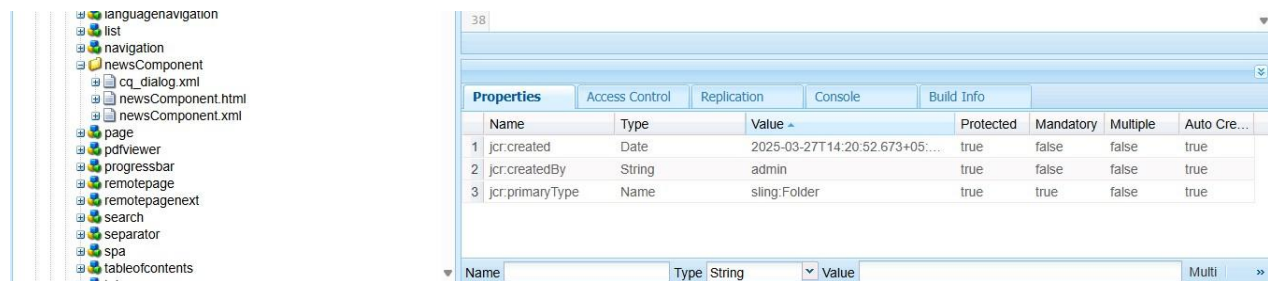


## 1. Create a New Component for News

- ❖ Navigate to CRXDE Lite → **/apps/myTraining/components/**
- ❖ Create a new component folder named **newsComponent**
- ❖ Inside the newsComponent folder, create the following files:
  - newsComponent.html** → HTL template for rendering the component
  - newsComponent.xml** → Defines the component properties
  - \_cq\_dialog.xml** → Configures the dialog for authoring
  - NewsComponent.java** → Sling Model (in core bundle)



### NewsComponent.java (Sling Model)

```
package com.myTraining.core.models;
import
com.adobe.cq.sightly.WCMUsePojo;
import com.adobe.cq.wcm.core.components.models.Component;
import org.apache.sling.models.annotations.Default;
import org.apache.sling.models.annotations.Model;
import org.apache.sling.models.annotations.injectorspecific.ValueMapValue;
import javax.inject.Inject;
```

```
@Model(adaptables = { Component.class })
```

```
public class NewsComponent {

    @Inject
    @Default(values = "Default
    Title") private String title;

    @Inject
    @Default(values = "Default News
    Detail") private String newsDetail;

    @Inject
    @Default(values = "2025-03-
    20") private String
    publishedDate;

    @Inject
    @Default(values = "Default Source")
    private String source;

    public String getTitle()
    { return title;
    }

    public String getNewsDetail()
    { return newsDetail;
    }

    public String getPublishedDate()
    { return publishedDate;
    }

    public String getSource()
    { return source;
    }
}
```

### **newsComponent.html (HTL Template)**

```
<div class="cop-news-component">
  <h2 class="news-title">${properties.title}</h2>
  <p class="news-detail">${properties.newsDetail}</p>
  <p class="news-date">Published on: ${properties.publishedDate}</p>
  <p class="news-source">Source: ${properties.source}</p>
</div>
```

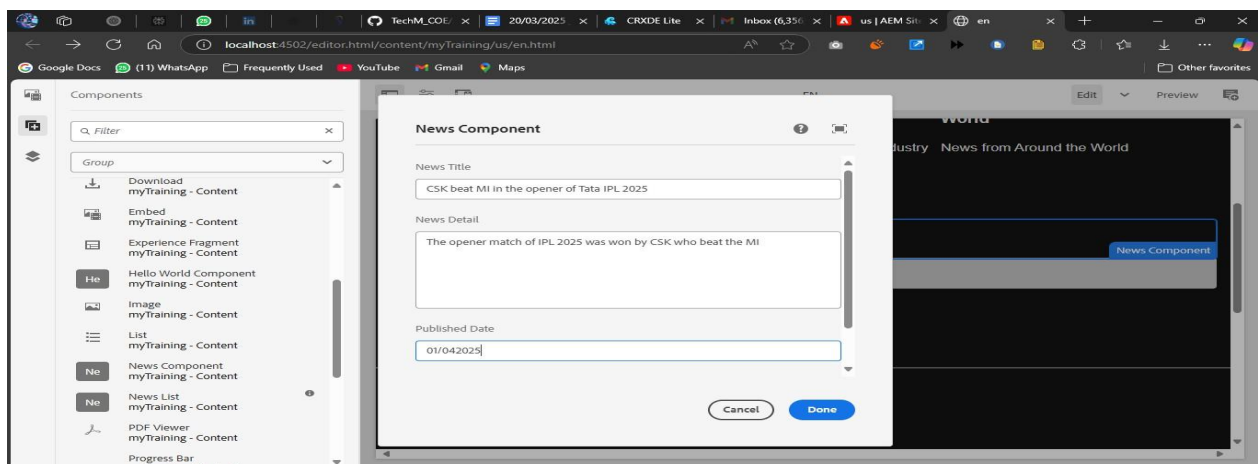
### **cq\_dialog.xml (Dialog for Authoring)**

```
<?xml version="1.0" encoding="UTF-8"?>
<jcr:root
  xmlns:sling="http://sling.apache.org/jcr/sling/
  1.0"
  xmlns:cq="http://www.day.com/jcr/cq/1.0"
  xmlns:jcr="http://www.jcp.org/jcr/1.0"
  jcr:primaryType="cq:Dialog"
  cq:dialogMode="floating"
  title="News Component"
  helpPath="help/using/newsComponent.html">
  <items jcr:primaryType="cq:Panel">
    <items jcr:primaryType="cq:WidgetCollection">
      <title
        jcr:primaryType="cq:Widg
        et" fieldLabel="News
        Title" name="./title"
        xtype="textfield"/>
    <newsDetail
      jcr:primaryType="cq:Widget"
      fieldLabel="News Detail"
      name="./newsDetail"
      xtype="textarea"/>
```

```

<publishedDate
  jcr:primaryType="cq:Widget"
  fieldLabel="Published Date"
  name="./publishedDate"
  xtype="datetime"/>
<source
  jcr:primaryType="cq:Widget"
  fieldLabel="Source"
  name="./source"
  xtype="textfield"/>
</items>
</items>
</jcr:root>

```



## 2. Create a Multifield Component (Multiple News)

1. Create a new component folder **multiNewsComponent**
2. Add a **multiNewsComponent.html** file.
3. Modify the **Sling Model** to support a multifield.

### MultiNewsComponent.java

```
package com.myTraining.core.models;
```

```

import org.apache.sling.models.annotations.Model;
import org.apache.sling.models.annotations.injectorspecific.ChildResource;
import javax.inject.Inject;
import java.util.List;
@Model(adaptables = Resource.class)
public class MultiNewsComponent {
    @ChildResource
    private List<NewsItem> newsList;
    public List<NewsItem>
    getNewsList() {
        return newsList;
    }
    public static class NewsItem
    { @Inject
    private String title;

    @Inject
    private String source;

    public String getTitle()
    { return title;
    }
    public String getSource()
    { return source;
    }
    }
}

```

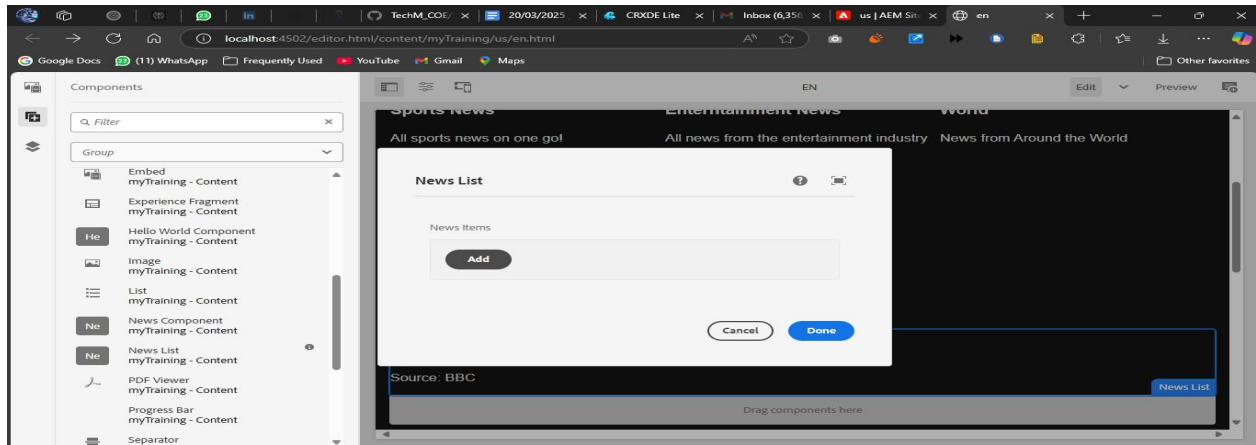
### **multiNewsComponent.html**

```

<div class="multi-news-component">
    <ul>
        <sly data-sly-list.news="${resource.children}">
            <li>
                <h2>${news.title}</h2>
                <p>Source: ${news.source}</p>
            </li>
        </sly>
    </ul>

```

</div>



**3. Create Clientlibs and Apply Green colour to the heading(h2) and yellow colour to news detail(<p>) and date should be black in colour.**

**Create a clientlib folder:**

/apps/myTraining/clientlibs/newsComponent

**clientlibs/css/style.css**

```
.cop-news-component .news-title
{ color: green;
}
```

```
.cop-news-component .news-
detail { color: yellow;
}
```

```
.cop-news-component .news-date
{ color: black;}
```

**clientlibs/js/script.js**

```
console.log("News Component Loaded");
```

**clientlibs/.content.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<jcr:root
```

```

xmlns:sling="http://sling.apache.org/jcr/sling/1.0"
xmlns:jcr="http://www.jcp.org/jcr/1.0"
jcr:primaryType="cq:ClientLibraryFolder"
categories="myTraining.newsComponent"
dependencies="cq.jquery">
</jcr:root>

```

## 4. Create a Base Page Component

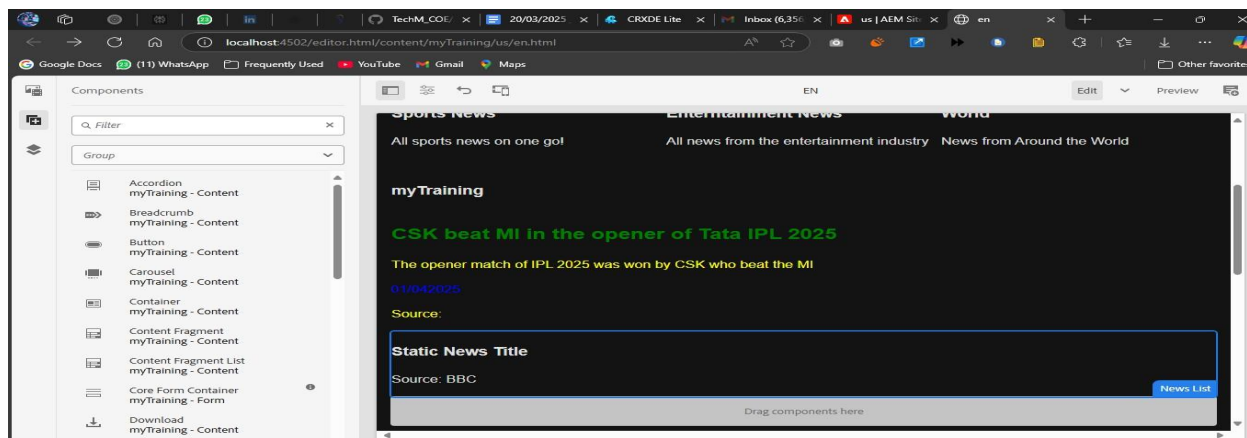
Create a **basePage** component.

### basePage.html

```

<head>
  <meta property="og:title" content="{properties.ogTitle}">
  <meta property="og:description" content="{properties.ogDescription}">

```



```

  <meta property="og:image" content="{properties.ogImage}">
</head>

```

## 5. Create Global Page Properties

Modify the page properties to include OG metadata.

### \_cq\_dialog.xml

```

<items jcr:primaryType="cq:WidgetCollection">
  <ogTitle jcr:primaryType="cq:Widget" fieldLabel="OG Title" name="/ogTitle"
xtype="textfield"/>

  <ogDescription jcr:primaryType="cq:Widget" fieldLabel="OG Description"
name="/ogDescription" xtype="textarea"/>

  <ogImage jcr:primaryType="cq:Widget" fieldLabel="OG Image

```

```
Path" name="./ogImage" xtype="pathfield"/>
</items>
```

## 6. What is extraClientLibs?

- ❖ extraClientLibs is an **AEM feature** that allows including additional client libraries at runtime.
- ❖ It is useful when you need to **load specific client libraries dynamically**.
- ❖ You can add it in **multiNewsComponent**.

### Adding extraClientLibs to Multifield Component

Modify **\_cq\_dialog.xml**:

```
<extraClientLibs
  jcr:primaryType="nt:unstructured"
  extraClientLibs="[myTraining.newsCompon
  ent]" xtype="checkbox"
  fieldLabel="Include Extra Clientlibs"/>
```