



RESPONSIVE UI RELEASE NOTES

Version 2023.0.0

2/20/2023

In this release...

- [Upgraded dependency packages](#)
- [Updated @jha/rui-angular **jha-app** to support PathLocationStrategy](#)
- [Accessibility enhancements](#)
- [Adjusted grid line styles for data grid, tree grid and complex grid](#)
- [Added support for showClear option for multiple PrimeNG components](#)
- [Resolved existing memory leaks](#)
- [Updated rui-display-block to add right padding if content is scrollable](#)
- [Enabled jhaToggleValChange event for reactive jha-toggle component](#)
- [Bugs fixed in this release](#)

Upgraded dependency packages

Multiple packages were upgraded, some including breaking changes. Packages of note that were upgraded include:

- @angular (15.1.1)
- @angular/material (15.1.4)
- ag-grid (29.0.0)
- chart.js (4.2.1)
- jodit (3.24.5)
- @fullcalendar (6.1.4)
- ngx-bootstrap (10.2.0)
- primeng (14.1.2)
- typescript (4.9.5)

Use the Angular Update Guide to upgrade your local application:

<https://update.angular.io/?l=3&v=14.0-15.0>.

The @angular/material packages has some significant breaking changes in 15.x. All of the components have been completely rewritten, changing both structure and class names. For this release, we're moving up to 15.x, **but importing the legacy versions of the components**.

To do that, the existing imports in the app.module will go from this:

```
import { MatTabsModule } from '@angular/material/tabs';  
import { MatChipsModule } from '@angular/material/chips';  
import { MatSelectModule } from '@angular/material/select';
```

To importing the legacy modules like this:

```
import { MatLegacyChipsModule } from '@angular/material/legacy-chips';  
import { MatLegacySelectModule } from '@angular/material/legacy-select';  
import { MatLegacyTabsModule } from '@angular/material/legacy-tabs';
```

There are also breaking changes for moving from @fullcalendar/* from 5.x to 6.x. **Version 6.x requires the additional package of @fullcalendar/core**. Make sure to add this to the package.json dependency list.

Also, in 5.x, the dayGridPlugin, timeGridPlugin and interactionPlugin are imported into the app.module and registered using:

```
FullCalendarModule.registerPlugins([  
  dayGridPlugin,  
  timeGridPlugin,  
  interactionPlugin
```

```
]);
```

In `@fullcalendar 6.x` remove those elements from the `app.module` and add the following to the view typescript files that use `FullCalendar`:

```
import dayGridPlugin from '@fullcalendar/daygrid';
import timeGridPlugin from '@fullcalendar/timegrid';
import interactionPlugin from '@fullcalendar/interaction';
```

```
ngOnInit() {
  this.options = {
    plugins: [
      dayGridPlugin,
      timeGridPlugin,
      interactionPlugin
    ],
    ...
  }
}
```

In `@fullcalendar 6.x`, the styling is now imported with the module and no longer needs to be bundled with the application css. **You can remove references to:**

- "node_modules/@fullcalendar/common/main.css"
- "node_modules/@fullcalendar/daygrid/main.css"
- "node_modules/@fullcalendar/timegrid/main.css"

Updated `@jha/rui-angular jha-app` to support `PathLocationStrategy`

Routing for Angular/Web Component (AWC) applications (using `@jha/rui-angular` and `@jha/rui-wc`) was previously forced to use `HashLocationStrategy`. We have made an update to the `@jha/rui-angular jha-app` component to support `PathLocationStrategy`.

If you are using the default `PathLocationStrategy` in your Angular application, set the **`routeMatchPrefix`** property to an empty string in your `jha-app` component: `<jha-app routeMatchPrefix="" ...>`. This will listen to click events from `@jha/rui-wc` components (header and nav buttons, mega menu components, button menu options/templates, etc), and if they have an `href` attribute value that doesn't start with 'http', it will use the Angular router instead of reloading the page with the updated URL.

As before, when using generic html elements such as anchor tags, you can continue to use the **`routerLink`** attribute to trigger Angular routing.

Accessibility enhancements

Numerous accessibility enhancements were implemented, some including breaking changes.

1. If the page does not contain any breadcrumbs, a <nav> and element were still output to the view without any elements, creating an accessibility error. This is no longer the case.
2. Aria-label attributes have been added to nav and header components to ensure they're acknowledged by screen readers.
3. Previously elements (nav, header, buttons) with text and an icon output an sr-only span with the iconType that was leading to duplicate messages from screen readers. This sr-only span is now only rendered when no text is provided.
4. Within the **jha-header-right** element, child elements are placed within an element. Elements such as **jha-header-dropdown** and **jha-header-buttons** contained elements, but because of the component, these were not direct children of the . To be semantically correct, **the elements have been removed from the header components and will need to be wrapped around the jhresponsive components to be semantically accurate.**

For example, in your main-header.component.html file this snippet:

```
<jha-header ...>
  <jha-header-left></jha-header-left>
  <jha-header-right>
    <jha-header-mega-menu ...></jha-header-mega-menu>
    <jha-header-dropdown ...></jha-header-dropdown>
    <jha-header-button ...></jha-header-button>
  </jha-header-right>
</jha-header>
```

Should now look like this:

```
<jha-header ...>
  <jha-header-left></jha-header-left>
  <jha-header-right>
    <li><jha-header-mega-menu ...></jha-header-mega-menu></li>
    <li><jha-header-dropdown ...></jha-header-dropdown></li>
    <li><jha-header-button ...></jha-header-button></li>
  </jha-header-right>
</jha-header>
```

Adjusted grid line styles for data grid, tree grid and complex grid

Previously, row grid lines used in the default *minimal* appearance were very light and could be tough to discern. We've updated the grid row border colors for all themes to be more prominent. This change applies to the p-table, p-treeTable and the ag-grid. As a part of this update, there is a new `-table-row-border` CSS var available for theming.

Added support for showClear option for multiple PrimeNG components

Added support for showClear option for PrimeNG dropdown, inputNumber, multiSelect, treeSelect and calendar components.

Resolved existing memory leaks

The previous releases contained some unsubscribed subscriptions that were leading to memory leaks in the framework. These have been unsubscribed and cleaned up.

Updated rui-display-block to add right padding if content is scrollable

The existing rui-display-block contained no internal padding, so at certain times the content would lie right up against the right scrollbar with no padding. The logic has been updated so that if the content is taller than the display block content area, a right padding is added to give the content breathing room.

Enabled jhaToggleValChange event for reactive jha-toggle component

The existing jha-toggle component only exposed the jhaToggleValChange event when the jhalsReactive property was set to false. Now, the event is available for reactive implementations too.

Bugs fixed in this release

- ANGULAR COMPONENTS
 - Updated Master/Detail styles to be more consistent.
 - Updated removeStackedView in jha-app to test for jha-function-view before attempting to update classList.
 - Corrected styles for p-radioButton within p-dataTable.

- WEB COMPONENTS
 - Updated rui-form-field to contain PrimeNG components that included dropdowns.
 - Updated error states of rui-form-field to correct border color for prefix, suffix and action buttons.
 - Reinstated **rui-selected-tab-change** event for rui-tabset.
 - Updated cursor for display block back-to-top button to be **pointer**.
 - Updated height of popup notification with button to be consistent with other notifications.
 - Updated header and navigation z-index values when function view isBusy = true so dropdown and mega menus can flow over spinner background.
 - Update rui-focus-trap to test for elements before removing event listeners to resolve errors.
 - Updated Master/Detail styles to be more consistent.
 - Updated error state title for jha-group-box.
 - Updated focus state for rui-form-field inputs when using action buttons.
 - Updated removeStackedView in jha-app to test for rui-function-view before attempting to update attribute list.
 - Set alignment in rui-button-menu-submenu toLower() to catch case differences.
 - Corrected styles for p-radioButton within p-dataTable.
 - Synch up data dense styles for rui-form-field and rui-record-detail.
 - Correct rui-subheader styles for data dense mode.