

API REST Unica Collaborate



Table des matières

Chapitre 1. Collaborate API REST.....	1
API REST - Modèles Collaborate.....	
API REST - Dossiers de modèles Collaborate.....	
API REST - Formulaire Collaborate.....	
API REST - Dossiers de formulaires Collaborate.....	
API REST - Listes Collaborate.....	
API REST - Listes avancées Collaborate.....	
API REST - Campagnes à la demande Collaborate.....	

Chapitre 1. Collaborate API REST

Les API REST Collaborate vous permettent de manipuler des modèles, des dossiers, des formulaires, des listes et des campagnes à la demande.

Présentation

L'API se base sur les principes REST (Representational State Transfer). Il vous est ainsi plus facile d'écrire et de tester des applications. Vous pouvez utiliser votre navigateur pour accéder aux URL, et vous pouvez utiliser n'importe quel client HTTP standard dans n'importe quel langage de programmation pour interagir avec l'API. Tous les appels d'API doivent être intégralement en minuscules.

Adresse URL de base

L'adresse URL de base est `http://<hostname>:<port>/collaborate/api/collaborate/v1`.

Ajoutez un préfixe à l'URL de base pour chaque API répertoriée. Par exemple :

`http://host:port/collaborate/api/collaborate/v1/lists/{project_id}/projectforms/{form_Id}`

Liste des API

Tableau 1. API REST pour Collaborate

Modèles	
get: /templates	Répertorie tous les modèles correspondants au type de modèle indiqué.
get: /templates/{templateId}	Trouve un modèle correspondant à l'ID indiqué.
Dossiers de modèles	
get: /templatefolders	Répertorie l'ensemble des dossiers utilisés pour les modèles.
get: /templatefolders/{folderId}	Trouve un dossier correspondant à l'ID.
post: /templatefolders	Crée un nouveau dossier.
put: /templatefolders/{folderId}	Met à jour les détails d'un dossier.
delete: /templatefolders/{folderId}	Supprime un dossier existant.

get: /templatefolders/{folderId}/templates	Récupère une liste de tous les modèles du dossier indiqué.
get: /templatefolders/{folderId}/templates/{templateId}	Trouve le modèle correspondant à l'ID.
get: /templatefolders/{parentFolderId}/subfolders	Récupère les sous-dossiers d'un dossier parent.
post: /templatefolders/{parentFolderId}/subfolders	Crée un sous-dossier sous le dossier parent.
<u>Formulaires</u>	
get: /forms	Répertorie tous les formulaires.
get: /forms/{formId}	Trouve le formulaire correspondant à l'ID.
get: /forms/{formId}/attributes	Répertorie tous les attributs correspondants à l'ID de formulaire.
get: /forms/{formId}/attributes/{attrID}	Extrait un attribut de formulaire qui correspond à l'ID d'attribut et l'ID de formulaire.
get: /forms/{formId}/offers	Répertorie toutes les offres correspondantes à l'ID de formulaire donné.
get: /forms/{formId}/offers/{offerID}	Extrait une offre selon son ID.
<u>Dossiers de formulaires</u>	
get: /formfolders	Répertorie tous les dossiers utilisés pour les formulaires.
get: /formfolders/{folderId}	Trouve un dossier correspondant à l'ID.
post: /formfolders	Crée un nouveau dossier.
put: /formfolders/{folderId}	Met à jour les détails d'un dossier.
delete: /formfolders/{folderId}	Supprime un dossier existant.
get: /formfolders/{folderId}/forms	Récupère une liste de tous les formulaires du dossier indiqué.
get: /formfolders/{folderId}/forms/{formId}	Trouve un formulaire correspondant à l'ID.
get: /formfolders/{folderId}/forms/{formId}/attributes	Répertorie tous les attributs correspondants à l'ID de formulaire.

get: /formfolders/{folderId}/forms/{formId}/attributes/{attrId}	Extrait un attribut de formulaire correspondant à l'ID d'attribut.
get: /formfolders/{folderId}/forms/{formId}/offers	Répertorie tous les ID d'offres relatifs à l'ID de formulaire indiqué.
get: /formfolders/{folderId}/forms/{formId}/offers/{offerId}	Extrait une offre correspondante à l'ID d'offre indiqué.
get: /formfolders/{parentFolderId}/subfolders	Récupère les sous-dossiers d'un dossier parent.
post: /formfolders/{parentFolderId}/subfolders	Crée un sous-dossier sous le dossier parent.

Listes

get: /lists	Répertorie toutes les listes simples.
post: /lists	Crée une nouvelle campagne de liste.
get: /lists/{projectId}	Trouve une liste simple.
put: /lists/{projectId}	Met à jour les attributs standard d'une campagne de liste simple.
put: /lists/{projectId}/changestate	Met à jour le statut d'une campagne simple.
get: /lists/{projectId}/projectforms	Répertorie tous les formulaires associés à un projet.
get: /lists/{projectId}/projectforms/{formId}	Donne des détails sur un formulaire, ainsi que les valeurs actuelles d'attributs personnalisés.
put: /lists/{projectId}/projectforms/{formId}	Met à jour les valeurs d'attributs et d'offres personnalisés.
get: /lists/{projectId}/workflowrows	Répertorie toutes les tâches de flux de travaux et étapes d'un projet.
get: /lists/{projectId}/workflowrows/{rowId}	Trouve une tâche de flux de travaux ou une étape.
post: /lists/{projectId}/workflowrows/{rowId}/run	Exécute un diagramme.
get: /lists/{projectId}/workflowrows/{rowId}/run/{jobId}	Donne les résultats d'une exécution diagramme qui correspond à l'ID de tâche.
get: /lists/{projectId}/recurrence	Récupère une configuration de récurrence pour une liste.

put: /lists/{projectId}/recurrence	Met à jour une configuration de récurrence pour une liste.
get: /lists/{projectId}/attachments	Répertorie toutes les pièces jointes à un projet.
post: /lists/{projectId}/attachments/	Enregistre une pièce jointe en fonction des détails passés, puis renvoie un objet JSON qui contient les détails de la pièce jointe.
delete: /lists/{projectId}/attachments/{attachmentId}	Supprime une pièce jointe correspondante à l'ID de pièce jointe passé.
get: /lists/{projectId}/attachmentfolders	Répertorie tous les dossiers de pièces jointes d'un projet.
get: /locals/{projectId}/attachmentfolders/{folderId}	Renvoie les détails d'une pièce jointe correspondante à l'ID de dossier passé.
Listes avancées	
get: /advlists	Répertorie toutes les listes avancées.
post: /advlists	Crée une nouvelle liste avancée.
get: /advlists/{projectId}	Trouve une liste avancée.
put: /advlists/{projectId}	Met à jour les attributs standard d'une liste avancée.
put: /advlists/{projectId}/changestate	Met à jour le statut d'une liste avancée.
get: /advlists/{projectId}/projectforms	Répertorie tous les formulaires associés à un projet.
get: /advlists/{projectId}/projectforms/{formId}	Donne des détails sur un formulaire, ainsi que les valeurs actuelles d'attributs personnalisés.
put: /advlists/{projectId}/projectforms/{formId}	Met à jour les valeurs d'attributs et d'offres personnalisés.
get: /advlists/{projectId}/workflowrows	Répertorie toutes les tâches de flux de travaux et étapes d'un projet.
get: /advlists/{projectId}/workflowrows/{rowId}	Trouve une tâche de flux de travaux ou une étape.
post: /advlists/{projectId}/workflowrows/{rowId}/run	Exécute un diagramme.

get: /advlists/{projectId}/ workflowrows/{rowId}/run/{jobId}	Donne les résultats d'une exécution diagramme qui correspond à l'ID de tâche.
get: /advlists/{projectId}/recurrence	Récupère une récurrence d'une liste avancée.
put: /advlists/{projectId}/recurrence	Met à jour une récurrence d'une liste avancée.
get: /advlists/{projectId}/ attachments	Répertorie toutes les pièces jointes à un projet.
get: /advlists/{projectId}/ attachments/{attachmentId}	Renvoie en tant que fichier téléchargeable une pièce jointe valide correspondante à l'ID
post: /advlists/{projectId}/ attachments/	Enregistre une pièce jointe en fonction des détails passés, puis renvoie un objet JSON qui contient les détails de la pièce jointe.
delete: /advlists/{projectId}/ attachments/{attachmentId}	Supprime une pièce jointe correspondante à l'ID de pièce jointe passé.
get: /advlists/{projectId}/ attachmentfolders	Répertorie tous les dossiers de pièces jointes d'un projet.
get: /advlists/{projectId}/ attachmentfolders/{folderId}	Renvoie les détails d'une pièce jointe correspondante à l'ID de dossier passé.
<u>Campagnes à la demande</u>	
get: /locals	Répertorie toutes les campagnes à la demande.
post: /locals	Crée une nouvelle campagne à la demande.
get: /locals/{projectId}	Trouve une campagne à la demande.
put: /locals/{projectId}	Met à jour des attributs standard d'une campagne à la demande.
put: /locals/{projectId}/changestate	Met à jour le statut d'une campagne à la demande.
get: /locals/{projectId}/projectforms	Répertorie tous les formulaires associés à un projet.
get: /locals/{projectId}/ projectforms/{formId}	Donne des détails sur un formulaire, ainsi que les valeurs actuelles d'attributs personnalisés.
put: /locals/{projectId}/ projectforms/{formId}	Met à jour les valeurs d'attributs et d'offres personnalisés.
get: /locals/{projectId}/ workflowrows	Répertorie toutes les tâches de flux de travaux et étapes d'un projet.

get: /locals/{projectId}/ workflowrows/{rowId}	Trouve une tâche de flux de travaux ou une étape.
post: /locals/{projectId}/ workflowrows/{rowId}/run	Exécute un diagramme.
get: /locals/{projectId}/ workflowrows/{rowId}/run/{jobId}	Donne les résultats de l'exécution diagramme qui correspond à l'ID de tâche.
get: /locals/{projectId}/recurrence	Récupère une récurrence d'une campagne à la demande.
put: /locals/{projectId}/recurrence	Met à jour une récurrence d'une campagne à la demande.
get: /locals/{projectId}/attachments	Répertorie toutes les pièces jointes à un projet.
get: /locals/{projectId}/ attachments/{attachmentId}	Renvoie en tant que fichier téléchargeable une pièce jointe valide correspondante à l'ID.
post: /locals/{projectId}/ attachments/	Enregistre une pièce jointe en fonction des détails passés, puis renvoie un objet JSON qui contient les détails de la pièce jointe.
delete: /locals/{projectId}/ attachments/{attachmentId}	Supprime une pièce jointe correspondante à l'ID de pièce jointe passé.
get: /locals/{projectId}/ attachmentfolders	Répertorie tous les dossiers de pièces jointes d'un projet.
get: /locals/{projectId}/ attachmentfolders/{folderId}	Renvoie les détails d'une pièce jointe correspondante à l'ID de dossier passé.

Codes d'état HTTP

Pour chaque demande, les API REST de Collaborate tentent de renvoyer les codes HTTP adéquats. Par exemple :

200 - OK 201 - Nouvel objet créé sur le serveur. Vérifiez l'en-tête de réponse de l'emplacement pour obtenir l'URI de cet objet. 204 - La demande de mise à jour/suppression a abouti. 400 - Demande incorrecte (corrigez l'erreur avant de soumettre à nouveau) 401 - L'utilisateur n'est pas autorisé à effectuer cette action. Demande non autorisée. 404 - Objet demandé introuvable. 422 - Les données non valides sont fournies dans la demande. 500

- Erreur de serveur interne (Vérifiez [<Collaborate_home>/logs/system.log](Collaborate_home/logs/system.log) sur le serveur de l'application Web Collaborate pour plus de détails.) 404 - Introuvable

Plusieurs ressources en ligne fournissent des informations relatives aux codes HTTP.

Infrastructure de sécurité des API HCL Unica

Platform fournit l'infrastructure de sécurité des API implémentées par les produits HCL Unica.

Un ensemble de propriété de configuration sur la page **Paramètres > Configuration** permet aux développeurs de définir la sécurité suivante pour les API fournies par les produits HCL Unica.

- Pour une API de produit spécifique, vous pouvez bloquer l'accès au produit.
- Pour une API de produit spécifique, vous pouvez exiger HTTPS pour la communication entre l'API spécifiée et le produit.
- Pour une API de produit spécifique, vous pouvez exiger l'authentification pour la communication entre l'API spécifiée et le produit.

Les propriétés de configuration qui contrôlent la sécurité de l'API se trouvent sous la catégorie **Unica Platform | Sécurité | Gestion de l'API**. Chaque produit dispose d'un modèle de propriété de configuration que vous pouvez utiliser pour créer de nouveaux paramètres de sécurité pour les API fournies par ce produit.

Vous pouvez définir et modifier les paramètres de sécurité d'une API de manière appropriée pour les tests d'unités ou le déploiement ou au cours du cycle de vie des API.

L'infrastructure de sécurité prend actuellement uniquement en charge les API de Campaign.

L'infrastructure de sécurité Platform prend en charge les deux options d'authentification suivantes pour l'accès aux API protégées. Vous pouvez utiliser l'une ou l'autre, en fonction de votre environnement.

- Les utilisateurs internes qui sont enregistrés sur Platform peuvent être authentifiés à l'aide de leurs données d'identification de connexion Platform pour obtenir un jeton sécurisé.

- Les utilisateurs externes qui font partie d'une fédération Platform qui est configuré pour utilisation, peuvent être authentifiés via le serveur du fournisseur d'identité.

Authentification d'utilisateurs internes avec l'API de connexion Platform

Pour authentifier des utilisateurs internes dans des applications client, utilisez l'API Platform **login** afin de générer des jetons sécurisés. Vous pouvez ensuite appeler n'importe quelle API protégée en transmettant les paramètres requis dans l'en-tête de requête, en plus des paramètres attendus par l'API.

Le filtre de sécurité intercepte ces demandes protégées, les valide et les transmet pour traitement.

Une fois l'utilisateur Platform authentifié, le filtre de sécurité Platform ajoute le nom de connexion de l'utilisateur à la demande en tant qu'attribut de la clé `USER_NAME_STRING` avant de le transmettre au produit pour traitement.

Les jetons sécurisés ont un cycle de vie par défaut de 15 secondes. Une fois ce délai expiré, le jeton ne peut plus être utilisé pour appeler une API protégée. Chaque fois que l'API Platform **login** est appelée pour un utilisateur, l'intégralité des jetons de sécurité précédents pour cet utilisateur est invalidée.

Vous pouvez changer le cycle de vie des jetons sécurisés en définissant la valeur de la propriété **Token lifetime** située sur la page **Paramètres > Configuration** sous la catégorie **Général | Divers**.

Exemple d'URL

```
http[s]://host:port/unica/api/manager/authentication/login/
```

Paramètres d'en-tête

Tableau 2. Paramètres d'en-tête de l'API de connexion avec des utilisateurs internes

Paramètre	Description
m_user_name	Nom de connexion Platform de l'utilisateur interne.
m_user_password	Mot de passe Platform en texte en clair de l'utilisateur interne.

Réponse

Lorsque la connexion aboutit, la réponse est HTTP 200 avec les données JSON suivantes.

- `m_tokenId` - jeton généré de manière aléatoire
- `m_user_name` - nom de l'utilisateur connecté
- `createDate` - horodatage au format affiché dans l'exemple suivant, où le fuseau horaire est IST (heure normale de l'Inde) :

```
Mon Jul 06 18:23:35 IST 2015
```

Lorsque la connexion échoue avec des données d'identification incorrectes, la réponse est HTTP 401 (non autorisé). Lorsque l'API de `login` est configurée pour être bloquée, la réponse est 403 (interdit). Lorsque l'API de `login` est configurée pour utiliser HTTPS, et si elle est appelée sur HTTP, la réponse est 403 (interdit).

Pour déconnecter des utilisateurs internes, utilisez l'API de Platform `logout`.

Déconnexion d'utilisateurs internes avec l'API de déconnexion Platform

Utilisez l'API de Platform `logout` pour déconnecter des utilisateurs internes et supprimer le jeton sécurisé.

L'API de `logout` est protégée par défaut. Les paramètres d'authentification sont prévus dans l'en-tête de requête en regard des clés prédéfinies.

Exemple d'URL

```
http[s]://host:port/unica/api/manager/authentication/logout/
```

Paramètres d'en-tête

Tableau 3. Paramètres d'en-tête de l'API de déconnexion

Paramètre	Description
<code>m_user_name</code>	Nom de connexion Platform de l'utilisateur interne.
<code>m_tokenId</code>	Jeton sécurisé obtenu via l'authentification.
<code>api_auth_mode</code>	Utilisez la valeur <code>manager</code> pour les utilisateurs internes.

Réponse

Lorsque l'authentification aboutit, la réponse est HTTP 200 et le jeton sécurisé est supprimé. Si la réponse est HTTP 200, l'application client doit confirmer la déconnexion.

Lorsque l'authentification échoue, la réponse est HTTP 401.

Authentification des utilisateurs externes et déconnexion via une fédération

Lorsque Platform est intégré avec une fédération prise en charge, les utilisateurs peuvent se connecter à leur propre système et l'application client obtient un jeton via le serveur IdP (Identity Provider) fourni par Platform.

Une fois qu'un utilisateur fédéré est authentifié, son nom de connexion Platform correspondant est ajouté à la requête comme attribut de la clé `USER_NAME_STRING`.

La déconnexion doit être effectuée sur le serveur IdP.

Paramètres d'en-tête

Le tableau suivant décrit les paramètres d'en-tête à utiliser lors de l'authentification via le serveur IdP fourni par Platform.

Tableau 4. Paramètres d'en-tête avec une fédération

Paramètre	Description
<code>f_userId</code>	ID utilisateur dans la fédération.
<code>f_clientId</code>	ID client dans la fédération.
<code>f_spId</code>	ID de fournisseur de services dans la fédération.
<code>f_tokenId</code>	Jeton de connexion unique du serveur IdP.
<code>api_auth_mode</code>	Utilisez la valeur <code>f_sso</code> pour l'authentification fédérée.

Réponse

La réponse est HTTP 200, avec des éléments supplémentaires en fonction de l'API.