

alfaview

Developer Handbook



support.alfaview.com

February 19, 2026

Contents

API access	4
Constants	4
Error handling	5
Notes	5
Create group links	6
Summary	6
Request	6
Response	7
Notes	7
Create guest links	8
Summary	8
Request	8
Response	9
Notes	9
Create rooms and departments	10
Summary	10
Request	10
Response	11
Notes	11
Delete guest and group links	12
Summary	12
Request	12
Response	12
Notes	13
Initial authentication with API Key	14
Summary	14
Request	14
Response	14
Notes	15

Initial authentication with Username/Password	16
Summary	16
Request	16
Response	16
Notes	17
List companies	18
Summary	18
Request	18
Response	18
Notes	19
List permission groups.....	20
Summary	20
Request	20
Response	21
Notes	22
List rooms.....	23
Summary	23
Request	23
Response	24
Notes	24
Subsequent authentication with Refresh Token	25
Summary	25
Step 1: Extract refresh token	25
Step 2: Store refresh token	25
Step 3: Generate access token from refresh token	26
Switch company	27
Summary	27
Generate join url from group link	28
Summary	28
Existing group link	28
Request	28
Response	29

Assemble join url	29
Command line options	30
Legal Notice	32

API access

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/api-access/>

- API base URL `https://apis.alfaview.com/json/v1/`
- Common API endpoints
 - Authentication Service `https://apis.alfaview.com/json/v1/authenticationService`
 - Business Logic Service `https://apis.alfaview.com/json/v1/businessLogicService`
 - Company Service `https://apis.alfaview.com/json/v1/companyService`
 - Guest Service `https://apis.alfaview.com/json/v1/guestService.v2`
 - Meeting Service `https://apis.alfaview.com/json/v1/meetingService`
 - Quota Service `https://apis.alfaview.com/json/v1/quotaService`
 - Room Service `https://apis.alfaview.com/json/v1/roomService`
 - User Service `https://apis.alfaview.com/json/v1/userService`
- Username/Password credentials can be created via user invite in the admin interface at <https://app.alfaview.com/#/accounts/>
- API keys need to be created via account management in the admin interface at <https://app.alfaview.com/#/settings/>

Constants

- Define a `requestId` function that generates unique identifiers for debugging requests later on
- DEPRECATED: Define a constant for guest access codes (random string)
- DEPRECATED: Define a constant for join link base url `https://app.alfaview.com/#/join/`
- DEPRECATED: Define a constant for roles (Should now be retrieved from the *Business Logic Service*)
 - Moderator: `{ "voice": true, "video": true, "promote": true, "join": true, "screen": true }`
 - Participant: `{ "voice": true, "video": true, "join": true, "screen": true, "chat": true }`
 - Spectator: `{ "join": true, "chat": true }`

Error handling

Examples

```
{  
  "replyInfo": {  
    "requestId": "z3ai9B11509",  
    "statusMessage": "OK"  
  },  
  ...  
}
```

```
{  
  "code": 16,  
  "message": "token expired",  
  "details": [  
    {  
      "@type": "type.googleapis.com/common.ReplyInfo",  
      "requestId": "tn5BoQ10239",  
      "statusCode": "TOKEN\_OUTDATED",  
      "statusMessage": "token expired"  
    }  
  ]  
}
```

Notes

- API Responses contain a `replyInfo`
- Check containing `statusMessage` to be `OK`
- If error says `token expired` you will have to reauthenticate
- If error says `context deadline exceeded` there was a timeout and the request needs to be reissued

Last updated on July 3rd, 2025, 12:09 pm

Create group links

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/create-group-links/>

Summary

- Group links can be shared with multiple recipients
- The endpoint can create multiple group links at once
- Required API calls: `guestService.v2/createGroupLinks`

Request

Request `POST` to <https://apis.alfaview.com/json/v1/guestService.v2/createGroupLinks>

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  },  
  "roomId": "### ROOM ID ###",  
  "groupLinkCreations": [  
    {  
      "description": "### GROUP LINK TITLE ###",  
      "validFrom": "2022-08-01T13:15:00Z",  
      "validUntil": "2022-08-02T14:00:00Z",  
      "permissionGroupId": "### PERMISSION GROUP ID ###",  
    }  
  ]  
}
```

Response

```
{  
  "replyInfo": {  
    ...  
    "statusMessage": "OK"  
  },  
  "groupLinks": [  
    {  
      ...  
      "joinLink":  
        ↳ "https://app.alfaview.com/#/join/01FHOEX0VQQ378MV7Z5NAGHESK/4173dafb-5cde5c85-9  
        ↳ ,  
      "accessKey": "NcHef8FIc17a5jpiWy4fJ0cw2dCRpbFG"  
    }  
  ]  
}
```

Notes

- `groupLinkCreations` can contain multiple objects, contains the following attributes
 - `description` the title of this group link, not visible for the recipient
 - `validFrom` and `validUntil` define the time frame in which the group link is valid. If only `validFrom` is defined the time range has an open end. If none are defined time range is unlimited. Both values are optional.
 - retrieve the desired `permissionGroupId` from the Business Logic Service
- The response contains a corresponding array of group links, which then contain an individual `joinLink`

Last updated on March 14th, 2025, 10:44 am

Create guest links

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/create-guest-links/>

Summary

- Guest links are non-shareable
- Guest links are for individual recipients
- Can create multiple guest links
- Optionally sends an email in the specified language (DE or EN)
- Required API calls: `guestService.v2/createGuestLinks`

Request

Request `POST` to <https://apis.alfaview.com/json/v1/guestService.v2/createGuestLinks>

```
{
  "accessInfo": {
    "requestId": "### REQUEST ID ###",
    "accessToken": "### ACCESS TOKEN ###"
  },
  "roomId": "### ROOM ID ###",
  "guestLinkCreations": [
    {
      "email": "### GUEST EMAIL ###",
      "displayName": "### GUEST NAME ###",
      "validFrom": "2022-07-25T07:25:48.057Z",
      "validUntil": "2022-07-25T07:25:48.057Z",
      "locale": "### GUEST LOCALE ###",
      "sendEmail": true,
      "permissionGroupId": "### PERMISSION GROUP ID ###",
      "skipRoomPermissionsUpdate": false
    }
  ]
}
```

Response

```
{  
  "replyInfo": {  
    ...  
    "statusMessage": "OK"  
  },  
  "guestLinks": [  
    {  
      ...  
      "joinLink":  
        ↳ "https://app.alfaview.com/#/join/01FHOEX2VQQ378MV7Z5NAGHESK/4173dafb-5cde5c85-9  
        ↳ ,  
      "accessKey": "PHLlurwVkiY91KA1XVc8WtNsKBoRwStK"  
    }  
  ]  
}
```

Notes

- `guestLinkCreations` can contain multiple objects, contains the following attributes
 - `displayName` the name the guest will display during a meeting, optional
 - `validFrom` and `validUntil` define the time frame in which the guest link is valid. If only `validFrom` is defined the time range has an open end. If none are defined time range is unlimited. Both values are optional.
 - set `sendEmail` to true if an invitation email needs to be sent out to the guest
 - `locale` applies to the invitation email, can be `de` or `en`, defaults to `en`
 - retrieve the desired `permissionGroupId` from the *Business Logic Service*
 - `skipRoomPermissionsUpdate` should be set to `false` when creating guest links
- The response contains a corresponding array of guest links, which then contain an individual `joinLink`
- As of today, guest links created with this endpoint will not show up in the administration interface for a rooms guest links (known issue, will be fixed soon)

Last updated on July 3rd, 2025, 11:28 am

Create rooms and departments

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/create-rooms-and-departments/>

Summary

- Create a room or a department anywhere in the room structure
- Required API calls: `roomService/create`

Request

Request `POST` to <https://apis.alfaview.com/json/v1/roomService/create>

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  },  
  "room": {  
    "displayName": "Conference Room 1",  
    "mode": "ROOM\_MODE\_SPECTATOR",  
    "type": "ROOM\_TYPE\_ROOM",  
    "quotas": {  
      "name": "### QUOTA NAME ###"  
    },  
    "permissions": {  
      "### USER ID ###": "### MODERATOR ROLE ###",  
      "### USER ID ###": "### PARTICIPANT ROLE ###",  
      "### USER ID ###": "### PARTICIPANT ROLE ###",  
      ...  
    },  
    "defaultPermissions": "### PARTICIPANT ROLE ###",  
    ...  
  }  
}
```

Response

```
{  
...  
"roomId": "b00430bd-a7ba-59db-aad9-2034464e92b9",  
...  
}
```

Notes

- `type` should be `ROOM_TYPE_ROOM` when creating a room or `ROOM_TYPE_DEPARTMENT` when creating a department
- `mode` should be `ROOM_MODE_SPECTATOR` in order to allow for spectators, or `ROOM_MODE_NORMAL` to disallow spectators (applies to room creation only)
- `parentId` can be set to place the room or department under a parent room or department in the room structure, optional
- `quota` should reflect the customer quota in numbers, or the name of the quota set retrieved from the quota service
- `permissions` can contain a set of user ids and their associated role in this room, optional
- `defaultPermissions` can be set to allow all registered users of the account to enter this room in a default permission set

Last updated on July 3rd, 2025, 12:04 pm

Delete guest and group links

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/delete-guest-and-group-links/>

Summary

- Same process for guest and group links
- Endpoint can only delete one link at a time
- Required API calls: `guestService.v2/deleteLink`

Request

Request `POST` to <https://apis.alfaview.com/json/v1/guestService.v2/deleteLink>

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  },  
  "roomId": "### ROOM ID ###",  
  "id": "### GUEST / GROUP LINK ID ###",  
  "skipRoomPermissionsUpdate": false  
}
```

Response

```
{  
  "replyInfo": {  
    ...  
    "statusMessage": "OK"  
  }  
}
```

Notes

- you can get a list of guest and group links and their ids with the `listGroupLinks` request

Last updated on July 3rd, 2025, 12:07 pm

Initial authentication with API Key

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/initial-authentication-with-api-key/>

Summary

- Creates an access token which can be used throughout the process
- `companyId` is required
- Required API calls: `authenticationService/authenticate`

Request

Request `POST` to `https://apis.alfaview.com/json/v1/authenticationService/authenticate`

```
{  
  "requestId": "### REQUEST ID ###",  
  "authorizationCodeCredentials": {  
    "clientId": "### API KEY ALIAS ###",  
    "code": "### API KEY SECRET ###",  
    "companyId": "### COMPANY ID ###"  
  }  
}
```

Response

```
{  
  ...  
  "accessToken": "...",  
  "expiresAt": "...",  
  ...  
}
```

Notes

- Store the access token for subsequent requests
- Remember to re-authenticate before `expiresAt` timestamp, or you will receive an error that the token has expired

Last updated on July 3rd, 2025, 11:38 am

Initial authentication with Username/Password

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/initial-authentication-with-username-password/>

Summary

- Creates an access token which can be used throughout the process
- Can be checked for expiry
- `companyId` is optional, API response contains all accessible `companyId`s
- Required API calls: `authenticationService/authenticate`

Request

Request `POST` to <https://apis.alfaview.com/json/v1/authenticationService/authenticate>

```
{  
  "requestId": "### REQUEST ID ###",  
  "usernamePasswordCredentials": {  
    "username": "### USERNAME ###",  
    "password": "### PASSWORD ###",  
    "companyId": "### COMPANY ID ###"  
  }  
}
```

Response

```
{  
  ...  
  "accessToken": "...",  
  "expiresAt": "...",  
  ...  
}
```

Notes

- Store the access token for subsequent requests
- Remember to re-authenticate before `expiresAt` timestamp, or you will receive an error that the token has expired

Last updated on July 3rd, 2025, 11:34 am

List companies

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/list-companies/>

Summary

- Translate the company ids from authentication response into company display names
- Required API calls: `companyService/list`

Request

Request `POST` to `https://apis.alfaview.com/json/v1/companyService/list`

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  },  
  "companyIdList": [  
    "### COMPANY ID ###"  
  ]  
}
```

Response

```
{  
  ...  
  "companies": {  
    "A1B2C3D4E5F6G7H8I9J0": {  
      "displayName": "My Company",  
      ...  
    }  
  },  
  ...  
}
```

Notes

- API response for authentication request contains all company ids that are accessible
- Add all company ids from authentication response to the `companyIdList` array

Last updated on July 3rd, 2025, 11:53 am

List permission groups

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/list-permission-groups/>

Summary

- Request all permission groups for the authenticated company
- Permission groups are required for other API calls (e.g. guest link creation)
- Required API calls: `businessLogicService/permissionGroups/list`

Request

Request `POST` to `https://apis.alfaview.com/json/v1/businessLogicService/permissionGroups`

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  }  
}
```

Response

```
{  
  "replyInfo": {  
    "statusMessage": "OK"  
  },  
  "permissionGroups": {  
    "1390a746-5fda-454a-9a10-b21af32e38ed": {  
      "name": "Moderator",  
      "order": 1,  
      "permissions": {  
        "voice": true,  
        "video": true,  
        "promote": true,  
        "join": true,  
        "screen": true,  
        "chat": true,  
        "chatBroadcast": true  
      }  
    },  
    "6067b76c-5450-4fab-b70d-c7ba99eddcf2": {  
      "name": "Participant",  
      "order": 2,  
      "permissions": {  
        "voice": true,  
        "video": true,  
        "join": true,  
        "screen": true,  
        "chat": true  
      }  
    },  
    "885e356d-02dd-4eec-960c-cd647f4b859a": {  
      "name": "Spectator",  
      "order": 3,  
      "permissions": {  
        "join": true,  
        "chat": true  
      }  
    },  
    ...  
  }  
}
```

Notes

The `permissionGroups` array carries the permission group Ids as the key

Last updated on July 3rd, 2025, 12:02 pm

List rooms

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/list-rooms/>

Summary

- Request all rooms in order to populate a dropdown menu for room selection
- Required API calls: `roomService/list`

Request

Request `POST` to `https://apis.alfaview.com/json/v1/roomService/list`

```
{  
  "accessInfo": {  
    "requestId": "### REQUEST ID ###",  
    "accessToken": "### ACCESS TOKEN ###"  
  },  
  "limit": 0,  
  "offset": 0,  
  "filterRoomIds": [  
    "### ROOM ID ###"  
  ]  
}
```

Response

```
{  
  ...  
  "rooms": {  
    "2bed6af5-51b9-4188-be98-25b6623b7a0f": {  
      "displayName": "Conference room 2",  
      ...  
    },  
    "b495c2b7-7f43-49c5-81b2-e44d429bb215": {  
      "displayName": "Conference room 3",  
      ...  
    },  
    "c134df6f-990a-4f99-8020-8290295e840f": {  
      "displayName": "Conference room 1",  
      ...  
    }  
  }  
}
```

Notes

- `limit`, number of results, default:100 max:1000
- `offset`, for paging, default:0
- `filterRoomIds`, list of room IDs if you want to filter your query
- Note that rooms are not ordered in any way

Last updated on July 3rd, 2025, 11:55 am

Subsequent authentication with Refresh Token

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/subsequent-authentication-with-refresh-token/>

Summary

- Retrieve and store refresh token from response header of initial authentication
- Generate an access token from the refresh token
- Use access token for subsequent API operations
- `companyId` is required
- Required API calls: `authenticationService/authenticate`

Step 1: Extract refresh token

- Upon successful initial authentication the refresh token will be issued in the set-cookie response header of the authentication request

Response header set-cookie example

- The refresh token is the value of the `auth` key

```
auth=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX; Path=/;
Domain=app.alfaview.com; Expires=Sat, 19 Nov 2022 15:18:15 GMT; HttpOnly;
Secure
```

Step 2: Store refresh token

- Browser based applications use cookies to store the refresh token
- Other applications may use operating system specific solutions, such as the Data Protection API on Windows or the KeyChain Access on macOS

Step 3: Generate access token from refresh token

Request **POST** to <https://apis.alfaview.com/json/v1/authenticationService/authenticate> including a **COOKIE** with key **auth** and value of the refresh token

```
{  
  "requestId": "### REQUEST ID ###",  
  "refreshTokenCredentials": {  
    "companyId": "### COMPANY ID ###",  
  }  
}
```

Response

```
{  
  ...  
  "accessToken": "...",  
  "expiresAt": "...",  
  ...  
}
```

Notes

- A refresh token is always tied to a **companyId**. Store them as a pair.

Last updated on July 3rd, 2025, 11:49 am

Switch company

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/switch-company/>

Summary

- If the initial authentication is done without the optional `companyId`, the authentication might log you into the wrong company (first returned accessible company will be used)
- In order to switch the company, simply authenticate again, and pass the desired `companyId`
- Applies to Username/Password authentication and Refresh Token authentication
- Required API calls: `authenticationService/authenticate`

Last updated on July 3rd, 2025, 11:51 am

Generate join url from group link

Source: <https://support.alfaview.com/en/developers/request-your-api-credentials/usage-examples/generate-join-url-from-group-link/>

Summary

- Extract `companyId`, `roomId`, and `accessKey` from existing group link
- Authenticate with extracted information
- Assemble join url
- Required API calls: `guestService.v2/authenticate`

Existing group link

```
// Format:  
https://app.alfaview.com/#/join/<COMPANYID>/<ROOMID>/<ACCESSKEY>/v2  
// Example:  
https://app.alfaview.com/#/join/01EZYCFW9MJ10P15YJ63NZ8S47/5726d492-6eb2-  
5f59-9ee8-203181250dc4/rgOUSiy2pFWkRFNfnkXuJRVXcD72Go0b/v2  
// Extracted:  
companyId: 01EZYCFW9MJ10P15YJ63NZ8S47  
roomId: 5726d492-6eb2-5f59-9ee8-203181250dc4  
accessKey: rgOUSiy2pFWkRFNfnkXuJRVXcD72Go0b
```

Request

Request `POST` to <https://apis.alfaview.com/json/v1/guestService.v2/authenticate>

```
{  
  "companyId": "string", // extracted companyId  
  "roomId": "string", // extracted roomId  
  "accessKey": "string", // extracted accessKey  
  "displayName": "string", // guest name  
  "externalId": "string" - // optional, can be used to derive a  
  // reproducible userId for groupLinks to prevent simultaneous use.  
  // The caller is responsible to provide a unique value per user;  
  // validate="optional,min=32,max=256"  
}
```

Response

```
{  
  "replyInfo": {  
    "statusMessage": "OK"  
  },  
  "accessToken": "string", // use this to assemble the join url  
  "tokenExpiresAt": "string", // expires 12 hours after creation  
  "userId": "string",  
  "linkInfo": {  
    "type": "TYPE\_GROUP\_LINK"  
  }  
}
```

Assemble join url

```
// Format for web client:  
https://webclient.alfaview.com/?--url=alfaview://apis.alfaview.com:443?comp  
↳ anyId=<COMPANY>&roomId=<ROOMID or MEETINGID>&token=<ACCESSTOKEN>  
// Format for native client:  
alfaview://apis.alfaview.com:443?companyId=<COMPANYID>&roomId=<ROOMID or  
↳ MEETINGID>&token=<ACCESSTOKEN>  
// Example:  
alfaview://apis.alfaview.com:443?  
companyId=01EZYCFW9MJ10P15YJ63NZ8S47&roomId=5726d492-6eb2-5f59-9ee8-  
203181250dc4&token=IRXPAVVc1nZC0u5+mKTfXeUQr5bf/C1bSJt3Pd0ibtpDSdineL5trH0  
BuArpIqfBo41sh7S5gPxntpEBYl0wAIu6pT700f1nx2p8Idc7CDZUy6aVhYTa70jc3Pzu0k6UZ  
twi43ixjqhKvSIY0JYdQrExlvjWhUZVuIGf6Wnk50D73z4YmQnVLQ9c1ye6/u4l5p6fe0rgTP1  
6mKCMnH/CXb5Y21HHGdxT4ENS9Vpg0wI5ihI4xZ7Rc3VjkRpWEUDtmeKwSn1NhQkcuFezlWBQr  
sI5kvxp9R0vE1aw
```

Last updated on March 14th, 2025, 10:46 am

Command line options

Source: <https://support.alfaview.com/en/developers/command-line-options/>

Command	Usage
--allowmultiple	This command allows you to run multiple alfaview instances simultaneously on Windows and Linux operating systems. Note: This command does not work if you join an alfaview room using an invitation link.
--autoacceptwelcome	This command prevents the accessibility settings, general settings as well as audio and video settings from being set up before joining an alfaview room for the first time. Changes to these settings can be made in the app or via the webclient later on.
--bypasssettings	This command prevents the saved settings from being applied. Instead, alfaview opens with the default settings. Any settings made during this session are not saved.
--disableleaveroom	This command prevents users from leaving the alfaview room and returning to the login page or the room and meeting list. The Leave room icon is not displayed to users in the alfaview room.
--help	This command displays all available help information.
--loglevel	This command can be used to output log entries by level. Valid values are: debug / info / warning / error / critical. The command value --loglevel=debug corresponds to the command --verbose. Note: You will receive all log entries of the level info (default) until the level you set is set by the alfaview application.
--nocolor	This command deactivates the coloring in the command center terminal.

Command	Usage
--noupdate	This command prevents alfaview from automatically checking for new updates. Note: Update alfaview regularly. Older alfaview versions are gradually excluded from usage.
--skipportcheck	This command prevents alfaview from checking ports and determining whether a connection can be established. alfaview is launched directly. We do not recommend this command if it is not absolutely necessary.
--skipquicksetup	This command deactivates the Quick settings before a user joins an alfaview room. Instead, the user will directly join the alfaview room, where they can adjust the settings.
--url	This command is automatically set for invitation links. It opens the alfaview room for which the join permission has been granted. For an example usage, please refer to page Generate join url from group link .
--verbose	This command activates the extended debug logging (more details).
--version	This command displays the alfaview version currently in use.

Last updated on March 21st, 2025, 09:13 am

Legal Notice

Publisher

alfaview GmbH

Kriegsstr. 100
76133 Karlsruhe, Germany

Website: www.alfaview.com

Email: info@alfaview.com

Phone: +49 721 35450-450

Fax: +49 721 35450-69

Managing Director: Nikolaos Fostopoulos

Court Registration: District court Mannheim HRB 730903

VAT ID: DE318911962

Technical Service Provider

alfatraining Bildungszentrum GmbH

Kriegsstraße 100
76133 Karlsruhe, Germany

Managing Director: Nikolaos Fostopoulos

Court Registration: District court Mannheim HRB 730739

VAT ID: DE318522740

Copyright

The website design is protected by copyright. All images on this website are protected by copyright. The downloading of pictures, their further publication, further transmission, reproduction or other use is prohibited.

Dispute Resolution

The European Commission provides an Online Dispute Resolution platform at <https://ec.europa.eu/consumers/odr>. alfaview GmbH does not participate in consumer arbitration procedures.