

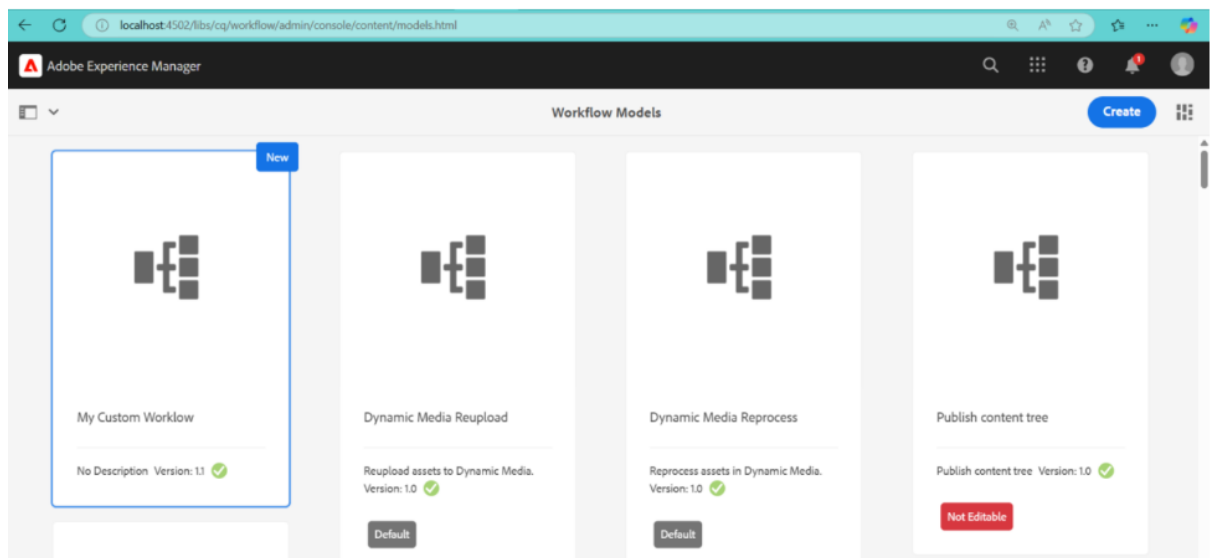
DAY_6_TASK_AEM

-25/03/25

Custom AEM Setup Documentation

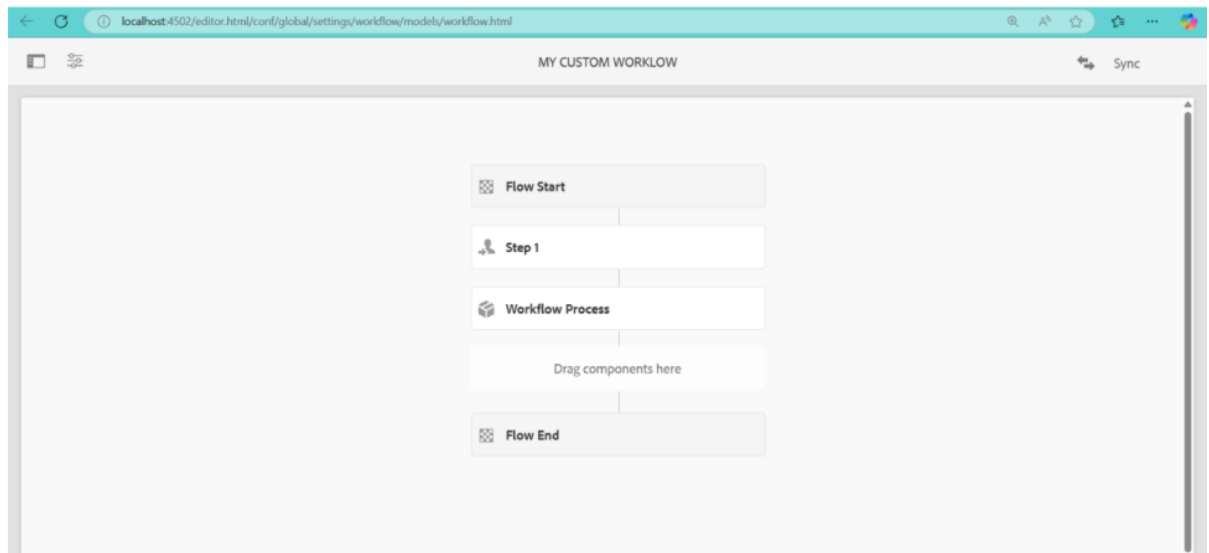
Custom Workflow Creation

- Accessed the Workflow Models section and initiated a new workflow model.
- Assigned the name "My Custom Workflow."
- Incorporated a Process Step, configured its settings, saved the changes, and activated the workflow.



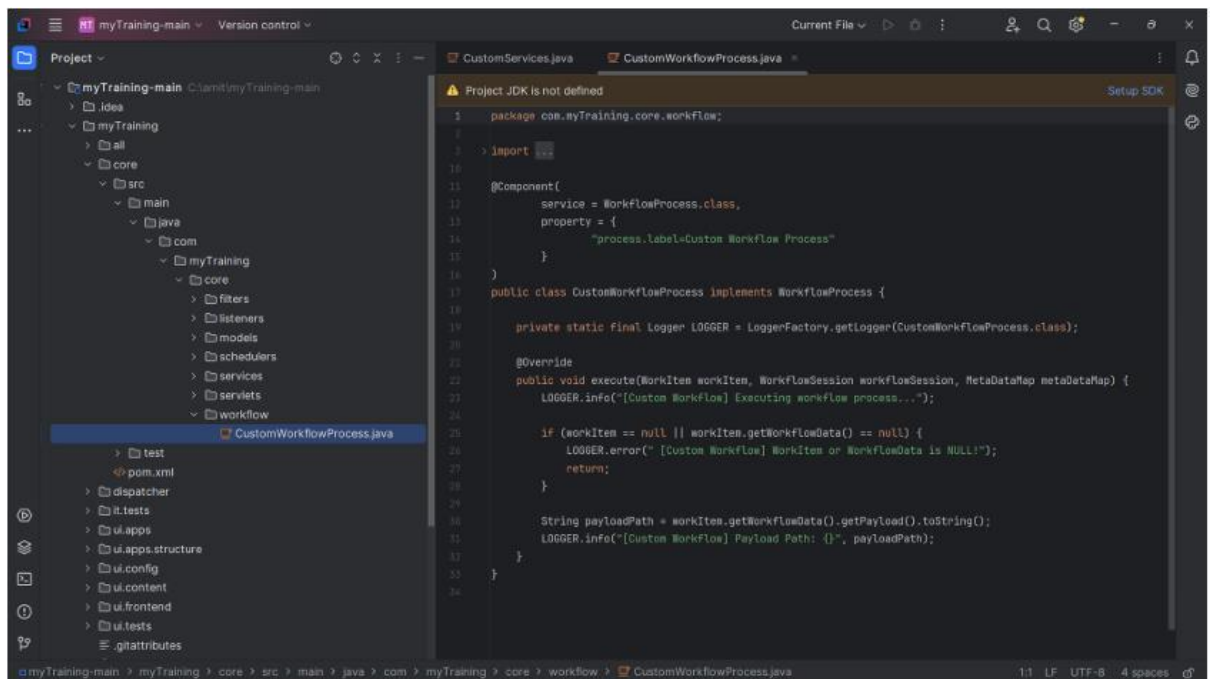
Implementation of a Custom Workflow Process

- Designed a process step to log the title of the page.
- Successfully deployed the process and executed the workflow on a selected page.



Event Handler Development

- Created an event listener to monitor ResourceChanged events.
- Deployed the listener to capture and log modified resource paths.
- Validated the logs upon resource modification.



Create a Sling Job to Print "Hello World" in Logs

Description: A Sling Job will be scheduled to log a simple message

```

import org.osgi.service.component.annotations.Component;
import org.osgi.service.event.Event;
import org.osgi.service.event.EventHandler;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

@Component(service = EventHandler.class, property = { 1 usage
    "event.topics=org/apache/sling/api/resource/Resource/ADDED",
    "event.topics=org/apache/sling/api/resource/Resource/CHANGED",
    "event.topics=org/apache/sling/api/resource/Resource/REMOVED"
})
public class CustomEventHandler implements EventHandler {
    private static final Logger LOG = LoggerFactory.getLogger(CustomEventHandler.class); 1 usage

    @Override
    public void handleEvent(Event event) {
        LOG.info("Resource changed: {}", event.getProperty("path"));
    }
}

```

Scheduler Configuration (Runs Every 5 Minutes)

- Developed a scheduled task utilizing a cron expression.
- Successfully deployed the scheduler.

```

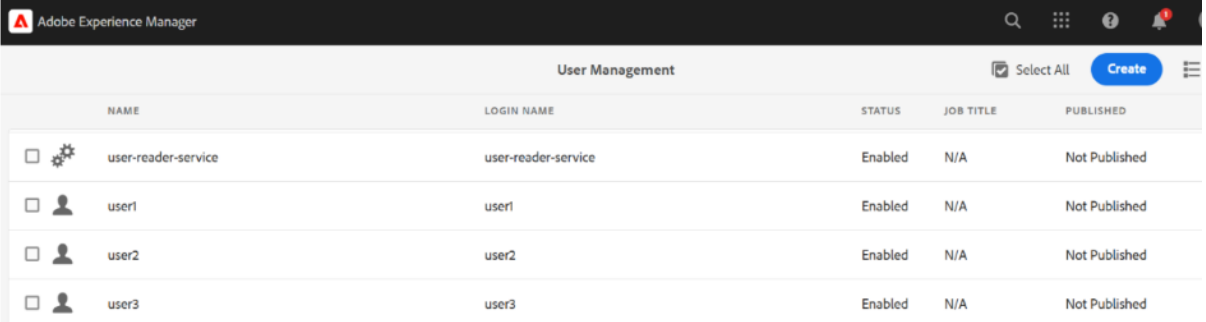
10 import org.slf4j.Logger;
11 import org.slf4j.LoggerFactory;
12
13 import java.util.concurrent.atomic.AtomicBoolean;
14
15 @Component(service = Runnable.class, immediate = true) 2 usages
16 @Designate(ocd = SimpleScheduler.Config.class)
17 public class SimpleScheduler implements Runnable {
18
19     private static final Logger LOG = LoggerFactory.getLogger(SimpleScheduler.class); 2 usages
20
21     private final AtomicBoolean schedulerActivated = new AtomicBoolean(initialValue: false); 1 usage
22
23     @ObjectClassDefinition(name = "Custom AEM Scheduler Configuration" 2 usages
24     public @interface Config {
25         @AttributeDefinition( no usages
26             name = "Cron Expression",
27             description = "Define the cron expression for execution (every 5 minutes)"
28         )
29     }

```

User and Group Configuration

User Creation

- Navigated to Tools > Security > Users.
- Created three distinct users: user1, user2, and user3.

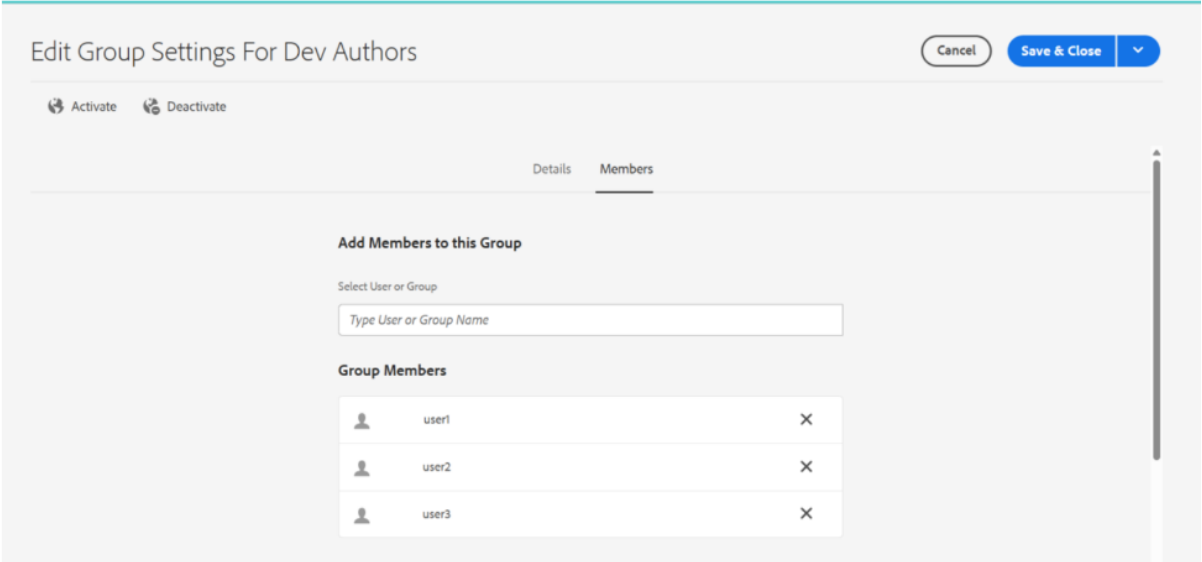


The screenshot shows the Adobe Experience Manager User Management interface. At the top, there's a header with the Adobe Experience Manager logo and a search bar. Below the header, there's a 'User Management' section with a 'Select All' checkbox and a 'Create' button. The main content is a table with columns: NAME, LOGIN NAME, STATUS, JOB TITLE, and PUBLISHED. The table lists four users: user-reader-service, user1, user2, and user3. All users are 'Enabled' and have 'N/A' for 'JOB TITLE' and 'Not Published' for 'PUBLISHED'.

	NAME	LOGIN NAME	STATUS	JOB TITLE	PUBLISHED
<input type="checkbox"/> ⚙️	user-reader-service	user-reader-service	Enabled	N/A	Not Published
<input type="checkbox"/> 👤	user1	user1	Enabled	N/A	Not Published
<input type="checkbox"/> 👤	user2	user2	Enabled	N/A	Not Published
<input type="checkbox"/> 👤	user3	user3	Enabled	N/A	Not Published

Group Formation

- Accessed Tools > Security > Groups.
- Established a new group named "Dev Authors."
- Added the three previously created users to this group.



The screenshot shows the 'Edit Group Settings For Dev Authors' dialog box. At the top, there's a title bar with 'Cancel' and 'Save & Close' buttons. Below the title bar, there are 'Activate' and 'Deactivate' buttons. The main content area has two tabs: 'Details' and 'Members'. The 'Members' tab is selected, showing a section titled 'Add Members to this Group' with a search bar labeled 'Select User or Group' and a text input field 'Type User or Group Name'. Below this, there's a section titled 'Group Members' with a table listing three users: user1, user2, and user3. Each user entry has a person icon and a close button (X).

Edit Group Settings For Dev Authors

Cancel Save & Close

Activate Deactivate

Details Members

Add Members to this Group

Select User or Group

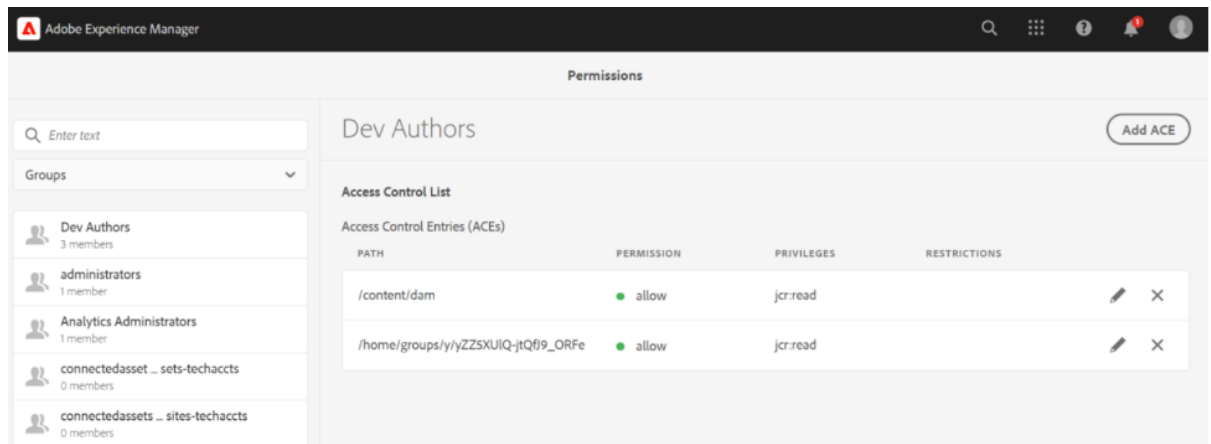
Type User or Group Name

Group Members

👤	user1	×
👤	user2	×
👤	user3	×

Permission Assignments

- Opened User Administration > Permissions.
- Granted Read access for both /content and /dam directories.
- Assigned Replication Access to the "Dev Authors" group.



Final Verification

- Successfully executed workflows, monitored event triggers, and reviewed logs.
- Confirmed scheduled tasks ran as expected.
- Verified user permissions and replication access to ensure proper configurations.