

Using Adaptive Cards Designer control from the PnP control gallery in SPFx solutions

Fabio Franzini – CEO Apvee Solutions – Microsoft MVP

Agenda

- Recap of Adaptive Card Host
- Introduction to the Adaptive Card Designer control, how it was implemented and how to use it in SPFx
- Demo
- Next Steps
- References

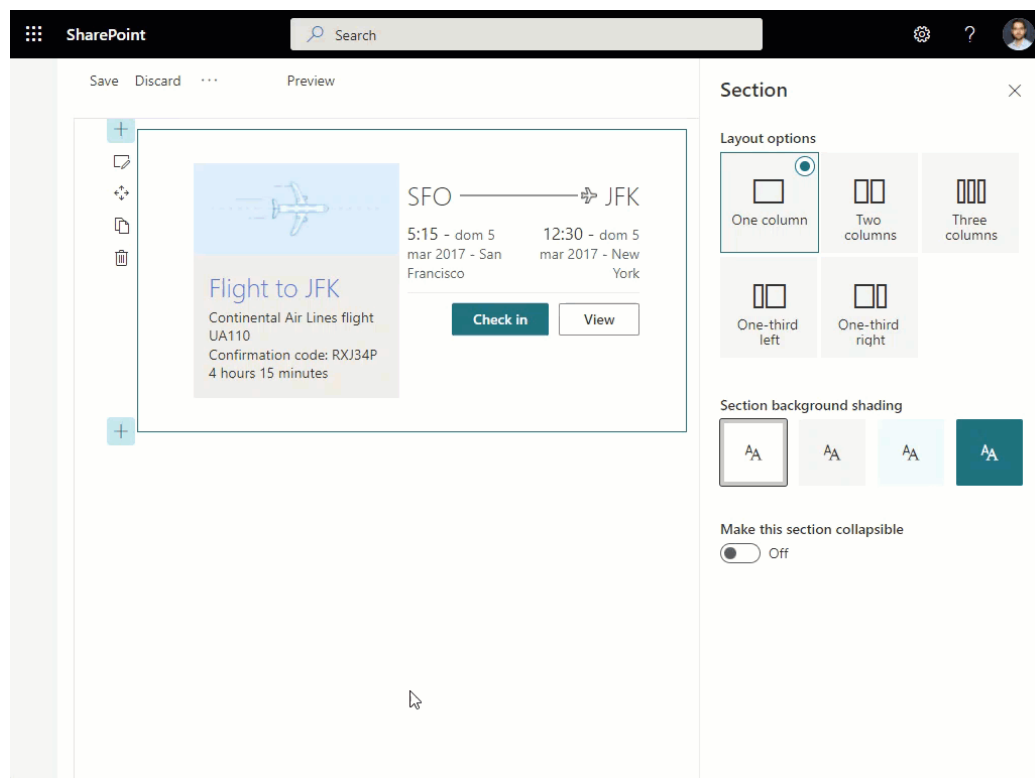
Recap of Adaptive Card Host

- React Control that render an Adaptive Card using the SDK
- Using a brand-new elements that use the Fluent UI Theme both for SharePoint and Microsoft Teams
- Auto apply the AC Template if data object is passed to the control
- Callbacks to intercept Actions and Errors
- Callbacks to set Custom Elements and Custom Actions
- Callback to set custom Host Capability properties

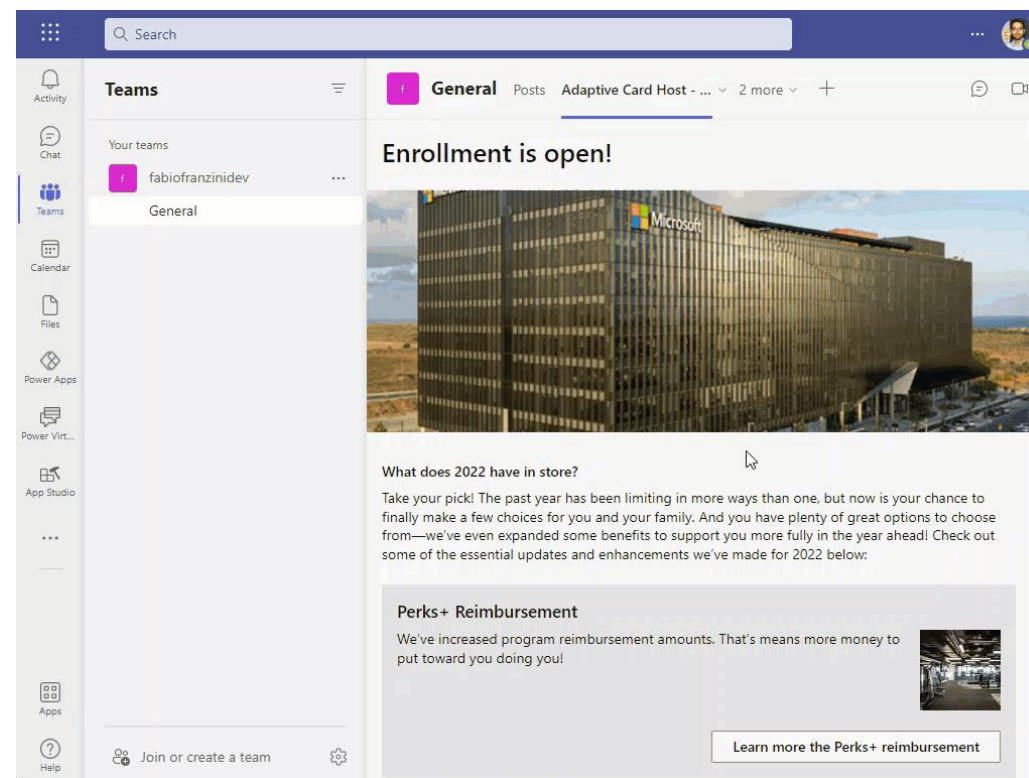
```
<AdaptiveCardHost
  card={card}
  data={data}
  style={null}
  className={null}
  theme={props.theme}
  themeType={themeType}
  hostConfig={null}
  onInvokeAction={(action) => alert(JSON.stringify(action))}
  onError={(error) => alert(error.message)}
  onSetCustomElements={(registry: CardObjectRegistry<CardElement>) => { }}
  onSetCustomActions={(registry: CardObjectRegistry<Action>) => { }}
  onUpdateHostCapabilities={(hostCapabilities: HostCapabilities) => {
    hostCapabilities.setCustomProperty("__customProperty", Date.now);
  }}
  context={props.context}
/>
```

Recap of Adaptive Card Host: Same code, different scenarios

SharePoint - SPFx Web Part



Teams Tab - SPFx Web Part



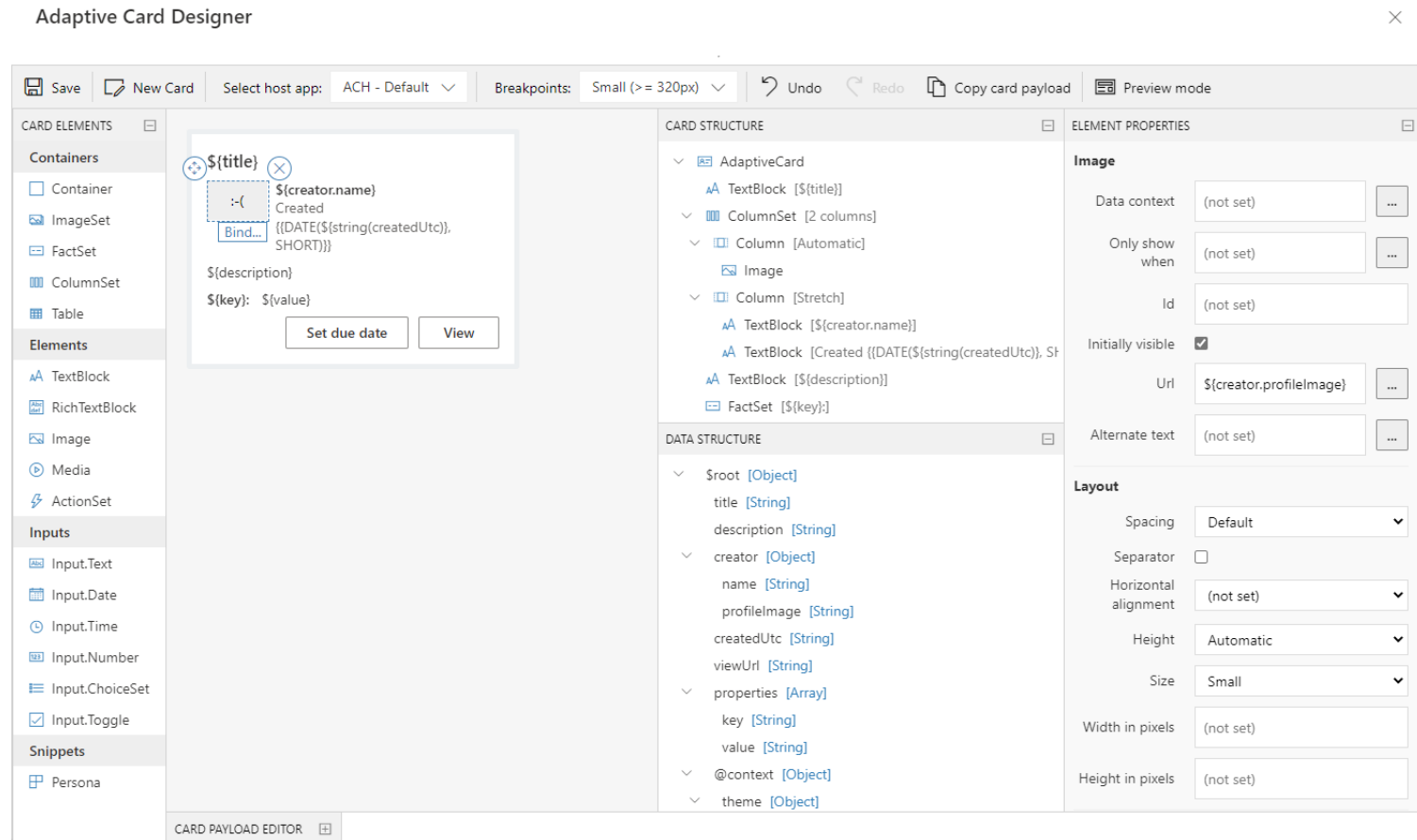
Introduction to the Adaptive Card Designer control

- React Control that render an Adaptive Card Designer using the SDK
- Open the Designer as a Fluent UI Panel
- Fix all problems on hosting the Designer in React controls and in SPFx
- Use all Elements and Actions from the Adaptive Card Host control
- Same “Data Structure” of the Adaptive Card Host control
- Easy to use!
- Possibility to add snippets of cards
- Available from the next version of the PnP Reusable React controls library

**NOT YET
AVAILABLE**

```
<AdaptiveCardDesignerHost
  cardPayload={card}
  data={data}
  theme={props.theme}
  context={props.context}
  enableDataBindingSupport={true}
  showDataStructureToolbox={true}
  showTargetVersionMismatchWarning={true}
  bindingPreviewMode={BindingPreviewMode.GeneratedData}
  snippets={snippets}
  onSave={({payload: object}) => setCard(payload)}
  hideCopyToJsonToolbarCommand={false}
/>
```

Introduction to the Adaptive Card Designer control



DEMO & CODE

Next Steps

- ~~Implement the Adaptive Card Host component for the PnP Reusable React Controls Library (Available from version 3.6.0)~~
- ~~Implement the Adaptive Card Designer component for the PnP Reusable React Controls Library (Available from the next version of the library)~~
- Create a set of new Adaptive Cards Elements (targeted only for JS SDK)
 - Maybe using Fluent UI React
 - Maybe using Fluent UI Web Components
 - Maybe using MGT (Thanks to João Mendes)

About Me

I'm CEO at Apvee Solutions, startup focused on building solutions on top of Microsoft 365 ecosystem.

I'm a Microsoft MVP in Office Development and Business Applications categories.

- Mail: fabio@apvee.com
- Twitter: [@franzinifabio](https://twitter.com/franzinifabio)
- LinkedIn: www.linkedin.com/in/fabiofranzini
- GitHub: <https://github.com/fabiofranzini>



References

- <https://adaptivecards.io/>
- <https://pnp.github.io/sp-dev-fx-controls-react/controls/AdaptiveCardHost/>
- <https://github.com/pnp/sp-dev-fx-webparts/tree/main/samples/react-adaptive-card-host-control>

Thank you!

Fabio Franzini – CEO Apvee Solutions – Microsoft MVP