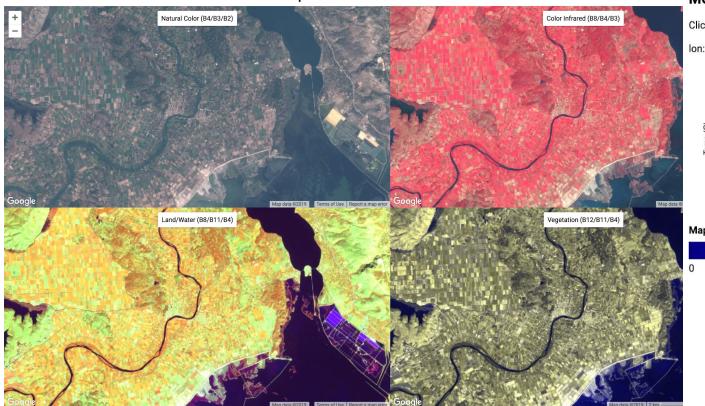
[https://earthoutreachonair.withgoogle.com/events/ge
oforgood22?talk=day1-track3-talk1](https://earthoutreachonair.withgoogle.com/events/geoforgood22?talk=day1-track3-talk1)



Earth Engine Apps



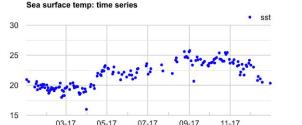
September 2018 Sentinel-2 Visualizations



MODIS Ocean Temperature - Time Series Inspector

Click a location to see its time series of ocean temperatures.

lon: -19.16 lat: 22.31



Map Legend: median 2017 ocean temp (C)



Collection Mosaic Editor

This app allows you to interactively explore the effects of mosaicking different Landsat 8 TOA Images using ee.Reducer.median(). To change which tiles are included in the mosaic, check or uncheck the thumbnails. To mosaic another area, pan/zoom and click on the map.

LC08_018030_20170216

LC08_018030_20170304

LC08_018030_20170523



LC08_018030_20170608

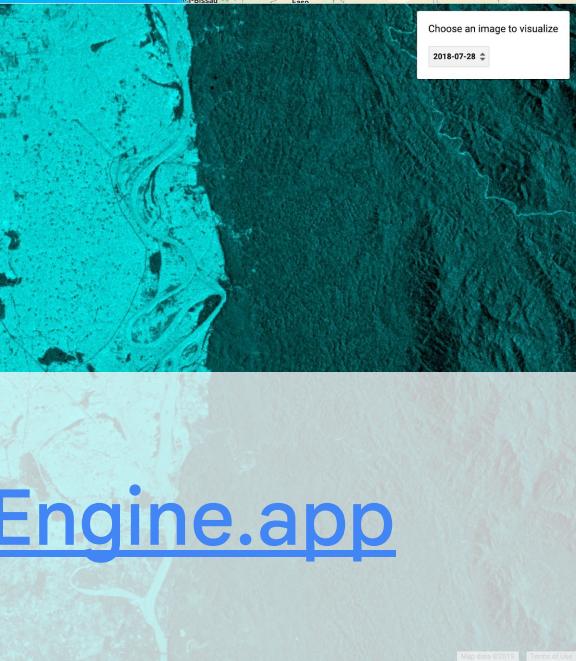
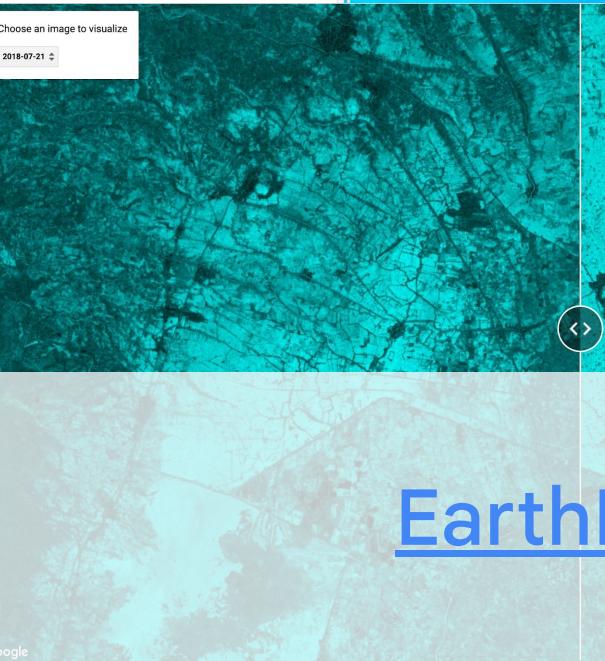
LC08_018030_20170624

LC08_018030_20170827



LC08_018030_20170912

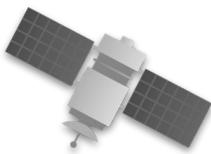
LC08_018030_20171201



EarthEngine.app

What is an Earth Engine app?

EE apps are **dynamic** user interfaces for **publishing** and **sharing** your Earth Engine analyses and results.



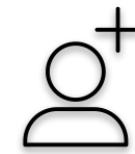
Dynamic

If your Earth Engine script is dynamic (i.e. as new satellite data is ingested), then your app will be, too!



Published online

You can publish your results online for easy visualization and exploration.



Shareable

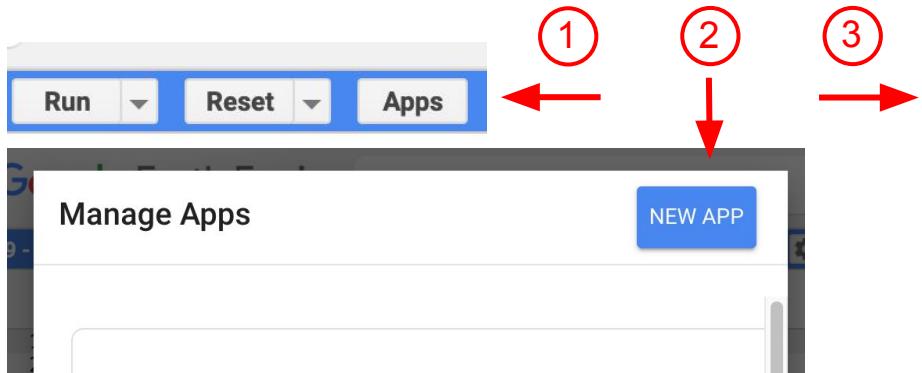
People don't have to be Earth Engine users to view it.

Earth Engine Apps don't have a console!

When printing to the console is not enough

Earth Engine Apps let you communicate your analyses to a broader audience outside of the Earth Engine developer community.

We want to let users supply their own inputs into your analysis code in order to let them **explore what they're interested in**.



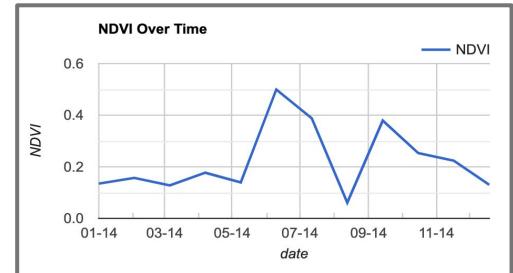
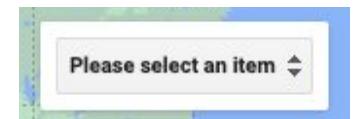
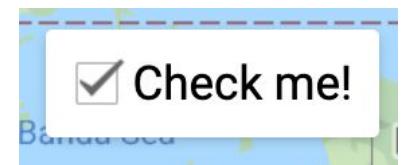
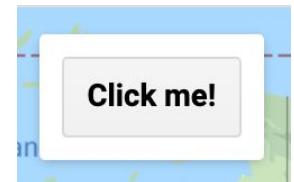
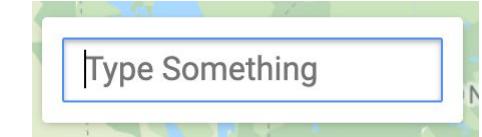
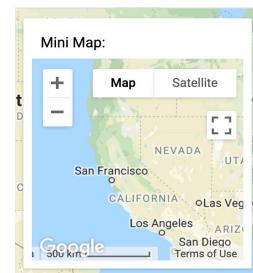
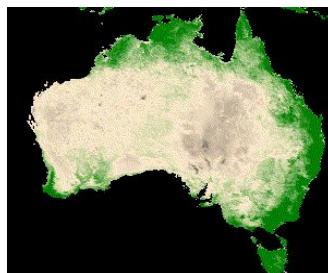
A screenshot of the 'Publish New App' dialog box. At the top, it says 'Publish New App' and 'Name and URL'. There are three tabs at the bottom: 'Name and URL' (selected), 'Source Code', and 'Publication Details'. Each tab has a red circle with a number above it: '1' for 'Name and URL', '2' for 'Source Code', and '3' for 'Publication Details'. Below the tabs is a section titled 'App Name and URL' with a sub-section 'App Name' containing the text 'My App'.

Earth Engine Apps are built from Widgets

Earth Engine User Interfaces and Apps

Widgets are user interface components that let users of your app interact with your script.

Whenever a user interacts with one, a widget generates an event that your script listens for and executes a callback function that you supply.



Widgets



What kind of Widgets do we have in Earth Engine?

Earth Engine User Interfaces and Apps

1. Widgets that let you **display** information.
2. Widgets that let users **input parameters** or **interact** with your script.
3. Widgets that let you **layout** other widgets.

What kind of Widgets do we have in Earth Engine?

Earth Engine User Interfaces and Apps

1. Widgets that let you **display** information.
2. Widgets that let users **input parameters or interact** with your script.
3. Widgets that let you **layout** other widgets.

Display Widget: ui.Label

Earth Engine User Interfaces and Apps

ui.Label

```
var label = ui.Label("Hello!");  
  
label.setValue("Changed label.");
```



ui.Chart

```
label.setUrl("https://earthengine.google.com");  
  
label.setImageUrl("data:image/gif;base64,...");
```

ui.Thumbnail

```
Map.add(label);
```

New!



#GeoForGood22

Display Widget: ui.Chart

Earth Engine User Interfaces and Apps

ui.Label

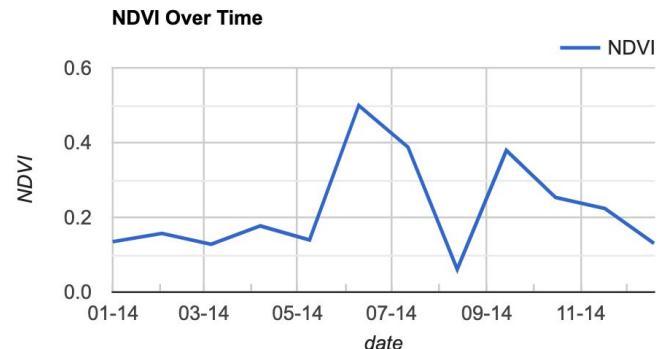
```
var chart = ui.Chart.image.series(  
    ndvi, point, ee.Reducer.mean(), 200);  
chart.setOptions({
```

ui.Chart

```
    title: 'NDVI Over Time',  
    vAxis: {title: 'NDVI'},  
    hAxis: {  
        title: 'date',  
        format: 'MM-yy',  
        gridlines: {count: 7},  
    },  
});
```

ui.Thumbnail

```
Map.add(chart);
```



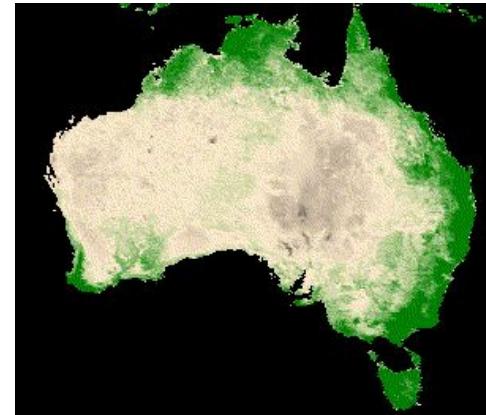
#GeoForGood22

Display Widget: ui.Thumbnail

Earth Engine User Interfaces and Apps

See: [Animated Thumbnail Example](#)

```
ui.Label      ui.Thumbnail({  
    image: collection,  
    params: {  
        crs: 'EPSG:3857',  
        dimensions: '300',  
        region: rect,  
        min: -2000,  
        max: 10000,  
        palette: 'black, blanchedalmond, green, green',  
        framesPerSecond: 12,  
    }  
});
```



What kind of Widgets do we have in Earth Engine?

Earth Engine User Interfaces and Apps

1. Widgets that let you **display** information.
2. Widgets that let users **input parameters** or **interact** with your script.
3. Widgets that let you **layout** other widgets.

Display Widget: ui.Buttons

Earth Engine User Interfaces and Apps

ui.Button

```
var button = ui.Button("Click me!");
```

ui.Checkbox

```
button.onClick(function() {  
    print("I was clicked");
```

ui.Slider

```
});
```

ui.DateSlider

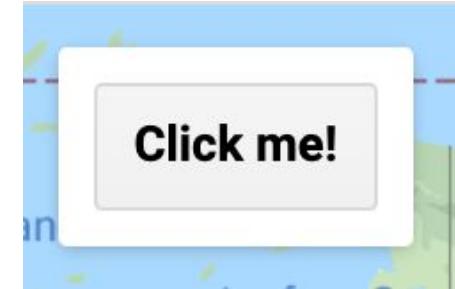
```
Map.add(button);
```

ui.Select

```
button.setImageUrl("data:image/gif;base64,...");
```

ui.Textbox

New!



Display Widget: ui.Checkbox

Earth Engine User Interfaces and Apps

ui.Button

```
var checkbox = ui.Checkbox("Check me!");
```

ui.Checkbox

```
checkbox.onChange(function(isChecked) {
```

ui.Slider

```
    if (isChecked) {
```

ui.DateSlider

```
        print("I'm checked");
```

ui.Select

```
    } else {
```

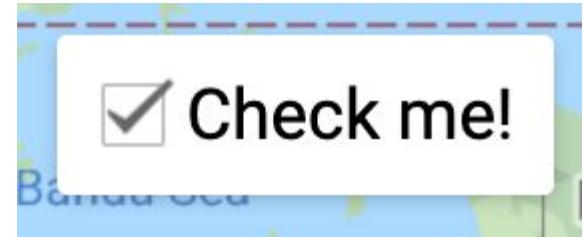
ui.Textbox

```
        print("I'm unchecked");
```

```
}
```

```
});
```

```
Map.add(checkbox);
```



Display Widget: ui.Slider

Earth Engine User Interfaces and Apps

ui.Button

```
var slider = ui.Slider(0, 10, 5, 1);
```

ui.Checkbox

```
slider.onChange(function(value) {  
    print(value);  
});
```

ui.Slider

```
});
```

ui.DateSlider

```
Map.add(slider);
```

ui.Select

ui.Textbox



Display Widget: ui.DateSlider

Earth Engine User Interfaces and Apps

ui.Button

ui.Checkbox

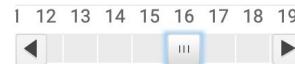
ui.Slider

ui.DateSlider

ui.Select

ui.Textbox

```
var dateSlider = ui.DateSlider({  
  start: "2019-09-10",  
  end: "2019-09-20",  
  value: "2019-09-16",  
});  
  
dateSlider.onChange(function(dateRange) {  
  print(dateRange);  
});  
  
Map.add(dateSlider);
```



Sep 16, 2019 [Jump to date](#)

« September »							« 2019 »		
Sun	Mon	Tue	Wed	Thu	Fri	Sat			
36	1	2	3	4	5	6	7		
37	8	9	10	11	12	13	14		
38	15	16	17	18	19	20	21		
39	22	23	24	25	26	27	28		
40	29	30	1	2	3	4	5		
41	6	7	8	9	10	11	12		
Today							None		

Display Widget: ui.Select

Earth Engine User Interfaces and Apps

ui.Button

```
var select = ui.Select({  
  items: ["item1", "item2", "item3"],  
  placeholder: "Please select an item",
```

ui.Slider

```
  value: "item1",
```

ui.DateSlider

```
});  
  
select.onChange(function(value) {
```

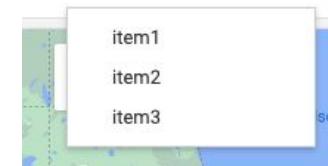
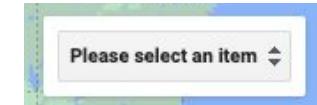
ui.Select

```
  print(value);
```

ui.Textbox

```
});
```

```
Map.add(select);
```



Display Widget: ui.Textbox

Earth Engine User Interfaces and Apps

ui.Button

```
var textbox = ui.Textbox("Type  
Something");
```

ui.Checkbox

```
textbox.onChange(function(text) {
```

ui.Slider

```
    print("Typed: " + text);
```

ui.DateSlider

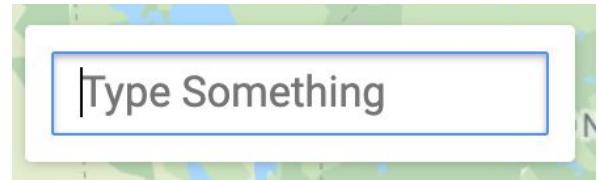
```
});
```

```
print(textbox.getValue());
```

ui.Select

```
Map.add(textbox);
```

ui.Textbox



What kind of Widgets do we have in Earth Engine?

Earth Engine User Interfaces and Apps

1. Widgets that let you **display** information.
2. Widgets that let users **input parameters or interact** with your script.
3. Widgets that let you **layout** other widgets.

Display Widget: ui.Map

Earth Engine User Interfaces and Apps

ui.Map

```
var map = ui.Map();
```

ui.Panel

```
ui.root.clear();
```

ui.SplitPanel

```
ui.root.add(map);
```

```
map.addLayer(...);
```

```
map.onClick(...);
```



Display Widget: ui.Panel

Earth Engine User Interfaces and Apps

ui.Map

```
var panel = ui.Panel();  
  
var title = ui.Label("Mini Map:");
```

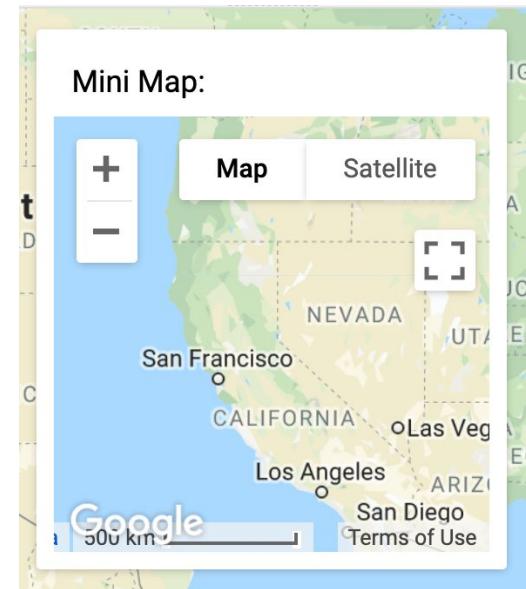
ui.Panel

```
panel.add(title);
```

ui.SplitPanel

```
panel.add(ui.Map());  
  
panel.setLayout(  
    ui.Panel.Layout.flow("vertical"))
```

```
Map.add(panel);
```



Display Widget: ui.SplitPanel

Earth Engine User Interfaces and Apps

ui.Map

```
var leftMap = ui.Map();  
var rightMap = ui.Map();
```

ui.Panel

```
var splitPanel = ui.SplitPanel({  
    firstPanel: leftMap,
```

ui.SplitPanel

```
    secondPanel: rightMap,  
    wipe: true, // false for resizable panels  
    orientation: "horizontal",
```

```
});
```

```
ui.root.clear();  
ui.root.add(splitPanel);
```



How do I work with widgets?



But first, some JavaScript!



Functions

Earth Engine User Interfaces and Apps

A function is a block of code that executes when you call it by its name. They may also accept some inputs:

```
function sayHello(name) {  
    return "Hello, my name is " + name;  
}  
  
print(sayHello("Sufyan"));
```

Console Output:

```
> Hello, my name is Sufyan
```

Dictionaries

Earth Engine User Interfaces and Apps

An object is a data structure that allows you store key-value pairs.
Input a key and it returns the value that you passed in:

```
var imageCollections = {  
  landsat: 'LANDSAT/LC08/C01/T1_RT',  
  sentinel: 'COPERNICUS/S5P/NRTI/L3_N02',  
}  
  
print(imageCollections.landsat)  
print(imageCollections['sentinel'])  
imageCollections.srtm = 'CGIAR/SRTM90_V4'
```

Console Output:

```
> LANDSAT/LC08/C01/T1_RT  
> COPERNICUS/S5P/NRTI/L3_N02
```

How do I work with widgets?



Widget Workflow

Earth Engine User Interfaces and Apps

1. Make a widget:

```
var button = ui.Button("Click Me");
```

2. Add event listeners:

```
button.onClick(function() {  
  print("I was clicked.");  
});
```

3. Add it to another widget:

```
Map.add(button);
```

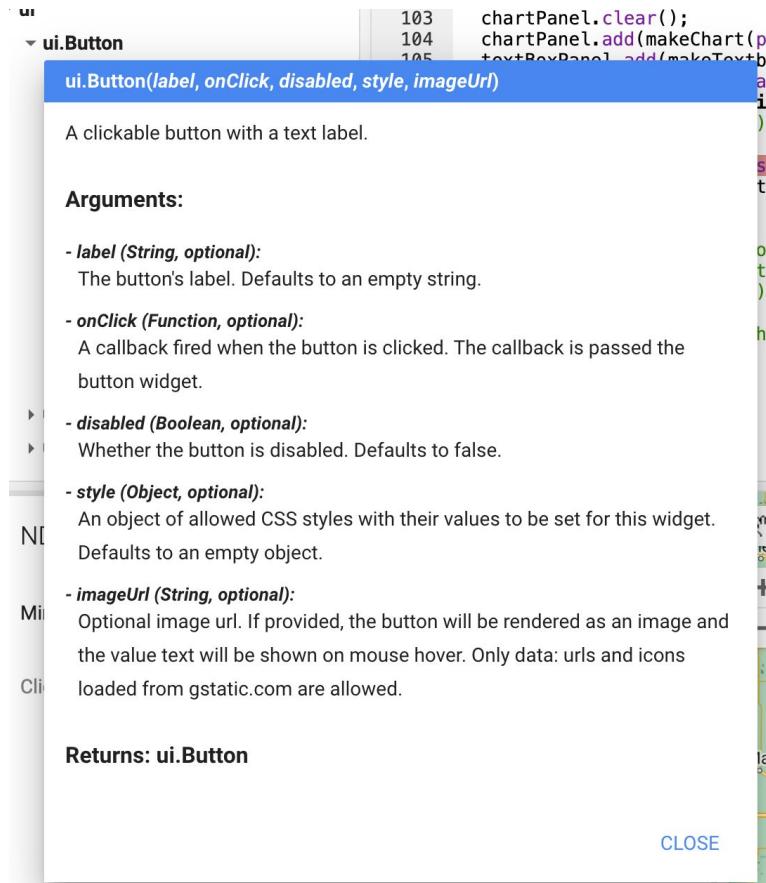
4. Style it:

```
button.style().set('color', 'red');  
button.style().set('position', 'top-right');
```

A more compact form:

Earth Engine User Interfaces and Apps

```
var button = ui.Button({  
  label: "Click Me",  
  onClick: function() {  
    print("I was clicked.");  
  },  
  style: {  
    color: 'red',  
    position: 'top-right',  
  },  
});  
Map.add(button);
```



Result:

Earth Engine User Interfaces and Apps



#GeoForGood22

How do I make my widgets useful?



Event Handling: making your widgets interactive

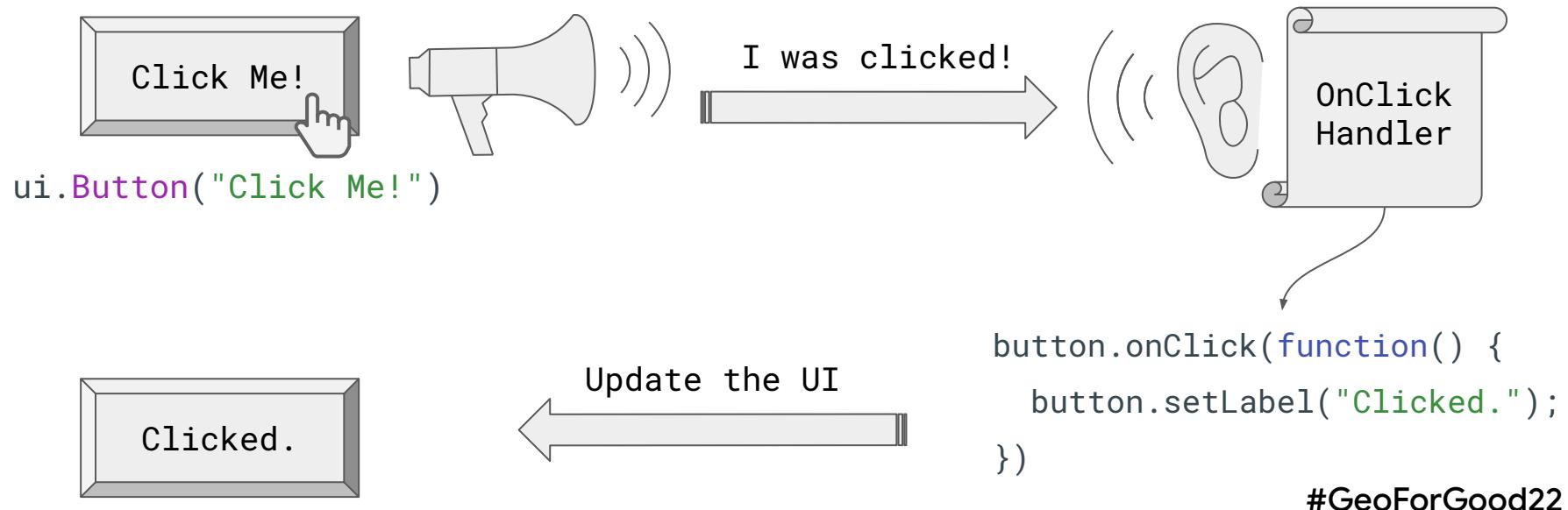
Earth Engine User Interfaces and Apps

By binding an event handler, or callback, to a user event, you can execute a function when the user does something.

Event Handling: making your widgets interactive

Earth Engine User Interfaces and Apps

By binding an event handler, or callback, to a user event, you can execute a function when the user does something.



Event Handling Patterns

Earth Engine User Interfaces and Apps

Most of our widgets (except for ui.Label) have one of two event handlers:
onClick or **onChange**, depending on the widget.

All of the **onChange** callback functions share the same pattern for inputs: the first parameter is the new input value and the second is the widget itself:

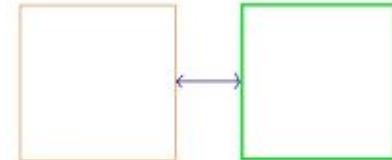
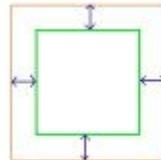
```
textbox.onChange(function(value, widget) {  
  print("Widget: " + widget + " has value: " + value);  
});
```

How do I make my app pretty?



Styling Widgets

Earth Engine User Interfaces and Apps



Easy styling properties (they use CSS syntax)

Padding

Margin

Property	Function	Example Values
padding	Puts padding inside the widget, surrounding it	'10px' '10px 0px' '10px 5px 0px 5px'
margin	Adds space between widgets	'10px' '10px 0px' '10px 5px 0px 5px'
border	Adds a styled border	'2px solid gray' '1px dashed #333'
color	Changes the font color	'white' 'rgb(255,255,255)' '#ffffff'
backgroundColor	Changes the background color	'white' 'rgb(255,255,255)' '#ffffff'
fontWeight	Changes font weight	'bold' 'lighter' '100'
position	Changes position on the map	'bottom-right', 'top-left', 'top-center'

Styling Widgets

Earth Engine User Interfaces and Apps

You can style widgets two ways:

1. Supply a **style** property in the widget constructor (emulates CSS):

```
style: {  
    height: '100%',  
    width: '30%',  
    position: 'bottom-right'  
}
```

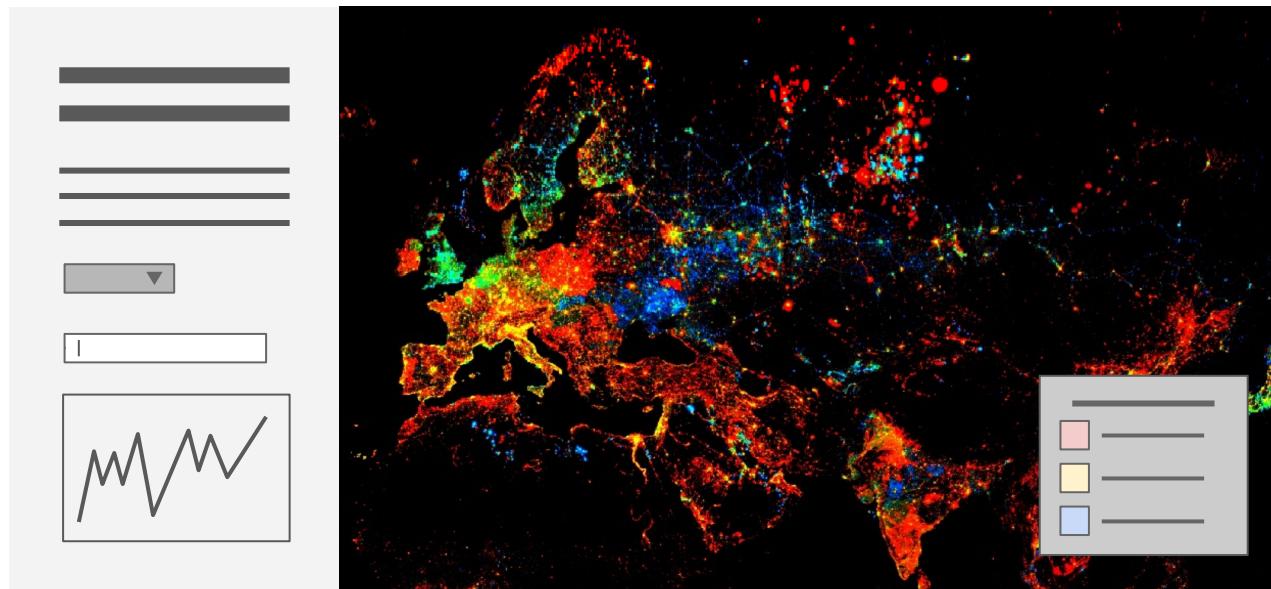
2. Explicitly set a style property using **myWidget.style().set(...)**:

```
myWidget.style().set('position': 'bottom-right');
```

Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

EE Apps “Bread and Butter” Template in 4 steps:



#GeoForGood22

Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

Step 1: Make the side panel. Add widgets to it.



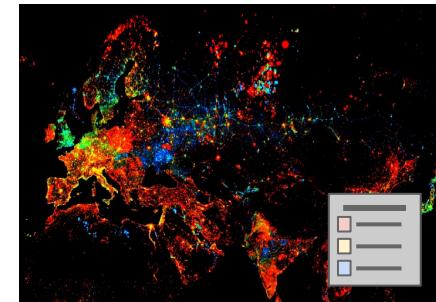
```
var sidePanel = ui.Panel({  
  layout: ui.Panel.Layout.flow('vertical'),  
  style: {  
    height: '100%',  
    width: '30%',  
  },  
})  
sidePanel.add(ui.Label(...));
```

Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

Step 2: Make the map and legend.

```
var map = ui.Map();  
  
var legendPanel = ui.Panel({  
  widgets: [ui.Label("Legend")],  
  layout: ui.Panel.Layout.flow('vertical'),  
  style: {  
    position: 'bottom-right',  
  },  
});  
  
map.add(legendPanel);
```



#GeoForGood22

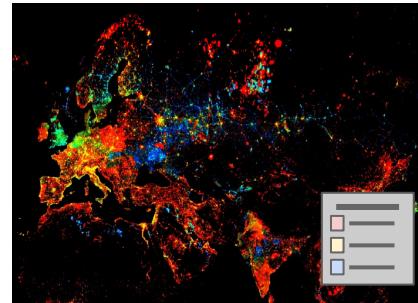
Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

Step 3: Combine the side panel and map into a ui.SplitPanel:



```
var splitPanel = ui.SplitPanel({  
  firstPanel: sidePanel,  
  secondPanel: map,  
});
```



Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

Step 4: Clear the ui.root and add the split panel:

Removes the default map from the screen.

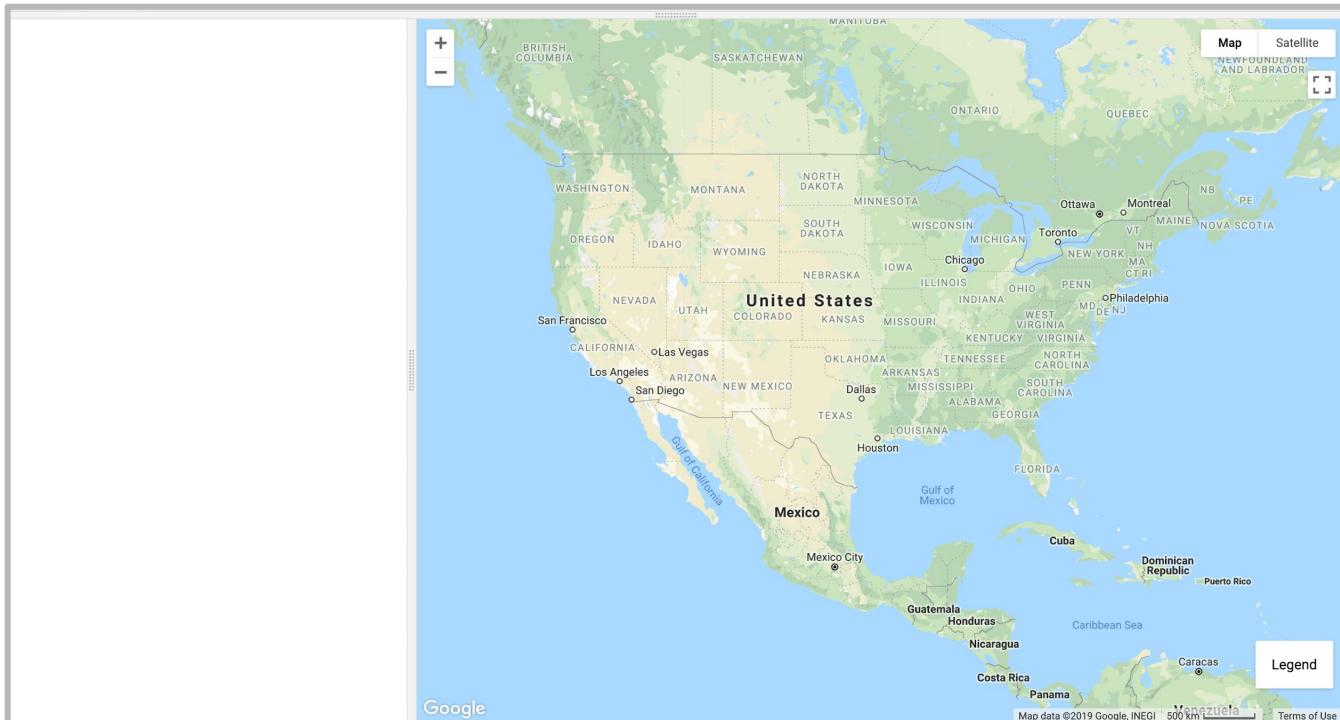
```
ui.root.clear();  
  
ui.root.add(splitPanel);
```

Make the split panel the only widget on the screen.

Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

All that's left...



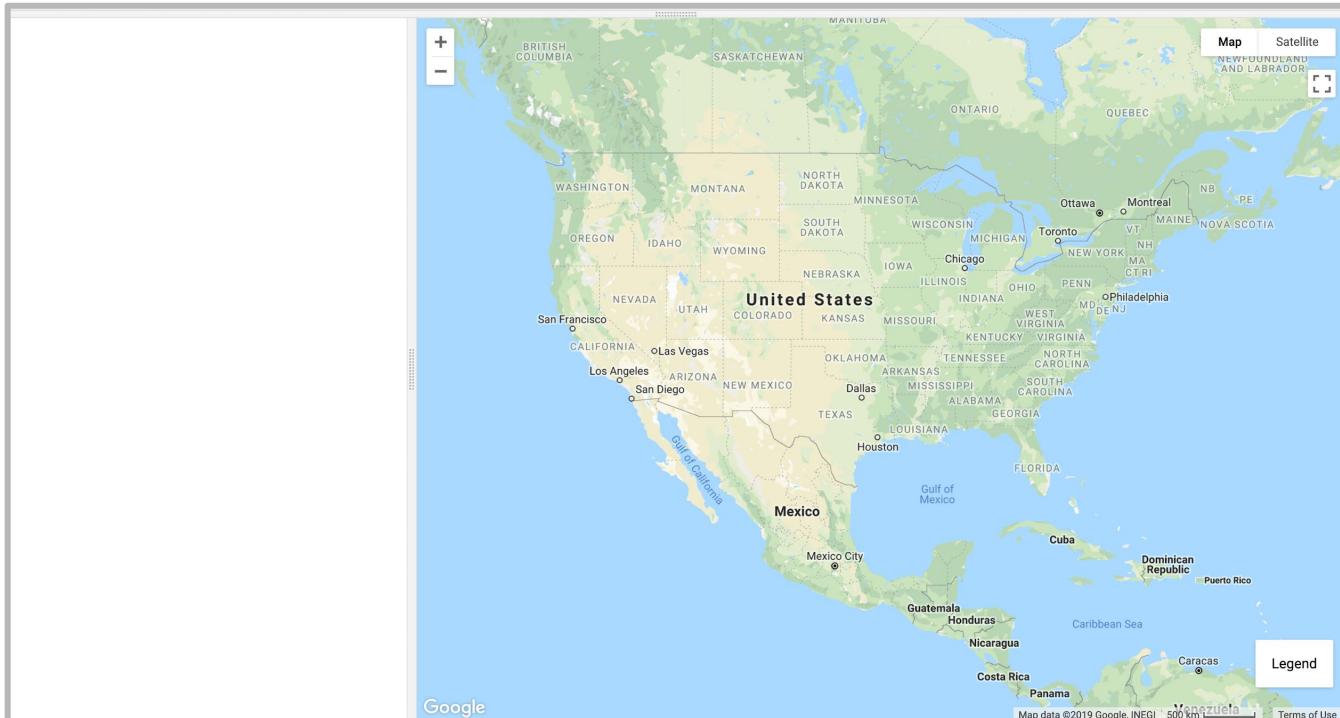
#GeoForGood22

Making Layouts with ui.Panel and ui.Map

Earth Engine User Interfaces and Apps

All that's left...

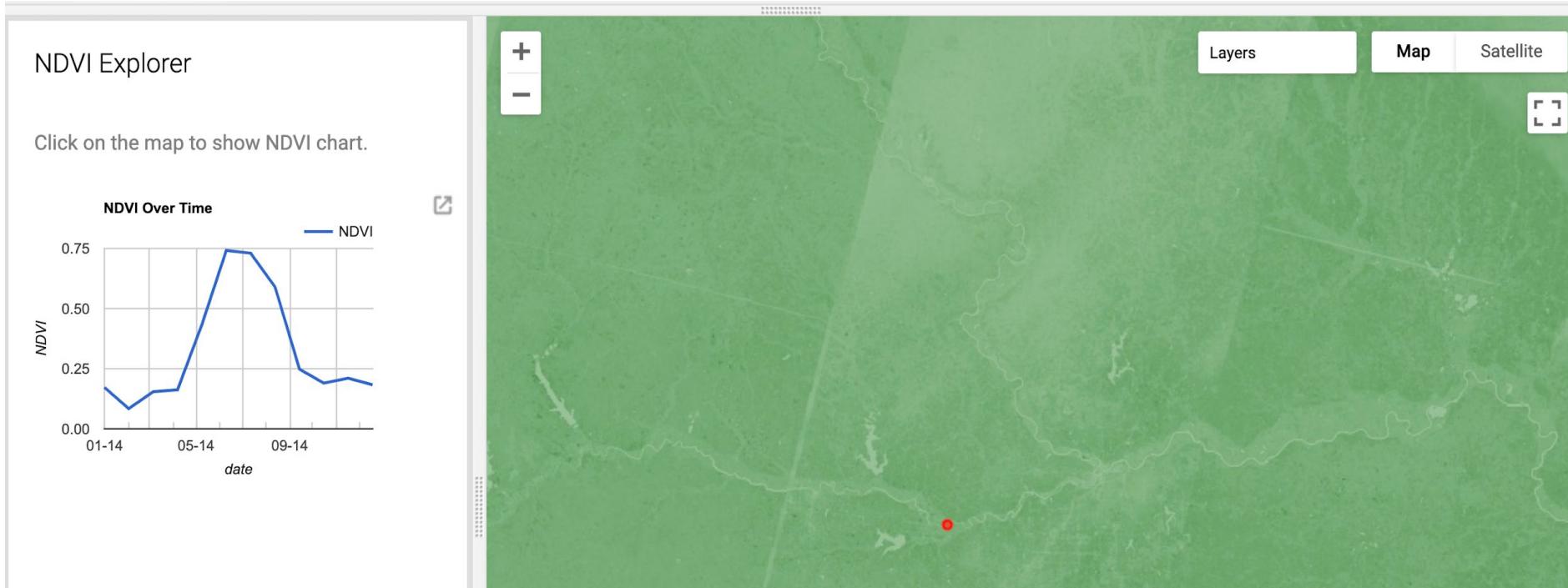
is the hard part!



#GeoForGood22

Example: NDVI Explorer (see Logistics slide)

Earth Engine User Interfaces and Apps



Want to learn more? Start here!

Earth Engine User Interfaces and Apps

Resource	When
<u>Publishing an Earth Engine App to Accompany your Research Paper</u> by Tyler Erickson and Justin Braaten	2021
<u>Publishing and Earth Engine App to Accompany your Research Paper</u> Resources - accompanying code snippets for the talk above	2021
<u>Hey Earth Engine, How Do I? - Apps Edition</u> by Tyler Erickson and Justin Braaten and others	2021
<u>Making a Simple Earth Engine App: Cooking Class with Michael & Karin</u> - a fun way of approaching building apps!	2020
<u>Earth Engine Widgets “Tasting Menu”</u>	2020
<u>Earth Engine Developer Docs on Earth Engine Apps and Widgets</u>	-

Earth Engine User Interfaces and Apps

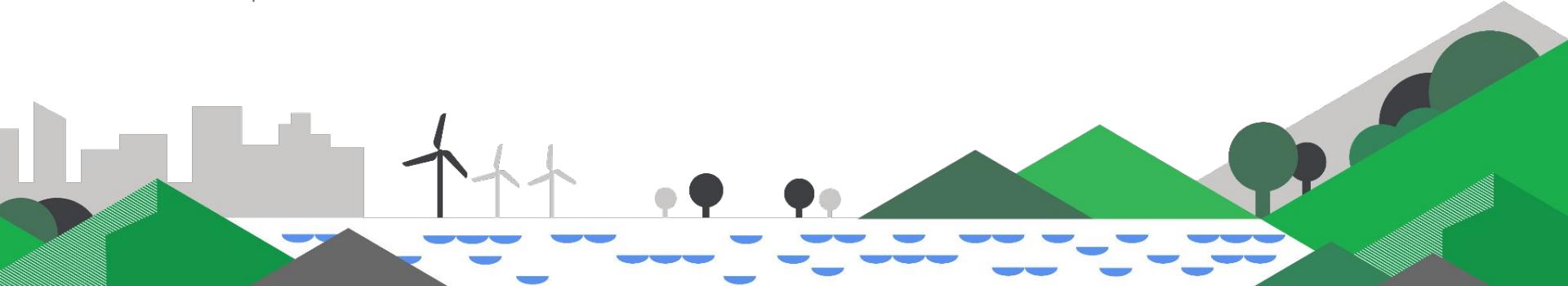
How to build an awesome UI in Earth Engine



Jorge Garcia and Sufyan Abbasi

Earth Engine Frontend Team

October 2022 | #GeoForGood22





Thank you!



Geo for Good Summit 2022

#GeoForGood22