

API Integratome – Documentation Technique

Modèle REST avec RPC encapsulé (MessageType)

Version :2.1 – Juin 2026

Auteur : Ugo HABERMAN

Table des matières

Table des matières

API Integratome – Documentation Technique.....	1
Table des matières	2
Introduction et périmètre.....	3
Règles d'intégration.....	3
Schéma des échanges (Séquence).....	3
Authentification.....	4
Environnements.....	4
Payload JSON	4
Réponse	4
Endpoint métier : getRapportPharmacie.....	5
Environnements.....	5
Payload JSON	5

Introduction et périmètre

Cette documentation spécifie les protocoles d'échange par Webservice entre les plateformes Totum et Integratome. L'architecture applicative repose sur un modèle REST sécurisé.

- Le traitement des données par Integratome respecte les principes suivants :
- Absence de persistance : Aucune donnée transmise n'est stockée par Integratome.
- Traitement à la volée : Les rapports d'analyse d'Integratome sont générés en se basant uniquement sur les données reçues dans le corps de la requête.
- Mode synchrone : Les réponses aux requêtes sont traitées et renvoyées de manière synchrone.

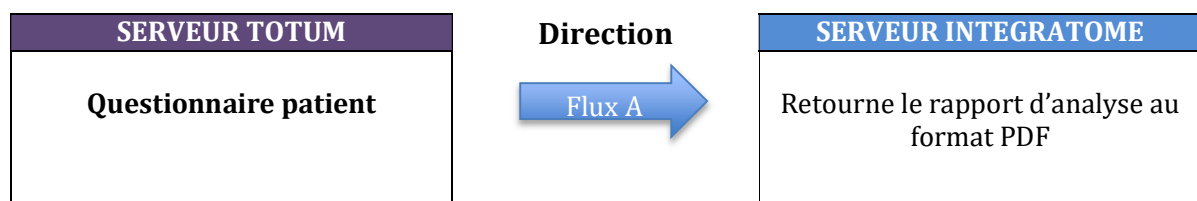
Règles d'intégration

- Pour garantir la sécurité et la conformité des échanges, le client doit se conformer aux règles techniques suivantes :
- Authentification : Tous les appels aux endpoints métiers nécessitent un jeton d'authentification de type Bearer Token valide.
- Gestion du Token : La durée de validité du jeton est strictement limitée à 3600 secondes (1 heure).
- Traçabilité (external_ref) : Le champ external_ref doit être fourni dans les requêtes ; il sert d'identifiant de corrélation unique entre les systèmes Integratome et **Totum**.
- Format des dates : Toutes les dates et horodatages doivent être transmis au format ISO 8601 (ex: YYYY-MM-DDTHH:mm:ssZ).
- Encodage : L'ensemble des payloads (requêtes et réponses) doit être impérativement encodé en UTF-8.

Schéma des échanges (Séquence)

Le tableau ci-dessous représente la séquence logique du flux entre les deux serveurs.

Sens : Sortant Totum | Méthode : POST (WS Integratome)



Authentification

Environnements

Environnement	Base URL
Qualification	https://auth.integratome.fr
Production	https://auth.integratome.com

Payload JSON

```
{
  "client_id": "<fourni par Integratome>",
  "client_secret": "<fourni par Integratome >"
}
```

Voici l'exemple en qualification :

```
[
  'server' => [
    'serverAuth'  => 'https://auth.integratome.fr' // Serveur d'autorisation
    , 'serverApi'  => 'https://api.integratome.fr' // Serveur de ressources
    , 'CUSTOMREQUEST' => 'POST' // Mode d'échange
  ]
  , 'authPayload' => [
    'grant_type'  => 'client_credentials'
    , 'client_id'  => 'TotumQual' // Identifiant du client
    , 'client_secret' => 'a8f4b2c1d9e6f3a0b5c7d4e9f1a8b3c5' // Secret associé à
l'identifiant client
    , 'scope'      => 'pdf_totum:read'
  ]
]
```

Réponse

```
{
  "access_token": qmLP2BwehSYUmPlx1RLvUszWGb7xAJEMoc",
  "token_type": "Bearer",
  "expires_in": 3600,
  "username": "Totum"
}
```

Endpoint métier : getRapportPharmacie

Environnements

Environnement	Base URL
Qualification	https://api.integratome.fr
Production	https://api.integratome.com

Headers

```
{
  'Authorization: Bearer <BearerRecuPrecedement>'
  'Content-Type: application/json',
  'Accept: application/json',
}
```

Payload JSON

```
{
  "MessageType": "PDF_TOTUM",
  "Version": "2.1.0",
  "i18n": "fr_FR",
  "ControlId": "Totum20260520114500",
  "EncodedData": "BASE64_DU_JSON_ENCODE de DATA"
}
DATA
{
  "patientContext": {
    "person": {
      "gender": "M",
      "birth_datetime": "1930-04-26",
      "i18n": "fr_FR"
    },
  },
  "questionnaire": {
    "F3": 1,
    "F6": 62,
    "F77": 165,
    "F78": 68.5,
    "F82": 0,
    "F89": 1
  }
}
```

```
}
```

Réponse

Si message en erreur :

```
{
    'errorCode' : 400,
    'message' : 'Encodage Base64 est invalide'
}
```

Si message traité avec succès :

```
(
    [status] => Accepted
    [results] => Array
        (
            [data] => Array
                (
                    [title] => Rapport d'Analyse
                    [pdf] =>
JVBERi0xLjMKMyAwIG9iago8PC9UeXB1IC9QYWdlCi9QYXJlbnQgMSAwIFIKL1Jlc291cmNlcyAyIDAgUgovQ29udGVu
dHMgNCwAwIFI+PgplbmRvYmoKNCAwIG9iago8PC9GaWw0ZXIgL0ZsYXRlRGVjb2RlIC9MZW5ndGggMTM2Mj4+CnN0cmVh
bQp4nI2ZzXLbNhCA73kK3JoeghAA8XDMHOfQSadTS90Znjq0RKvs0KRDUM7ySH20vkz1FgUJwIAMSnyLbGuobxe7H1cA
TdFPbwrMJfr7zcctev+ZIMJxUaDtA7rdzm8RpXGpkNQ1VhJt9+jtXfX01A8T2v/woava720NNtNw3E3H4d9/fkTbv+YP
FpgQhcyrIOaVMOKGw0sAGgJQhRlHkgus00LEXFqid/bNoUYf0VdEzMXF2etwQIbECCYaSaZxYd06+XC3/eWPze+b7e3P
txuXCPrVpKJLbpLQSYpaigVnWRifvAC2T/eEY1V6RMpluA2tqkYRdJUSqkl9pfm0NUhpFmOLDyLSPNjdRlGqFwKHNpu
67Y9BR6lBqg80NCFvsyjb0lUxLs7jcd2qqYxIiqGSwlcLi0lFiwmf4ac5nxoBqqZjK/9kfU9V3gM0KwItByUoplGfM/
RaRytghK4gSzs0xvi5IwCAJtS8nMfXCW1JfToe72UZsXiShSiUSFwkIkuW5e5arm+EILzMqFL2RikAUlqb4iaYYLHpNu
qmHqD0P19GdTo309ovH70P33WI+pUjbCa6U2qVIFiSNsdr257LH5NtWpVJew/2LF5Qtcsy9f4HoIK0zJJNzNU56aGMI
pbCiwI5JgUtt06Zed8yBVjsmMJExawMbHbVsG/652rcHdv5Hksa5gKsNYxTrIs4AEnaBFsz1RyX9ALJ9RNYPcqwgGb4R
iSZ30ZBkVkdITGLJXQ5sRGMGLy9j1KeqG1FbmXE2PKaeGLiWQE/mHnHriU48saC19KScZ1BEmj05GU+MxP1uaLqm71A9
ocfTVN33bfp1MDHghGtaE0JlJAGt3huTJXlJYHV0xgRSxhgYyRkTSB1jYC3xxgTudWNKjQkDGsPEXJyZKYvEGAtaS4+Y
3RXTMcors2809Tg1D6kg1gwUJIAzgoAW6wXJkrwgsLI5QQIpIwiM5AQJpIwgsA54QQLquiBUY6qBghAxK7oIQhJBLGH1
pHCsWUx68e0IHnqzXzDjZN4yPJvfr3TftM38PZTuHFy4NwtMzbiIw5kte0oNqAbem8DKeA0rpvMmkDLewEjOm0DKeANr
jPcmoK57U5hzGYd5w7XEHfpvaOKNBa0PFobpGcrvWQ6nrpmqNjXEgmGGRNwCq9QQ0GqdIRErYwisbtaQiJQxBEayhkSk
jCGwFjhDItrVQ7g5uYv0yJQ3x0w0qD3mSpbaEAdaNaTgiyEB5UdL2+/6x36qj0MiiWMDJQlognkiCwzBXPkI1UgCLJ2T
JJBSSYAkj0kgpZIAuzCfedjZ8n5rDk1bdtUEKHn/GGCcIlLe4aSZSKIBQH0qVTGJD9Bunp4ro/flp1s3eVmiYsA26NE
EdI9CmzRXpIsyUsCK5+TJJYaksBITpJAykgC64SfJAF1fZKY208Dj8fc+Gc0XUThiSgWBjkkhYhRfPKYnchTM1RTnzsb
Ozpw1gR4bksCW7LXLJLAymsCK5zQJpIwmMJLTJJAYmsD64DUJqOuasAUH04RILO35SYpEEwta10QvmgSU12Ssu7Efjjoz
SCwa0EgCOTNIQKv1hmRJ3hBY3ZwhgZQxBEZyhgRSxhBYC7whAXXdkPkJP/A0XJptjnan4eTJqA0tb1oJl1ixG+a+c49B0
2QdsjgwbIhE400RAY3WKRKyMIrDCWUUuUkYRGmKqEpEYisB64BSJUFcVKXUXP8CAKSKV/++PTB7F0tC6Imp+Fhuhbsxe
6TT0bX9oUj0cFahHgBZYJHrAlur1CKxUD2DRnB6B10oBjDk9AinVA1h/r0dAXdDjF0kgX2gKZW5kc3RyZWFTcmVuZG9i
agoxIDAgb2JqCjw8L1R5cGUgL1BhZ2VzCi9LaWRzIFszIDAgUiBdCi9Db3VudCAxCi9NZWRpYUJveCBbMCAwIDU5NS4y
OCA4NDEuODlkcj4+
                )
            )
        )

    [MessageType] => PDF_TOTUM
    [extractedPatient] => N/A
    [processingQueueId] => ADT-QUEUE-1782194482
    [tokenInfo] => Array
        (
            [subject_client_id] => TotumQual
            [expires_at] => 2026-06-23 09:01:22
        )
    )
)
```

Le contenu 'pdf' est un binaire encodé en Base64.

Pour ouvrir le PDF (ou l'afficher dans un navigateur), vous devrez décoder cette chaîne.

Voici un exemple HTML pour afficher le contenu au format PDF

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <title>Visualiseur PDF</title>
</head>
<body>

  <button id="openPdfBtn">Ouvrir le PDF dans un nouvel onglet</button>

  <script>
    // Récupérez vos données depuis votre API (exemple ici avec la variable PHP injectée)
    const base64Data = "<?php echo $apiResponse['results']['data']['pdf']; ?>";

    document.getElementById('openPdfBtn').addEventListener('click', function() {

      // 1. Convertir le Base64 en tableau d'octets (Uint8Array)
      const byteCharacters = atob(base64Data);
      const byteNumbers = new Array(byteCharacters.length);
      for (let i = 0; i < byteCharacters.length; i++) {
        byteNumbers[i] = byteCharacters.charCodeAt(i);
      }
      const byteArray = new Uint8Array(byteNumbers);

      // 2. Créer un objet Blob à partir des données binaires
      const blob = new Blob([byteArray], { type: 'application/pdf' });

      // 3. Créer une URL temporaire pointant vers le Blob
      const blobUrl = URL.createObjectURL(blob);

      // 4. Ouvrir dans un nouvel onglet
      window.open(blobUrl, '_blank');

      // Optionnel : Nettoyer l'URL après un délai pour libérer la mémoire
      setTimeout(() => URL.revokeObjectURL(blobUrl), 10000);
    });
  </script>
</body>
</html>
```