

20.3.25

Create a News Component Using Sling Model

1. latest-news.html

```
<div class="latest-news-article">

  <h2>${newsDetails.headline}</h2>

  <p class="news-content">${newsDetails.content}</p>

  <p class="news-source">Source: ${newsDetails.sourceName}</p>

  <p class="publish-date">Published on: ${newsDetails.publishDate}</p>

</div>
```

2. LatestNewsModel.java

```
package com.news.core.models;

import org.apache.sling.api.resource.Resource;
import org.apache.sling.models.annotations.DefaultInjectionStrategy;
import org.apache.sling.models.annotations.Model;
import org.apache.sling.models.annotations.injectorspecific.ValueMapValue;

@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)

public class LatestNewsModel {

    @ValueMapValue
    private String headline;

    @ValueMapValue
    private String content;

    @ValueMapValue
    private String publishDate;

    @ValueMapValue
    private String sourceName;

    public String getHeadline() {

        return headline;

    }

}
```

```

    }

    public String getContent() {
        return content;
    }

    public String getSourceName() {
        return sourceName;
    }

    public String getPublishDate() {
        return publishDate;
    }
}

```

3. dialog.xml

```

<jcr:root xmlns:jcr="http://www.jcp.org/jcr/1.0"
    xmlns:sling="http://sling.apache.org/jcr/sling/1.0"
    xmlns:core="http://www.day.com/jcr/cq/1.0"
    jcr:primaryType="cq:Dialog"
    cq:dialogMode="floating"
    jcr:title="Latest News Configuration">
    <content jcr:primaryType="nt:unstructured">
        <items jcr:primaryType="nt:unstructured">
            <headline
                jcr:primaryType="nt:unstructured"
                sling:resourceType="granite/ui/components/coral/foundation/form/textfield"
                fieldLabel="News Headline"
                name="./headline"/>
        </items>
    </content>
    <content
        jcr:primaryType="nt:unstructured"
        sling:resourceType="granite/ui/components/coral/foundation/form/textarea"
        fieldLabel="News Content"

```

```

        name="./content"/>

<source

    jcr:primaryType="nt:unstructured"

    sling:resourceType="granite/ui/components/coral/foundation/form/textfield"

    fieldLabel="News Source"

    name="./sourceName"/>

<publishDate

    jcr:primaryType="nt:unstructured"

    sling:resourceType="granite/ui/components/coral/foundation/form/textfield"

    fieldLabel="Date Published"

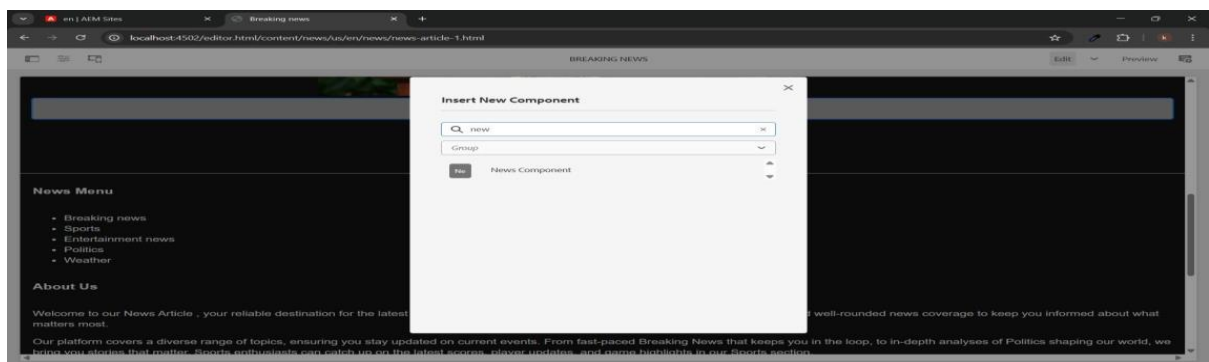
    name="./publishDate"/>

</items>

</content>

</jcr:root>

```



2. Create a Multi-Field Component Using Sling Model

New Component Path:

/apps/news/components/multi-news

1. news-list.html

```

<div data-sly-use="newsListModel" class="news-collection">

    <div data-sly-list="newsItem"="${newsListModel.newsItems}">

        <h2 class="news-heading">${newsItem.headline}</h2>

        <p class="news-source">Source: ${newsItem.sourceName}</p>
    
```

</div>

</div>

2. Sling Model: NewsListModel.java

```
package com.news.core.models;

import org.apache.sling.api.resource.Resource;
import org.apache.sling.models.annotations.DefaultInjectionStrategy;
import org.apache.sling.models.annotations.Model;
import org.apache.sling.models.annotations.injectorspecific.ChildResource;
import org.apache.sling.models.annotations.injectorspecific.ValueMapValue;
import java.util.List;

@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)
public class NewsListModel {

    @ChildResource
    private List<NewsEntry> newsItems;

    public List<NewsEntry> getNewsItems() {
        return newsItems;
    }

    public static class NewsEntry {

        @ValueMapValue
        private String headline;

        @ValueMapValue
        private String sourceName;

        public String getHeadline() {
            return headline;
        }

        public String getSourceName() {
            return sourceName;
        }
    }
}
```

```
}  
}  
}
```

3. Dialog: news-multifield-dialog.xml

```
<jcr:root xmlns:jcr="http://www.jcp.org/jcr/1.0"  
    xmlns:sling="http://sling.apache.org/jcr/sling/1.0"  
    xmlns:core="http://www.day.com/jcr/cq/1.0"  
    jcr:primaryType="cq:Dialog"  
    cq:dialogMode="floating"  
    jcr:title="Manage News Entries">  
    <content jcr:primaryType="nt:unstructured">  
        <items jcr:primaryType="nt:unstructured">  
            <newsItems  
                jcr:primaryType="nt:unstructured"  
                sling:resourceType="granite/ui/components/coral/foundation/form/multifield"  
                fieldLabel="News Entries"  
                name="./newsItems">  
            <field jcr:primaryType="nt:unstructured"  
                sling:resourceType="granite/ui/components/coral/foundation/form/fieldset">  
            <items jcr:primaryType="nt:unstructured">  
                <headline  
                    jcr:primaryType="nt:unstructured"  
                    sling:resourceType="granite/ui/components/coral/foundation/form/textfield"  
                    fieldLabel="News Headline"  
                    name="./headline"/>  
            <sourceName  
                jcr:primaryType="nt:unstructured"  
                sling:resourceType="granite/ui/components/coral/foundation/form/textfield"  
                fieldLabel="News Source"
```

```

        name="./sourceName"/>
    </items>
</field>
</newsItems>
    </items>
</content>
</jcr:root>

```

Create a Base Page Component with Metadata

```

<sly data-sly-use="pageMetadataModel" data-sly-test="${pageMetadataModel}">
    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>${pageMetadataModel.pageTitle}</title>
        <!-- Open Graph Meta Tags for Social Sharing -->
        <meta property="og:title" content="${pageMetadataModel.socialMediaTitle}"/>
        <meta property="og:description"
content="${pageMetadataModel.socialMediaDescription}"/>
        <meta property="og:image" content="${pageMetadataModel.socialMediaImage}"/>

        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <header>
            <h1>${pageMetadataModel.pageTitle}</h1>

```

```
</header>

<main>

    <p>${pageMetadataModel.pageContent}</p>

</main>

</body>

</html>

</sly>
```

Create Custom Page Properties :

Create Custom Page Properties under "Global Properties" with the following fields:

- **Open Graph Title (og:title):** A catchy title that represents your page when shared on social media.
- **Open Graph Description (og:description):** A brief description that gives context about your page content.
- **Open Graph Image (og:image):** The image that will appear when your page is shared. This should be a high-quality, engaging visual.

What is extraClientLibs & How to Use It?

In **Adobe Experience Manager (AEM)**, **extraClientLibs** is a **configuration property** used to include **additional client libraries (CSS, JS, etc.)** that are not part of the primary client library of a component or page.

- **Reusability:** Share client libraries across multiple components.
- **Separation of Concerns:** Keep core styles/scripts separate from custom ones.
- **Performance:** Load additional libraries only when needed.