



PUBLIC

Document Version: 2608 – 2026-08

User Qualification API

SAP Ariba Sourcing

SAP Ariba APIs

SAP Ariba developer portal



Content

User Qualification API.	3
Using the User Qualification API.	4
Developing an Application to Create or Change User Qualification Values.	6
Error Messages Returned by the User Qualification API.	11
Version History for User Qualification API.	15
API-Specific Disclaimers and Legal Information.	16

User Qualification API

This guide applies to:

- SAP Ariba Strategic Sourcing Suite
- SAP Ariba Sourcing

The User Qualification API enables you to create or update user information for the user qualification feature. The user information specifies user qualification field values for specific users. If at least one user field value for each user qualification field defined for a site matches the field value in a project, the user is qualified for the project and can be added to the project team and added to approval flows in the project.

Note

By default, the user qualification feature is not enabled. The user qualification feature is enabled using the site configuration parameter **Enable user qualification**.

Endpoints included in the User Qualification API

The User Qualification API includes the following endpoints:

Endpoint	Description
POST /qualifications	Create user qualification data records for users.
PUT /qualifications	Modify existing user qualification data records for users. All values in the user qualification data records for the specified users are deleted and replaced with the specified values.

Using the User Qualification API

This procedure describes how to use the User Qualification API.

Prerequisites

The user qualification feature must be enabled for your site (the site configuration parameter **Enable user qualification** must be set to **Yes**).

Context

Procedure

1. A member of the **Customer Administrator** group uses the **Import User Qualification Fields** task to define user qualification fields for your site, then uses the **Import User Qualification Field Mappings** task to map these fields to sourcing project header fields.
2. A User Qualification API application sends user values for qualification fields to SAP Ariba.

Results

If an action is performed that can give a user access to a project or document and the user has user qualification fields configured, SAP Ariba compares the project field values with the user's qualification field values to determine if the action can complete or if the action is blocked.

Actions that can give a user access to a project or document include:

- Adding a user to a project team, including adding a user to the **Project Owner** group for a project.
- Adding a user to an approval flow. Active approvers (or reviewers) in an approval flow are given temporary access to the document.
- Delegating capabilities to a user.

Related Information

[Specifying User Qualification Fields and Mapping These Fields to Sourcing or Contract Project Header Fields](#)

Developing an Application to Create or Change User Qualification Values

Use these steps to develop an application to create or change user qualification values.

Prerequisites

- To create an application and request access to use the SAP Ariba APIs relevant to this feature, you must have access to the SAP Ariba Developer Portal. For details refer to [Steps to Start Using the SAP Ariba APIs in Help for the SAP Ariba Developer Portal](#).
- All queries made to this API must be authenticated using OAuth authentication. For details refer to [SAP Ariba Developer Portal Authentication](#) in [Help for the SAP Ariba Developer Portal](#).

Context

Note

In the sample URLs in this topic, replace `{{runtime_url}}` with the desired runtime URL from the **Environment Details** table on the SAP Ariba Developer Portal discovery page for this API.

Procedure

1. Obtain the following information:
 - The ID (user ID) of the user who is creating or changing user qualification field values, such as a member of the **Customer User Admin** group.
 - The ID of each user that you want to configure user qualification field values for.
 - The password adapter for each user. SAP Ariba solutions use password adapters to distinguish between users that have same user names. The default password adapter values are **PasswordAdapter1** (for internal users at your company, the buyer organization) and **SourcingSupplierUser** (for external supplier and customer organization users).
 - The unique name for each user qualification field configured on your site. The unique names are specified in the **UniqueName** column in file imported by the **Import User Qualification Fields** task.
 - The user qualification field values that you want to match for each user.
2. Configure user qualification values for users.

Post a request to the User Qualification API `qualifications` endpoint to create user qualification data records:

```
POST {{runtime_URL}}/qualifications?
user=user&passwordAdapter=passwordAdapter&realm=realm
```

Body:

```
{
  "values": [
    {
      "uniqueName": "userId",
      "passwordAdapter": "passwordAdapter",
      "qualificationInfo": [
        {
          "uniqueName": "userQualificationField",
          "values": [ fieldValues... ]
        }...
      ]...
    }...
  ]
}
```

Table 1: Query String Parameters (Used in All Queries for This API)

Query String Parameter	Description
<code>user</code>	User ID of the person creating (or updating) user qualification data records.
<code>passwordAdapter</code>	Password adapter used to authenticate the <code>user</code> , such as <code>PasswordAdapter1</code> .
<code>realm</code>	Your site name.

Table 2: Request Body Parameters

Name	Description
<code>values</code>	Array with entries for users. Each user entry specifies the user qualification field values used to validate the user for a project.
<code>"uniqueName": "<i>userId</i>"</code>	Specifies the user ID of the user for whom the user qualification field values are defined.
<code>"passwordAdapter": "<i>passwordAdapter</i>"</code>	Specifies the password adapter for the user for whom the user qualification field values are defined, such as <code>PasswordAdapter1</code> .
<code>qualificationInfo</code>	Array with entries for the user qualification fields and field values used to validate the user for a project (determine if the user can be added to the project team or approval flows in the project).

Name	Description
"uniqueName": " <i>userQualificationField</i> "	Specifies the unique name of the user qualification field. The <i>userQualificationField</i> value must match the unique name of a user qualification field configured for your site, as specified in the UniqueName column in the file imported by the Import User Qualification Fields task.
values	Array of user qualification field values for the user. These values are compared with the value or values for the project field mapped to the user qualification field. For simple data types, such as text or a value from an enumerated list of strings, enter the value. For complex data types, such as flex master data objects (records), specify the UniqueName for the object, which is shown in the ID column of the chooser for the project field.

POST request:

```
POST {{runtime_URL}}/qualifications?
user=admin1&passwordAdapter=PasswordAdapter1&realm=mySite
```

Request body:

```
{
  "values": [
    {
      "uniqueName": "UserA",
      "passwordAdapter": "PasswordAdapter1",
      "qualificationInfo": [
        {
          "uniqueName": "Field1",
          "values": [
            "AAA"
          ]
        },
        {
          "uniqueName": "Field2",
          "values": [
            "BBB",
            "CCC"
          ]
        }
      ]
    },
    {
      "uniqueName": "UserB",
      "passwordAdapter": "PasswordAdapter1",
      "qualificationInfo": [
        {
          "uniqueName": "Field1",
          "values": [
            "AAA"
          ]
        },
        {
          "uniqueName": "Field2",
          "values": [
            "DDD"
          ]
        }
      ]
    }
  ]
}
```

```
}
  ]
}
]
```

In this example, the site has 2 user qualification fields defined, `Field1` and `Field2`, that are mapped to custom sourcing project header fields with the same names. The request creates user qualification data records for the users `UserA` and `UserB` with the following results:

- `UserA` is validated for a project (can be added as a project team member or to an approval flow in the project) if both the following conditions are met:
 - The project field `Field1` is set to any of the following values:
 - `AAA`
 - No value (an empty project field value matches any user qualification field value)
 - The project field `Field2` is set to any of the following values:
 - `BBB`
 - `CCC`
 - `BBB and CCC`
 - No value (empty field)
 - `UserB` is validated for a project if both the following conditions are met:
 - The project field `Field1` is set to any of the following values:
 - `AAA`
 - No value
 - The project field `Field2` is set to any of the following values:
 - `DDD`
 - No value (empty field)
3. **Optional:** If you want to change values in a user qualification data record, use a PUT request using the same body syntax for creating user qualification data records.

A PUT request deletes all values in the user qualification data records for the specified users before adding the information in the request, so a PUT request must include values for all user qualification fields for each user entry in the request.

In a PUT request, the `qualificationInfo` for a user can also add entries for new user qualification fields.

Every user specified in the PUT request must have user qualification data record that was previously created with a POST request.

PUT request:

```
PUT {{runtime_URL}}/qualifications?
user=admin1&passwordAdapter=PasswordAdapter1&realm=mySite
```

Request body:

```
{
  "values": [
    {
      "uniqueName": "UserA",
      "passwordAdapter": "PasswordAdapter1",
      "qualificationInfo": [
```

```

    {
      "uniqueName": "Field1",
      "values": [
        "ZZZ"
      ]
    },
    {
      "uniqueName": "Field2",
      "values": [
        "BBB",
        "CCC"
      ]
    }
  ]
}

```

This PUT request changes the configuration created in [2](#) so that **UserA** is validated for projects which have the **Field1** value **ZZZ** instead of **AAA**. Because a PUT request deletes all values in the user qualification data records for the users in the request before adding the new user values in the request, you must also include the current **UserA** values for the **Field2** field.

4. **Optional:** If you no longer want to use the user qualification mechanism for a user, you must delete all values in the user qualification data record for that user. Use a PUT request using the same body syntax for creating a user qualification data record but with an empty **qualificationInfo** array for the user.

PUT request:

```

PUT {{runtime_URL}}/qualifications?
user=admin1&passwordAdapter=PasswordAdapter1&realm=mySite

```

Request body:

```

{
  "values": [
    {
      "uniqueName": "UserB",
      "passwordAdapter": "PasswordAdapter1",
      "qualificationInfo": [
      ]
    }
  ]
}

```

This PUT request turns off user qualification checking for **UserB** (as configured in [2](#)) so that **UserB** can be added to any sourcing project or approval flow. The request contains an entry for **UserB** with an empty **qualificationInfo** array.

Error Messages Returned by the User Qualification API

Use this information to troubleshoot your queries to the User Qualification API.

HTTP Response Code	HTTP Message	Error Code (Body)	Message (Body)	Additional Information
400	The API key passed in the request is not valid.			The API key passed in the query headers does not match the API key in the OAuth token.
400		40001	No request body.	

HTTP Response Code	HTTP Message	Error Code (Body)	Message (Body)	Additional Information									
400		40002	Invalid user qualification info.	Error that contains a details array with additional errors. Possible error and message values for child errors are: <table><tr><th>Error</th><th>Message</th><th>Additional Information</th></tr><tr><td>4000201</td><td>Invalid field value for user <i>userId</i>. Field: <i>fieldName</i>. Value: <i>fieldValue</i>.</td><td>Check the valid values for this user qualification field. For complex data types, such as flex master data objects (records), specify the UniqueName for the object, which is shown in the ID column of the chooser for the project field in the user interface.</td></tr><tr><td>4000202</td><td>The field <i>fieldName</i> for the user <i>userId</i> is not a valid user qualification field.</td><td>Check the UniqueName values of the user qualification fields configured for your site. A member of the Customer</td></tr></table>	Error	Message	Additional Information	4000201	Invalid field value for user <i>userId</i> . Field: <i>fieldName</i> . Value: <i>fieldValue</i> .	Check the valid values for this user qualification field. For complex data types, such as flex master data objects (records), specify the UniqueName for the object, which is shown in the ID column of the chooser for the project field in the user interface.	4000202	The field <i>fieldName</i> for the user <i>userId</i> is not a valid user qualification field.	Check the UniqueName values of the user qualification fields configured for your site. A member of the Customer
Error	Message	Additional Information											
4000201	Invalid field value for user <i>userId</i> . Field: <i>fieldName</i> . Value: <i>fieldValue</i> .	Check the valid values for this user qualification field. For complex data types, such as flex master data objects (records), specify the UniqueName for the object, which is shown in the ID column of the chooser for the project field in the user interface.											
4000202	The field <i>fieldName</i> for the user <i>userId</i> is not a valid user qualification field.	Check the UniqueName values of the user qualification fields configured for your site. A member of the Customer											

HTTP Response Code	HTTP Message	Error Code (Body)	Message (Body)	Additional Information		
				Error	Message	Additional Information
						Administrator and the Customization Administrator groups can use the Export User Qualification Fields task to export these values.
400		40004	You must specify one or more user qualification fields.			No user qualification fields are configured for your site. A member of the Customer Administrator and the Customization Administrator groups must use the Import User Qualification Fields task to configure user qualification fields.
400		40005	You must send a POST request to create user qualification info before you can send a PUT request for this API.			You attempted to use a POST request to edit user qualification field information for one or more users, but there is no user qualification field information for this user or users. Use a PUT request to create information for the users. This error contains a details array with child entries for each entry in the values array that contains a user that does not have user information previously posted. Each entry in the details array contains: "code": 4000501 "message": "Missing user qualification info for user <i>userId</i>."

HTTP Response Code	HTTP Message	Error Code (Body)	Message (Body)	Additional Information
401		40101	Invalid user or users.	Error that contains a details array with child entries for each entry in the values array with an invalid user. Each entry in the details array contains: "code": 4010101 "message": "The user <i>userId</i> is not a valid user."
403		40301	This feature is not available.	The user qualification feature is not enabled for your site. A member of the Customer Administrator group must enable this feature using the Intelligent Configuration Manager (ICM).
429	Rate limit error			You have exceeded the maximum permitted query rate. Refer to the developer portal discovery page for this API for the applicable query rates.
500		50001	An internal error occurred while processing the request.	
500		50002	An internal error occurred while converting the JSON data object.	
504	Gateway Time-out			The server timeout time has been exceeded, possibly due to network traffic or reliability issues. Retry your query.

Version History for User Qualification API

The User Qualification API may have multiple versions, only one of which is active at this time. Previous versions are deprecated and might be decommissioned at a later date.

API Version	Status	As of Release
1	Active	2105 (Q2 2021)

Related Information

[API versioning and deprecation on the developer portal](#)

API-Specific Disclaimers and Legal Information

The SAP Ariba Developer Portal included in the SAP Ariba APIs product and the APIs made available on this site are provided solely at the discretion of SAP without warranty of any kind, and SAP may change, suspend, or cancel any or all features or functions of the SAP Ariba APIs product or revise the website at any time. Any production use of or commercialization of applications containing any APIs provided on this website is prohibited without a written agreement between your company and SAP governing such activities.

Access to this API is available to you as a subscriber to this solution as part of the SAP Cloud Service Level Agreement. However, it is not considered part of the solution. Use of this API is purely optional and is subject to restrictions stated in the documentation, including the Terms of Use and the documentation found at the SAP Ariba Developer Portal (see <https://developer.ariba.com/api>). If you wish to connect a third-party service using this API, first confirm that the company is participating in the SAP partner program and is authorized to provide a connection to this solution using this API. You will be required to submit written consent to SAP to authorize the exchange of data with the third-party service.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
 - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
 - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

Copyright © 2026 Ariba, Inc. All rights reserved.

This documentation, as well as the Ariba solutions, software and/or services described in it, contain proprietary information. They are provided under a license or other agreement containing restrictions on use and disclosure and are also protected by copyright, patent and/or other intellectual property laws. Except as permitted by such agreement, no part of the document may be reproduced or transmitted in any form by any means, electronic, mechanical or otherwise, without the prior written permission of Ariba, Inc.

Ariba, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in the documentation. The information contained in the documentation is subject to change without notice.

Ariba and Ariba products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Ariba, Inc. in the United States and other countries. Please see <http://www.ariba.com/legal/trademarks> for additional trademark information and notices.

Ariba Sourcing solutions (On Demand and software) are protected by one or more of the following patents, including without limitation: U.S. Patent Nos. 6,199,050; 6,216,114; 6,223,167; 6,230,146; 6,230,147; 6,285,989; 6,408,283; 6,499,018; 6,564,192; 6,871,191; 6,952,682; 7,010,511; 7,072,061; 7,130,815; 7,146,331; 7,152,043; 7,225,152; 7,277,878; 7,249,085; 7,283,979; 7,283,980; 7,296,001; 7,346,574; 7,383,206; 7,395,238; 7,401,035; 7,407,035; 7,444,299; 7,483,852; 7,499,876; 7,536,362; 7,558,746; 7,558,752; 7,571,137; 7,599,878; 7,634,439; 7,657,461; 7,693,747; 8,364,577; and 8,392,317. Patents pending.

Other Ariba product solutions are protected by one or more of the following patents:

U.S. Patent Nos. 6,199,050, 6,216,114, 6,223,167, 6,230,146, 6,230,147, 6,285,989, 6,408,283, 6,499,018, 6,564,192, 6,584,451, 6,606,603, 6,714,939, 6,871,191, 6,952,682, 7,010,511, 7,047,318, 7,072,061, 7,084,998; 7,117,165; 7,225,145; 7,324,936; 7,536,362; 8,364,577; and 8,392,317. Patents pending.

Certain Ariba products may include third party software or other intellectual property licensed from a third party. For information regarding software or other intellectual property licensed from a third party, go to <http://www.ariba.com/copyrights.cfm>.

