



External Card Issuing API Integration Manual

ALIVE Platform

Table of Contents

Revision History	4
Introduction	5
Important Introductory Data	5
Card Issuer Type.....	5
Card Issuing Mode	5
API Specification.....	5
Authentication	5
Content Type	6
Charset.....	6
Parameter and Attribute Validations	6
Listings Pagination	6
Responses Handling	6
Data Model	7
General Object Models.....	7
API Responses	7
Data Object Models	9
Cardholders.....	9
Graduations	11
Cards	12
Enums.....	13
Cardholders.....	13
Graduations	15
Cards	15
Resources	16
Authentication	16
POST /authentication.authenticate	16
Cardholders	17
POST /cardholders.set	17
POST /cardholders.setPhoto	21
POST /cardholders.anonymize	22
GET /cardholders.list.....	23
GET /cardholders.get	23
GET /cardholders.getPhoto.....	24
Graduations	24
POST /graduations.set	24
GET /graduations.list.....	25
GET /graduations.get	26
Cards	26
POST /cards.report.....	26
POST /cards.request	28

POST /cards.update	30
POST /cards.changeValidity	32
POST /cards.changeStatus	33
POST /cards.issuePlastic.....	34
GET /cards.list	36
GET /cards.get.....	37
Workflows.....	37
How to issue a new ISIC, ITIC, ALIVE STUDENT or ALIVE STAFF card.....	37
Digital and plastic cards were issued together.....	37
Plastic card was issued later than the digital card	37
How to report a student graduation and issue a new ALIVE ID ALUMNI card	38
How to update cardholder’s data stored in NCDB and displayed on a digital card	38
How to update card validity, status, smart technology or custom fields data	39
How to delete a cardholder if a wrong data were sent	39
Test Environment.....	40
Production Environment	40
Technical Support	40

Revision History

Date	Version	Author	Description of Changes
3. 8. 2020	0.8	Lukas Kabicek	Initial release.
4. 8. 2020	0.8.1	Lukas Kabicek	Added authentication service and endpoint details.
31. 8. 2020	0.8.2	Lukas Kabicek	Changed cardholder roles data model specification, address type enum values and cardholder's photo upload endpoint details.
1. 9. 2020	0.9 CZ Uni	Lukas Kabicek	Amendments specific for Czech universities.
27. 10. 2020	1.0 CZ Uni	Lukas Kabicek	Added parameter <i>expires_in</i> into the response body of the authentication endpoint. Added list of general numeric error codes.
14. 12. 2020	1.1 CZ Uni	Lukas Kabicek	Validation of an <i>external_id</i> attribute in the <i>[MODEL] Cardholder</i> was changed to be unique per card issuer. Cardholder's phone number attribute is not mandatory anymore.
17. 2. 2021	1.2 CZ Uni	Lukas Kabicek	Added parameter <i>institution_id</i> to the cardholder's data model and related API endpoint response samples. Workflows section was updated. Typo and wording corrections (image file uploads, authentication endpoint description etc.).
21. 2. 2022	1.3 CZ Uni	Lukas Kabicek	<ul style="list-style-type: none"> • A new data object model and enum for student graduations were added. • New API endpoints for reporting and reading student graduations were added. • A new ALIVE ID ALUMNI card type was added to the <i>[ENUM] Card Type</i>. • A new API endpoint <i>POST /cards.request</i> for requesting issuance of the ALIVE ID ALUMNI cards was added. • Information about ALIVE ID ALUMNI card issuance workflow were added. • Multiple workflow descriptions were updated. • An option to update card custom fields when calling <i>POST /cards.issuePlastic</i> endpoint was added. • The <i>[ENUM] Gender</i> was updated – value <i>OTHER</i> is now available. • Typo and wording corrections were done across the document. • A new visual identity of GTS ALIVE was applied.

Introduction

This document describes API for external card issuers in the information system ALIVE Platform. The API is intended for card issuers who issue cards by themselves (digital or plastic including plastic card printing) and would like to communicate with the National Card Database (NCDB) in an automated way. The API allows to manage cardholders, their graduations, report issued cards or request card issuance (difference is in the card number stock management, see below) as well as manage card validity, status and other related parameters.

Important Introductory Data

Card Issuer Type

Value	Description
INSTITUTION	An (academic) institution issuing card only for its students, teachers, and employees. All issued cards will have displayed or printed the same (academic) institution name which has been agreed with the ISIC account manager at the beginning of the cooperation. The (academic) institution guarantee eligibility to get the card for all its cardholders.

Card Issuing Mode

Value	Description
REPORT	Card issuer orders card numbers from local ISIC authority. Assigned card numbers are sent to card issuer who manages their stock on his side and assign card numbers to individual issued cards. Card issuer reports data about issued cards including assigned card numbers to NCDB. Card issuer prints plastic cards from his system (if they are issued). This card issuing mode is applicable for issuing of ISIC, ITIC, ALIVE STUDENT and ALIVE STAFF cards.
REQUEST	Card issuer orders card numbers from local ISIC authority. Assigned card numbers are not sent to card issuer. Card number stock is managed on the side of the local ISIC authority (in NCDB application), not on card issuer's side. Card issuer requests card issuance from NCDB by sending card data excluding card number which is assigned by NCDB and returned to the card issuer in an API response. This card issuing mode is applicable for issuing of ALIVE ID ALUMNI cards. This card type doesn't allow issuance of the plastic cards.

API Specification

Authentication

- All API **endpoints are secured** so the client must be successfully authenticated to be able to call them.
- Client authentication is secured by **Bearer Token** which must be sent in the header of each API request as **Authorization: Bearer xxx** (xxx substituted by the token). The API is **not session based**.
- To obtain valid token the client is requested to call [authentication service endpoint](#) using unique [authentication credentials](#) provided by his ISIC account manager.

- The **validity of the token** is very short (token validity is a part of the authentication endpoint response) thus we recommend to ask for a new token before every set of the API requests.
- Data transfers are encrypted using **SSL certificate** of domain validation (DV) authentication level.

Content Type

- Most of the requests and responses use **JSON** data format (**application/json** content-type). Clients should also include request header parameter **Accept: application/json** to receive correct response data format.
- **Images** are **uploaded** using **multipart/form-data** encoding type. Besides a path to the image stored in the local device (as permanent or temporary file) the request must contain also `cardholder_id` attribute to identify the cardholder to whom the image is being uploaded. Base64 encoded images are not supported. A direct image data format is also not supported.
- **Images** are **downloaded** as **general binary data** (octet-stream) with appropriate MIME Content-Type header. Base64 encoded images are not supported.

Charset

- All requests should be sent with **UTF-8** charset specified in the header.

Parameter and Attribute Validations

- The API validates query parameters and attributes of created or modified objects.
- Any errors are in the body of HTTP 422 response (see object model “Errors”).
- The error messages are only in English language.
- The API uses logical operator AND between query parameters.
- Date data type attributes are expected and returned in format “yyyy-MM-dd”.
- Datetime data type attributes uses ISO 8601 extended format.
 - In API requests, date and time values are expected in local timezone with specification of the offset from UTC (e.g. for UTC+5 timezone the format should be “2020-08-03T14:35:26+05:00” meaning you would like to get data for specified day and time with time shift of 5 hours ahead of UTC).
 - In API responses, date and time values are always in UTC.

Listings Pagination

- The API supports “limit” and “offset” parameters in data listings to reduce data traffic for individual requests.
- The “limit” parameter is mandatory, its maximal value is not restricted.
- The “offset” parameter is zero-based index of the first returned record.

Responses Handling

- Clients are requested to ignore all unknown attributes in API responses to preserve compatibility in the future.

- The order of parameters in JSON responses is not significant and may change over time. Clients are requested to not rely on it.

Data Model

General Object Models

API Responses

API responses contain from HTTP code and response body with pre-defined structure as described below. In case of error the human readable error messages should be used only for debugging purposes and are not meant to be displayed to the end users.

NOTE: A list of error codes and messages will be provided in one of the future versions of this API integration manual.

[MODEL] API Response

Attribute Name	Data Type	Description
message	String	Human readable API request result (provided only in English).
data	Array	Nested array of objects based on API endpoint relation (e.g. [MODEL] Card).
code	Integer	In case of error this attribute contains numeric error code.
errors	Array	Optional nested array of objects representing individual validation or error messages (see [MODEL] Errors).

[MODEL] Errors

Attribute Name	Data Type	Description
property_name	String	Optional name of the attribute which the error message is related to.
message	String	Human readable error message (provided only in English).
code	String	Internal alphabet error code.

General Numeric Error Codes

Code Range	Description
1001	Invalid data.
1002	Given JWT token is malformed and does not contain required attributes.
1003	Given media type is not supported.

2001	Invalid credentials.
2002	Authentication required.
3001	Entity not found.

Sample Response Body – Success

```

1 {
2   "message": "OK",
3   "data": [
4     {
5       "card_id": 12345
6     },
7     {
8       "card_id": 127455
9     }
10  ]
11 }
```

Sample Response Body – Error

```

1 {
2   "message": "Cardholder not found.",
3   "data": [],
4   "code": 3001
5 }
```

OR

```

1 {
2   "message": "Invalid data.",
3   "data": [],
4   "code": 1001,
5   "errors": [
6     {
7       "property_name": "access_key_id",
8       "message": "This value should not be blank.",
9       "code": "IS_BLANK_ERROR"
10    },
11    {
12      "property_name": "secret_access_key",
13      "message": "This value should not be blank.",
14      "code": "IS_BLANK_ERROR"
15    }
16  ]
17 }
```


Data Object Models

Cardholders

[MODEL] Cardholder

Attribute Name	Data Type	Validations	Description
cardholder_id	Integer	-	Unique, primary cardholder's identifier.
external_id	String	Unique per card issuer	Cardholder's identifier from the third-party's system.
institution_id	Integer	-	Internal identifier of the academic institution (client) in NCDB.
academic_degree_before_name	String	Free	Cardholder's pre-nominal academic degrees.
first_name	String	Free	Cardholder's first name.
last_name	String	Free	Cardholder's last name.
academic_degree_after_name	String	Free	Cardholder's post-nominal academic degrees.
date_of_birth	Date	Not in future	Cardholder's date of birth.
birth_number	String	Valid national format	Cardholder's birth number.
gender	String	Enum	Cardholder's gender.
email	String	Unique per card issuer	Cardholder's e-mail address.
phone_number	String	Regex [0-9, (,), +, -]	Cardholder's phone number.
language	String	Enum	Preferred communication language, 2 letter language code.
personal_id	String	Free	Free cardholder's identifier (student ID, membership ID, personal ID).
class_name	String	Free	Name of the school class.
grade	Integer	Values 1-9	Study grade.
roles	Object	-	Nested object (see [MODEL] Role).
student_status	String	Enum	If student, cardholder's student status.
addresses	Array	-	Nested array of objects (see [MODEL] Address).
legal_representatives *	Array	-	Nested array of objects (see [MODEL] Legal Representative).
custom_fields **	Array	-	Nested array of objects (see [MODEL] Custom Field).

* At least one Legal Representative's data must be specified in case the cardholder is younger than an age required by local law (15 years in the Czech Republic).

** Custom fields may be set as mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account.

[MODEL] Role

Roles which are not included in the request are considered as the client doesn't know such information.

NOTE: Crossed out values will be supported in the future API development phases.

Attribute Name	Data Type	Validations	Description
employee	Boolean	-	Recent employee of the institution which creates a cardholder.
graduate	Boolean	-	Successfully graduated person.
student	Boolean	-	Active student with all student privileges granted by local law.
teacher	Boolean	-	Active teacher.

[MODEL] Address

Attribute Name	Data Type	Validations	Description
type	String	Enum	Address type.
address_line_1	String	Free	First address line, usually street and house number.
address_line_2	String	Free	Second address line.
city	String	Free	City.
district	String	Enum	State/Province/County.
postal_code	String	Free	Postal (ZIP) code.
country	String	Enum	ISO 3166-1 alpha-3 country code (3 letter). (https://www.iso.org/iso-3166-country-codes.html)

[MODEL] Legal Representative

NOTE: Crossed out values will be supported in the future API development phases.

Attribute Name	Data Type	Validations	Description
type	String	Enum	Legal Representative's relationship with the cardholder.
academic_degree_before_name	String	Free	Legal Representative's pre-nominal academic degree.
first_name	String	Free	Legal Representative's first name.
last_name	String	Free	Legal Representative's last name.

academic_degree_after_name	String	Free	Legal Representative's post-nominal academic degree.
date_of_birth	Date	Not in future	Legal Representative's date of birth.
birth_number	String	Valid national format	Legal Representative's birth number.
gender	String	Enum	Legal Representative's gender.
email	String	Free	Legal Representative's e-mail address.
phone_number	String	Regex [0-9, (,), +, -]	Legal Representative's phone number.
language	String	Enum	Preferred communication language, 2-letter language code.
addresses	Array	-	Nested array of objects (see [MODEL] Address).

[MODEL] Custom Field

Custom fields are created and assigned to individual card issuers by ISIC account managers based on the mutual agreement. Their setup range related to this API is key, mandatory/optional, and min. and max. allowed length of the string.

Attribute Name	Data Type	Validations	Description
name	String	Assigned field	Custom field's name.
value	String	Field settings	Custom field's value.

Graduations

[MODEL] Graduation

Attribute Name	Data Type	Validations	Description
graduation_id	Integer	-	Unique, primary identifier of the graduation.
cardholder_id	Integer	-	Unique, primary cardholder's identifier.
faculty_name	String	Free	Faculty name.
field_of_study	String	Free	Field of study.
graduation_date	Date	Not in future	Date of graduation.
academic_degree	String	Enum	Attained academic degree.
pre_nominal_academic_degree	String	Free	Cardholder's pre-nominal academic degree obtained on that graduation.
post_nominal_academic_degree	String	Free	Cardholder's post-nominal academic degree obtained on that graduation.

Cards

[MODEL] Card

Attribute Name	Data Type	Validations	Description
card_id	Integer	-	Unique, primary card identifier.
cardholder_id	Integer	-	Unique, primary cardholder's identifier.
number	String	Assigned card number ranges	Card number.
type	String	Enum	Card type.
form	String	Enum	Card form.
status	String	Enum	Card status.
valid_from	Date	Available card validity cycles	Card validity start.
valid_to	Date	Available card validity cycles	Card validity end.
valid_to_plastic	Date	Available card validity cycles	Plastic card validity end (may be the same or shorter than the digital card validity end).
design	String	Assigned card designs	Card design part number.
design_plastic	String	Assigned card designs	Plastic card design part number.
institution_id	Integer	NCDB's institution database	ID of the institution where the cardholder studies/teaches/works as stored in NCDB's institution database.
institution_name_line_1	String	Free	First line of the institution name where the cardholder studies/teaches/works as stated on a card.
institution_name_line_2	String	Free	Second line of the institution name where the cardholder studies/teaches/works as stated on a card.
smart_technologies	Array	-	Nested array of objects (see [MODEL] Smart Technology).
custom_fields	Array	-	Nested array of objects (see [MODEL] Custom Field).

[MODEL] Smart Technology

Attribute Name	Data Type	Validations	Description
type	String	Enum	Smart technology type.
value	String	Unique per type across the country	Smart technology value (RFID's UID, values stored in magnetic stripe etc.).

[MODEL] Custom Field

Custom fields are created and assigned to individual card issuers by ISIC account managers based on the mutual agreement. Their setup range related to this API is key, mandatory/optional, and min. and max. allowed length of the string.

Attribute Name	Data Type	Validations	Description
name	String	Assigned to card issuer	Custom field name.
value	String	Field setup	Custom field value.

Enums

Cardholders

[ENUM] Gender

Value	Description
FEMALE	Female.
MALE	Male.
OTHER	Other.

[ENUM] Language

Preferred language for communication with a cardholder.

Value	Description
cs	Czech.
en	English.

[ENUM] Student Status

Value	Description
FULL_TIME	Full time student.
PART_TIME	Part time student.

[ENUM] Address Type

NOTE: Crossed out values will be supported in the future API development phases.

Value	Description
BILLING	Billing address (for invoicing purposes).
DELIVERY	Delivery address (where the cardholder lives, where to send a mail).
REGISTERED	Registered address (as stated on cardholder's ID card).

[ENUM] District (State/Province/County)

Dependent on a country where the card issuer's profile is created. For the list of available values ask your ISIC account manager.

[ENUM] Country

ISO 3166-1 alpha-3 country codes (<https://www.iso.org/iso-3166-country-codes.html>).

[ENUM] Legal Representative Type

Value	Description
MOTHER	Mother.
FATHER	Father.
SISTER	Sister.
BROTHER	Brother.
GRANDMOTHER	Grandmother.
GRANDFATHER	Grandfather.
AUNT	Aunt.
UNCLE	Uncle.
GUARDIAN	Guardian.
STEPMOTHER	Stepmother.
STEPFATHER	Stepfather.
FOSTER_MOTHER	Foster-mother.
FOSTER_FATHER	Foster-father.
GOVERNESS	Governess.
OTHER	Other.

Graduations

[ENUM] Academic Degree

Value	Description
BACHELOR	Bachelor's degree.
MASTER	Master's degree.
DOCTORATE	Doctorate.

Cards

[ENUM] Card Type

Value	Description
ISIC	ISIC card (for full-time students).
ISIC_SCHOLAR	ISIC SCHOLAR card (in the Czech Republic it is also known as ISIC Školák).
ITIC	ITIC card (for teachers).
IYTC	IYTC card (for youth people).
ALIVE	ALIVE card (for members of sport associations and other companies, issued only in the Czech Republic).
ALIVE_STUDENT	ALIVE Student card (for part-time students, issued only in the Czech Republic).
ALIVE_STAFF	ALIVE Staff card (for non-teaching staff of schools, issued only in the Czech Republic).
ALIVE_ALUMNI	ALIVE Alumni card (for university graduates, issued only in the Czech Republic).
ALIVE_ID_ALUMNI	ALIVE ID Alumni card (for university graduates).

[ENUM] Form

Value	Description
DIGITAL	Digital card.
DIGITAL_PLASTIC	Digital and plastic card.

[ENUM] Status

Value	Description
-------	-------------

ACTIVE	Active card (does not inform about card validity period!).
CANCELLED	Cancelled card (e.g., damaged, lost, stolen).
RETURNED	Returned card (its number will be used for new cardholder).

[ENUM] Smart Technology Type

NOTE: Crossed out values will be supported in the future API development phases.

Value	Description
CONTACT_CHIP	Contact chip.
CONTACTLESS_BLUETOOTH_NFC	Digital key used by mobile phone's NFC or Bluetooth.
CONTACTLESS_EM_MARINE	Contactless RFID EM-Marine chip.
CONTACTLESS_MIFARE	Contactless RFID Mifare chip (including DESFire subtypes).
MAGNETIC_STRIPE	Magnetic stripe.

Resources

Authentication

The authentication service is accessible without any credentials. In the request body the client must send unique pair of *authentication credentials* provided by his ISIC account manager.

POST /authentication.authenticate

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
access_key_id	String	Yes	Unique API access key identifier (username equivalent).
secret_access_key	String	Yes	Unique API secret access key identifier (password equivalent).

Response

In the response the client may find an attribute *expires_in* informing about the token's validity (in seconds).

- HTTP 200 – OK + Bearer Token

Sample Request Body

```
1 {
2   "access_key_id": "externalIssuers.api.test",
3   "secret_access_key": "LHKwwZ4f-FNW4qa"
4 }
```

Sample Response Body

```
1
2 {
3   "message": "OK",
4   "data": {
5     "token": "<Bearer Token>",
6     "expires_in": 300
7   }
8 }
9
```

Cardholders

POST /cardholders.set

Create or update cardholder. For update you must include *cardholder_id* parameter in the request body. Cardholder updates are not differential. Within every request the client must send all cardholder's data. **Data missing in the request will be considered as for removal and will be deleted with no way to recover.**

NOTE: Crossed out values will be supported in the future API development phases.

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
cardholder_id	Integer	depends ->	Applicable only in case of the cardholder's update, then mandatory. Anonymized cardholders cannot be updated.
external_id	String	No	
academic_degree_before_name	String	No	
first_name	String	Yes	
last_name	String	Yes	
academic_degree_after_name	String	No	
date_of_birth	Date	Yes	Date cannot be in future. Format "yyyy-MM-dd".
birth_number	String	No	
gender	String	No	

email	String	Yes	Valid e-mail address format. E-mail must be unique per all cardholders within the card issuer. All cards of a single person should be linked to the same cardholder identified by the <i>cardholder_id</i> .
phone_number	String	No	
language	String	No	
personal_id	String	No	
class_name	String	No	
grade	Integer	depends ->	Mandatory in case of ISIC SCHOLAR card order by a school involved in the ISIC for Schools project in the Czech Republic.
roles	Object	Yes	
employee	Boolean	No	
graduate	Boolean	Yes	
student	Boolean	Yes	
teacher	Boolean	No	
student_status	String	Yes	
addresses	Array	No	Addresses are not mandatory but if sent, it must contain all its parameters marked below as mandatory.
type	String	Yes	
address_line_1	String	Yes	
address_line_2	String	No	
city	String	Yes	
district	String	No	
postal_code	String	Yes	
country	String	Yes	
legal_representatives	Array	depends ->	At least one Legal Representative's data must be specified in case the cardholder is younger than an age required by local law (in European Union countries it is usually 15 or 16 years). Ask your ISIC account manager for the age applicable for your country. If mandatory, it must contain all its parameters marked below as mandatory.
type	String	Yes	Relationship of the legal representative to the cardholder. Always mandatory. See [ENUM] Legal Representative Type for the list of available values.
academic_degree_before_name	String	No	
first_name	String	Yes	
last_name	String	Yes	

academic_degree_after_name	String	No	
date_of_birth	Date	No	Must be older than an age required by local law (see above). Date cannot be in future. Format "yyyy-MM-dd".
birth_number	String	No	
gender	String	No	
email	String	Yes	Valid e-mail address format. E-mail does not have to be unique.
phone_number	String	No	
language	String	No	
addresses	Array	No	Address is not mandatory but if sent, it must contain all its parameters marked below as mandatory. Individual parameters all the same as for the main cardholder's address listed above.
custom_fields	Array	depends ->	Custom fields may be mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account. If mandatory, it must contain all its parameters marked below as mandatory.
name	String	Yes	
value	String	Yes	

Response

- HTTP 200 – OK + [\[MODEL\] Cardholder](#)

Sample Request Body

```

1 {
2   "external_id": "ST44211",
3   "academic_degree_before_name": "Mgr.",
4   "first_name": "Petr",
5   "last_name": "Janík",
6   "academic_degree_after_name": "CSc.",
7   "date_of_birth": "1994-04-15",
8   "birth_number": "940415/0123",
9   "gender": "MALE",
10  "email": "petr.janik@test.cz",
11  "phone_number": "+420123456789",
12  "language": "en",
13  "personal_id": "4844V2",
14  "class_name": "7.B",
15  "grade": 7,
16  "roles": {
17    "student": true,
18    "graduate": false
19  },
20  "student_status": "FULL_TIME",
21  "addresses": [

```

```

22     {
23         "type": "REGISTERED",
24         "address_line_1": "Husova 1204/2a",
25         "address_line_2": "V Tower, 2. patro",
26         "city": "Praha 6",
27         "district": "Hlavní město Praha",
28         "postal_code": "16000",
29         "country": "CZE"
30     }
31 ],
32 "legal_representatives": [
33     {
34         "type": "MOTHER",
35         "first_name": "Jana",
36         "last_name": "Janíková",
37         "birth_number": "725109/0123",
38         "email": "jana.janikova@test.cz",
39         "phone_number": "+420123456789"
40     }
41 ],
42 "custom_fields": [
43     {
44         "name": "membership_code",
45         "value": "7110EQ4"
46     }
47 ]
48 }

```

Sample Response Body

```

1 {
2     "message": "OK",
3     "data": [
4         {
5             "cardholder_id": 1694452,
6             "external_id": "ST44211",
7             "institution_id": 29707,
8             "academic_degree_before_name": "Mgr.",
9             "first_name": "Petr",
10            "last_name": "Janík",
11            "academic_degree_after_name": "CSc.",
12            "date_of_birth": "1994-04-15",
13            "birth_number": "940415/0123",
14            "gender": "MALE",
15            "email": "petr.janik@test.cz",
16            "phone_number": "+420123456789",
17            "language": "en",
18            "personal_id": "4844V2",
19            "class_name": "7.B",
20            "grade": 7,
21            "roles": {
22                "student": true,
23                "graduate": false
24            },
25            "student_status": "FULL_TIME",

```

```

26     "addresses": [
27         {
28             "type": "REGISTERED",
29             "address_line_1": "Husova 1204/2a",
30             "address_line_2": "V Tower, 2. patro",
31             "city": "Praha 6",
32             "district": "Hlavní město Praha",
33             "postal_code": "16000",
34             "country": "CZE"
35         }
36     ],
37     "legal_representatives": [
38         {
39             "type": "MOTHER",
40             "first_name": "Jana",
41             "last_name": "Janíková",
42             "birth_number": "725109/0123",
43             "email": "jana.janikova@test.cz",
44             "phone_number": "+420123456789"
45         }
46     ],
47     "custom_fields": [
48         {
49             "name": "membership_code",
50             "value": "7110EQ4"
51         }
52     ]
53 }
54 ]
55 }

```

POST /cardholders.setPhoto

Upload cardholder's photo. When a photo is already present the next request will overwrite it. **The previous photo will not be available to recover.**

Based on the contract the photo is mandatory.

Validations

- JPG, JPEG or PNG file with RGB color space. PNG images with transparent background are transferred to an image with white background.
- Minimal image resolution is limited to 450 x 540 pixels (recommended at 300 DPI).
- Maximal image file size is limited to 3 MB.

Recommendations

- For best photo quality displayed/printed on a card the recommended image aspect ratio is 5:6.
- Photo should be in color containing a whole head of the cardholder. No sunglasses, no head covers.
- Photo background should be neutral (one color or soft color gradient).

Request Body

Request must be sent as **multipart/form-data** encoding type.

Parameter Name	Data Type	Mandatory	Note
cardholder_id	Integer	Yes	
photo	File	Yes	Address to the photo file in the client's folder structure.

Response

- HTTP 200 – OK

Sample Response Body

```
1 {  
2   "message": "OK",  
3   "data": []  
4 }
```

POST /cardholders.anonymize

Request anonymization of the cardholder identified by *cardholder_id* sent in the request body. The API will accept or reject the request. In case of the acceptance the cardholder's data (including data of his/her cards) will be anonymized within next 48 hours.

Validations

- Cardholder does not own any valid card (see *valid_to* property on the [\[MODEL\] Card](#)).
- Cardholder has not been anonymized yet.

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
cardholder_id	Integer	Yes	

Response

- HTTP 200 – OK

Sample Request Body

```
1 {  
2   "cardholder_id": 123  
3 }
```

Sample Response Body

```
1 {  
2   "message": "OK",  
3   "data": []  
4 }
```

GET /cardholders.list

Get data of cardholders filtered based on specified query parameters. Returned data are sorted by date of last modification in descending order (last modified record first).

Query Parameters

All query parameters of string data type are case-insensitive.

Parameter Name	Data Type	Mandatory	Description
external_id	String	No	Cardholder's external identifier.
first_name	String	No	Cardholder's first name.
last_name	String	No	Cardholder's last name.
date_of_birth	Date	No	Cardholder's date of birth.
email	String	No	Cardholder's e-mail address.
phone_number	String	No	Cardholder's phone number.
modified_from	Datetime	No	Return records created or modified after specified date and time (for option of incremental updates on the API client's side).
offset	Integer	No	Result paging offset (zero based index of the first returned item).
limit	Integer	Yes	Result page size (how many items to return).

Response

- HTTP 200 – OK + collection of [\[MODEL\] Cardholder](#)

GET /cardholders.get

Get data of the cardholder specified in the query parameter.

Query Parameters

Parameter Name	Data Type	Mandatory	Description
cardholder_id	Integer	Yes	Unique cardholder's identifier.

Response

- HTTP 200 – OK + [\[MODEL\] Cardholder](#)

GET /cardholders.getPhoto

Get photo of the cardholder specified in the query parameter.

Query Parameters

Parameter Name	Data Type	Mandatory	Description
cardholder_id	Integer	Yes	Unique cardholder's identifier.

Response

- HTTP 200 – OK + cardholder's photo as direct PNG or JPG image file data (octet-stream) with appropriate content-type header ([image/png](#) or [image/jpg](#)).

Graduations

POST /graduations.set

Create or update cardholder's graduation report. For update you must include [graduation_id](#) parameter in the request body. Graduation updates are not differential. Within every request the client must send all data about the graduation. **Data missing in the request will be considered as for removal and will be deleted with no way to recover.**

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
graduation_id	Integer	depends ->	Applicable only in case of the graduation update, then mandatory. Graduations of anonymized cardholders cannot be updated.
cardholder_id	Integer	depends ->	Unique identifier of the cardholder to whom the graduation is reported. Mandatory when creating a new graduation report. Optional for the graduation report update.
faculty_name	String	Yes	Faculty name.
field_of_study	String	Yes	Field of study.
graduation_date	Date	Yes	Date of graduation, cannot be in the future.
academic_degree	String	Yes	Attained academic degree. See [ENUM] Academic Degree for the list of available values.
pre_nominal_academic_degree	String		Cardholder's pre-nominal academic degree obtained on that graduation.

post_nominal_academic_degree	String	Yes – at least one of these parameters.	Cardholder's post-nominal academic degree obtained on that graduation.
------------------------------	--------	---	--

Response

- HTTP 201 – OK + [\[MODEL\] Graduation](#)

Sample Request Body

```

1 {
2   "cardholder_id": 154766,
3   "faculty_name": "Fakulta sociálních věd",
4   "field_of_study": "Politologie a veřejná politika",
5   "graduation_date": "2022-04-11",
6   "academic_degree": "BACHELOR",
7   "pre_nominal_academic_degree": "Bc."
8 }
```

Sample Response Body

```

1 {
2   "message": "OK",
3   "data": [
4     {
5       "graduation_id": 1,
6       "cardholder_id": 154766,
7       "faculty_name": "Fakulta sociálních věd",
8       "field_of_study": "Politologie a veřejná politika",
9       "graduation_date": "2022-04-13",
10      "academic_degree": "BACHELOR",
11      "pre_nominal_academic_degree": "Bc."
12    }
13  ]
14 }
```

GET /graduations.list

Get data of cardholder's graduations filtered based on specified query parameters. Returned data are sorted by date of last modification in descending order (last modified record first).

Query Parameters

All query parameters of string data type are case-insensitive.

Parameter Name	Data Type	Mandatory	Description
cardholder_id	Integer	No	Unique cardholder's identifier.
offset	Integer	No	Result paging offset (zero based index of the first returned item).

limit	Integer	Yes	Result page size (how many items to return).
-------	---------	-----	--

Response

- HTTP 200 – OK + collection of [\[MODEL\] Graduation](#)

GET /graduations.get

Get data of the cardholder’s graduation specified in the query parameter.

Query Parameters

Parameter Name	Data Type	Mandatory	Description
graduation_id	Integer	Yes	Unique identifier of the graduation.

Response

- HTTP 200 – OK + [\[MODEL\] Graduation](#)

Cards

POST /cards.report

Report a card issued to a cardholder identified by [cardholder_id](#). Card number must be assigned by the client and sent in the API request.

This endpoint should be called when issuing ISIC, ITIC, ALIVE STUDENT or ALIVE STAFF cards.

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
cardholder_id	Integer	Yes	Unique identifier of the cardholder to whom the card was issued.
type	String	Yes	Allowed values are based on client’s agreement with the ISIC account manager. Ask your ISIC account manager for the list of card types allowed for your account. See [ENUM] Card Type for the list of available values.
number	String	Yes	Card number must match to some of the card ranges you have assigned in NCDB based on your card number orders and must be free to use.
form	String	Yes	See [ENUM] Form for the list of available values.
valid_from	Date	Yes	Date must comply with card validity cycles available for selected card type at the time when the card is reported. Format “yyyy-MM-dd”.
valid_to	Date	Yes	Date must comply with card validity cycles available for selected card type at the time when the card is reported. Format “yyyy-MM-dd”.

smart_technologies	Array	Yes	Smart technologies related to the card.
type	String	Yes	See [ENUM] Smart Technology Type for the list of available values.
value	String	Yes	Value must be unique per smart technology type across the country.
custom_fields	Array	depends ->	Custom fields may be mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account. If mandatory, it must contain all its parameters marked below as mandatory.
name	String	Yes	
value	String	Yes	

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Sample Request Body

```

1 {
2   "cardholder_id": 14,
3   "type": "ISIC",
4   "number": "S420123456789A",
5   "form": "DIGITAL",
6   "valid_from": "2019-09-01",
7   "valid_to": "2020-12-31",
8   "smart_technologies": [
9     {
10      "type": "CONTACTLESS_MIFARE",
11      "value": "A14E5D5F"
12    },
13    {
14      "type": "CONTACTLESS_EM_MARINE",
15      "value": "8100A7BC5D"
16    }
17  ],
18  "custom_fields": [
19    {
20      "name": "esc_number",
21      "value": "abc123"
22    }
23  ]
24 }
```

Sample Response Body

```

1 {
2   "message": "OK",
3   "data": [
4     {
```

```

5     "card_id": 35502,
6     "cardholder_id": 14,
7     "type": "ISIC",
8     "number": "S420123456789A",
9     "form": "DIGITAL",
10    "status": "ACTIVE",
11    "design": "SA-420-026",
12    "valid_from": "2019-09-01",
13    "valid_to": "2020-12-31",
14    "institution_id": 4655,
15    "institution_name_line_1": "VŠ aplikované informatiky,",
16    "institution_name_line_2": "Kolín",
17    "smart_technologies": [
18      {
19        "type": "CONTACTLESS_MIFARE",
20        "value": "A14E5D5F"
21      },
22      {
23        "type": "CONTACTLESS_EM_MARINE",
24        "value": "8100A7BC5D"
25      }
26    ],
27    "custom_fields": [
28      {
29        "name": "esc_number",
30        "value": "abc123"
31      }
32    ]
33  }
34 ]
35 }

```

POST /cards.request

Request to issue a card to a cardholder identified by *cardholder_id*. Card number is assigned by the NCDB and returned in the API response.

This endpoint should be called when requesting issuance of the ALIVE ID ALUMNI cards. Prior implementation of this endpoint please read the [ALIVE ID ALUMNI card issuance workflow description](#).

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
cardholder_id	Integer	Yes	Unique identifier of the cardholder to whom the card should be issued.
type	String	Yes	For the alumni partnership the value is always <i>ALIVE_ID_ALUMNI</i> .
form	String	Yes	For the alumni partnership the value is always <i>DIGITAL</i> .
valid_from	Date	Yes	Day, when the card validity cycle starts. The ALIVE ID ALUMNI card validity cycle starts on the day of card issuance and last until the end of the same year (31/12/yyyy). Format "yyyy-MM-dd".

custom_fields	Array	depends ->	Custom fields may be mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account. If mandatory, it must contain all its parameters marked below as mandatory.
name	String	Yes	
value	String	Yes	

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Sample Request Body

```

1 {
2   "cardholder_id": 154766,
3   "type": "ALIVE_ID_ALUMNI",
4   "form": "DIGITAL",
5   "valid_from": "2022-04-14",
6   "custom_fields": [
7     {
8       "name": "library_id",
9       "value": "441556222104"
10    }
11  ]
12 }
```

Sample Response Body

```

1 {
2   "message": "OK",
3   "data": [
4     {
5       "card_id": 674330,
6       "cardholder_id": 154766,
7       "number": "CZA987654321TV",
8       "type": "ALIVE_ID_ALUMNI",
9       "form": "DIGITAL",
10      "status": "ACTIVE",
11      "valid_from": "2022-04-14",
12      "valid_to": "2022-12-31",
13      "design": "AID-CZ-0040",
14      "smart_technologies": [],
15      "institution_id": 19896,
16      "institution_name_line_1": "Vš aplikované informatiky",
17      "institution_name_line_2": "Kladno",
18      "custom_fields": [
19        {
20          "name": "library_id",
21          "value": "441556222104"
22        }
23      ]
24    }
25  ]
26 }
```

```

23         ]
24     }
25 ]
26 }

```

POST /cards.update

For update of card smart technology and custom fields data. Card updates are not differential. Within every request the client must send all smart technology and custom fields data as with newly issued card. **Data missing in the request will be considered as for removal and will be deleted with no way to recover.**

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
card_id	Integer	Yes	Cancelled, returned, and anonymized cards cannot be updated.
smart_technologies	Array	Yes	Smart technologies related to the card.
type	String	Yes	See [ENUM] Smart Technology Type for the list of available values.
value	String	Yes	Value must be unique per smart technology type across the country.
custom_fields	Array	depends ->	Custom fields may be mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account. If mandatory, it must contain all its parameters marked below as mandatory.
name	String	Yes	
value	String	Yes	

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Sample Request Body

```

1 {
2   "card_id": 4233658,
3   "smart_technologies": [
4     {
5       "type": "CONTACTLESS_MIFARE",
6       "value": "A14E5D5F"
7     },
8     {
9       "type": "CONTACTLESS_EM_MARINE",
10      "value": "8100A7BC5D"
11    }
12  ],
13  "custom_fields": [

```

```

14     {
15         "name": "esc_number",
16         "value": "abc123"
17     },
18     {
19         "name": "library_id",
20         "value": "441556222104"
21     }
22 ]
23 }

```

Sample Response Body

```

1 {
2     "message": "OK",
3     "data": [
4         {
5             "card_id": 4233658,
6             "cardholder_id": 445562,
7             "number": "H420123456789A",
8             "type": "ISIC_SCHOLAR",
9             "form": "DIGITAL",
10            "status": "ACTIVE",
11            "valid_from": "2019-09-01",
12            "valid_to": "2020-12-31",
13            "design": "HA-420-145",
14            "smart_technologies": [
15                {
16                    "type": "CONTACTLESS_MIFARE",
17                    "value": "A14E5D5F"
18                },
19                {
20                    "type": "CONTACTLESS_EM_MARINE",
21                    "value": "8100A7BC5D"
22                }
23            ],
24            "institution_name_line_1": "Základní škola Kladrubská,",
25            "institution_name_line_2": "Praha 10",
26            "custom_fields": [
27                {
28                    "name": "esc_number",
29                    "value": "abc123"
30                },
31                {
32                    "name": "library_id",
33                    "value": "441556222104"
34                }
35            ]
36        }
37     ]
38 }

```

POST /cards.changeValidity

Card validity extension. The end of card validity will be replaced by new date. In case the plastic card was issued and its validity can be extended both digital and plastic card validities will be extended. It is not possible to shorten the card validity.

This endpoint may be called only for ISIC, ITIC, ALIVE STUDENT and ALIVE STAFF cards. ALIVE ID ALUMNI cards have firm validity which is automatically extended by one (1) year executed 8 days before the card validity end.

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
card_id	Integer	Yes	Validity of cancelled, returned, and anonymized cards cannot be changed.
valid_to	Date	Yes	Date must comply with card validity cycles available for selected card type at the time when the card is reported. Format "yyyy-MM-dd".

Response

- HTTP 200 – OK + *[MODEL] Card*

Sample Request Body

```
1 {
2   "card_id": 198665,
3   "valid_to": "2021-12-31"
4 }
```

Sample Response Body

```
1 {
2   "message": "OK",
3   "data": [
4     {
5       "card_id": 198665,
6       "cardholder_id": 214552,
7       "number": "S420123456789A",
8       "type": "ISIC",
9       "form": "DIGITAL",
10      "status": "ACTIVE",
11      "valid_from": "2019-09-01",
12      "valid_to": "2021-12-31",
13      "design": "SA-420-071",
14      "smart_technologies": [
15        {
16          "type": "CONTACTLESS_MIFARE",
17          "value": "A14E5D5F"
18        },
19        {
20          "type": "CONTACTLESS_EM_MARINE",
```



```

21         "value": "8100A7BC5D"
22     }
23 ],
24     "institution_id": 10896,
25     "institution_name_line_1": "Vš aplikované informatiky,",
26     "institution_name_line_2": "Kladno",
27     "custom_fields": [
28         {
29             "name": "esc_number",
30             "value": "abc123"
31         }
32     ]
33 }
34 ]
35 }

```

POST /cards.changeStatus

Card status change. In case of card return the card will be locked for any future changes. Card number will become available for new usage. All returned cards are strictly monitored to avoid misuse of this feature!

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
card_id	Integer	Yes	Status of returned and anonymized cards cannot be changed.
status	Date	Yes	See [ENUM] Status for the list of available values. Status of returned cards cannot be changed.

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Sample Request Body

```

1 {
2     "card_id": 198665,
3     "status": "CANCELLED"
4 }

```

Sample Response Body

```

1 {
2     "message": "OK",
3     "data": [
4         {
5             "card_id": 198665,
6             "cardholder_id": 214552,
7             "number": "S420123456789A",
8             "type": "ISIC",

```

```

9         "form": "DIGITAL",
10        "status": "CANCELLED",
11        "valid_from": "2019-09-01",
12        "valid_to": "2021-12-31",
13        "design": "SA-420-071",
14        "smart_technologies": [
15            {
16                "type": "CONTACTLESS_MIFARE",
17                "value": "A14E5D5F"
18            },
19            {
20                "type": "CONTACTLESS_EM_MARINE",
21                "value": "8100A7BC5D"
22            }
23        ],
24        "institution_id": 10896,
25        "institution_name_line_1": "VŠ aplikované informatiky,",
26        "institution_name_line_2": "Kladno",
27        "custom_fields": [
28            {
29                "name": "esc_number",
30                "value": "abc123"
31            }
32        ]
33    }
34 ]
35 }

```

POST /cards.issuePlastic

Plastic card issuance report. During a card lifetime an unlimited count of plastic cards may be issued. Client's are requested to report all plastic cards issued (first plastic card for digital card, next new plastic cards or their duplicates for different reasons etc.). With every plastic card issuance report you can report smart technology changes. These card updates are not differential. Within every request the client must send all card's smart technology data as with newly issued card. **Data missing in the request will be considered as for removal and will be deleted with no way to recover. This includes smart technologies reported earlier when digital card was issued.**

This endpoint may be called only for ISIC, ITIC, ALIVE STUDENT and ALIVE STAFF cards. ALIVE ID ALUMNI cards don't have an option of plastic card issuance, they are only digital.

Request Body Parameters

Parameter Name	Data Type	Mandatory	Note
card_id	Integer	Yes	Plastic card cannot be issued to cancelled, returned, and anonymized cards.
design_plastic	String	No	Requested card design must be enabled for the card issuer. If not specified, a default card issuer's card design for the card type will be used.

valid_to_plastic	Date	No	May be the same or shorter than the digital card validity end. If not specified, the end of the plastic card validity will be set the same as the digital card's validity. Format "yyyy-MM-dd".
smart_technologies	Array	Yes	Smart technologies related to the card.
type	String	Yes	See [ENUM] Smart Technology Type for the list of available values.
value	String	Yes	Value must be unique per smart technology type across the country.
custom_fields	Array	depends ->	Custom fields may be mandatory or optional based on client's agreement with the ISIC account manager. Ask your ISIC account manager for the list of mandatory custom fields applicable for your account. If mandatory, it must contain all its parameters marked below as mandatory.
name	String	Yes	
value	String	Yes	

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Sample Request Body

```

1 {
2   "card_id": 198665,
3   "design_plastic": "SA-420-168",
4   "valid_to_plastic": "2021-12-31",
5   "smart_technologies": [
6     {
7       "type": "CONTACTLESS_MIFARE",
8       "value": "A14E5D5F"
9     },
10    {
11      "type": "CONTACTLESS_EM_MARINE",
12      "value": "8100A7BC5D"
13    }
14  ],
15  "custom_fields": [
16    {
17      "name": "esc_number",
18      "value": "abc123"
19    }
20  ]
21 }
```

Sample Response Body

```

1 {
2   "message": "OK",
3   "data": [
4     {
```

```

5      "card_id": 198665,
6      "cardholder_id": 214552,
7      "number": "S420123456789A",
8      "type": "ISIC",
9      "form": "DIGITAL_PLASTIC",
10     "status": "ACTIVE",
11     "valid_from": "2019-09-01",
12     "valid_to": "2021-12-31",
13     "valid_to_plastic": "2021-12-31",
14     "design": "SA-420-071",
15     "design_plastic": "SA-420-168",
16     "smart_technologies": [
17       {
18         "type": "CONTACTLESS_MIFARE",
19         "value": "A14E5D5F"
20       },
21       {
22         "type": "CONTACTLESS_EM_MARINE",
23         "value": "8100A7BC5D"
24       }
25     ],
26     "institution_id": 10896,
27     "institution_name_line_1": "VŠ aplikované informatiky,",
28     "institution_name_line_2": "Kladno",
29     "custom_fields": [
30       {
31         "name": "esc_number",
32         "value": "abc123"
33       }
34     ]
35   }
36 ]
37 }

```

GET /cards.list

Get data of cards filtered based on specified query parameters. Returned data are sorted by date of last modification in descending order (last modified record first).

Query Parameters

All query parameters of string data type are case-insensitive.

Parameter Name	Data Type	Mandatory	Description
cardholder_id	Integer	No	Unique cardholder's identifier.
number	String	No	Card number.
status	String	No	Card status (see [ENUM] Status).
type	String	No	Card type (see [ENUM] Card Type).

modified_from	Datetime	No	Return records created or modified after specified date and time (for option of incremental updates on the API client's side).
offset	Integer	No	Result paging offset (zero based index of the first returned item).
limit	Integer	Yes	Result page size (how many items to return).

Response

- HTTP 200 – OK + collection of [\[MODEL\] Card](#)

GET /cards.get

Get data of the card specified in the query parameter.

Query Parameters

Parameter Name	Data Type	Mandatory	Description
card_id	Integer	Yes	Unique card identifier.

Response

- HTTP 200 – OK + [\[MODEL\] Card](#)

Workflows

How to issue a new ISIC, ITIC, ALIVE STUDENT or ALIVE STAFF card

Digital and plastic cards were issued together

1. Authenticate yourself.
2. Get to know if the cardholder already exists in NCDB (to get his/her ID) or use cardholder's ID stored in your system.
3. Create or update cardholder (including a photo if necessary) to secure his/her data stored in NCDB are actual.
4. Report card issuance.

Please note: The card issuance have to be the last step of the process, otherwise the card will be issued with old cardholder's data and photo. Cardholder's data are automatically transferred from the cardholder object to the card object at the moment of card issuance or update.

Plastic card was issued later than the digital card

1. Authenticate yourself.
2. Get card ID from NCDB if it is not already stored in your system.

3. Report plastic card issuance by calling [POST /cards.issuePlastic](#) API endpoint. Card number remains the same as of the digital card.

The same process applies for all plastic cards issued during the card lifetime. Data of previous plastic card are overwritten by a new plastic card's data. Cardholder should not have more than one valid plastic card at the same time.

How to report a student graduation and issue a new ALIVE ID ALUMNI card

1. Authenticate yourself.
2. Get to know if the cardholder already exists in NCDB (to get his/her ID) or use cardholder's ID stored in your system.
3. Create or update cardholder (including a photo if necessary) to secure his/her data stored in NCDB are actual. If a new cardholder was created, save his/her cardholder ID.
4. Report cardholder's graduation.
5. Request ALIVE ID ALUMNI card issuance by calling [POST /cards.request](#) API endpoint.
Please note: The ALIVE ID ALUMNI card cannot be issued if there is no graduation record linked to the cardholder to whom the card issuance is requested.
6. Save the returned card ID and card number assigned by NCDB.

Important: Each cardholder (graduate) may have issued only one ALIVE ID ALUMNI card regardless of how many times he/she has graduated from the university. In case of a new graduation (e.g. master's degree graduation after previously reported bachelor's degree graduation and ALIVE ID ALUMNI issuance request) you should only report a new graduation to the existing cardholder. In case you want to update some of the data displayed on a digital card (academic titles, photo etc.), you should update these data on the cardholder level and then call [POST /cards.update](#) API endpoint to make sure they are updated also on the digital card.

How to update cardholder's data stored in NCDB and displayed on a digital card

1. Authenticate yourself.
2. Get cardholder's ID from NCDB if it is not already stored in your system.
3. Update the cardholder.
4. Update the card by calling [POST /cards.update](#) API endpoint. By doing that, data of the cardholder will be automatically updated on the digital form of this card. Don't forget to include information about card's custom fields and smart technologies, otherwise these data will be deleted from the card record in NCDB.
Please note: If the card has also its plastic version and a new plastic card with updated cardholder's data (surname, photo etc.) was issued, you have to send an API request to the endpoint [POST /cards.issuePlastic](#) instead of [POST /cards.update](#) to secure the information about newly issued plastic card is added to the evidence in NCDB.

How to update card validity, status, smart technology or custom fields data

1. Authenticate yourself.
2. Get card ID from NCDB if it is not already stored in your system.
3. Update the card using the appropriate API endpoint. Doing so, data of the cardholder will be automatically updated on a digital card.

How to delete a cardholder if a wrong data were sent

1. Authenticate yourself.
2. Get cardholder's ID from NCDB if it is not already stored in your system.
3. Ask for cardholder's data anonymization. Only cardholder without valid cards (valid to date of all his/her cards is in the past) may be anonymized.

Test Environment

The following details are necessary for web service integration testing. Please ask your ISIC account manager for authentication details and card issuer's setup based on your contract specification.

Documentation in Postman: will be specified after its completion

Base URL: <https://api.test.aliveplatform.com/v1>

Access Key ID: ask your ISIC account manager

Secret Access Key: ask your ISIC account manager

Card Issuer Type: *INSTITUTION*

Card Issuing Mode for ISIC, ITIC, ALIVE STUDENT and ALIVE STAFF Cards: *REPORT*

Card Issuing Mode for ALIVE ID ALUMNI Cards: *REQUEST*

Production Environment

Access details to the production environment will be provided **after full implementation and successful tests against the test environment.**

Base URL: <https://api.aliveplatform.com/v1>

Access Key ID: ask your ISIC account manager

Secret Access Key: ask your ISIC account manager

Card Issuer Type: *INSTITUTION*

Card Issuing Mode for ISIC, ITIC, ALIVE STUDENT and ALIVE STAFF Cards: *REPORT*

Card Issuing Mode for ALIVE ID ALUMNI Cards: *REQUEST*

Technical Support

For all support queries please contact the ALIVE Platform support team at support@aliveplatform.com.