Year in review

ZOHO CREATOR'S 2019

INDEX

IMPROVEMENTS & ENHANCEMENTS 1

FORMS, PAGES, AND REPORTS

CONNECTIONS AND WORKFLOWS

APPLICATION LEVEL UPDATES

MOBILE

ZIA UPDATES 10

DELUGE UPDATES 17

NEW FEATURES 21

REFERENCE LINKS 35

IMPROVEMENTS & ENHANCEMENTS

All of these features and improvements are available in Zoho Creator 5, unless specified otherwise.

FORMS, REPORTS & PAGES

Address and Location fields

The physical address entered by the user will immediately fetch its geo-coordinates using map services. The displayed address's coordinates will allow users to adjust to the exact location on a map.

Display options for location data in reports

The display option in reports provides users the option to define if the data in these fields is to be displayed as a physical address or as a set of coordinates.

Navigational buttons in calendar and timeline reports

View specific pickers and reposition navigation buttons to improve usability.

Custom actions after a confirmation

There's now an option to get confirmation before running a custom action workflow, to avoid executions from accidental clicks or taps.

Interactivity in reference-listing pop-ups

Pop-ups that list references while deleting a field or app will now show a link to the referred component, so users don't have to write it all down before handling them.

Auto-refresh of related elements in pages

Page elements in an embedded form will automatically be refreshed when the user creates, updates, and deletes records in that form.

Drill down to data points in pivot charts

This enables users to visualize underlying data points as new charts.

Field-level permissions when viewing related data

Admins can now permit users to only view related data blocks when they have permission to view data in the corresponding form.

Panel builder

This is a point-and-click interface for customizing panels. It enables users to add and customize multiple elements within panels.

Gauge

Gauges visually indicate progress towards goals or KPIs. A gauge on a page will display an aggregate value that's derived from the data in a form.

Kanban report customizations

The user can hide or show the EMPTY column, or show it with a custom label and filter the columns that the Kanban report is to display.

CONNECTIONS & WORKFLOWS

Custom OAuth connectors

We've extended support for third-party authentications with the OAuth 2.0 connector.

Form workflows revamped

Event-based handling of form workflows make navigation easier and more efficient.

Zia functions in Deluge

Zia now brings more intelligence to apps, with functions for sentiment analysis, language prediction, anomaly detection, and phone number parsing.

APPLICATION LEVEL UPDATES

Domain restriction

To ensure that an email address from an external domain does not get added as a user or developer, domain restriction can be enabled at the app level.

DKIM email authentication

Zoho Creator now supports DKIM and SPF email authentication for outgoing emails.

MOBILE

"Open in app" from Recents

On the Recents screen, users can now swipe left on a component to open it in the corresponding app, allowing them to access the component from that app.

Siri shortcuts in Creator iOS app

The forms, reports, and pages in apps will now appear in iOS device search results. Users will need to access the apps once for search to read the components they contain.

ZIA POWERED SMART DATA IMPORT

This improvement enabled:

- Data cleansing capabilities powered by Zia—Zoho's AI assistant for business
- Support for importing subform, name, and address data
- An import interface with a spreadsheet app-like experience

* **Note:** These improvements are available during app creation and not when creating a form by importing a file.

Zia-powered data cleansing capabilities

Before uploading files and creating an app, it's best practice to tweak the data in it, so that it fits the model of tables in a relational database. To assist here, the new Zia-powered import interface enables:

- **Filtering** the data in a sheet. Only the filtered data will be imported into the application.
- **Finding** and **replacing** data
- Performing data-cleansing operations in sheets, which are grouped under the Refine Data option:
  - Fix inconsistent spellings
  - Fill in missing values
  - Split the data in a column into two by defining their delimiter
  - Merge columns
  - Explore the frequency distribution of values in a column

Importing subform data

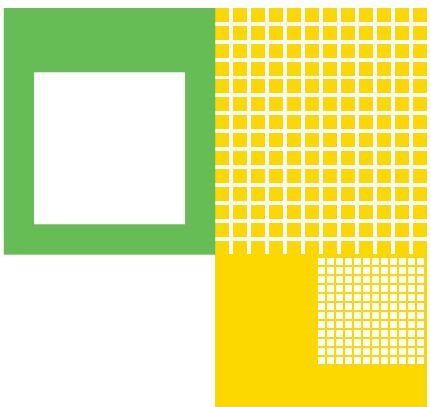
Before uploading a file, the main or parent form data and subform data must be in separate sheets.

In the import interface, checking a sheet's **Make it as sub-form** checkbox let's Creator know that it contains subform data. Upon doing so, users must:

- Select the sheet that contains its parent data.
  This will become the parent or main form after the import operation.

- Select the column in the parent sheet that identifies each row uniquely.

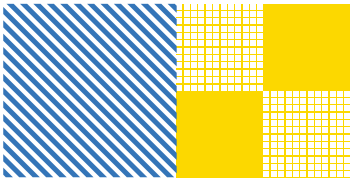
- Select the column that's the parent row identifier in the current sheet (which is being marked as subform)



Importing name and address data

Users can map the Name or Address field to a column via its **Field type** option.

Name or address data can be present in a single column or span multiple columns. When it spans multiple columns, users can map the name subfields (Prefix, First Name, Last Name, Suffix) and address subfields (Address Line 1, Address Line 2, etc.) to the respective columns in the sheet.



Spreadsheet-like import interface

Users can map the Name or Address field to a column via its **Field type** option.

Currently, upon importing, the sheets are displayed as collapsible sections that show a preview of the data in them. Instead, they're now displayed as tabs—like sheets in a spreadsheet. In addition to existing abilities, such as renaming and selecting/deselecting sheets, users can:

- View the data in every row in the sheet
- Sort the data in sheets. (This sorting won't be applied in the app itself. It's only to let users view their data on their own.)
- Delete rows
- Undo and redo actions

Spreadsheet app-like import interface

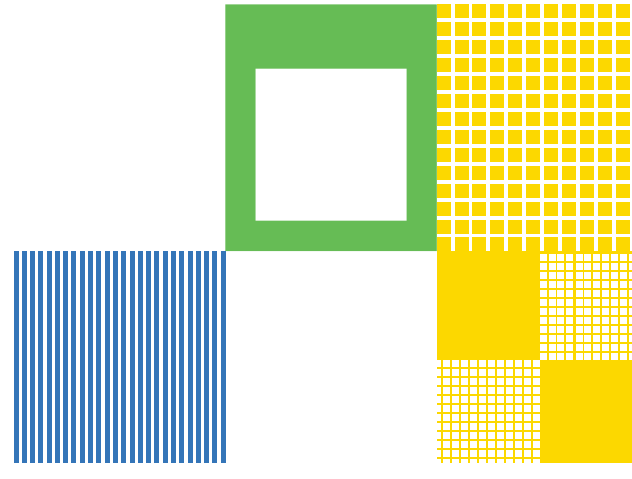
Zia-powered smart data import capabilities are now available during form creation, as well.

Additionally, the following abilities were introduced:

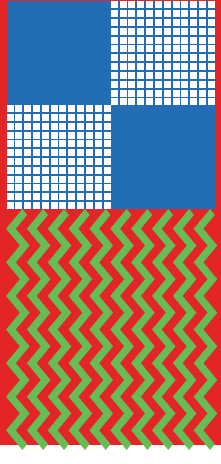
- Edit a cell's value by double-clicking it
- Edit identical cells in a column by right-clicking
- Enable the ***No duplicate values*** property for a column
- Enable the Mandatory property for a column
- Establish relationships with data in existing forms via a Lookup field

Zia Search interface

Users can now launch the Zia Search interface using keyboard shortcuts: **Ctrl** + / on Windows, and **Command** + / (⌘ + /) on Mac. These will work both when accessing a Creator account's home page, and while editing apps.



DELUGE UPDATES



subform.insert() and subform.clear()

These Deluge tasks enable dynamically adding and managing data in subforms.

Insert data into a subform when adding a record in the main form

The ***add record*** Deluge task now supports adding rows in subforms:

```
// declaring the row
<row1_variable> = <form_link_name>.<subform_field_link_name>();

// assigning values for various subform fields in the row
<row1_variable>.<field1_link_name> = <value>;
<row1_variable>.<field2_link_name> = <value>;

// declaring another row (declare as many rows as required)
<row2_variable> = <form_link_name>.<subform_field_link_name>();

// assigning values for various subform fields in the row
<row2_variable>.<field1_link_name> = <value>;
<row2_variable>.<field2_link_name> = <value>;

// declare a variable to hold the collection of rows
<row_collection_variable> = collection();
<row_collection_variable>.insert(<row1_variable>, <row2_variable>);

// insert subform rows using the add record task
record_id = insert into <form_link_name>
[
 Added_User = <value>
 <subform_field_link_name> = <row_collection_variable>
];
```

Dynamically insert rows in subforms when users add or edit data

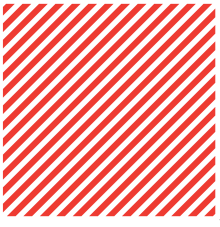
```
// declaring the row
<row1_variable> = <form_link_name>.<subform_field_link_name>();

// assigning values for various subform fields in the row
<row1_variable>.<field1_link_name> = <value>;
<row1_variable>.<field2_link_name> = <value>;

// declaring another row (declare as many rows as required)
<row2_variable> = <form_link_name>.<subform_field_link_name>();

// assigning values for various subform fields in the row
<row2_variable>.<field1_link_name> = <value>;
<row2_variable>.<field2_link_name> = <value>;

// declare a variable to hold the collection of rows
<row_collection_variable> = collection();
<row_collection_variable>.insert(<row1_variable>, <row2_vari-
able>);

// insert the rows into the subform on a form workflow
input.<subform_field_link_name>.insert(<row_collection_vari-
able>);
```

Delete data from subforms

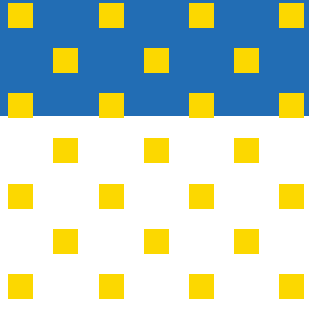
```
// delete rows from a subform
input.<subform_field_link_name>.clear();
```

This task will behave differently based on the type of subform:

- When operating the clear() task on a blank/inline sub form, it'll delete all the rows in that subform.
- When operating the clear() task on a subform that itself is a form, it'll just delete the relationship between the main form's record and related subform records. Records in the subform will remain intact.

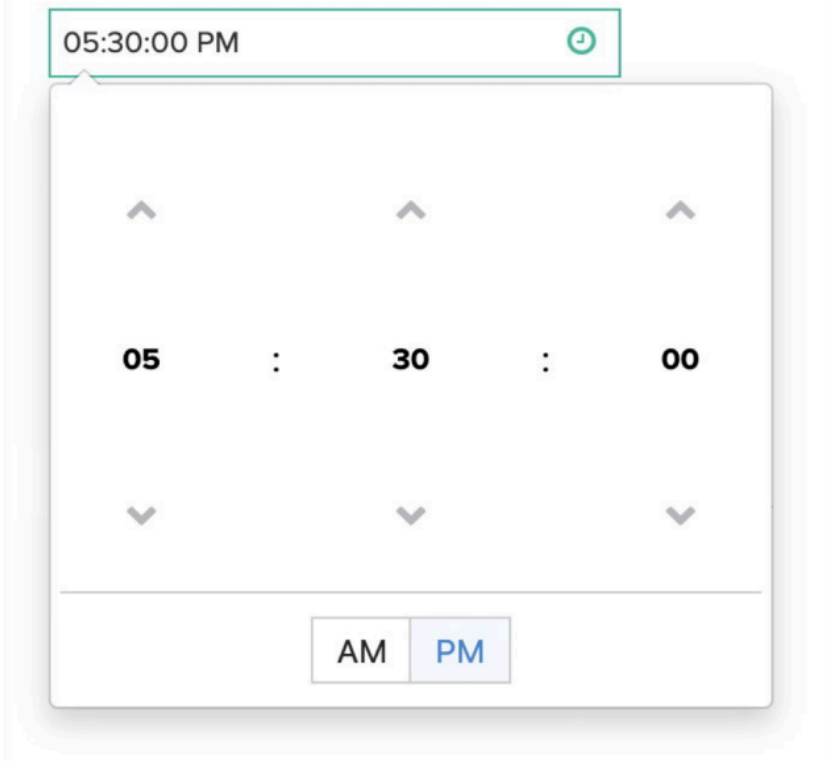


NEW FEATURES



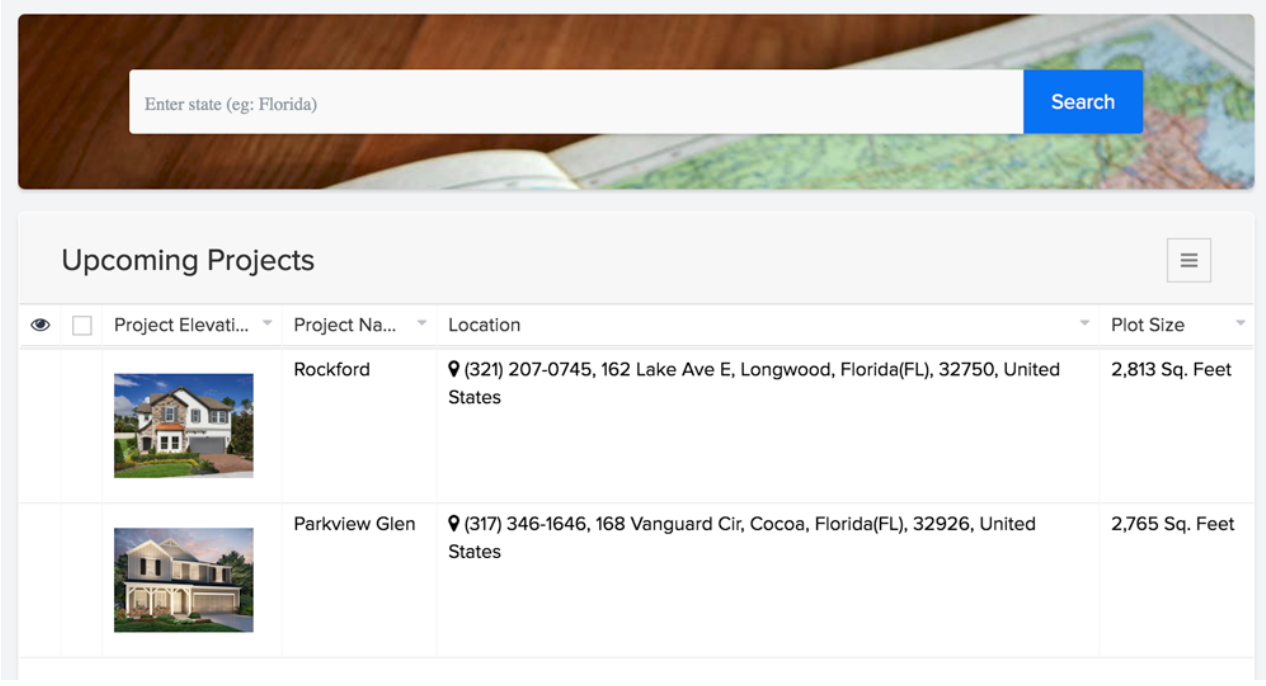
Time field and 12-hour format

The Time field records time independent of date, allowing fine-grained control over time specifications, such as the use of 12 or 24 hour format and time interval capture. New controls also apply for the Date-Time field.



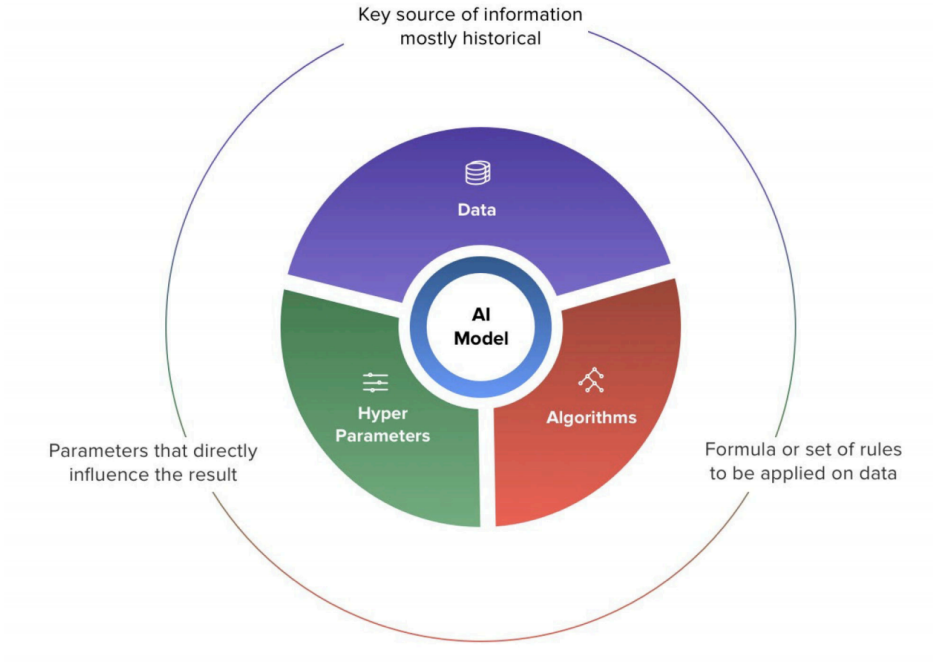
Search elements in pages

The Search element in a page enables users to search for data across the applications in a Zoho Creator account. The search results are shown by filtering a report or page using the search input.



AI fields

AI in Creator helps build smarter and more predictive applications that can solve specific business challenges. Built-in finite AI models are trained to process large amounts of data, identify patterns, and accomplish tasks, such as object detection, data clearing, sentiment analysis, and keyword extraction.



AI Deluge tasks

Zoho's AI can be used via Deluge to:

Parse a phone number

The zoho.ai.parsePhoneNumber task examines the given phone number to provide its device, network, location, and formatting information.

Syntax

```
<response_variable> = zoho.ai.parsePhoneNumber(<phone_number>,
<country_code>);
```

Example

```
response = zoho.ai.parsePhoneNumber("92597XXXXX", "US");
```

Response


```
{
 "extension":"",
 "italian_leading_zero":"false",
 "international_format":"+1 925-97X-XXXX",
 "national_format":"(925) 97X-XXXX",
 "timezone":"[\"America/Los_Angeles\"]",
 "e164_format":"+192597XXXXX",
 "national_number":"92597XXXXX",
 "type":"FIXED_LINE_OR_MOBILE",
 "outside_from_CH":"00 1 925-97X-XXXX",
 "country_code":"1",
 "carrier":"",
 "original_format":"(925) 97X-XXXX",
 "outside_from_US":"1 (925) 97X-XXXX",
 "location":"California",
 "raw_input":"92597XXXXX",
 "region":"US",
 "country_code_source":"FROM_DEFAULT_COUNTRY"
 }
```

Parse an address

The **zoho.ai.parseAddress** task is used to split the given address into individual components, such as road, house number, city, and district.

Syntax

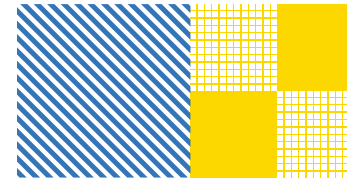
```
<response_variable> = zoho.ai.parseAddress(<address>);
```

Example

```
address_list = List();
address_list.add(”Estancia IT Park, GST Road, Vallancherry Village, Chengalpattu, Kanchipuram district, Tamil Nadu 603202”);
address_list.add("4708 HWY 71 E Del Valle, TX 78617");

response = zoho.ai.parseAddress(address_list);
```

Response


```
{
  "data":{
     "items":[
        {
           "road":"gst road vallancherry village",
           "city":"chengalpattu",
           "state_district":"kanchipuram district",
           "postcode":"603202",
           "state":"tamil nadu",
           "house":"estancia it park"
        },
        {
           "country":"usa",
           "unit":"suite 200",
           "road":"burleson rd building",
           "city":"austin",
           "postcode":"78744",
           "house_number":"310",
           "state":"tx"
        }
     ]
  },
  "message":"OK",
  "status":200
}
```

Identify the language of text

The **zoho.ai.predictLanguage** task identifies the language of a given piece of text.

Syntax

```
<response_variable> = zoho.ai.predictLanguage(<sentence>);
```

Example

```
response = zoho.ai.predictLanguage(”Hai tout, comment ça va?");
```

Response


```
{
   "data":{
      "items":[
         {
            "predictionProbabilities":{
               "fr":0.9999907117029719
            },
            "queryValue":"Hai tout, comment ça va?"
         }
      ]
   },
   "message":"OK",
   "status":200
}
```

Find named entities

The **zoho.ai.findNamedEntities** task extracts parameters relevant to a specified intent from a given text.

Syntax

```
<response_variable> = zoho.ai.findNamedEntities(<sentence>,<entity_type>);
```

Example

```
text = "Hello, I'm Shawn. My interests are data science and Internet Of Things";

entity_types = List();

entity_types.add(”person”);

entity_types.add(”skills”);

response = zoho.ai.findNamedEntities(text,entity_types);
```

Response


```
{
   "data":{
      "items":[
         {
            "skills":[
               "data science",
               "Internet Of Things"
            ],
            "person":[
               "Shawn"
            ]
         }
      ]
   },
   "message":"OK",
   "status":200
}
```

Sentiment analysis

The **zoho.ai.analyseSentiment** task interprets a given piece of text to predict if it has positive, negative, or neutral emotion.

Syntax

```
<response_variable> = zoho.ai.analyseSentiment(<sentence>);
```

Example

```
response = zoho.ai.analyseSentiment("The session was in-
triguing.");
```

Response

```
{
   "Prediction":"Positive",
   "Probability":{
      "Neutral":"16%",
      "Negative":"50%",
      "Positive":"79%"
   }
}
```

Text recognition

The **zoho.ai.recognizeText** task performs OCR (optical character recognition) on a specified file.

Syntax

```
<response_variable> = zoho.ai.recognizeText(<file>,
[<model_type>], [<language>]);
```

Example

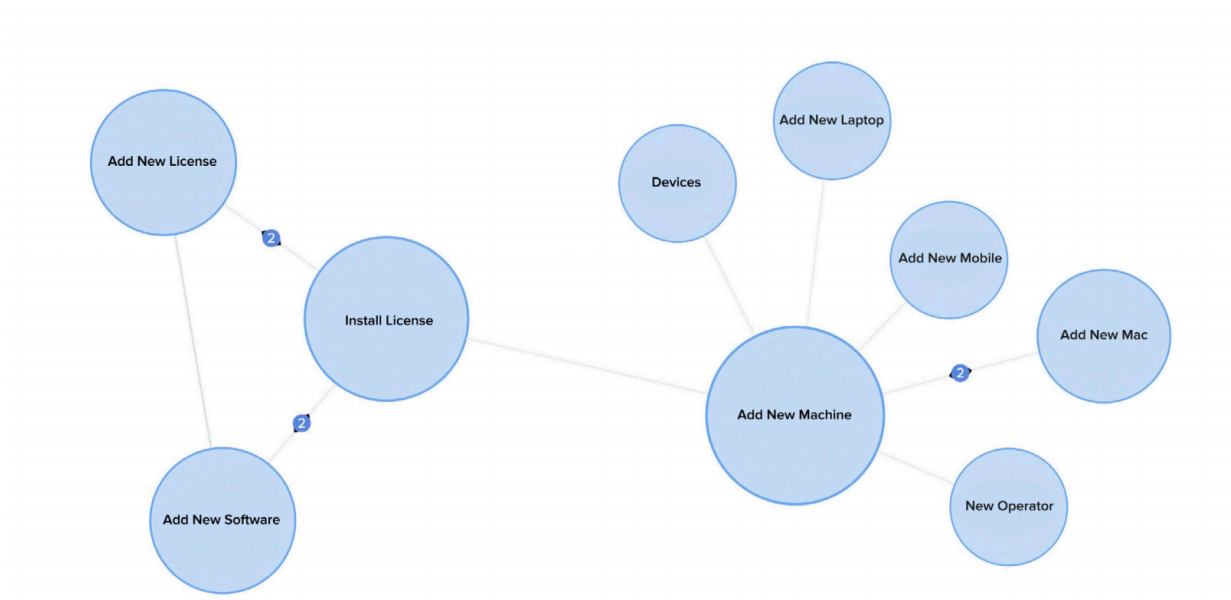
```
// To fetch the image file from the web
image = invokeurl
   [
      url: "http://www.thenonlinearpath.com/wp-content/up-
loads/2016/05/GoodVibesOnly.png"
      type: GET
   ];
// To extract the text from the image file
 response = zoho.ai.recognizeText(image);
```

Response

```
{
   "text":"GOOD\n\nVIBES\n\nONLY\n"
}
```

Schema builder

The schema builder is a graphical tool that lets users visualize their data model, create new relationships, or edit existing ones as needed. Replacing the ER diagram, its web of circles is designed to be clutter-free and scalable, for complex applications with large forms and many relationships. The number of fields in a form and the number of relationships attached to it determine the size of the circles.



Reference Links

Product update catalog

zoho.com/creator/release-notes/

Extensive help documentation

zoho.com/creator/newhelp/

Ebooks and case studies

zoho.com/creator/help/ebooks/

For technical support, reach out to us!

support@zohocreator.com