

Module Invitation

- [Fonctionnalités](#)
- [Installation et paramétrage](#)
- [Utilisation avec Shibboleth](#)

Fonctionnalités

Le module invitation est un développement que nous avons demandé à Nuxeo pour permettre notamment le positionnement de droits sur des utilisateurs qui ne se sont jamais connectés dans un contexte shibboleth.

Bien que pensé pour un contexte shibboleth, ce plugin est également utilisable dans un autre contexte. Il peut permettre par exemple à un administrateur d'un workspace de positionner un droit à un utilisateur et d'en informer de dernier en lui envoyant un mail en même temps.

Le détails de l'implémentation technique et fonctionnelle est décrite dans ce [document](#).

Installation et paramétrage

Récupérez les sources (nuxeo-platform-registration-core et nuxeo-user-platform-registration-web) sur : [<https://github.com/nuxeo/nuxeo-platform-user-registration/>]

```
https://github.com/nuxeo/nuxeo-platform-user-registration/]
```

Personnalisez les messages envoyés en modifiant le fichier **nuxeo-platform-user-registration-web/src/main/resources/OSGI-INF/user-registration-contrib.xml** et les templates ftl associés.

Exemple :

```
<extension target="org.nuxeo.ecm.user.registration.UserRegistrationService"
  point="configuration">
  <configuration>
    <requestDocType>UserRegistration</requestDocType>
    <container>
      <docType>UserRegistrationContainer</docType>
      <parentPath>/default-domain</parentPath>
      <name>registrationRequests</name>
      <title>Registration Requests Container</title>
    </container>
    <validationEmail>
      <title>Espace collaboratif : nouvelle invitation</title>
      <template>skin/views/userRegistration/ValidationEmailTemplate.ftl</template>
    </validationEmail>
    <reviveEmail>
      <title>Invitation sur l'espace collaboratif</title>
      <template>skin/views/userRegistration/ReviveEmailTemplate.ftl</template>
    </reviveEmail>
    <validationRelUrl>site/userRegistration/validate/</validationRelUrl>
  </configuration>
</extension>
```

supprimez éventuellement certaines fonctionnalités en modifiant le fichier **nuxeo-platform-user-registration-web/src/main/resources/OSGI-INF/user-registration-actions-contrib.xml**

Utilisez ensuite maven pour faire un jar (mvn package) que vous placerez ensuite dans le répertoire plugins ou bundles du template custom.

Redémarrez Nuxeo : vous devriez alors voir apparaître 3 sous-onglets permettant de gérer les invitations dans l'onglet administration des workspaces.

Le comportement global du module se paramètre au niveau de l'admin center

Utilisation avec Shibboleth

Pour obtenir le comportement attendu avec Shibboleth (l'utilisateur que l'on invite reçoit un mail avec une url de validation pour accepter l'invitation. Lorsqu'il accède à cette dernière, il doit s'authentifier (il devient donc connu de Nuxeo et les droits peuvent alors être positionnés sur son identité "réelle"), il faut procéder à quelques modification sur le module de base.

1 - Dans **nuxeo-platform-user-registration-web/src/main/resources/OSGI-INF/user-registration-contrib.xml**, supprimez ou commentez les lignes suivantes qui permettent un accès à l'url de validation sans authentification :

```
<extension target="org.nuxeo.ecm.platform.ui.web.auth.service.PluggableAuthenticationService"
  point="openUrl">
  <openUrl name="RegisterUser_validate">
    <grantPattern>${org.nuxeo.ecm.contextPath}/site/userRegistration/validate/.*/grantPattern>
  </openUrl>
</extension>
```

2 - Au contraire, pour forcer l'authentification, ajoutez le point d'extension suivant qui pourra être placé par exemple dans le fichier **esup-login-config.xml** qui définit l'authentification shib :

```
<extension target="org.nuxeo.ecm.platform.ui.web.auth.service.PluggableAuthenticationService"
  point="specificChains">
  <specificAuthenticationChain name="invite">
    <urlPatterns>
      <url>(./)/validate/.*/</url>
    </urlPatterns>
    <replacementChain>
      <plugin>BASIC_AUTH</plugin>
      <plugin>SHIB_AUTH</plugin>
    </replacementChain>
  </specificAuthenticationChain>
</extension>
```

3 - Modifiez le fichier **nuxeo-platform-user-registration-web/src/main/java/org/nuxeo/ecm/user/registration/webengine/UserRegistrationObject.java**,

Ajouter :

```
import org.nuxeo.ecm.core.api.CoreSession;
```

puis (ligne 48) :

```
public Object validateTrialForm(@PathParam("requestId")
String requestId) throws Exception {
  UserRegistrationService usr = Framework.getLocalService(UserRegistrationService.class);
  String redirectUrl = ctx.getServerURL() + "/" + BaseURL.getWebAppName();
  try {
    Map<String, Serializable> additionnalInfo = buildAdditionnalInfos();
    // ADD ESUP
    CoreSession coreSession = ctx.getCoreSession();
    String newUser = coreSession.getPrincipal().getName();
    additionnalInfo.put("userinfo:login", newUser);
    // END ADD
    ...
  }
}
```