Webform API – Venue-First – Integration Guide

1. Who This Guide Is For

This guide is for customers exploring or planning to use the Venue-First WebForm API.

Your account is linked either to a Scheme Provider Group (Contract Area) containing multiple Scheme Providers, or to a single Scheme Provider managing multiple schemes, depending on your organisation's setup.

Healthcare / Community Professionals or Self-Referrals select a venue first; the API returns the Scheme Provider ID associated with that venue, which is then used to retrieve the available schemes.

If your WebForm API account is associated with a single Scheme Provider, use the Standard Integration Guide instead.

2. Your API Account

When your Venue-First account is created, ReferAll will provide the following details:

Username

Password

Scheme Provider Group ID

Use the username and password to authenticate. The Scheme Provider Group ID is used only when retrieving the venues available to your account.

3. Why This Flow Is Different

A Standard account begins with a known Scheme Provider ID. A Venue-First account begins with a Scheme Provider Group ID because the specific Scheme Provider is not known until the Healthcare Professional or Self-Referral selects a venue.

The Venue-First flow therefore adds one initial call to retrieve the activity venues available across the Scheme Providers in your group. Each venue is returned with its Scheme Provider ID. Once a venue has been selected, the remaining API calls follow the Standard integration flow.

4. Integration Flow

Step	Call
1	POST /Authorization/Authenticate - get a JWT token
2	GET /Venues/GetAllVenues?schemeProviderGroupId={your group id} - list available venues and their Scheme Provider IDs
3	GET /Schemes/GetAvailableSchemes?schemeProviderId={id from selected venue} - list schemes
4	GET /SchemeConfiguredListItems/GetSchemeConfiguredListItems?schemeId={id} - load scheme-configured form values
5	POST /WebFormSubmit/Submit - submit the completed referral, including the selected venue

Step 2 is the additional step required for a Venue-First account.

5. Step by Step

Step 1 - Authenticate

Exchange the username and password supplied by ReferAll for a JWT token. Use this token as a Bearer token on every subsequent request.

```
POST /v1/Authorization/Authenticate
{ "username": "YourUsername", "password": "YourPassword" }

-> 200 OK
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
```

Step 2 - List available venues

Pass the Scheme Provider Group ID supplied with your account to retrieve the activity venues available across the Scheme Providers in your group.

```
GET /v1/Venues/GetAllVenues?schemeProviderGroupId=3
Authorization: Bearer {token}

-> 200 OK
[
  { "activityVenueId": 3687, "activityVenueName": "Bungola", "schemeProviderId": 5034 },
  { "activityVenueId": 7287, "activityVenueName": "Cally Snow House", "schemeProviderId": 5034 }
]
```

Present the venue list to the patient. When a venue is selected, use the returned schemeProviderId in Step 3 and retain the activityVenueId for submission. An empty array means no venues are currently available for the Scheme Provider Group.

Step 3 - List schemes for the selected venue's Scheme Provider

Use the schemeProviderId returned with the selected venue to retrieve the schemes available for that Scheme Provider.

```
GET /v1/Schemes/GetAvailableSchemes?schemeProviderId=5034
Authorization: Bearer {token}

-> 200 OK
[
  { "schemeId": 5054, "name": "Exercise on Referral",
    "schemeAllowsSelfReferral": true, "schemeAllowsHpReferral": true }
]
```

Step 4 - Load form values for the selected scheme

Pass the selected schemeId to retrieve the scheme-configured values required to build the form, including titles, genders, medical conditions, consent statements and ActivityVenues.

```
GET /v1/SchemeConfiguredListItems/GetSchemeConfiguredListItems?schemeId=5054
Authorization: Bearer {token}

-> 200 OK
{ "Titles": [...], "Genders": [...], "MedicalConditions": [...],
  "ActivityVenues": [...], "SchemeConsents": [...], ... }
```

Important: ActivityVenues is scheme-specific. Before submission, confirm that the activityVenueId selected in Step 2 appears in the ActivityVenues returned for the selected scheme. If it does not, that venue and scheme are not configured as a valid combination.

Step 5 - Submit the referral

Submit the completed referral using the selected schemeId and include the activityVenueId selected in Step 2 in miscellaneous.activityVenues.

```
POST /v1/WebFormSubmit/Submit
Authorization: Bearer {token}
{
  "schemeId": 5054,
  "referralType": 1,
  "patient": { ... },
  "miscellaneous": {
    "activityVenues": 3687,
    "schemeConsents": "1766,1767",
    ...
  }
}

-> 200 OK
"Data submitted successfully"
```

6. Important: Venue and Scheme Pairing

Venue availability is configured per scheme. A venue may be available under the selected Scheme Provider but not enabled for every scheme returned by GetAvailableSchemes.

To avoid an invalid venue/scheme combination, use the ActivityVenues collection returned by GetSchemeConfiguredListItems for the selected scheme and confirm that it contains the activityVenueId chosen in Step 2 before submitting the referral.

7. Common Errors

400 - Invalid schemeProviderGroupId

Your schemeProviderGroupId is not valid, or is not registered to your account. Verify that you are using the schemeProviderGroupId supplied with your API credentials.

400 - Invalid venue/scheme combination

The selected venue is not available for the selected scheme. Always retrieve the schemeProviderId from the selected venue, then obtain the available schemes for that Scheme Provider before submitting the referral.

400 - Validation failed on submit

One or more submitted fields did not match the scheme's configured options (for example, an ID not present in that scheme's configured list) or a required field was missing, such as a mandatory consent. The response body lists which fields failed.

401 - Unauthorized

Your token is missing, expired, or invalid, or the supplied schemeProviderGroupId is not associated with your account. Re-authenticate via Step 1 and verify that you are using the correct schemeProviderGroupId.

For the full technical request/response reference, see the accompanying OpenAPI specification, which can be imported into Postman, Swagger UI, or Scalar.