

# Building Data-driven Canvas Apps



# Agenda

- Screen Design Guidelines
- Displaying Repeating Data using Galleries
- Using Table Functions to Filter and Sort Data
- Updating Data using Edit Forms and Data Cards
- Customizing SharePoint List Forms



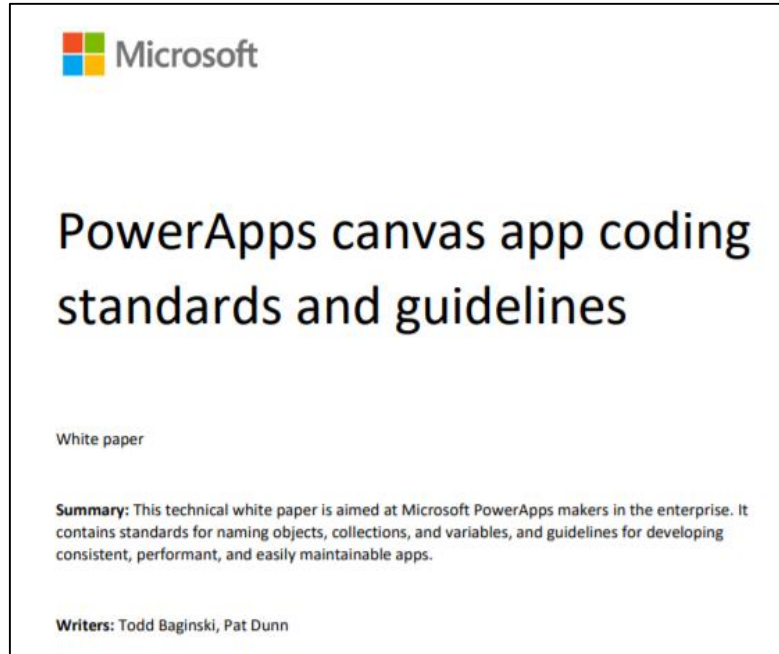
# Challenges with Building Canvas Apps

- Who is the typical PowerApps application maker?
  - A developer without a software development background
- What is the maker is responsible for?
  - designing, building, testing, deploying
- What are the challenges in deploying canvas apps?
  - Building projects that are easy to maintain and extend
  - Building consistency across team members
  - Building canvas apps with better performance



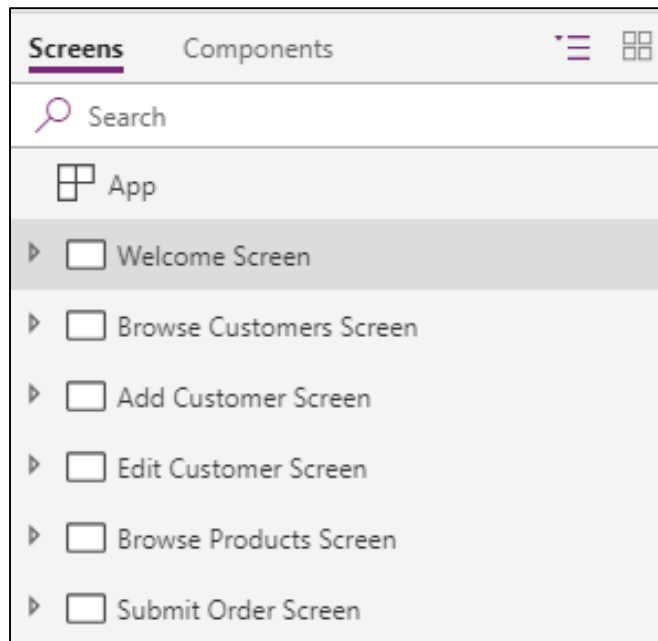
# Read the Canvas Apps Whitepaper

- PowerApps canvas app coding standards and guidelines  
<https://powerapps.microsoft.com/en-us/blog/powerapps-canvas-app-coding-standards-and-guidelines/>
- As a builder, you should strive for...
  - Simplicity, Readability , Performance, Supportability, Accessibility
  - Ease of deployment and administration



# Creating Screen Names

- It's import to create screen names correctly
  - Screen names will be read aloud by screen readers
  - Names should include spaces and avoid abbreviations
  - Screen name should end with the word "Screen"
  - Screen name should reflect purpose of screen



# Control Naming Convention

- Define and document a naming convention
  - You can use this list as a starting point

button	btn
camera control	cam
canvas	can
card	crd
collection	col
combo box	cmb
date picker	dte
drop down	drp
radio button	rad
form	frm
gallery	gal

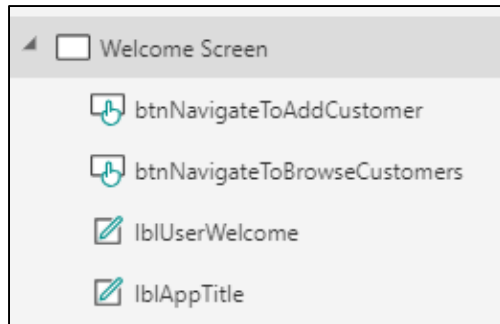
group	grp
header page shape	fdr
html text	html
icon	ico
image	img
label	lbl
page section shape	sec
shapes (rectangle, etc.)	shp
table data	tbl
text input	txt
timer	tim



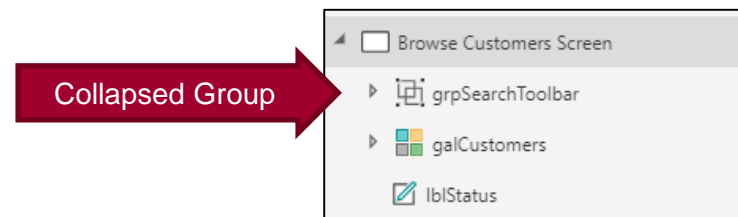
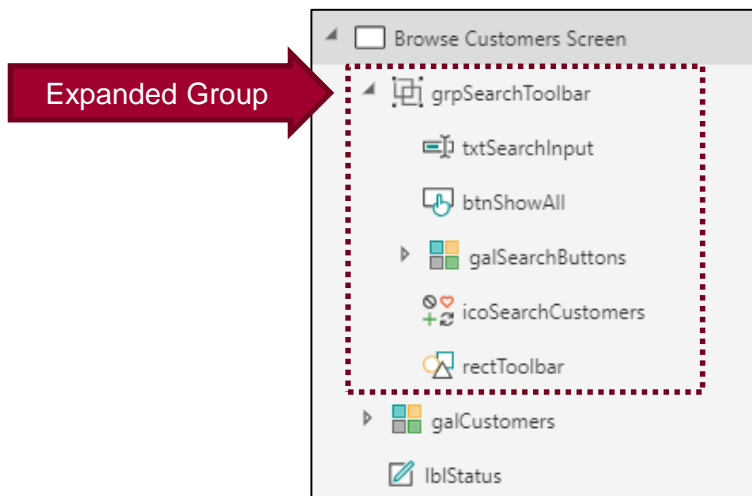


# Keeping Screens Maintainable

- Each control name should be easy to read & understand

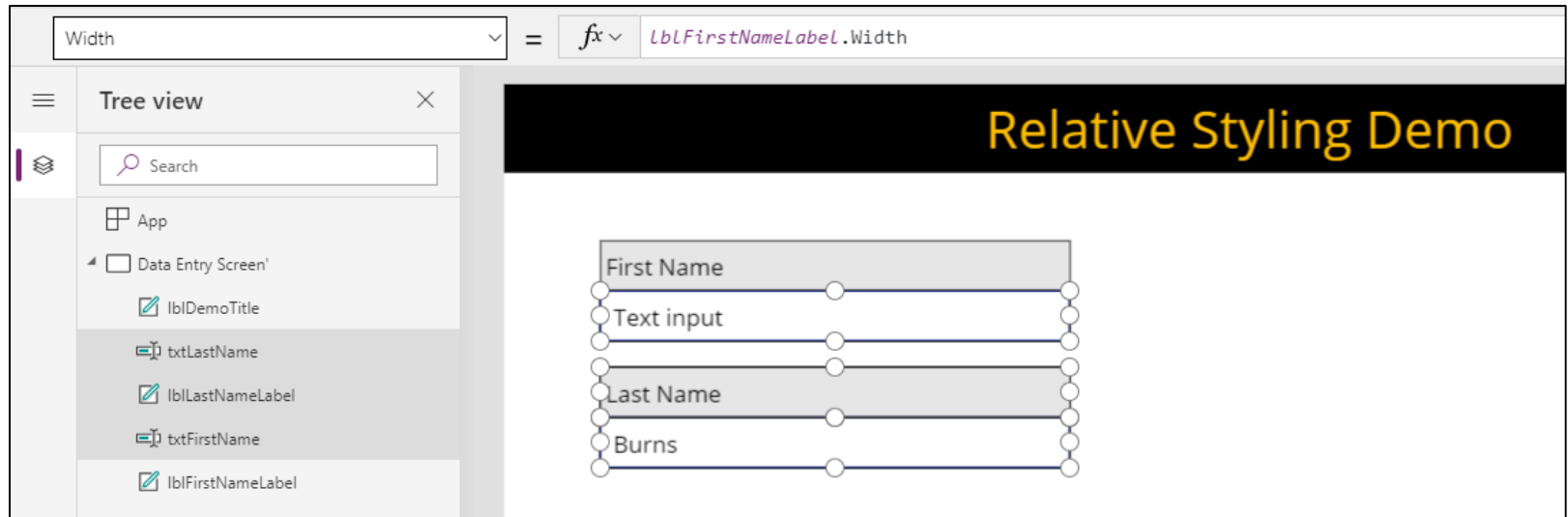


- Use grouping to isolate set of controls with single purpose



# Relative Styling

- Calculate control property values from other controls
  - Common to use properties like X, Y, Width, Height, Size, Fill, etc.





# Agenda

- ✓ Screen Design Guidelines
- Displaying Repeating Data using Galleries
  - Using Table Functions to Filter and Sort Data
  - Updating Data using Edit Forms and Data Cards
  - Customizing SharePoint List Forms



# Displaying Repeating Items using Galleries









- Galleries are used to display repeating items
  - A list of customers




 <b>Lionel Witt</b>    email: Lionel.Witt@SoylentCorporation.com phone: 1(305)888-6666	 <b>Casandra Vega</b>    email: Casandra.Vega@TailSpinToys.com phone: 1(480)444-8888	 <b>Jude Melton</b>    email: Jude.Melton@JupiterMiningCorp.com phone: 1(707)222-5555
 <b>Suzette Fernandez</b>    email: Suzette.Fernandez@GraysonSkyDomes.com phone: 1(404)333-1111	 <b>Dee Santana</b>    email: Dee.Santana@TradeFederation.com phone: 1(541)555-6666	 <b>Corrine Ortega</b>    email: Corrine.Ortega@TyrellCorporation.com phone: 1(210)333-5555
 <b>Gaston Knight</b>    email: Gaston.Knight@LucernePublishing.com phone: 1(801)333-3333	 <b>Elvis Knight</b>    email: Elvis.Knight@DoublemeatPalace.com phone: 1(505)111-2222	

- A set of command buttons



- A products list and a shopping cart

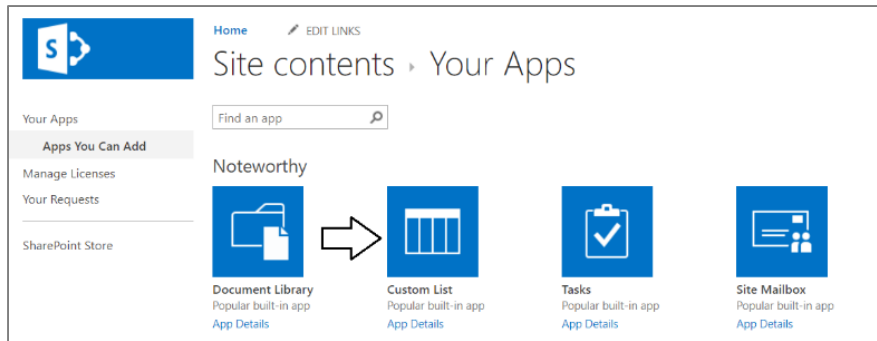
	<b>Batman Action Figure</b> A super hero who sometimes plays the role of a dark knight.	Quantity 1		ADD TO CART
	<b>Captain America Action Figure</b> A super action figure that protects freedom and the American way of life.	Quantity 1		ADD TO CART
	<b>GI Joe Action Figure</b> A classic action figure from the 1970s.	Quantity 1		ADD TO CART
	<b>Green Hulk Action Figure</b> An overly muscular action figure that strips naked when angry.	Quantity 2		ADD TO CART

Shopping Cart for Lionel Witt				
No	Product	List Price	Total	
1	Batman Action Figure	\$14.95	\$14.95	
2	Green Hulk Action Figure	\$9.95	\$19.90	
10	Twitter Follower Action Figure	\$1.00	\$10.00	



# Creating Lists in SharePoint Online

- PowerApps works best with modern lists
  - Best to create lists from the Custom list type



- Add site columns to track required data

Customers			
Last Name ▾	First Name ▾	Company ▾	E-Mail ▾
Smith	John	Some Company	SomeEmail@SomeDomain.com
Doe	Jane	Doe Consulting	janedoe@geemail.com



# Creating galCustomers

- Create a gallery to display customers

 <b>Lionel Witt</b>    email: Lionel.Witt@SoylentCorporation.com phone: 1(305)888-6666	 <b>Casandra Vega</b>    email: Casandra.Vega@TailSpinToys.com phone: 1(480)444-8888	 <b>Jude Melton</b>    email: Jude.Melton@JupiterMiningCorp.com phone: 1(707)222-5555
 <b>Suzette Fernandez</b>    email: Suzette.Fernandez@GraysonSkyDomes.com phone: 1(404)333-1111	 <b>Dee Santana</b>    email: Dee.Santana@TradeFederation.com phone: 1(541)555-6666	 <b>Corrine Ortega</b>    email: Corrine.Ortega@TyrellCorporation.com phone: 1(210)333-5555
 <b>Gaston Knight</b>    email: Gaston.Knight@LucernePublishing.com phone: 1(801)333-3333	 <b>Elvis Knight</b>    email: Elvis.Knight@DoublemeatPalace.com phone: 1(505)111-2222	



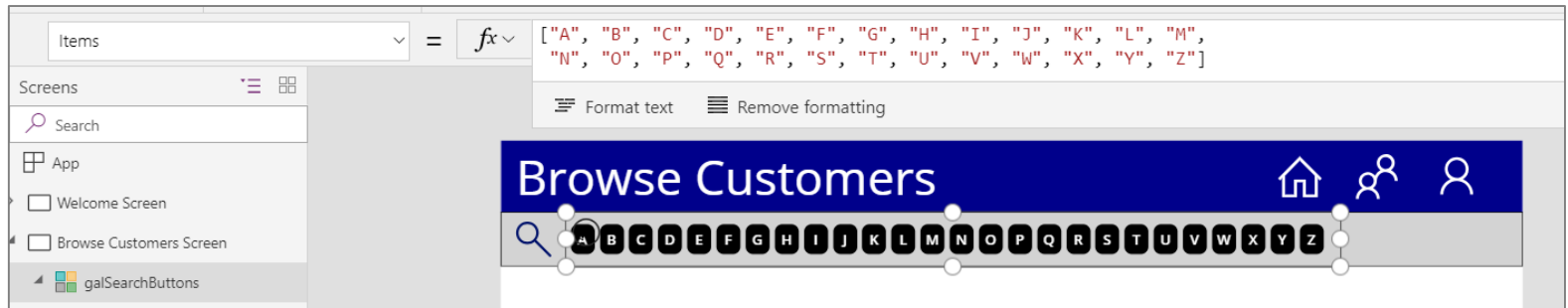
# Agenda

- ✓ Screen Design Guidelines
- ✓ Displaying Repeating Data using Galleries
- Using Table Functions to Filter and Sort Data
  - Updating Data using Edit Forms and Data Cards
  - Customizing SharePoint List Forms



# Creating galSearchButtons

- Create a search toolbar
  - Assign table of letters to **Items** property of **galSearchCustomers**



- Use **ThisItem.Value** for **Text** property of button inside gallery template



# Table Functions

- Each table represented as a value
  - Table can be passed as argument
  - Table can be returned by function
- Functions that return tables
  - Filtering: `Filter`, `Search`
  - Sorting: `Sort`, `SortByColumns`
  - Shaping: `AddColumns`, `DropColumns`, `RenameColumns`
  - Grouping: `GroupBy`, `Ungroup`





# Creating a Search Filter

- Assign filter string to local context variable



- Configure expression for **Items** property to set filtering

```
If(  
    IsBlank(locCustomerFilter),  
    Sort(Customers, 'Last Name'),  
    Sort(Filter(Customers, StartsWith('Last Name', locCustomerFilter)), 'Last Name')  
)
```





**DEMO**

## Filtering Data in a Table

# Agenda

- ✓ Screen Design Guidelines
- ✓ Displaying Repeating Data using Galleries
- ✓ Using Table Functions to Filter and Sort Data
- Updating Data using Edit Forms and Data Cards
- Customizing SharePoint List Forms



# Working with Edit Forms and Data Cards

- Form acts as a container for data cards
  - Each form binds to a single record
  - Within a form, each data card binds to an underlying field
  - Each data card contains an encapsulated set of child controls

The screenshot displays a Visual Studio IDE window titled 'frmAddCustomer'. On the left, a 'Solution Explorer' pane shows a list of data cards and other form components:

- First Name\_DataCard1
- Last Name\_DataCard1
- Company\_DataCard1
- Email Address\_DataCard1
- Work Phone\_DataCard1
- Home Phone\_DataCard1
- Address\_DataCard1
- City\_DataCard1
- State\_DataCard1
- Zipcode\_DataCard1
- ☐ Edit Customer Screen
- ☐ Browse Products Screen
- ☐ Submit Order Screen
- ☐ Order Confirmation Screen

On the right, the main design area shows the layout of the 'frmAddCustomer' form. It features two columns of data cards, each with a text box and a label. The labels are preceded by an asterisk (\*), indicating required fields. The data cards are arranged in two columns:

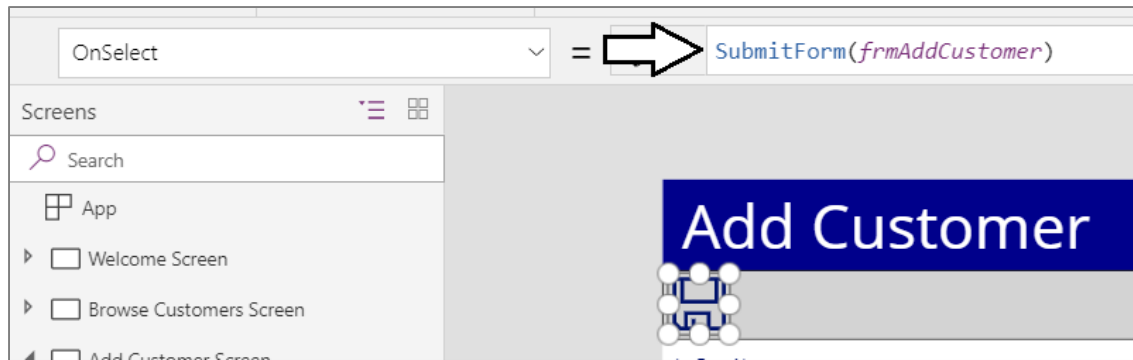
- Left Column:**
  - \* First Name
  - Company
  - Work Phone
  - Address
  - State
- Right Column:**
  - \* Last Name
  - \* Email Address
  - Home Phone
  - City
  - Zipcode





# Form Submission with Edit Forms

- Update with Edit Form triggered by calling **SubmitForm**



- Submit button should be disabled when form data is not valid



# Custom Form Validation with Edit Forms

- Data cards contain label control for error message



- Set **Text** property of label to display error message



# Agenda

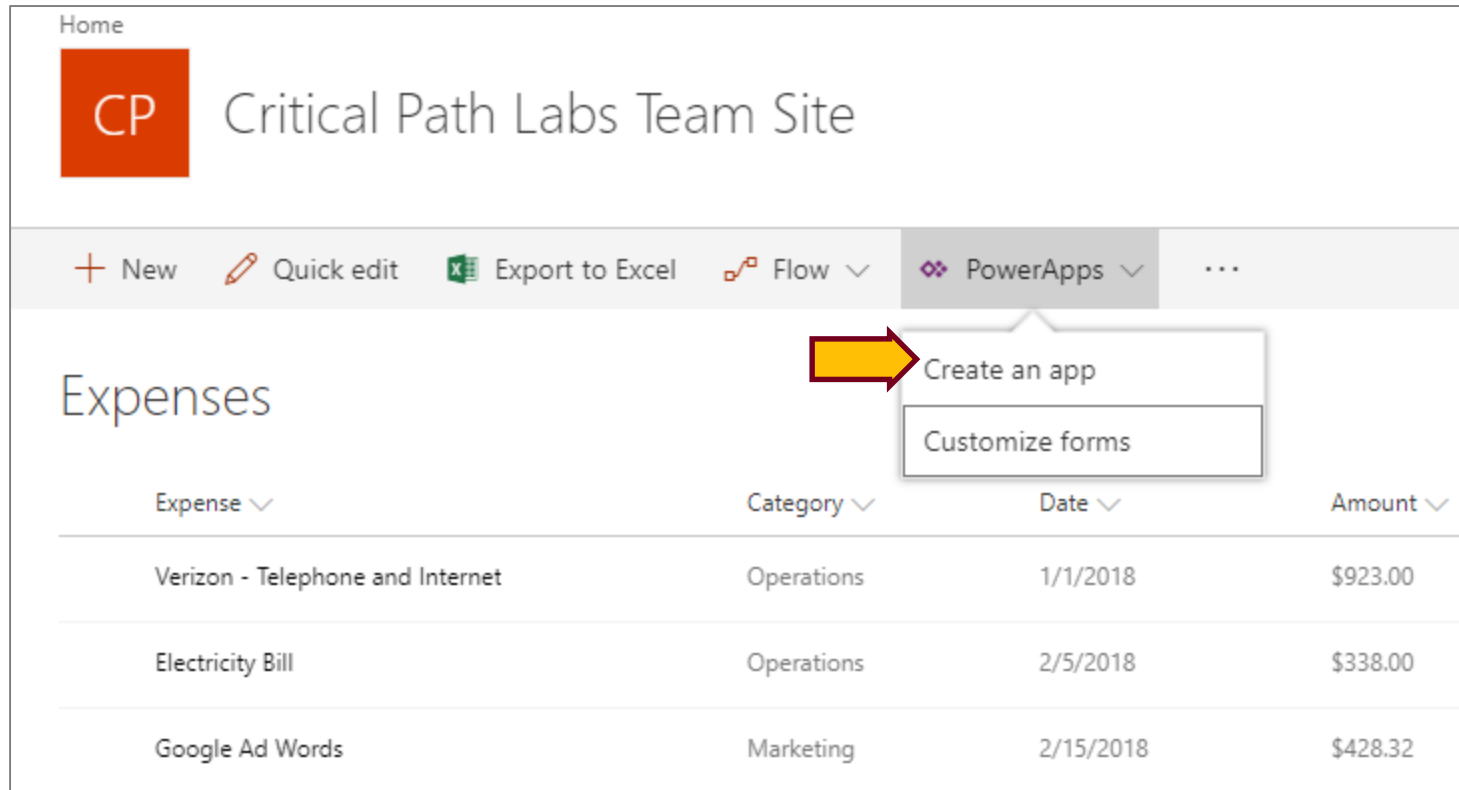
- ✓ Screen Design Guidelines
- ✓ Displaying Repeating Data using Galleries
- ✓ Using Table Functions to Filter and Sort Data
- ✓ Updating Data using Edit Forms and Data Cards
- Customizing SharePoint List Forms





# Customizing SharePoint List Forms

- Modern SharePoint lists allow forms customization
  - Customizing SharePoint list forms involves creating canvas app



Home

CP Critical Path Labs Team Site

+ New Quick edit Export to Excel Flow PowerApps ...

Expenses

Expense Category Date Amount

Verizon - Telephone and Internet	Operations	1/1/2018	\$923.00
Electricity Bill	Operations	2/5/2018	\$338.00
Google Ad Words	Marketing	2/15/2018	\$428.32

Create an app

Customize forms



# Customizing the Form

- SharePoint list data passed in connection

The screenshot displays the Microsoft PowerApps editor interface. At the top, the 'PowerApps' logo is visible. Below it, a navigation bar includes 'Back to SharePoint', 'File', 'Insert', 'View' (selected), 'Action', and 'Home'. A secondary bar shows 'Data sources', 'Media', 'Collections', 'Variables', and 'Advanced'. The main workspace is divided into three panels:

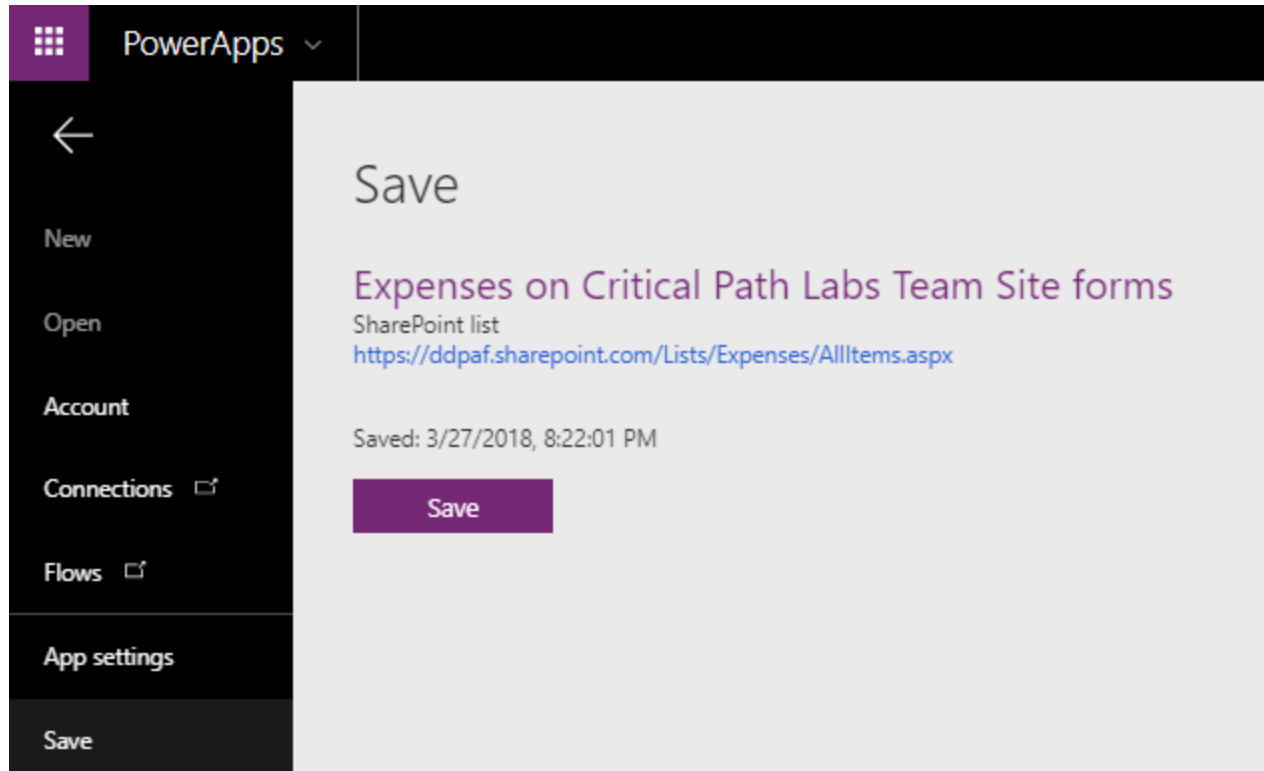
- Left Panel (Screens):** A tree view showing the project structure. It includes 'SharePointIntegration', 'FormScreen1', 'Label1', and 'SharePointForm1'. Under 'SharePointForm1', there are sub-items: 'ID\_DataCard1', 'Amount\_DataCard1', 'Date\_DataCard1', 'Title\_DataCard1', and 'DataCard2'.
- Center Panel (Form Design):** A visual representation of the custom form. It has a yellow header with the text 'This is a Custom Form'. Below the header, there are input fields for 'Expense' (containing 'Verizon - Telephone and Internet'), 'Category' (a dropdown menu with 'Operations' selected), 'Date' (containing '1/1/2018'), and 'Amount' (containing '923').
- Right Panel (Properties):** A panel for configuring the selected form. It has tabs for 'Properties', 'Rules', and 'Advanced'. The 'Properties' tab is active, showing settings for 'SharePointForm1'. The 'Data' section shows the data source as 'Expenses' (a SharePoint list). The 'Layout' section shows the form is set to 'Vertical'. The 'Default mode' is set to 'Edit'. The 'Position' and 'Size' sections show the form's dimensions (X: 0, Y: 60, Width: 444, Height: 730). The 'Color' and 'Border' sections show the form's styling. The 'Visible' property is set to 'On'.

Below the 'Properties' panel, there is a 'Fields' section that lists the fields from the 'Expenses' list. The fields are: 'ID' (value 123), 'Expense' (value abc), 'Category' (value Operations), 'Date' (value 1/1/2018), 'Amount' (value 923), and 'Modified' (value 1/1/2018). Each field has a checkbox to select it and a dropdown menu to choose its display format.

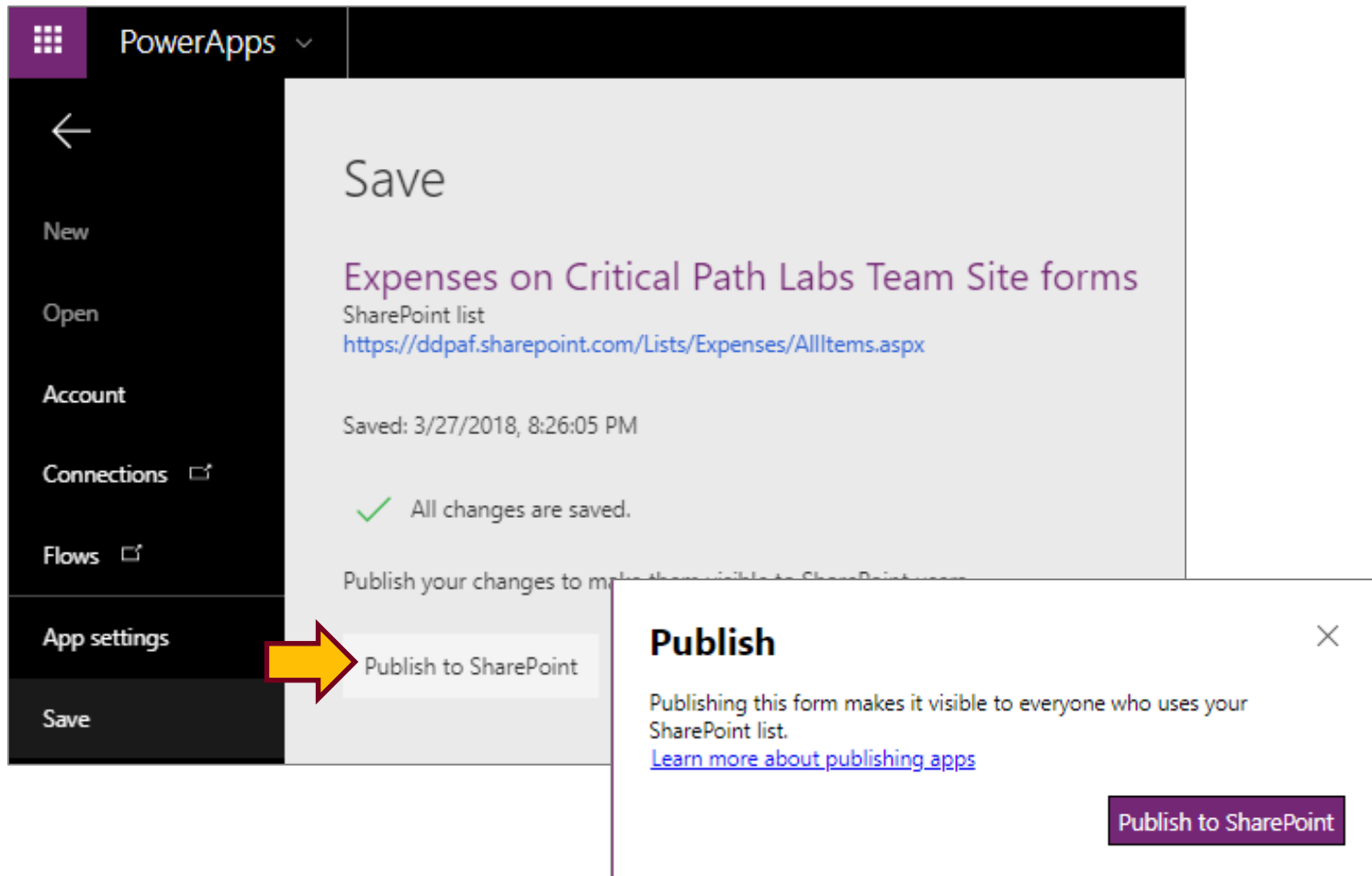


# Saving the Customized Form

- Canvas app is saved back into SharePoint site



# Publishing the Form Back to SharePoint



# Editing Items using a Customized Form

Home

CP

Critical Path Labs Team Site

Edit

Share

Copy link

Delete

Flow

## Expenses

Expense	Category	Date
Verizon - Telephone and Internet	Operations	1/1/2018
Electricity Bill	Operations	2/5/2018
Google Ad Words	Marketing	2/15/2018
Printer Paper	Office Supplies	3/1/2018
Azure HDInsight Subscription	Research & Development	3/15/2018

Save

Cancel

Copy link

Customize

This is a Custom Form

ID

2

\* Expense

Electricity Bill

Category

Operations

Date

2/5/2018

Amount

338







**DEMO**

# Customizing SharePoint List Forms

# Summary

- ✓ Screen Design Guidelines
- ✓ Displaying Repeating Data using Galleries
- ✓ Using Table Functions to Filter and Sort Data
- ✓ Updating Data using Edit Forms and Data Cards
- ✓ Customizing SharePoint List Forms

