



nordic.
summit

Custom Pages in Model-Driven Apps: The Ultimate Guide for App Makers

DIAMOND SPONSOR



The Digital
Neighborhood

PLATINUM SPONSOR



Microsoft

GOLD SPONSORS



SILVER SPONSORS



BRONZE SPONSORS



Thomas van der Waard, NL

Power Platform Consultant 

Hot Air Ballooning 



t.van.der.waard@indocs.nl



[/in/thomasvanderwaard](https://www.linkedin.com/in/thomasvanderwaard)




[@_PowerThomas](https://twitter.com/_PowerThomas)




[PowerThomas.com](https://www.PowerThomas.com)

Agenda


 Introduction and the Basics

 Getting Started (**Demo**)

 Important Considerations

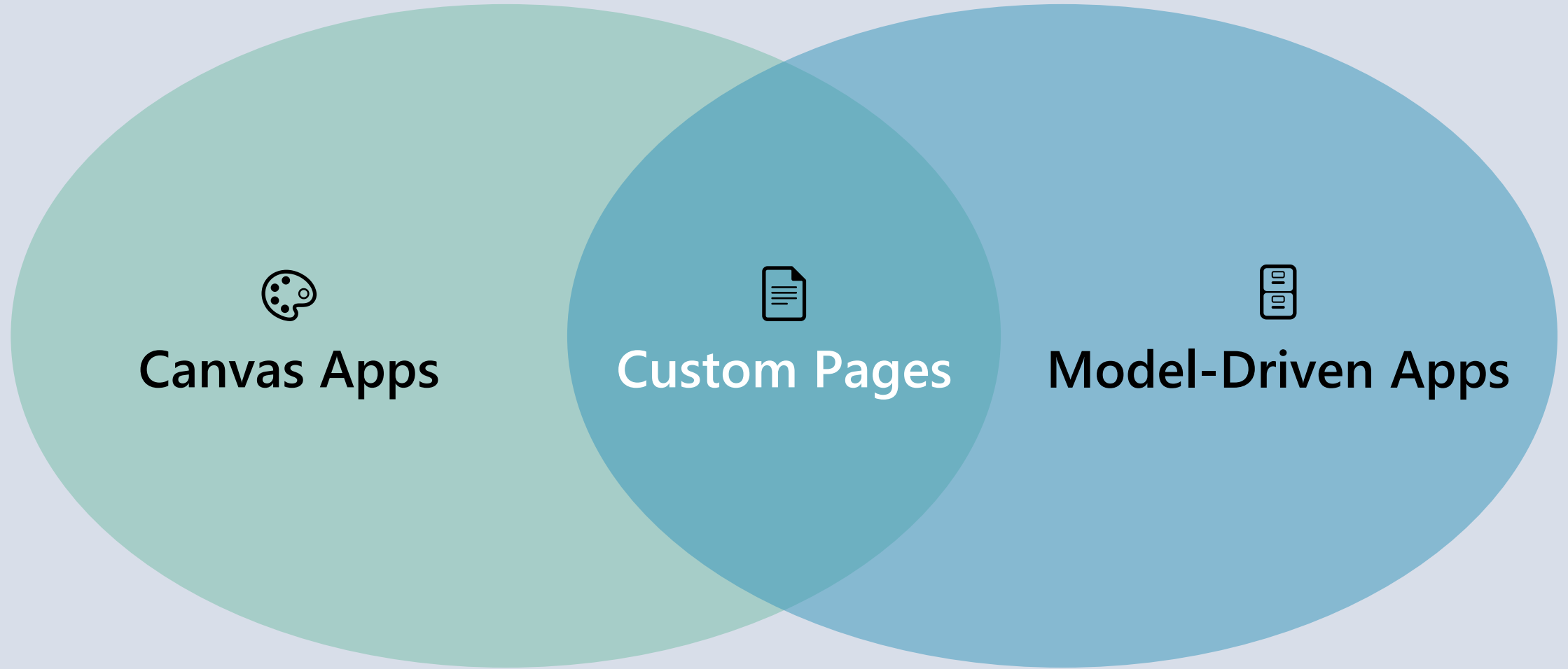
 Tips and Best Practices

 Advanced Features (**Demo**)

 Q&A and Wrap-Up

The What and the Why

Power Apps



The difference with Canvas Apps

Canvas Apps	Custom Pages
Stand-alone	Model-driven app component (Page)
More controls	Limited controls
Classic controls New Fluent UI controls (modern controls)	Classic controls can be activated separately Outdated Fluent UI controls (same as in D4T)
Navigating between screens within the app	Navigating to other MDA components
As an embedded app on model-driven forms	As a page or dialog within the app
Explicit app sharing	No need to share separately, goes through the MDA

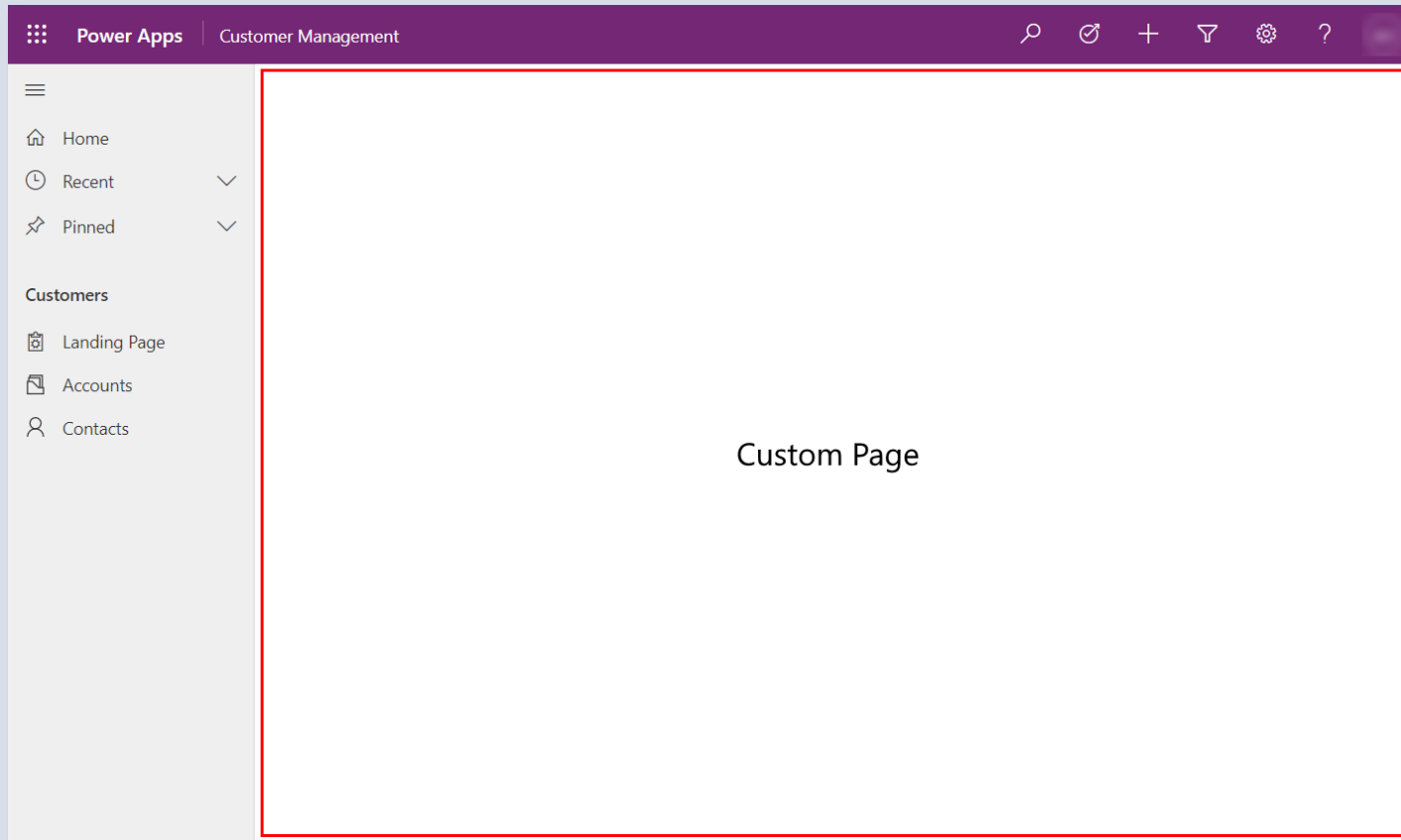
Why use Custom Pages?

- Going beyond out-of-the-box functionality
- More low-code, less pro-code
- Multiple tables on a single screen
- Connect with data sources other than Dataverse

Is this new?

No, it's not... GA since Nov '21

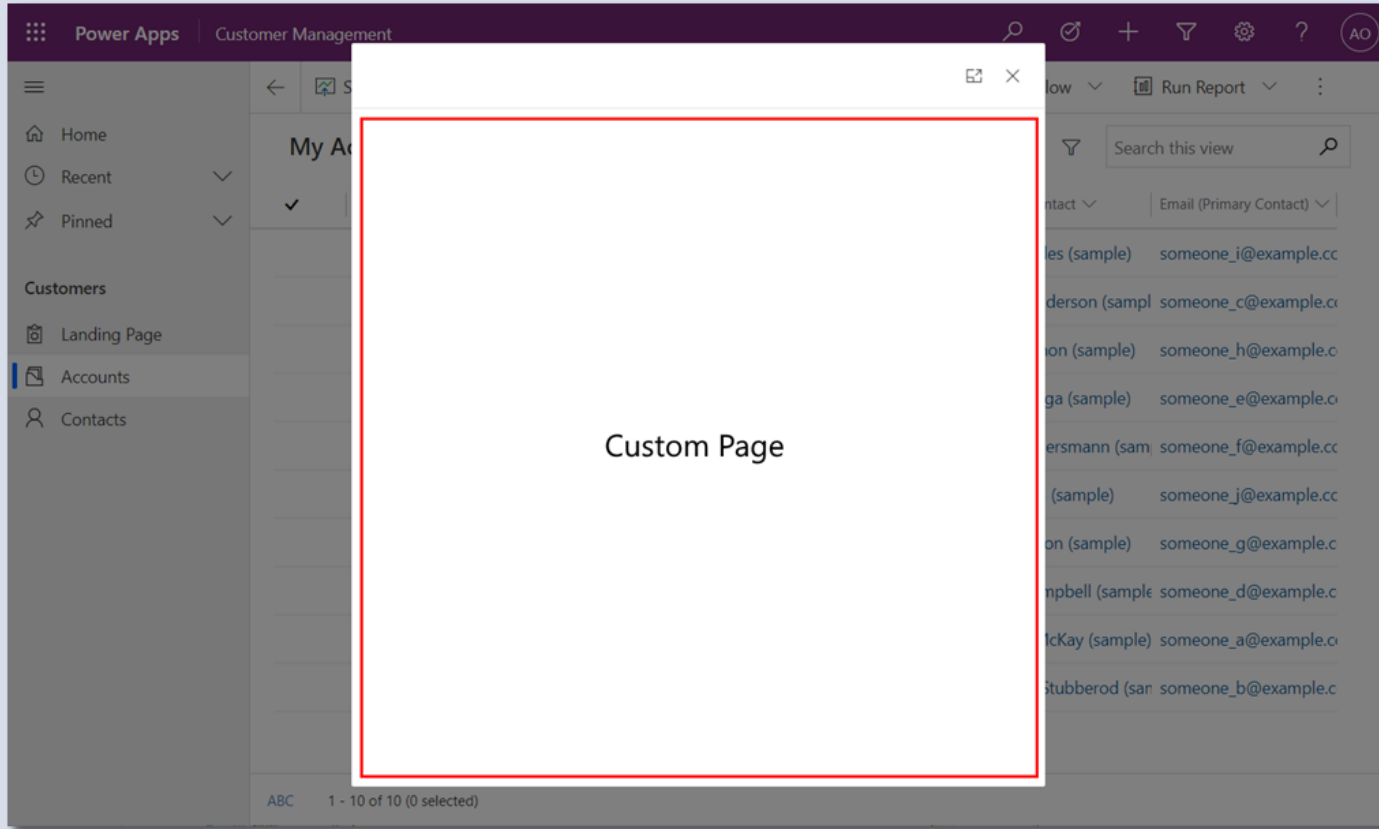
The Basics



Pages Types

Full Page

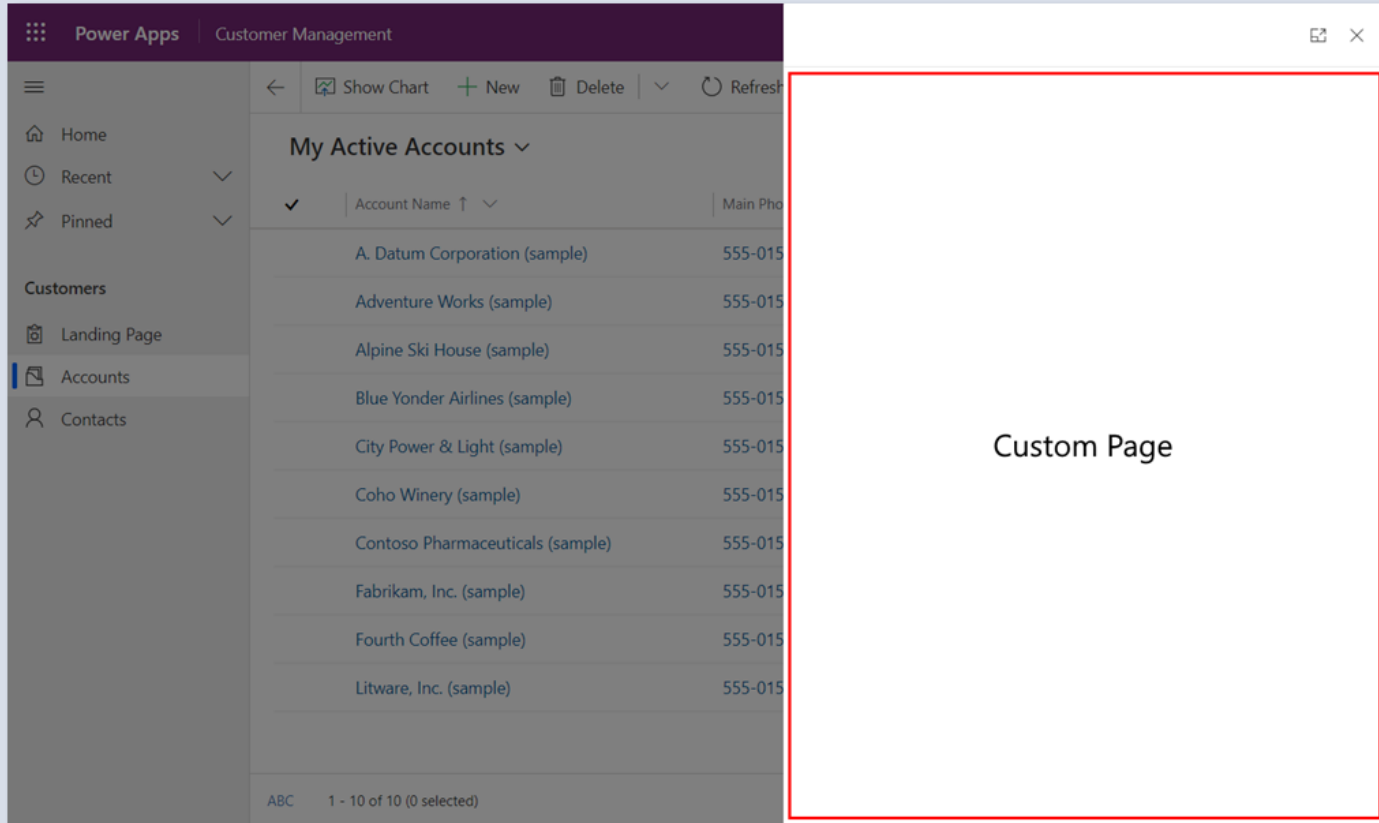
- No code required
- Can be added via the model-driven app designer
- Visible in the navigation menu (sitemap)



Pages Types

Center dialog (pop-up)

- Can be opened with a JS function
- Height and width adjustable (px or %)
- Title optional



Pages Types

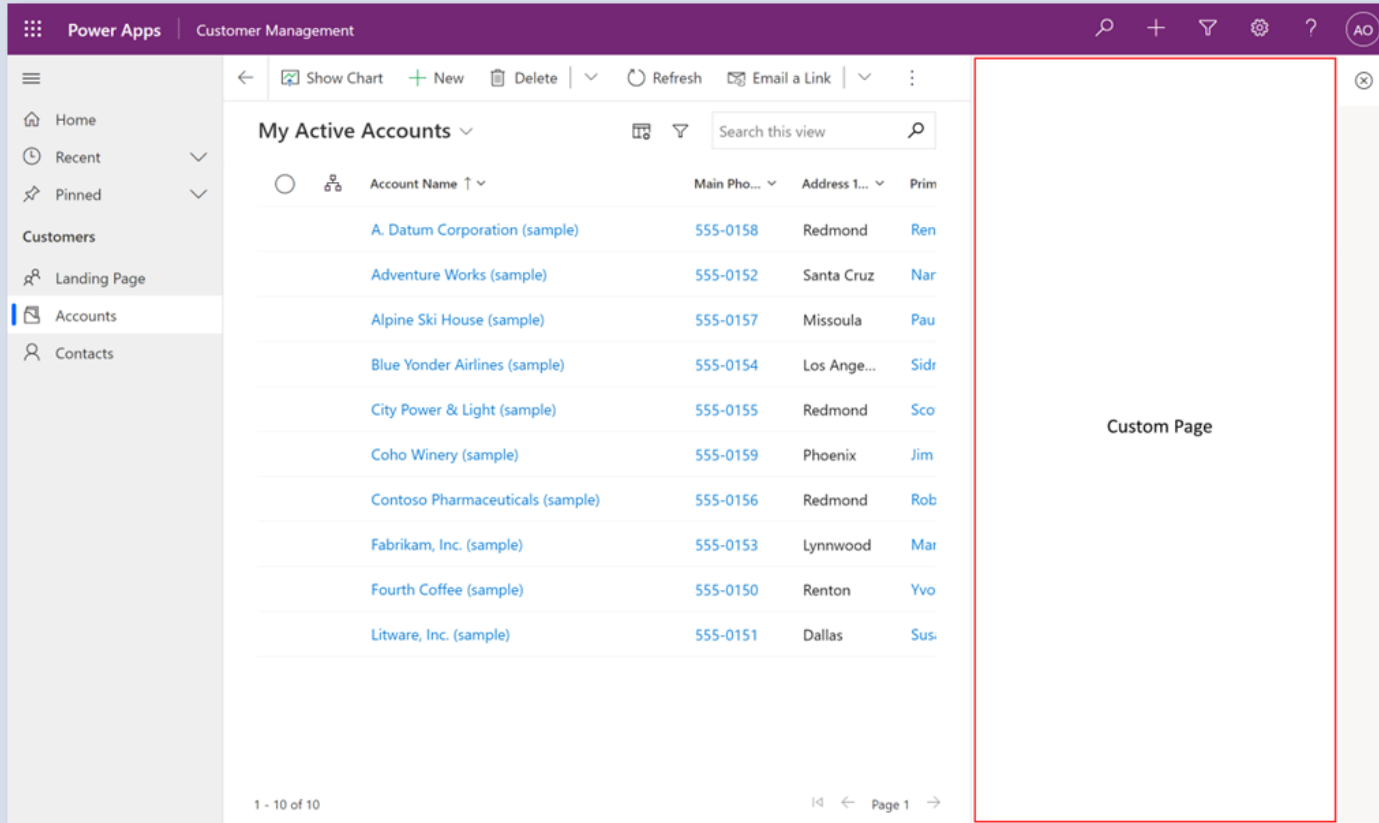
Side bar

- Can be opened with a JS function
- Width adjustable (px or %)
- Title optional

Pages Types

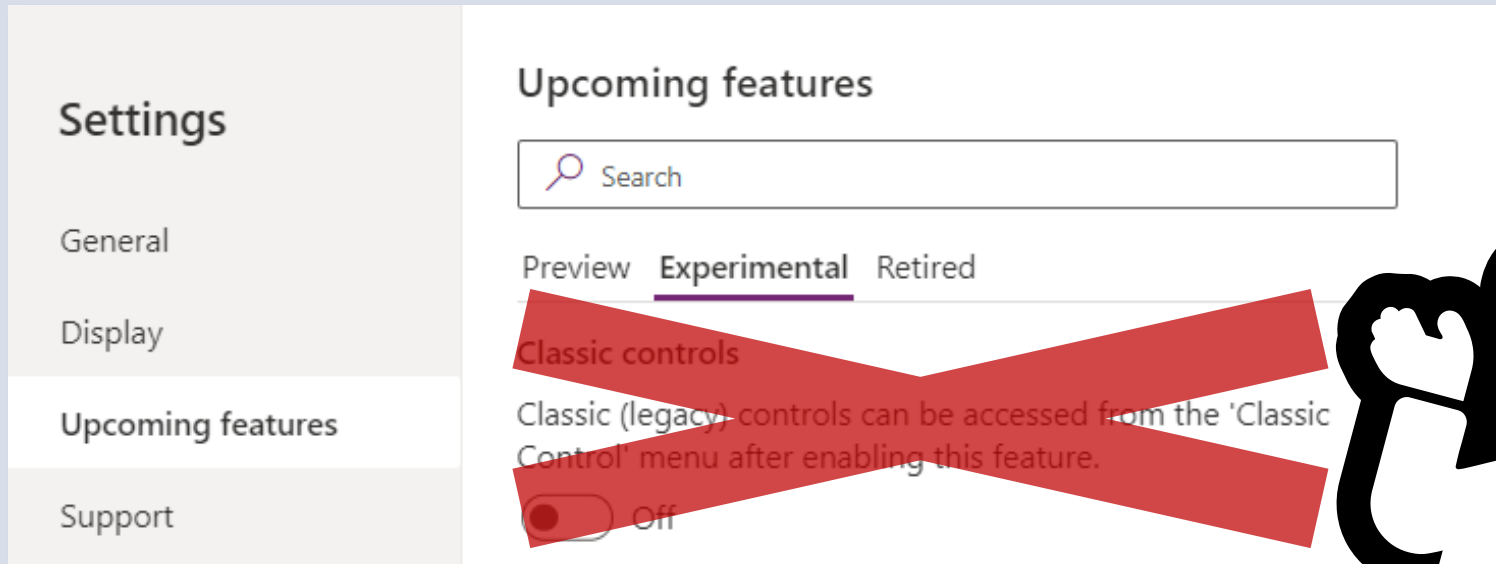
Side pane

- JS Function
- Persistent Across Pages
 - e.g., Webshop Cart
- Badge Icon Feature
- User-Closable Tabs

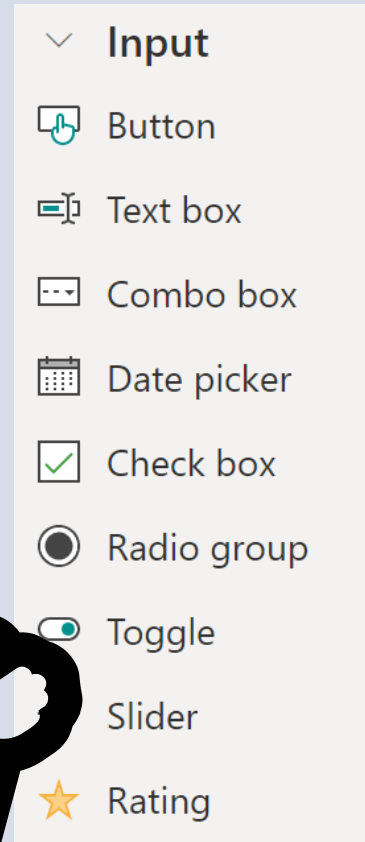


Available Controls

- Fluent UI styled (PCF)
 - First introduced in D4T
- Classic controls feature available under Settings



Fluent UI



- New Modern Controls (Preview)

Unavailable Controls

- Pen input
- Dropdown
- List box
- PDF viewer
- Bing Maps controls
- AI Builder controls
- Mixed Reality controls

A screenshot of a software interface showing a list of unavailable controls. The controls are organized into two main sections: 'AI Builder' and 'Mixed Reality'. Each control is represented by a small icon followed by its name. The 'AI Builder' section includes: Pen input (pen icon), Drop down (dropdown arrow icon), List box (list icon), PDF viewer (experimental) (PDF icon), Business card reader (business card icon), Receipt processor (receipt icon), Form processor (form icon), Object detector (object icon), and Text recognizer (text icon). The 'Mixed Reality' section includes: Address input (address icon), Map (map icon), View in MR (MR icon), View shape in MR (shape icon), Measure in MR (measure icon), and Markup in MR (markup icon).

- Pen input
- Drop down
- List box
- PDF viewer (experimental)
- Address input
- Map

AI Builder

- Business card reader
- Receipt processor
- Form processor
- Object detector
- Text recognizer

Mixed Reality

- View in MR
- View shape in MR
- Measure in MR
- Markup in MR

Navigation

Navigation

To a table's Default View:

```
Navigate([Dataverse table]);
```

```
Navigate(Accounts);
```

To a table's Specific View:

```
Navigate([Dataverse table views choice item]);
```

```
Navigate('Accounts (Views)'. 'My Active Accounts');
```

NOTE: Make sure to add the table as a datasource

Navigation

To a table's Default Form:

```
Navigate([Dataverse table record]);
```

```
Navigate(AccountGallery.Selected);
```

To a table's Specific Form:

```
Navigate([Dataverse table record], {Page: [table forms choice item]});
```

```
Navigate(  
    AccountGallery.Selected,  
    {Page: 'Accounts (Forms)'. 'Account for Interactive experience'}  
);
```

Navigation

To another Custom Page:

```
Navigate([Page Name]);
```

```
Navigate('Second Custom Page');
```

To another Custom Page with a record parameter:

```
Navigate([Dataverse table record], {Page: [Page Displayname]});
```

```
Navigate(  
    AccountGallery.Selected,  
    {Page: 'Second Custom Page'}  
);
```

Navigation

To a table's Form in Create mode:

```
Navigate(Defaults([Dataverse table]));
```

```
Navigate(Defaults(Accounts));
```

To a table's Form in Create mode with fields defaulted:

```
Navigate(Patch(Defaults([Dataverse table]),{[Create record]}));
```

```
Navigate(  
    Patch(  
        Defaults(Accounts),  
        {'Account Name': "My company", Phone: "555-3423" }  
    ); // Lookup columns don't seem to be passed  
);
```

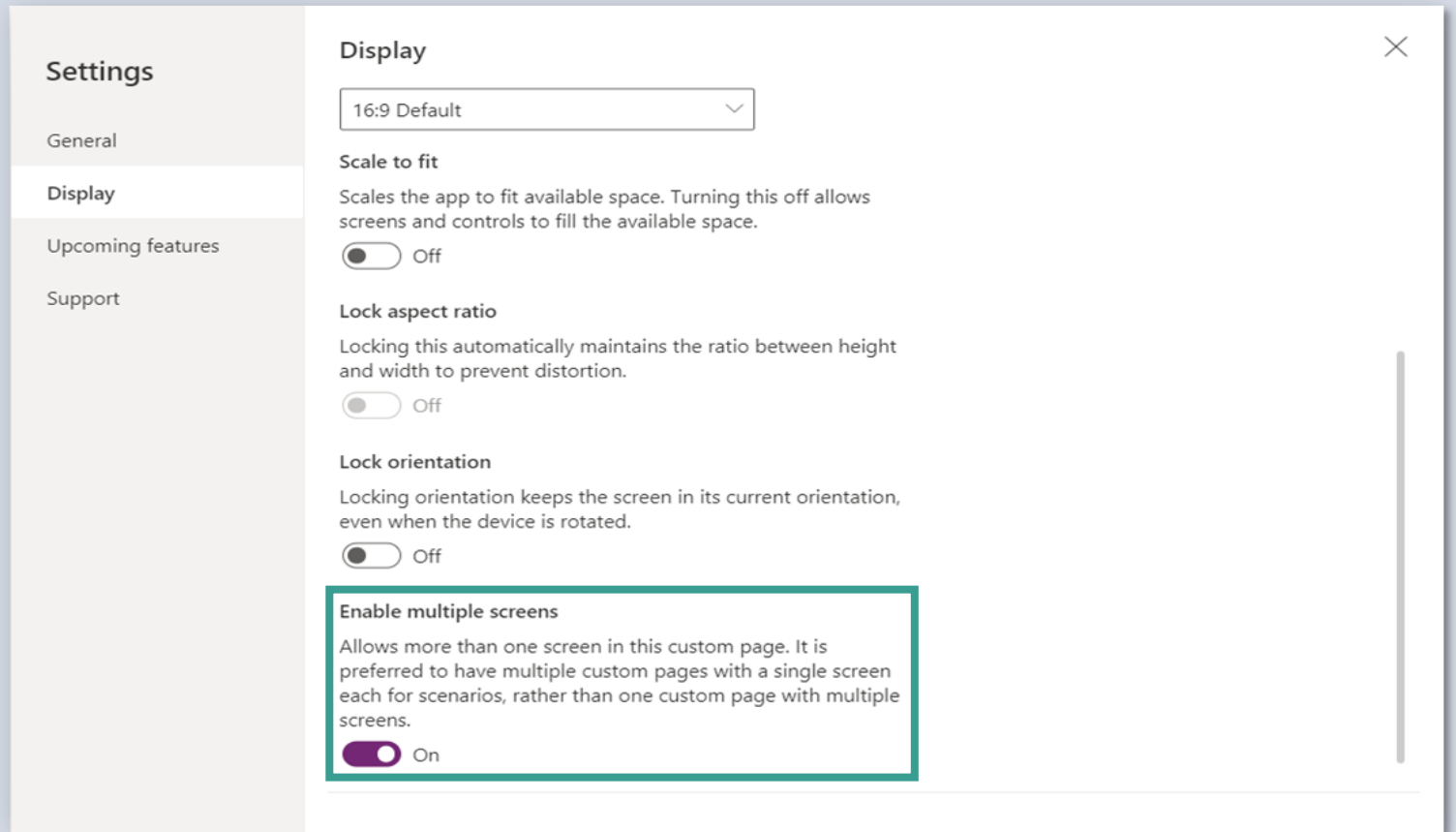
Navigation

To a previous **Model-driven page** or **Custom Page Screen** or **Close a Dialog**:

`Back();`

Multiple screens

- Single page by default
- Activate setting separately
- Not recommended:
 - This may affect the `Back()` function when you want to close dialogs.



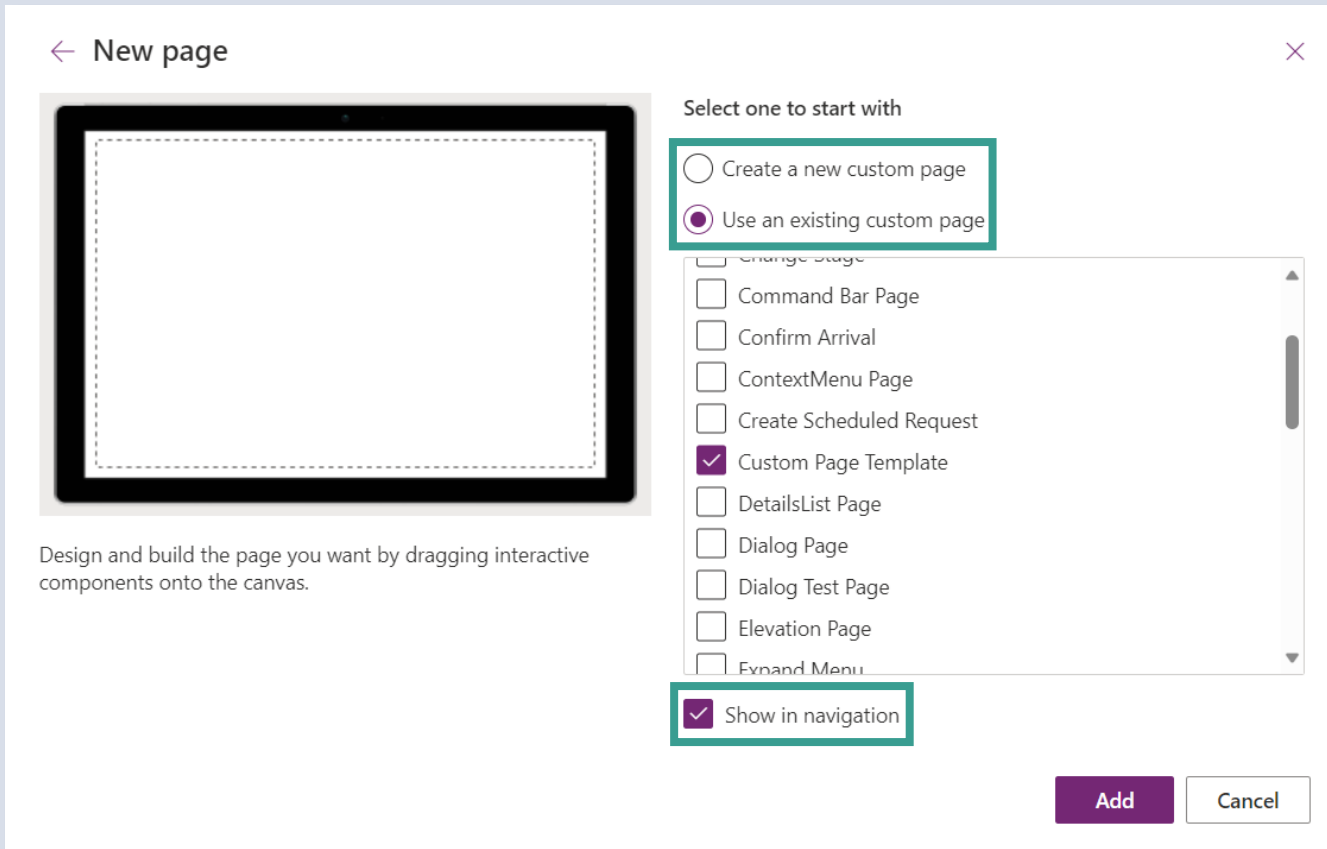
Let's Get Started!

DEMO: Creating a Full Page



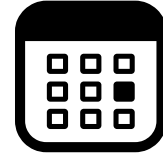
Recap

1. Created a Full Page from within the model-driven app designer
2. New or Existing
3. Show/Hide in Navigation



Important considerations

Limitations



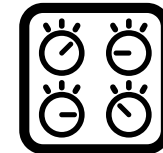
No current user
data formats



8-hr session vs
MDA longer
duration

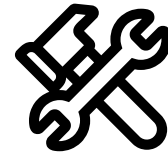
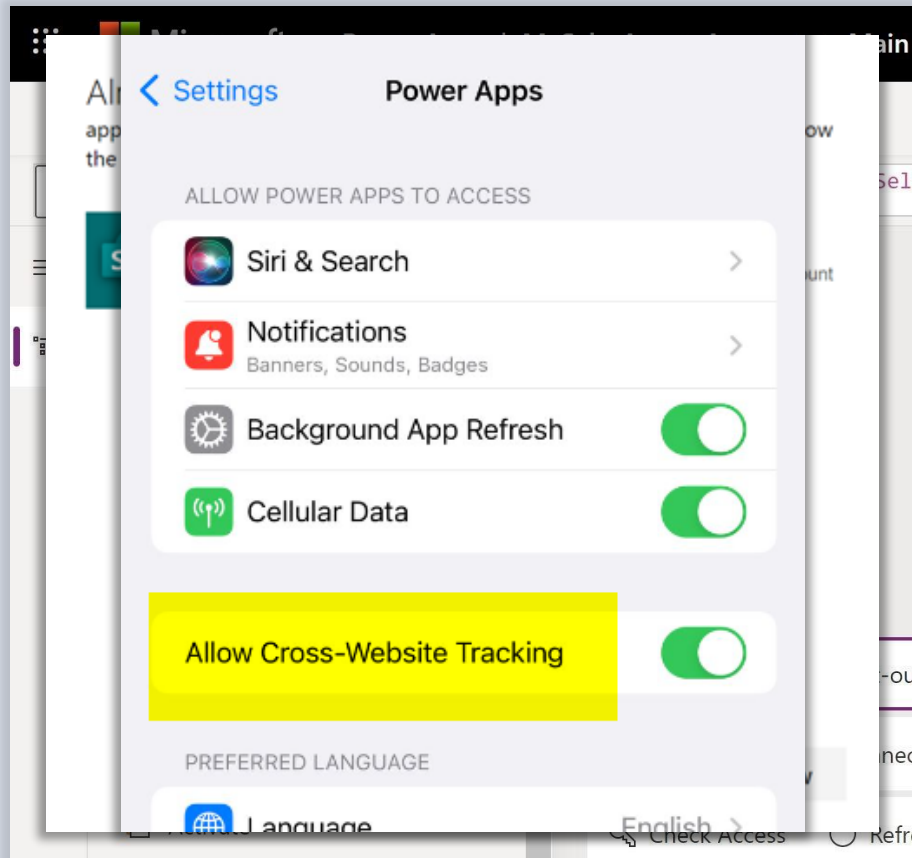


Must share for
multiple makers



Fewer Fluent UI
properties

Other considerations



Modern
Commanding



Evaluate other
MDA features first



Mobile in preview
& iOS cookies

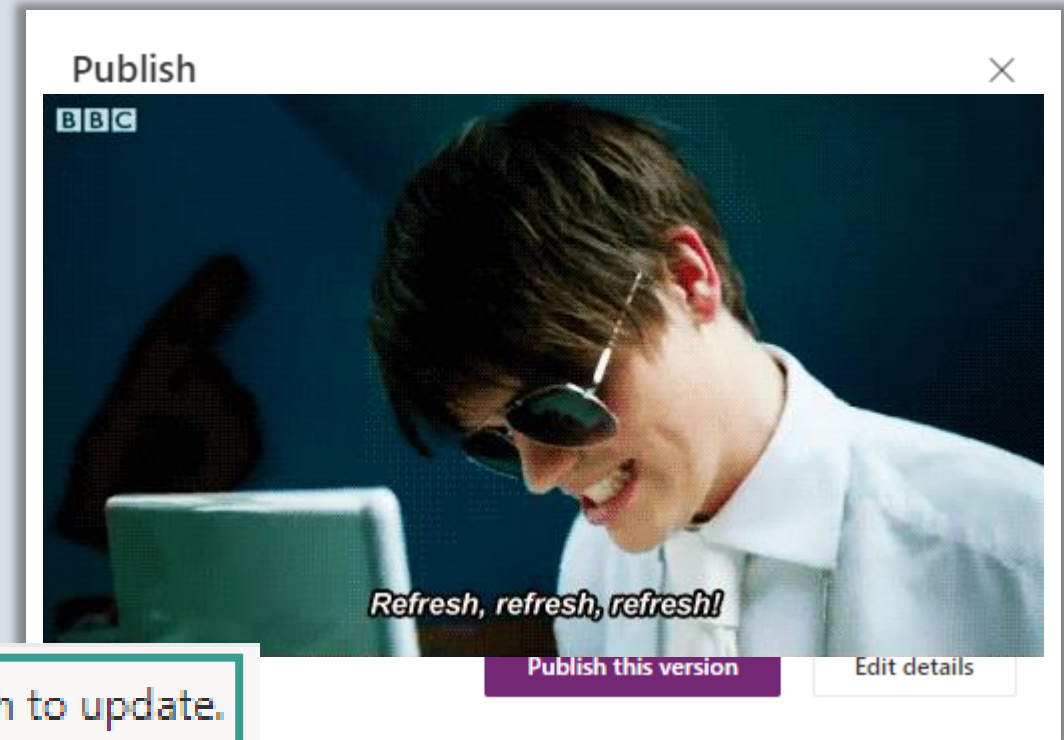


Universal data
source consent

Tips and Best practices

Publishing

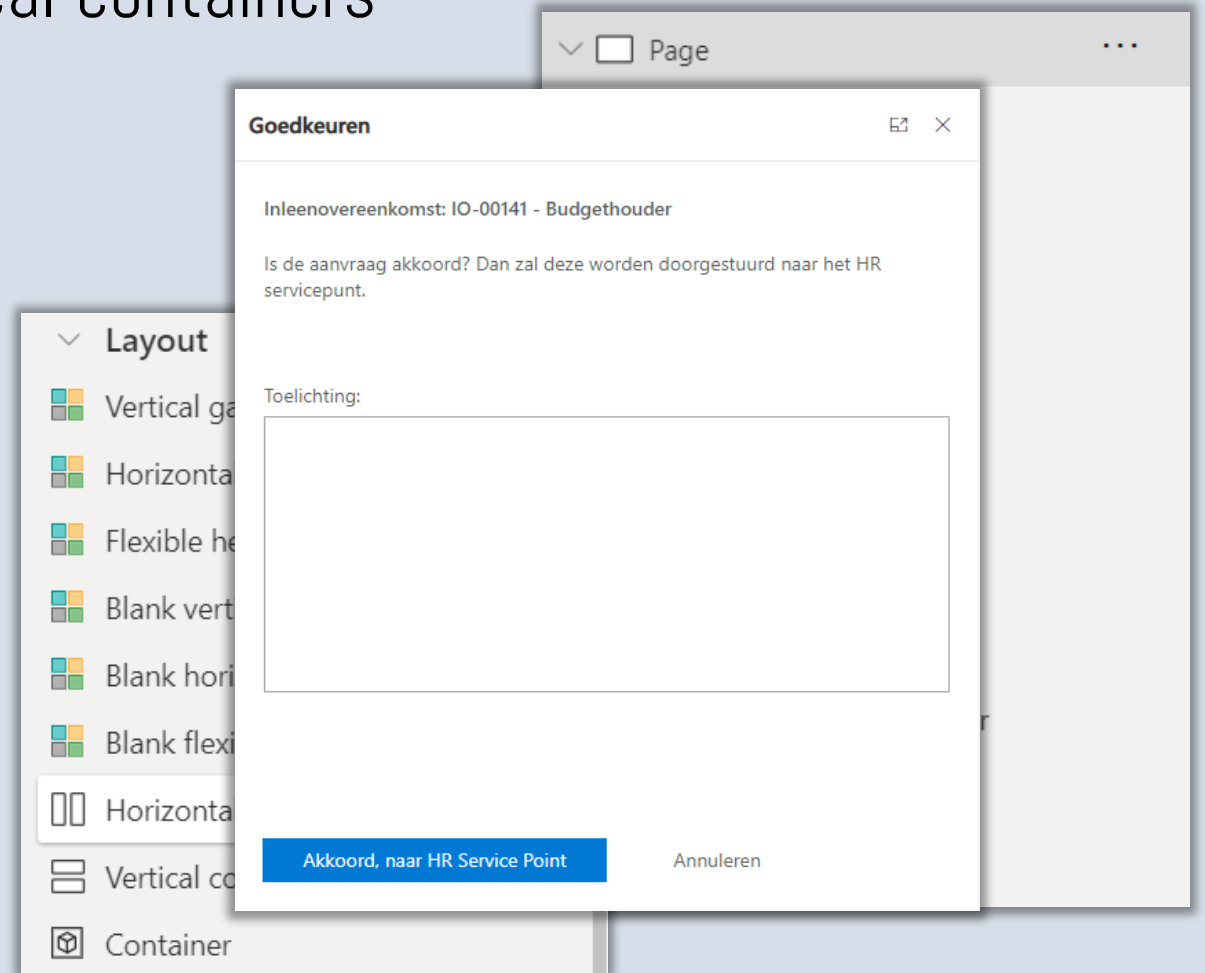
-  Save before first publish
-  No design-time publish response
-  Runtime new version notification







Responsive Layout

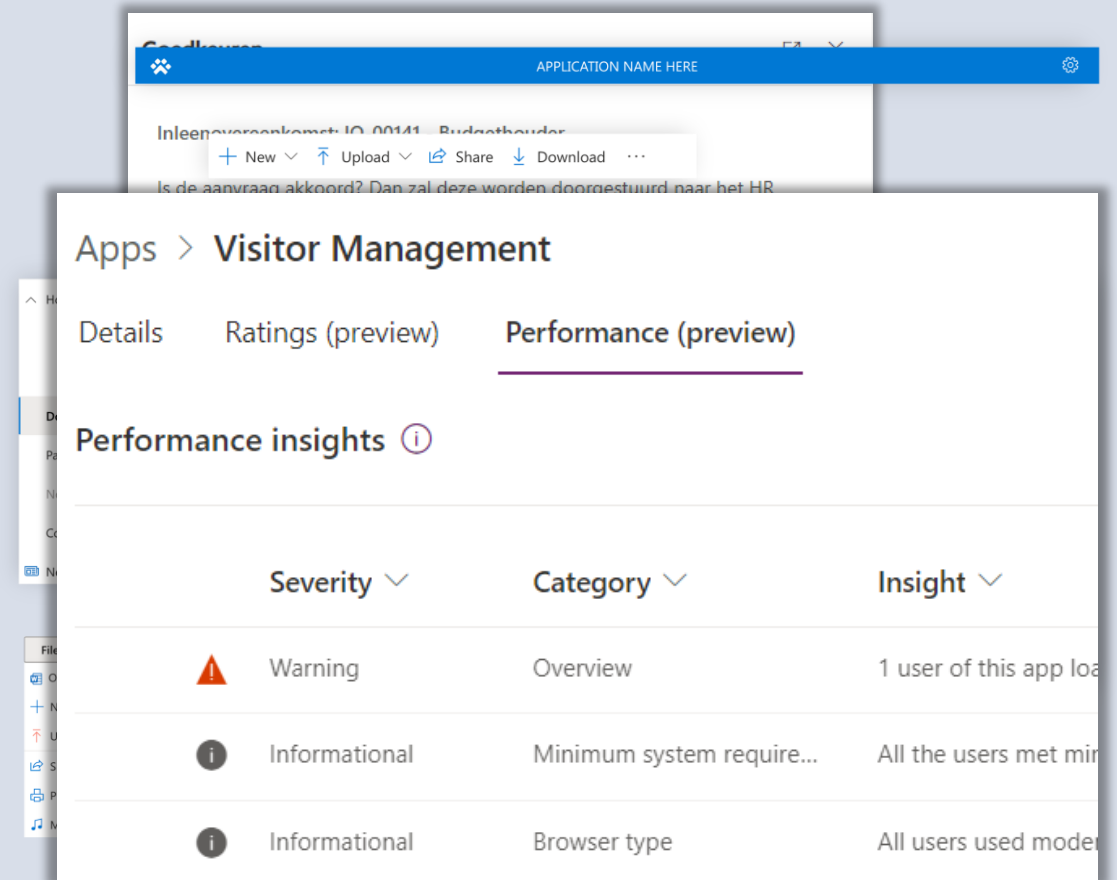
 Auto-layout with Horizontal/Vertical containers

 Consistent MDA styling



Performance

-  Light weight for performance
-  Limit complex components & controls
-  Use Creator Kit (or Modern Controls)
-  Try Performance Insights (Preview)

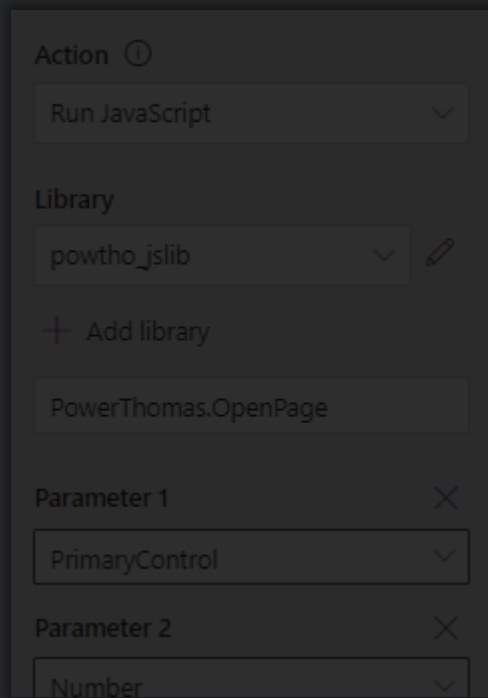


Severity	Category	Insight
Warning	Overview	1 user of this app loa
Informational	Minimum system require...	All the users met mir
Informational	Browser type	All users used moder

Command Bar (Ribbon)

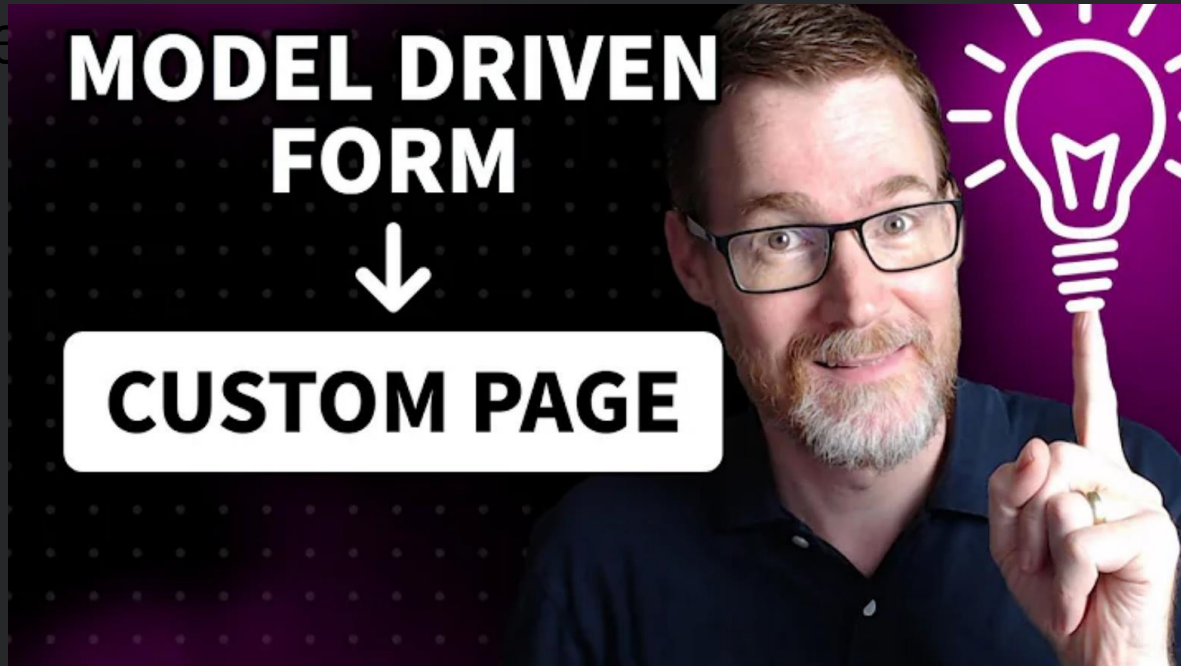
Modern commanding:

Open a custom page



Ribbon Workbench:

Open with Custom Page

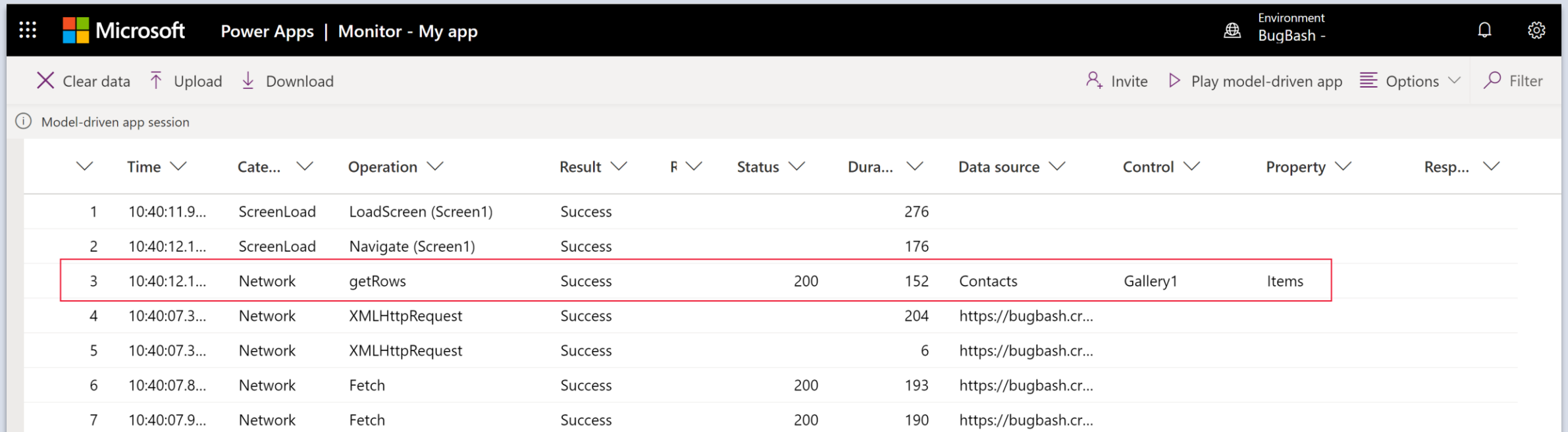


Scott Durow



Monitor is your best friend!

Use Monitor for real-time activity logging & debugging



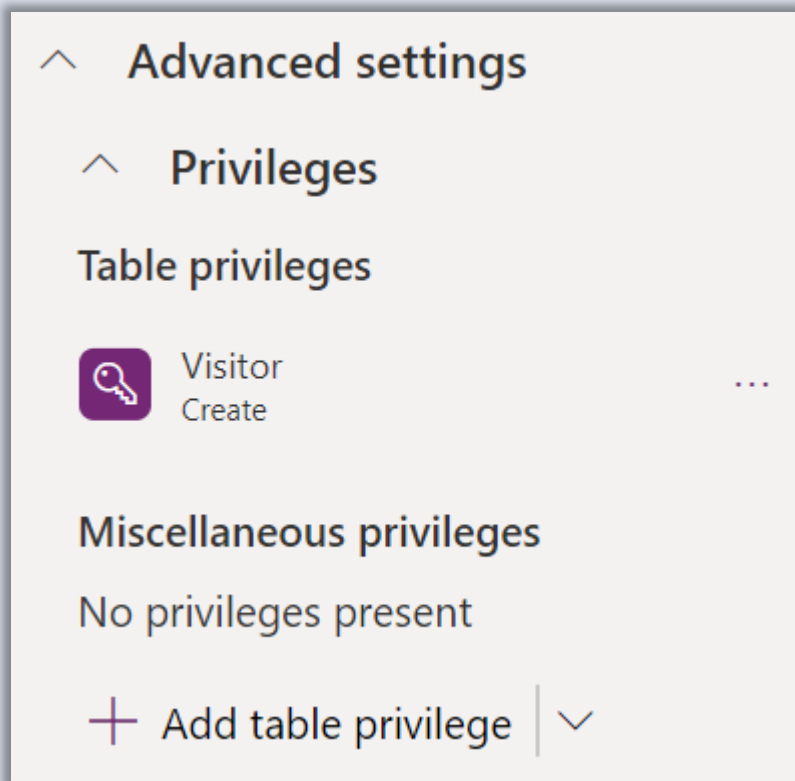
The screenshot shows the Microsoft Power Apps Monitor interface. At the top, it displays the Microsoft logo, 'Power Apps | Monitor - My app', and the environment 'BugBash'. Below the header, there are action buttons for 'Clear data', 'Upload', and 'Download'. On the right, there are buttons for 'Invite', 'Play model-driven app', 'Options', and 'Filter'. The main content area shows a table of activity logs for a 'Model-driven app session'. The table has columns for Time, Category, Operation, Result, R, Status, Duration, Data source, Control, Property, and Response. Row 3 is highlighted with a red border, showing a 'Network' operation 'getRows' that was successful, returning 200 status and 152 items from the 'Contacts' data source in the 'Gallery1' control.

	Time	Cate...	Operation	Result	R	Status	Dura...	Data source	Control	Property	Resp...
1	10:40:11.9...	ScreenLoad	LoadScreen (Screen1)	Success			276				
2	10:40:12.1...	ScreenLoad	Navigate (Screen1)	Success			176				
3	10:40:12.1...	Network	getRows	Success		200	152	Contacts	Gallery1	Items	
4	10:40:07.3...	Network	XMLHttpRequest	Success			204	https://bugbash.cr...			
5	10:40:07.3...	Network	XMLHttpRequest	Success			6	https://bugbash.cr...			
6	10:40:07.8...	Network	Fetch	Success		200	193	https://bugbash.cr...			
7	10:40:07.9...	Network	Fetch	Success		200	190	https://bugbash.cr...			

Advanced Features

Security

Sitemap visibility



Advanced settings

Privileges

Table privileges

Visitor
Create

Miscellaneous privileges

No privileges present

+ Add table privilege | v

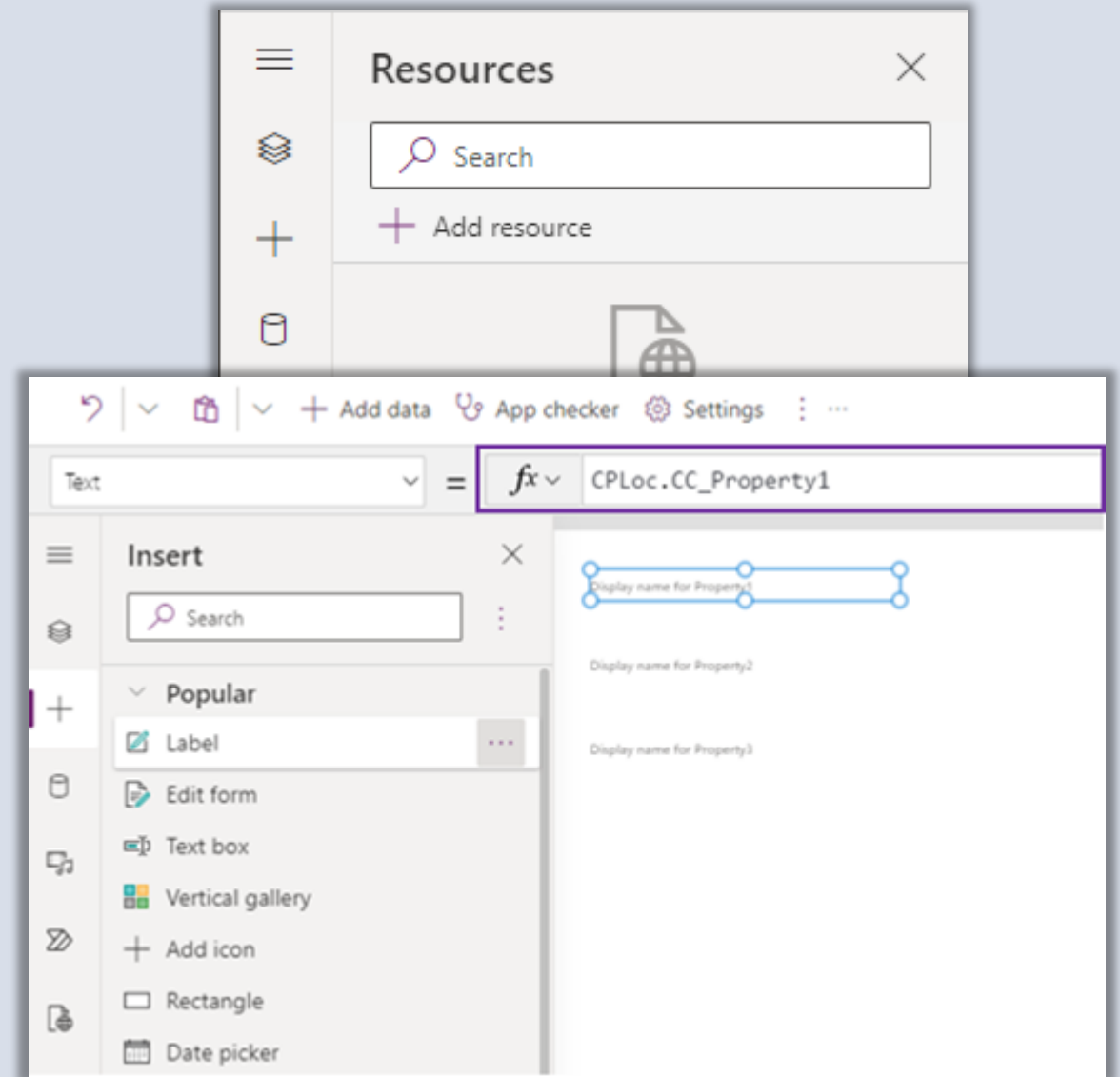
PowerFx permission checks

`DataSourceInfo()`

`RecordInfo()`


Localization

- 📁 Add **RESX** in Resources Tab
- 🌐 Set Localized Values via Expression




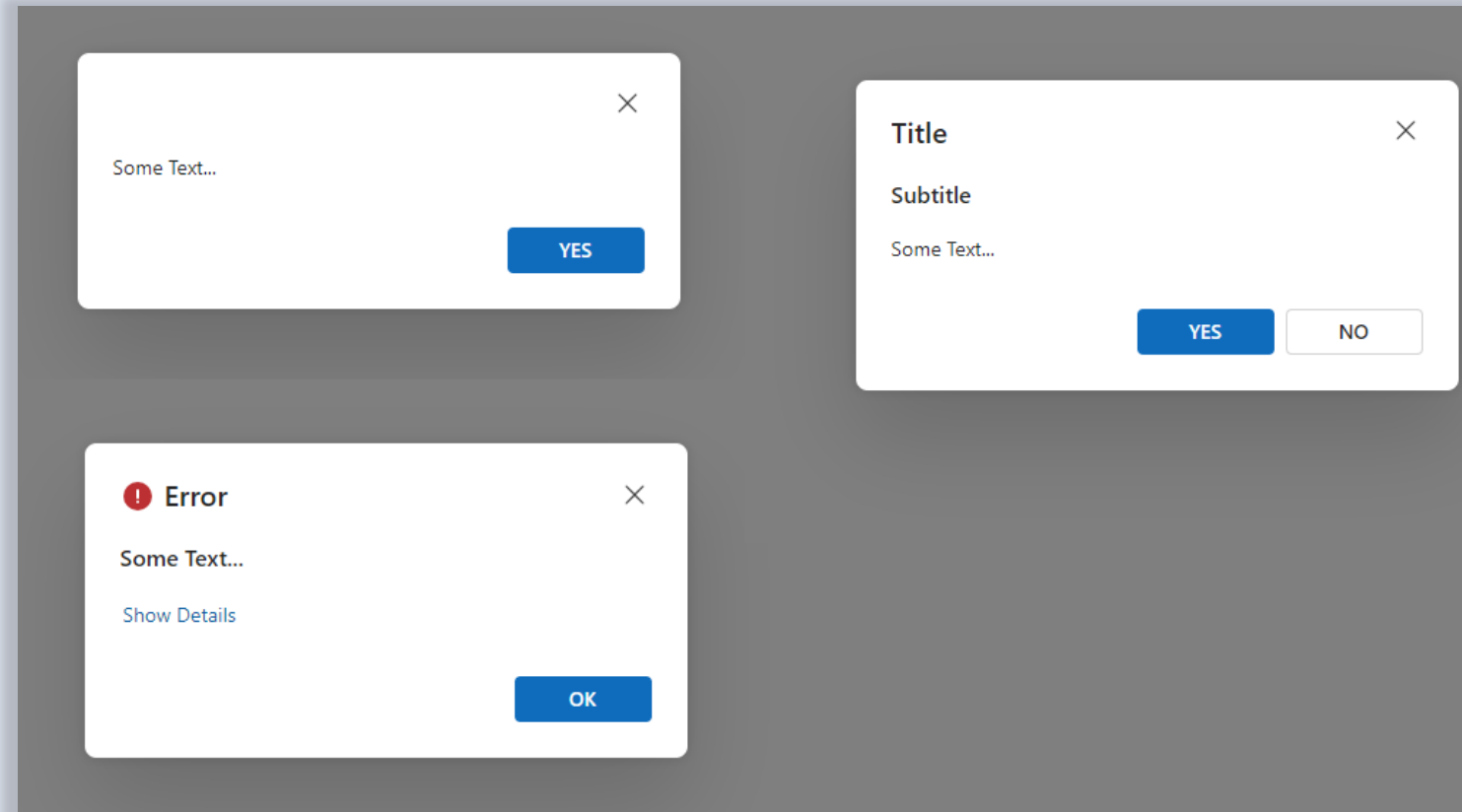
Canvas Components & Code Components

 Canvas Components via Library, not Direct

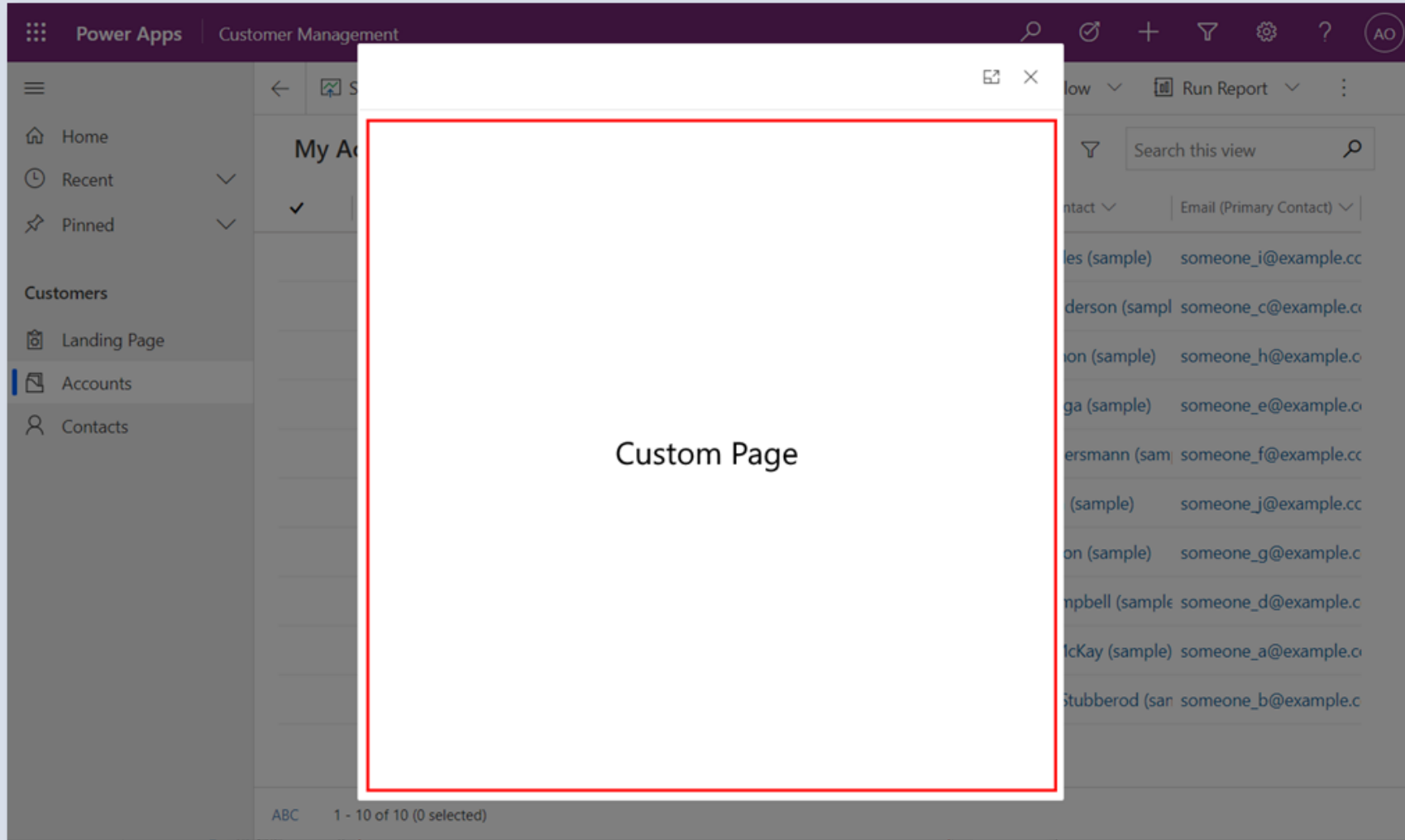
 Activate 'Modern Controls' in Library

 Code Components are supported

 Use PCF APIs: Navigation & WebAPI



Open a Custom Page as Dialog/Side bar



JavaScript

XRM Client API

 Use Examples from MS Learn

 Or try [my library from Github](#)

Action ⓘ

Run JavaScript

Library

powtho_jslib

+ Add library

PowerThomas.OpenPage

Parameter 1

PrimaryControl

Parameter 2

Number

Demo time!

Creating a Dialog with JS



Action ⓘ

Run JavaScript

Library

powtho_jslib

+ Add library

PowerThomas.OpenPage

Parameter 1 ✕

PrimaryControl

Parameter 2 ✕

Number

RECAP: OpenPage parameters

- ExecutionContext
- Title
- Logical name: *Custom page* and *Table*
- Target: *Full Page* or *Dialog*
- Position: *Centered* or *Sidebar*
- Dimensions: *Width* and *Height*
- Refreshing form when page is closed
- Return to default view when page is closed (like *'Save & Close'*)



nordic.
summit

Q&A



nordic.
summit

THANK YOU!