
Article

[Lorenzo Scalese](#) · Sept 5, 2022 9m de lecture

[Open Exchange](#)

Présentation de FHIRaaS

Introduction

Cet article vise à donner un aperçu du service FHIR Accelerator Service (FHIRaaS) piloté par l'implémentation de l'application iris-on-fhir, disponible dans OEX développé pour le concours FHIRaaS.

Un tutoriel de base vous guidera dans la configuration d'un déploiement FHIRaaS fonctionnel, comprenant une clé API et un serveur OAuth 2.0.

Une bibliothèque permettant d'utiliser les ressources FHIR par le biais de FHIRaaS est également brièvement évoquée.

Enfin, certaines fonctionnalités de l'application iris-on-fhir sont présentées dans des articles séparés. Vous pouvez consulter le code complet sur le référentiel github de l'application.

Ce contenu sera présenté dans une série de 3 articles.

Ce premier article semble un peu gros, mais ne vous inquiétez pas, c'est parce que j'ai mis beaucoup d'images pour vous aider dans vos étapes de configuration.

FHIRaaS

IRIS fournit déjà un environnement API FHIR [intégré dans IRIS for Health et IRIS Health Connect](#).

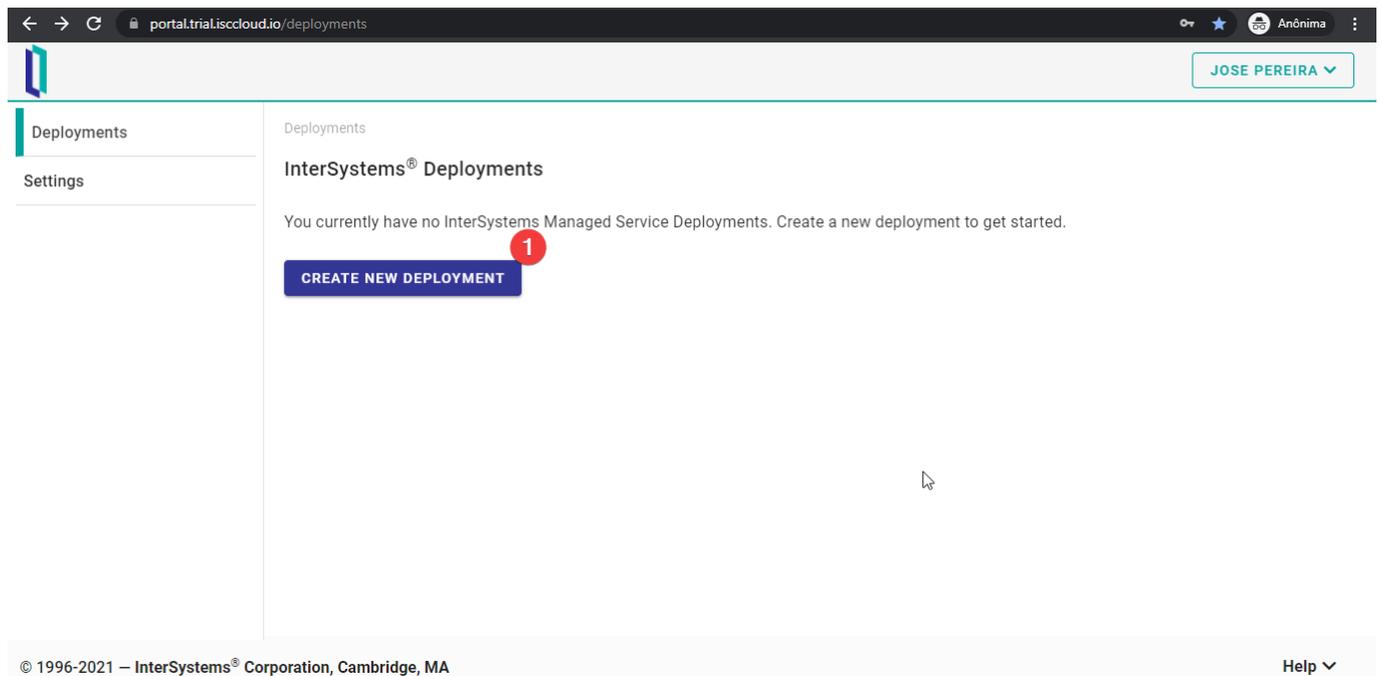
Mais si vous souhaitez profiter de l'environnement fiable, sécurisé et à faible maintenance offert par les services en nuage, vous pouvez désormais compter sur [InterSystems IRIS FHIR Accelerator Service \(FHIRaaS\)](#).

FHIRaaS est une infrastructure FHIR prête à l'emploi basée sur des services en nuage. Il vous suffit de mettre en place un déploiement et de commencer à utiliser l'API FHIR dans vos applications, quelle que soit leur nature : client JS (SMART on FHIR), backend traditionnel ou applications sans serveur.

Pour demander votre essai gratuit de FHIRaaS, veuillez contacter InterSystems.

Configuration du déploiement

Après la connexion, cliquez sur le bouton "CREATE NEW DEPLOYMENT".



Vous devrez passer par quelques étapes. La première consiste à choisir la taille du déploiement. Au moment de la rédaction de cet article, FHIRaaS ne propose qu'une seule option. Il suffit donc d'appuyer sur le bouton "CONTINUE".

L'étape suivante consiste à choisir le fournisseur de services en nuage qui sera utilisé. Là encore, une seule option était disponible au moment de la rédaction de cet article : AWS.

La dernière configuration est juste le nom du déploiement. Il y a quelques règles pour ce nom, et l'interface FHIRaaS vous alerte lorsqu'un nom invalide est fourni. Notez également que vous ne pouvez pas changer ce nom après la configuration. Cliquez sur le bouton "CONTINUE" après avoir sélectionné le nom.

Enfin, révisez votre configuration et démarrez votre déploiement FHIRaaS en cliquant sur le bouton "CREATE".

Si tout se passe bien, vous recevrez un message sympa indiquant que votre déploiement FHIRaaS est en cours de développement. Attendez quelques minutes jusqu'à la fin de ce processus.

Après quelques minutes, votre déploiement FHIRaaS est prêt à être utilisé. Il suffit d'appuyer sur le bouton de déploiement et de commencer à l'utiliser.

Après avoir cliqué sur le bouton de déploiement, l'onglet "Aperçu" est présenté. Notez que vous disposez de plusieurs onglets. Dans cet article, seuls les onglets "OAuth 2.0", "Credentials" et "API Development" seront couverts. Mais il ne s'agit que d'un aperçu, les autres ne sont pas du tout compliqués et vous pouvez facilement les explorer.

Contrôle d'accès

FHIRaaS prend en charge deux méthodes de contrôle d'accès : Clé API et OAuth 2.0. Passons en revue chacune d'entre elles.

Clé d'API

Les clés API sont des jetons générés par FHIRaaS qui permettent à vos applications d'interagir avec l'API sans interaction avec l'utilisateur. Cette méthode est donc idéale pour la communication entre le serveur FHIRaaS et les applications tierces autorisées, comme une API de chatbot, par exemple.

Pour créer une clé API, accédez d'abord à l'onglet "Credentials", puis cliquez sur le bouton "CRÉER UNE CLÉ API".

Choisissez un nom pour votre clé API et cliquez sur le bouton "ADD".

Génial ! Votre clé API a été créée ! Copiez-la et enregistrez-la dans un endroit sûr - vous ne pourrez plus accéder à cette information.

Après la création, vous ne pouvez que supprimer une clé API.

Pour utiliser cette clé Api, il suffit d'ajouter un en-tête x-api-key à une requête HTTP. Par exemple :

```
curl -X GET "https://fhir.lrwvcusn.static-test-account.isccloud.io/Patient" -H "accept: application/fhir+json" -H "x-api-key: your-apy-key"
```

OAuth 2.0 - Création d'un serveur OAuth 2.0 et ajout d'utilisateurs à celui-ci

Comme nous l'avons dit précédemment, la clé API permet d'utiliser l'API sans interaction avec l'utilisateur. Mais, lorsque vous créez une application pour vos utilisateurs, OAuth 2.0 et OpenID Connect sont aujourd'hui la norme industrielle pour l'authentification et l'autorisation.

Une petite remarque : bien que vous puissiez utiliser OAuth 2.0 et OpenID Connect indépendamment, il est tout à fait normal de voir les deux fonctionner côte à côte. Ainsi, lorsque je parle d'OAuth 2.0, je veux dire OAuth 2.0 pour l'autorisation et OpenID Connect pour l'authentification.

Alors, configurons un serveur OAuth 2.0. Tout d'abord, sélectionnez l'onglet "OAuth 2.0", puis cliquez sur le bouton "CREATE AUTHENTICATION SERVER"

L'étape suivante consiste à choisir un nom pour votre serveur OAuth 2.0 et à sélectionner le fournisseur d'identité (IdP) à utiliser. Au moment où cet article a été écrit, FHIRaaS ne supportait que AWS Cognito comme IdP. Il suffit donc d'appuyer sur le bouton "CREATE".

Si votre demande a été réussie, un message vous sera envoyé, comme dans l'image ci-dessous. Maintenant vous pouvez ajouter des utilisateurs à votre IdP en cliquant sur le bouton "ADD USER".

Vous serez redirigé vers l'onglet "Credentials". Pour ajouter un utilisateur, cliquez sur le bouton "CREATE USER".

Saisissez le nom d'utilisateur et son mot de passe, puis cliquez sur le bouton "CREER".

Si tout se passe bien, vous pouvez maintenant voir un utilisateur créé dans votre IdP. Cet utilisateur peut maintenant se connecter à des applications autorisées par ce serveur OAuth 2.0.

OAuth 2.0 - Ajout d'applications au serveur OAuth 2.0

Après la création d'un serveur OAuth 2.0 et l'ajout d'utilisateurs à celui-ci, ces utilisateurs peuvent utiliser les applications autorisées par ce serveur. Maintenant, nous allons ajouter une application au serveur.

Tout d'abord, accédez à l'onglet "OAuth 2.0", sélectionnez "Application", puis cliquez sur le bouton "CREATE APPLICATION".

Ensuite, choisissez un nom pour votre application dans le serveur et le serveur OAuth 2.0 qui vient d'être créé. Les URLs doivent diriger vers votre application. L'"URL de redirection" est l'adresse de destination lorsque les utilisateurs se connectent avec succès. L'"URL de déconnexion" est la page vers laquelle les utilisateurs sont redirigés, lorsqu'ils utilisent l'IdP pour se déconnecter.

Vous pouvez rediriger vers localhost pendant le développement, mais, bien sûr, pour la production, vous devez

fournir une URL accessible par l'Internet.

Les dernières étapes consistent à choisir les ressources FHIR (domaines d'application) que les utilisateurs doivent accepter de partager avec l'application. Pour ce test simple, toutes les ressources sont demandées, mais dans les applications réelles, vous pouvez contrôler chaque ressource FHIR, comme si l'application pouvait juste lire, juste écrire ou les deux. Si les utilisateurs ne sont pas d'accord avec cette demande d'autorisation, le serveur OAuth 2.0 refusera l'accès à ces ressources.

Après avoir correctement configuré les domaines, appuyez sur le bouton "CREER".

Si tout se passe bien, vous verrez apparaître un message vert. Maintenant, vous pouvez vérifier les paramètres de votre nouvelle application ou la supprimer en appuyant sur la case de l'application. Vous pouvez également créer plusieurs applications dont vous avez besoin.

Développement d'API

Parmi les fonctionnalités intéressantes de FHIRaaS, citons l'onglet Développement API. Il vous fournit un explorateur de spécification OpenAPI de l'API FHIR, vous permettant d'essayer facilement toutes les fonctionnalités FHIR.

Pour y accéder, cliquez sur l'onglet "API Development". Après son chargement, vous pouvez sélectionner la ressource FHIR que vous souhaitez explorer. Notez que FHIRaaS fournit la version R4 pour les ressources FHIR.

Maintenant, nous allons nous authentifier pour utiliser l'outil. Vous devez d'abord créer une clé API.

Bien, maintenant, nous allons avoir tous les patients dans cette instance de FHIRaaS :

Comme vous pouvez le voir dans l'animation ci-dessus, vous pouvez effectuer toutes les opérations CRUD sur la ressource Patient - de même pour toutes les autres ressources disponibles.

Ce qui est bien ici, c'est que vous n'avez pas besoin de connaître toute la structure des ressources pour essayer d'effectuer des opérations sur elles. Par exemple, si vous souhaitez créer un nouveau patient, l'outil vous fournit un modèle pour cette ressource :

Vous disposez de la même fonctionnalité pour les autres ressources FHIR.

À la fin de la page, l'outil vous offre une belle vue de toutes les ressources liées, sous forme de schémas :

Conclusion

Dans cet article, nous abordons certains aspects de FHIRaaS et mettons en place un déploiement fonctionnel

Dans le prochain article, nous examinerons quelques exemples simples de son utilisation dans des applications.

[#InterSystems IRIS for Health](#)

[Voir l'application sur InterSystems Open Exchange](#)