

User Extra

Endpoints Summary

Method	Path	Swagger
GET	/user_extra/	Swagger ↗
POST	/user_extra/	Swagger ↗
GET	/user_extra/{id}/	Swagger ↗
PUT	/user_extra/{id}/	Swagger ↗
PATCH	/user_extra/{id}/	Swagger ↗
DELETE	/user_extra/{id}/	Swagger ↗

“ The User Extra API provides endpoints for managing extended user profile information beyond basic authentication details. These endpoints allow you to store and retrieve additional user attributes such as contact information, demographics, custom fields, and location data, enabling rich user profiles for your GATE-integrated applications.

Base URL: <https://gate.zequenze.com/api/v1>

Authentication: All endpoints require a Bearer token:

```
Authorization: Bearer <your-api-token>
```

Overview

The User Extra API category enables you to manage supplementary user information that extends beyond the core user authentication data. This API is essential for applications that need to collect and maintain detailed user profiles, including personal information, contact details, demographic data, and custom business-specific fields.

Key Features:

- **Extended Profile Management:** Store up to 8 custom fields plus standard demographic and contact information
- **Flexible Data Storage:** Support for various data types including dates, strings, integers, and enumerations
- **Parent-Child Relationships:** Link extra information to parent user accounts and filter by organization
- **Device Information:** Optional inclusion of device-related data for enhanced user tracking
- **Full CRUD Operations:** Complete create, read, update, and delete capabilities

Common Use Cases:

- Building comprehensive user registration forms with extended profile fields
- Collecting demographic data for analytics and reporting
- Managing customer contact information for support and communication
- Storing event-specific attendee information
- Maintaining custom business fields specific to your organization's needs
- Tracking user preferences and locale settings

Data Model: The User Extra resource connects to a parent user account and can store personal details (name, birthday, gender), location information (address, city, country, postal code), contact details (telephone, email preferences), business information (company, tax ID, passport number), and flexible custom fields for organization-specific data.

Endpoints

GET /user_extra/

Description: Retrieves a paginated list of user extra information records. This endpoint supports filtering by parent user, organization, and includes optional device information. Use this endpoint to browse all extended user profiles, filter by specific users or organizations, or retrieve bulk user data for reporting and analytics.

Use Cases:

- Retrieve all user profiles within a specific organization for reporting
- Search for users with specific parent relationships
- Export extended user data for analytics or data migration
- Build user directories with detailed profile information

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/?parent__organization_id=123&limit=50&offset=0&device_info=true
```

Parameters:

Parameter	Type	In	Required	Description
parent	string	query	No	Filter by parent user ID. Returns only user extra records linked to the specified parent user.
parent__organization_id	string	query	No	Filter by organization ID. Returns user extra records where the parent user belongs to the specified organization.
limit	integer	query	No	Number of results to return per page. Default is typically 10-100 depending on server configuration. Use for pagination control.
offset	integer	query	No	The initial index from which to return results. Use with limit for pagination (e.g., offset=20&limit=10 returns records 21-30).
device_info	boolean	query	No	When set to true, includes device information associated with the user extra records in the response.

cURL Example:

```
curl -X GET
"https://gate.zequenze.com/api/v1/user_extra/?parent__organization_id=456&limit=25&offset=0" \
-H "Authorization: Bearer YOUR_API_TOKEN" \
-H "Content-Type: application/json"
```

Example Response:

```
{
  "count": 150,
  "next":
  "https://gate.zequenze.com/api/v1/user_extra/?limit=25&offset=25&parent__organization_id=456",
  "previous": null,
  "results": [
    {
      "id": 1,
      "parent": 42,
      "title": "Dr.",
      "gender": "f",
      "birthday": "1985-06-15",
      "age": 38,
      "age_range": "35-44",
      "telephone": "+1-555-0123",
      "address": "123 Main Street",
      "city": "San Francisco",
      "state": "CA",
      "postal_code": "94102",
      "country": "United States",
      "country_id": "US",
      "company": "Tech Solutions Inc",
      "tax_id": "123-45-6789",
      "passport_number": null,
      "hometown": "Seattle",
      "location": "San Francisco, CA",
      "locale": "en_US",
      "event": "TechConf 2024",
      "accept_terms": true,
      "custom_field_1": "Premium Member",
      "custom_field_2": "Referral: LinkedIn",
      "custom_field_3": null,
      "custom_field_4": null,
      "custom_field_5": null,
      "custom_field_6": null,
      "custom_field_7": null,
      "custom_field_8": null
    },
    {
      "id": 2,
```

```

    "parent": 43,
    "title": "Mr.",
    "gender": "m",
    "birthday": "1990-03-22",
    "age": 33,
    "age_range": "30-34",
    "telephone": "+1-555-0456",
    "address": "456 Oak Avenue",
    "city": "Austin",
    "state": "TX",
    "postal_code": "78701",
    "country": "United States",
    "country_id": "US",
    "company": "Startup Ventures LLC",
    "tax_id": "987-65-4321",
    "passport_number": "X12345678",
    "hometown": "Houston",
    "location": "Austin, TX",
    "locale": "en_US",
    "event": "TechConf 2024",
    "accept_terms": true,
    "custom_field_1": "Standard Member",
    "custom_field_2": "Referral: Website",
    "custom_field_3": "VIP Access",
    "custom_field_4": null,
    "custom_field_5": null,
    "custom_field_6": null,
    "custom_field_7": null,
    "custom_field_8": null
  }
]
}

```

Response Codes:

Status	Description
200	Success - Returns paginated list of user extra records
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - Authenticated user lacks permission to view these records

Status	Description
400	Bad Request - Invalid query parameters provided

POST /user_extra/

Description: Creates a new user extra information record. This endpoint allows you to add extended profile information for a user by linking it to a parent user ID. All fields except the parent ID are optional, providing flexibility to collect only the information relevant to your application. Use this when onboarding new users or collecting additional profile information.

Use Cases:

- Complete user registration by adding extended profile information after account creation
- Collect attendee information during event registration
- Add demographic and contact details for marketing segmentation
- Store custom business-specific fields during user onboarding

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/
```

Parameters:

Parameter	Type	In	Required	Description
data	object	body	Yes	JSON object containing the user extra information fields. Only the <code>parent</code> field is required.

Request Body Schema:

Field	Type	Required	Description
parent	integer	Yes	The ID of the parent user account this extra information belongs to
title	string	No	User's title or honorific (e.g., Dr., Mr., Ms., Prof.)
gender	string	No	User's gender. Allowed values: "m" (male), "f" (female), "o" (other)

Field	Type	Required	Description
birthday	string	No	User's date of birth in YYYY-MM-DD format
age	integer	No	User's age in years
age_range	string	No	Age range category (e.g., "25-34", "35-44")
telephone	string	No	User's telephone number
address	string	No	Street address
city	string	No	City name
state	string	No	State or province
postal_code	string	No	Postal or ZIP code
country	string	No	Country name
country_id	string	No	ISO country code (e.g., "US", "GB", "DE")
company	string	No	Company or organization name
tax_id	string	No	Tax identification number
passport_number	string	No	Passport number
hometown	string	No	User's hometown
location	string	No	Current location description
locale	string	No	Locale/language preference (e.g., "en_US", "es_ES")
event	string	No	Associated event name or identifier
accept_terms	boolean	No	Whether user has accepted terms and conditions
custom_field_1 to custom_field_8	string	No	Flexible custom fields for organization-specific data

cURL Example:

```
curl -X POST "https://gate.zequenze.com/api/v1/user_extra/" \
-H "Authorization: Bearer YOUR_API_TOKEN" \
-H "Content-Type: application/json" \
-d '{
  "parent": 42,
  "title": "Ms.",
  "gender": "f",
```

```
"birthday": "1992-08-14",
"telephone": "+1-555-0789",
"address": "789 Pine Street",
"city": "Portland",
"state": "OR",
"postal_code": "97201",
"country": "United States",
"country_id": "US",
"company": "Creative Agency Co",
"locale": "en_US",
"accept_terms": true,
"custom_field_1": "Gold Tier",
"custom_field_2": "Newsletter: Yes"
}'
```

Example Response:

```
{
  "id": 125,
  "parent": 42,
  "title": "Ms.",
  "gender": "f",
  "birthday": "1992-08-14",
  "age": null,
  "age_range": null,
  "telephone": "+1-555-0789",
  "address": "789 Pine Street",
  "city": "Portland",
  "state": "OR",
  "postal_code": "97201",
  "country": "United States",
  "country_id": "US",
  "company": "Creative Agency Co",
  "tax_id": null,
  "passport_number": null,
  "hometown": null,
  "location": null,
  "locale": "en_US",
  "event": null,
  "accept_terms": true,
```

```
"custom_field_1": "Gold Tier",
"custom_field_2": "Newsletter: Yes",
"custom_field_3": null,
"custom_field_4": null,
"custom_field_5": null,
"custom_field_6": null,
"custom_field_7": null,
"custom_field_8": null
}
```

Response Codes:

Status	Description
201	Created - User extra record successfully created
400	Bad Request - Invalid data provided (e.g., missing parent ID, invalid date format, invalid gender value)
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - User lacks permission to create user extra records
404	Not Found - Parent user ID does not exist

GET /user_extra/{id}/

Description: Retrieves a specific user extra information record by its unique ID. This endpoint returns all stored extended profile information for a single user, optionally including device information. Use this to display user profile details, verify stored information, or retrieve data for editing.

Use Cases:

- Display complete user profile information on a user details page
- Retrieve user data before updating to show current values in an edit form
- Verify user information for customer support or verification purposes
- Fetch user demographics for personalized content delivery

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/125/?device_info=true
```

Parameters:

Parameter	Type	In	Required	Description
id	integer	path	Yes	The unique identifier of the user extra record to retrieve
device_info	boolean	query	No	When set to true, includes device information associated with this user extra record

cURL Example:

```
curl -X GET "https://gate.zequenze.com/api/v1/user_extra/125/" \  
-H "Authorization: Bearer YOUR_API_TOKEN" \  
-H "Content-Type: application/json"
```

Example Response:

```
{  
  "id": 125,  
  "parent": 42,  
  "title": "Ms.",  
  "gender": "f",  
  "birthday": "1992-08-14",  
  "age": 31,  
  "age_range": "30-34",  
  "telephone": "+1-555-0789",  
  "address": "789 Pine Street",  
  "city": "Portland",  
  "state": "OR",  
  "postal_code": "97201",  
  "country": "United States",  
  "country_id": "US",  
  "company": "Creative Agency Co",  
  "tax_id": "555-66-7777",  
  "passport_number": null,  
  "hometown": "Eugene",  
  "location": "Portland, OR",  
  "locale": "en_US",  
  "event": "Design Summit 2024",  
  "accept_terms": true,  
  "custom_field_1": "Gold Tier",
```

```
"custom_field_2": "Newsletter: Yes",
"custom_field_3": "Preferred contact: Email",
"custom_field_4": null,
"custom_field_5": null,
"custom_field_6": null,
"custom_field_7": null,
"custom_field_8": null
}
```

Response Codes:

Status	Description
200	Success - Returns the requested user extra record
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - User lacks permission to view this record
404	Not Found - User extra record with specified ID does not exist

PUT /user_extra/{id}/

Description: Completely replaces an existing user extra information record with new data. This endpoint requires all fields to be provided, effectively performing a full update. Any fields not included in the request will be cleared (set to null). Use PUT when you want to completely replace the user's extra information or when you have a complete data set to update.

Use Cases:

- Completely refresh user profile data from an external source
- Implement a "replace all" profile update functionality
- Synchronize complete user records from another system
- Reset user profile to a known state

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/125/
```

Parameters:

Parameter	Type	In	Required	Description
-----------	------	----	----------	-------------

id	integer	path	Yes	The unique identifier of the user extra record to update
data	object	body	Yes	Complete JSON object containing all user extra information fields

Request Body Schema: Same as POST endpoint - all fields should be provided to avoid clearing existing data.

cURL Example:

```
curl -X PUT "https://gate.zequenze.com/api/v1/user_extra/125/" \
-H "Authorization: Bearer YOUR_API_TOKEN" \
-H "Content-Type: application/json" \
-d '{
  "parent": 42,
  "title": "Dr.",
  "gender": "f",
  "birthday": "1992-08-14",
  "age": 31,
  "age_range": "30-34",
  "telephone": "+1-555-0999",
  "address": "789 Pine Street, Apt 4B",
  "city": "Portland",
  "state": "OR",
  "postal_code": "97201",
  "country": "United States",
  "country_id": "US",
  "company": "Creative Agency Co",
  "tax_id": "555-66-7777",
  "passport_number": null,
  "hometown": "Eugene",
  "location": "Portland, OR",
  "locale": "en_US",
  "event": "Design Summit 2024",
  "accept_terms": true,
  "custom_field_1": "Platinum Tier",
  "custom_field_2": "Newsletter: Yes",
  "custom_field_3": "Preferred contact: Phone",
  "custom_field_4": "Emergency contact: Jane Doe",
```

```
"custom_field_5": null,  
"custom_field_6": null,  
"custom_field_7": null,  
"custom_field_8": null  
'
```

Example Response:

```
{  
  "id": 125,  
  "parent": 42,  
  "title": "Dr.",  
  "gender": "f",  
  "birthday": "1992-08-14",  
  "age": 31,  
  "age_range": "30-34",  
  "telephone": "+1-555-0999",  
  "address": "789 Pine Street, Apt 4B",  
  "city": "Portland",  
  "state": "OR",  
  "postal_code": "97201",  
  "country": "United States",  
  "country_id": "US",  
  "company": "Creative Agency Co",  
  "tax_id": "555-66-7777",  
  "passport_number": null,  
  "hometown": "Eugene",  
  "location": "Portland, OR",  
  "locale": "en_US",  
  "event": "Design Summit 2024",  
  "accept_terms": true,  
  "custom_field_1": "Platinum Tier",  
  "custom_field_2": "Newsletter: Yes",  
  "custom_field_3": "Preferred contact: Phone",  
  "custom_field_4": "Emergency contact: Jane Doe",  
  "custom_field_5": null,  
  "custom_field_6": null,  
  "custom_field_7": null,  
  "custom_field_8": null  
}
```

Response Codes:

Status	Description
200	Success - User extra record successfully updated
400	Bad Request - Invalid data provided (e.g., invalid date format, invalid gender value)
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - User lacks permission to update this record
404	Not Found - User extra record with specified ID does not exist

PATCH /user_extra/{id}/

Description: Partially updates an existing user extra information record. Unlike PUT, PATCH allows you to update only specific fields without affecting other stored data. Fields not included in the request remain unchanged. This is the recommended method for most update operations as it provides granular control and prevents accidental data loss.

Use Cases:

- Update a user's telephone number or address without changing other information
- Add or modify custom fields while preserving existing data
- Update specific demographic information based on user input
- Implement incremental profile completion workflows

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/125/
```

Parameters:

Parameter	Type	In	Required	Description
id	integer	path	Yes	The unique identifier of the user extra record to update
data	object	body	Yes	JSON object containing only the fields to update

Request Body Schema: Same fields as POST endpoint, but all are optional - include only the fields you want to update.

cURL Example:

```
curl -X PATCH "https://gate.zequenze.com/api/v1/user_extra/125/" \  
-H "Authorization: Bearer YOUR_API_TOKEN" \  
-H "Content-Type: application/json" \  
-d '{  
  "telephone": "+1-555-1234",  
  "address": "789 Pine Street, Suite 200",  
  "custom_field_1": "Diamond Tier"  
'
```

Example Response:

```
{  
  "id": 125,  
  "parent": 42,  
  "title": "Dr.",  
  "gender": "f",  
  "birthday": "1992-08-14",  
  "age": 31,  
  "age_range": "30-34",  
  "telephone": "+1-555-1234",  
  "address": "789 Pine Street, Suite 200",  
  "city": "Portland",  
  "state": "OR",  
  "postal_code": "97201",  
  "country": "United States",  
  "country_id": "US",  
  "company": "Creative Agency Co",  
  "tax_id": "555-66-7777",  
  "passport_number": null,  
  "hometown": "Eugene",  
  "location": "Portland, OR",  
  "locale": "en_US",  
  "event": "Design Summit 2024",  
  "accept_terms": true,  
  "custom_field_1": "Diamond Tier",  
  "custom_field_2": "Newsletter: Yes",  
  "custom_field_3": "Preferred contact: Phone",  
  "custom_field_4": "Emergency contact: Jane Doe",
```

```
"custom_field_5": null,  
"custom_field_6": null,  
"custom_field_7": null,  
"custom_field_8": null  
}
```

Response Codes:

Status	Description
200	Success - User extra record successfully updated
400	Bad Request - Invalid data provided for the fields being updated
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - User lacks permission to update this record
404	Not Found - User extra record with specified ID does not exist

DELETE /user_extra/{id}/

Description: Permanently deletes a user extra information record. This operation cannot be undone. Use this endpoint when a user requests data deletion, when cleaning up test data, or when a user account is being fully removed from the system. Note that this only deletes the extra information record, not the parent user account.

Use Cases:

- Comply with data deletion requests (GDPR, privacy regulations)
- Remove user profile data when deactivating or deleting user accounts
- Clean up test or duplicate records
- Reset user profiles by deleting and recreating extra information

Full URL Example:

```
https://gate.zequenze.com/api/v1/user_extra/125/
```

Parameters:

Parameter	Type	In	Required	Description
id	integer	path	Yes	The unique identifier of the user extra record to delete

cURL Example:

```
curl -X DELETE "https://gate.zequence.com/api/v1/user_extra/125/" \  
-H "Authorization: Bearer YOUR_API_TOKEN"
```

Example Response:

No content returned (204 status code indicates successful deletion)

Response Codes:

Status	Description
204	No Content - User extra record successfully deleted
401	Unauthorized - Invalid or missing authentication token
403	Forbidden - User lacks permission to delete this record
404	Not Found - User extra record with specified ID does not exist or already deleted

Common Use Cases

Use Case 1: Complete User Onboarding with Extended Profile

When onboarding a new user to your application, you first create the core user account through the user authentication endpoints in GATE, then immediately create an associated user extra record to capture additional profile information.

Workflow:

1. Create user account via authentication API (receives user ID)
2. POST to `/user_extra/` with the new user ID as parent
3. Collect and store demographic, contact, and custom business fields

This approach allows you to maintain a clean separation between authentication data and extended profile information while providing a seamless registration experience.

Use Case 2: Progressive Profile Completion

Implement a progressive profile completion system where users can gradually add information to their profiles over time. Start with minimal required fields and use PATCH operations to incrementally add details.

Workflow:

1. Initial registration: POST with only `parent` and `accept_terms`
2. First login: PATCH to add basic contact info (`telephone`, `city`, `country`)
3. Profile enhancement: PATCH to add demographic data and preferences
4. Event registration: PATCH to update `event` and event-specific custom fields

This pattern improves user experience by avoiding long registration forms while still collecting comprehensive data over time.

Use Case 3: Organization-Wide User Data Export

Generate reports or exports of user information across an entire organization for analytics, compliance, or migration purposes.

Workflow:

1. GET `/user_extra/?parent__organization_id=123&limit=100` to retrieve first batch
2. Process returned records and extract `next` URL from pagination
3. Continue following `next` URLs until all records are retrieved
4. Aggregate data for reporting, CSV export, or data warehouse loading

Use pagination to handle large datasets efficiently and avoid timeout or memory issues.

Use Case 4: Event Attendee Management

Collect and manage detailed attendee information for conferences, webinars, or events by leveraging the `event` field and custom fields for event-specific data.

Workflow:

1. During event registration: POST user extra records with `event` field populated
2. Store event-specific data in custom fields (e.g., dietary restrictions, session preferences)
3. Query attendees: GET `/user_extra/?parent__organization_id=123` filtered by event
4. Update attendee status: PATCH to modify custom fields as needed (check-in status, badge printed, etc.)

This provides a flexible system for managing varying event requirements without schema changes.

Use Case 5: Multi-Locale User Management

Support international users with different locale preferences and location-specific information by properly utilizing the locale, country, and location fields.

Workflow:

1. Detect user's locale during registration (browser settings, IP geolocation)
2. POST user extra record with `locale`, `country`, and `country_id` fields
3. Use locale information to customize communication preferences
4. PATCH to update location information when users move or travel

This enables personalized experiences based on user location and language preferences while maintaining accurate demographic data.

Best Practices

Data Validation:

- Always validate date formats (YYYY-MM-DD) before submitting birthday fields
- Ensure gender values are limited to "m", "f", or "o" to avoid validation errors
- Validate parent user IDs exist before creating user extra records
- Use appropriate field types (e.g., integers for age, strings for custom fields)

Efficient Querying:

- Use pagination (`limit` and `offset`) when retrieving large datasets to avoid performance issues
- Filter by `parent_organization_id` when working with organization-specific data
- Cache frequently accessed user extra records to reduce API calls
- Consider implementing local storage for user profile data that changes infrequently

Update Operations:

- Prefer PATCH over PUT for most update scenarios to prevent accidental data loss
- Only use PUT when you need to completely reset/replace a user's extra information
- Validate required fields before submission to avoid partial update failures
- Implement optimistic locking or version checking for concurrent update scenarios

Privacy and Security:

- Store only necessary personal information and comply with data privacy regulations (GDPR, CCPA)
- Implement proper access controls to ensure users can only access their own data or authorized records
- Use HTTPS for all API communications to protect sensitive user information in transit
- Regularly audit and purge unnecessary personal data using the DELETE endpoint
- Consider encrypting sensitive fields (passport numbers, tax IDs) at the application layer

Custom Fields Management:

- Document the purpose and format of each custom field used in your organization
- Maintain consistency in custom field usage across your application
- Consider implementing a naming convention for custom fields (e.g., custom_field_1 always contains membership tier)
- Use JSON schema validation in your application to ensure custom field data consistency

Error Handling:

- Implement retry logic with exponential backoff for transient network errors
- Log API errors with sufficient context for debugging (user ID, request parameters)
- Provide user-friendly error messages while logging detailed technical errors
- Handle 404 errors gracefully when querying potentially deleted records

Performance Optimization:

- Batch user extra record creation when onboarding multiple users simultaneously
- Use the `device_info` parameter only when device data is actually needed
- Implement caching strategies for frequently accessed user profile data
- Consider implementing webhooks or change notifications to invalidate caches when data changes

Revision #6

Created 2026-02-04 05:16:45 UTC by ipena@zequenze.com

Updated 2026-03-06 18:37:31 UTC by ipena@zequenze.com