

Create snapshot document custom script - Viewer Action

Imports

```
import java.util.*;
import com.altirnao.aodocs.common.script.*;
import com.altirnao.aodocs.custom.*;
```

User triggered custom code.

```
public List executeWithFeedback(Locale locale, ReadableDocument doc) {
    try {
```

```
        String snapshotClassName = getParam("SNAPSHOT_CLASS_NAME");
        String requesterFieldName = getParam("REQUESTER_FIELD_NAME");
        String requestDateFieldName = getParam("REQUEST_DATE_FIELD_NAME");
        String accessFieldName = getParam("ACCESS_FIELD_NAME");
        String ogcRole = getParam("SNAPSHOT_ROLE");
        String engagementSnapshotRelation = getParam("ENGAGEMENT_SNAPSHOT_RELATION");

        PermissionService permissionService = getPermissionService();
        String currentUser = permissionService.getCurrentUser();

        DocumentService documentService = getDocumentService();
        String documentTitle = doc.getTitle();

        Date today = new Date();
        Calendar calendar = Calendar.getInstance();
        calendar.setTime(today);
        int minute = calendar.get(Calendar.MINUTE);

        Document newSnapshotDocument = documentService.createDocument(snapshotClassName,
documentTitle + " - Snapshot " +
                calendar.get(Calendar.DAY_OF_MONTH) + "/" +
                (calendar.get(Calendar.MONTH) + 1) + "/" +
                calendar.get(Calendar.YEAR) + " " +
                calendar.get(Calendar.HOUR) + ":" +
                (minute < 10 ? "0" : "") + minute + " " +
                (calendar.get(Calendar.AM_PM) == Calendar.AM ? "AM" : "PM"));

        List<String> ogcPeople = permissionService.listRoleUsers(ogcRole);

        newSnapshotDocument.setField(requesterFieldName, currentUser);
        newSnapshotDocument.setField(requestDateFieldName, today);
        newSnapshotDocument.setMultiField(accessFieldName, ogcPeople);

        newSnapshotDocument.setState("Waiting for Snapshot library to be created");

        newSnapshotDocument.addFromRelatedDocument(engagementSnapshotRelation, doc);

        documentService.store(newSnapshotDocument);

        FeedbackHelper feedbackHelper = getFeedbackHelper();
        return feedbackHelper.combine(
            feedbackHelper.successMessage("Snapshot document successfully created"),
```

```
        feedbackHelper.refreshPage()  
    );
```

```
catch (Exception e) {  
    handleError(e, doc);  
    return getFeedbackHelper().errorMessage(e.getClass() +" "+  
e.getMessage());  
}  
}
```