

Excellware Angular Application Overview

Excellware provides an Angular Application as a part of the Dynamo Tools package. This app provides the user interface via a Chrome web browser. The backend application is based on BBj servlets. BBj is a version of java that includes compatibility with legacy business basic applications.

All communication between the client (Angular app) and the server (BBj servlets) is via json.

The application can be launched with or without a login process.

Login Process

The initial login process begins with a url:

[Scheme] ://[Server]/CD/[gui|guitest]?cc=XX&otherArguments

examples:

<https://dev.excellware.com/CD/guitest/?cc=CD>

<https://portal.ncdsinc.com/CD/gui/?cc=NC&usertype=e>

The Angular app passes the mode, gui or guitest, and the query string to the server using this url:

[Scheme] ://[Server] : [Port]/dt/login?mode=guitest&cc=XX&otherArguments

The Port is 8443 for https and 8888 for http.

examples:

<https://dev.excellware.com:8443/dt/login?mode=guitest&cc=CD>

<https://portal.ncdsinc.com:8443/dt/login?mode=gui&cc=NC&usertype=e>

<http://xyz.domain.com:8888/dt/login?mode=gui&cc=DS>

The server will reply with an Authorization Header that must be sent back to the server for all subsequent API calls to manage the active session.

No Login

In some cases, the user has already logged in from a non-web application. In this case the user does not need to login again. The initial URL includes session data provided in the query arguments.

Server Responses

The server response to all api calls is always contained in an array named "data:". Each object in the data array has a property named "type". The various "type" values are shown in the table below, and describe the basic type of action to be performed by the client.

data.type	Description
pageInit	Always present as the first element in the data array. Indicates if the page contents should be cleared or modified.
notify	Messages to the user presented in the form of temporary toast messages, or alert boxes which require an operator response.
formData	A new single-part form with input fields for text, numbers, dates, time, radio

	buttons, check boxes, etc.
formUpdate	Changes to a form including hiding or unhiding fields, changing the form field values, setting the focus on a particular field, etc.
buttons	Show, hide, enable buttons
setup	Information that is stored by the client including fields that appear on a menu page, default timeout for toast messages, etc.
dtWebMenu	A collection of buttons that launch applications
displayView	Display data in a two column grid
displayGrid	A grid of rows and columns which may have clickable rows
entryGrid	A grid with input elements including text, numbers, dates, time, radio buttons etc.
rowGridUpdate	Modify a specific grid row
action	Open a new browser tab to a specified URL or replace the active tab
authorization	Update the authorization header on subsequent api calls.
dialog	Display a simple dialog similar to an alert message
popup	Create a new window in front of existing page
image	Display an image
dialogFile	Select files for uploading
htmlEdit	Html Editor input box
progress	Display a progress meter
progressResponse	Update the progress
treeControl	A tree control
multiPartForm	A MultiPart Form with titles and expanders for each section