SCAPE

SCAlable Preservation Environments

Notification and Assessment API

Authors

Person	Role	Partner	Contribution
Michael Kraxner	DEV	TUW	

Distribution

Person	Role	Partner
PW Group		

Revision History

Version	Status	Author	Date	Changes
0.1		Michael Kraxner	2014-06-01	First draft

Table of Contents

1. Introduction

The Notification and Assessment API is a set of HTTP endpoints to integrate Planning and Watch. It allows to notify the Planning Component of potential risks or opportunities, and to ask for re-assessment of a preservation plan when one of the influencing factors has changed.

2. Use Cases of the Notification and Assessment API

2.1. Notifying the planner about new risks or opportunities

A planner can store watch triggers in Scout to be notified about changes in the environment. These notifications should also be visible in the planning component.

2.2. Reassessment of plans

In course of the preservation planning workflow a planner defines all criteria which influence her decision to adopt a certain preservation action. Scout can monitor if measurements exceed predefined ranges for a criteria, but cannot evaluate the impact of minor changes on the overall outcome of the evaluation. Therefore the planning component needs to provide a mean to re-assess a plan with respect to changed measurements.

3. Specification

3.1. Notification

A Notification can consist of following properties:

- Message: The message which is shown to the user
- plannerEmail: The email of the planner which should be notified
- planId: The repository identifier of the preservation plan the notification relates to. If provided all related users are notified.
- measureUri: The URI of the measure the notification relates to, in case a trigger was fired.(*optional*)
- value: The actual value of the measurement, which exceeded predefined boundaries. (*optional*)

3.2. Authentication & Authorization

Following the REST[1] recommendations authentication is done by using Basic and Digest Access Authentication mechanism, also known as HTTP Basic Authentication, which is well established and supported throughout the otherwise heterogeneous IT landscape. This implies that every HTTP request has to have a BASE64 encoded user name and password string in the Authentication Header. Therefore encryption by using HTTP over SSL/TLS is strongly recommended.

Authorization is done by the repository depending on the current user and the individual Representations' associated rights and permission metadata.

3.3. HTTP Status codes

The existing HTTP status codes with their individual semantics are used in the context of the connector API.

- **200 OK** This indicates success.
- **201 Created** means that an object has been created in the repository.
- **401 Unauthorized** The request requires authentication. The user should authenticate properly against the repository and repeat the request.
- **403 Forbidden** The server refuses to fulfill the request. Authorization will not help and the request should not be repeated.
- **404 Not Found** The requested resource cannot be found.
- **415 Unsupported Media Type** The media type sent with the requested was not valid**.**
- **500 Internal Server Error** In the case of runtime errors that might be happening while requesting a resource: e.g. Disk full.

3.4. HTTP endpoints

Following is a specification of the HTTP endpoints. These have to be implemented in the planning component, and invoked by the watch component.

3.5. Add a notification

An endpoint to store a notification in the planning component via HTTP PUT. An agent can store a notification, accepted content type: application/json.

Path

/notification

Method

HTTP/1.1 PUT

Consumes

Notification

3.6. Re-Assessment

Path

/assessment/<planid>?alt=<alt-name>&m=<measure Id>&value=<currentval>

Method

HTTP/1.1 GET

Parameter

planId: The repository identifier of the preservation plan.

alt-name: The name of the alternative the measurement relates to

measure Id: the ID of the measure the measurement relates to, e.g for http://purl.org/DP/quality/measures#98 this would be 98.

currentval: The current measurement which requires a re-assessment of the plan.

Produces

The name of the highest ranked alternative. if it differs from the previously selected alternative, otherwise an empty string.

4. Glossary

Representational state transfer (REST) is a style of software architecture for distributed hypermedia systems such as the World Wide Web. The term representational state transfer was introduced and defined in 2000 by Roy Fielding in his doctoral dissertation. Fielding is one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification versions 1.0 and 1.1.