

1. Selected api backend requests

1.1. GET Auth Me

Request: {{host}}/auth/me

Description: Login with existing session token either via api or via closing / reopening browser tab. Session should exist until session timeout setting in config-prod.json or server restart.

For partner platform integration first call 'Auth Me' request, if this fails perform 'Auth Login' in order to get new session token.

Sample Request

Auth Me

```
curl --request GET \  
  --url 'http://Request: {{host}}/auth/me' \  
  --header 'content-type: application/json' \  
  --header 'token: {{token-user}}'
```

Sample Response

```
{  
  "id": "911d5613-d86e-4565-b248-5a257d60675d",  
  "email": "demo@insic.de",  
  "createdAt": "2016-05-27T04:02:00.199Z",  
  "updatedAt": "2017-06-14T15:47:14.098Z",  
  "partnerId": "8fc1f75f-fb59-42af-88e1-6cfa3e078095",  
  "status": 1,  
  "role": "admin",  
  "isBlocked": 0,  
  "lang": "de",  
  "profile": {  
    "id": "ddcbb417-dd66-4217-b081-c2986f882e42",  
    "userId": "911d5613-d86e-4565-b248-5a257d60675d",  
    "greeting": "Mr",  
    "firstName": "Erwin",  
    "lastName": "Mustermann",  
    "birthday": "1964-11-22T00:00:00.000Z",  
    "zip": "22926",  
    "province": "Schleswig-Holstein",  
    "city": "Ahrensburg",  
    "street": "Bei der Doppeleiche",  
    "houseNumber": "3",  
    "phone": null,  
    "iban": "DE79200505501234567890",  
    "bic": "HASPDEHHXXX",  
    "agreement1": 1,  
    "agreement2": 0,  
    "agreement3": 0,  
    "agreement4": 0,  
    "placeOfBirth": "Ahrensburg",
```

```
"passportFrontId": null,  
"passportBackId": null  
}  
}
```

1.2. POST Auth Login

Request: {{host}}/auth/login

Description: Login with Username/Password with roles user, partneradmin, system administrator. This Login data is same as the data used in the Widget frontend.

BODY

```
{  
  "email": "example@gmail.com",  
  "password": "password"  
}
```

Sample Request

Auth Login

```
curl --request POST \  
  --url 'http://Request: {{host}}/auth/login' \  
  --header 'content-type: application/json' \  
  --data '{  
    "email": "example@gmail.com",  
    "password": "password"  
  }'
```

1.3. POST Add User

Request: {{host}}/user

Description: Adds a user with full data set and full validation. Adding a user already existing for a partner is not possible. The service returns a duplicate key message on email or on the combination of firstname, familyname and birthday.

BODY

```
{  
  "email": "demo@insic.de",  
  "password": "demopassword",  
  "profile": {
```

```

    "firstName": "Birgit",
    "greeting": "Mrs",
    "lastName": "Mustermann",
    "birthday": "1970-12-01",
    "zip": "22926",
    "street": "Bei der Doppeleiche",
    "houseNumber": "3",
    "phone": "+491721799161",
    "iban": "1352426736",
    "bic": "20050550",
    "city": "Ahrensburg",
    "placeOfBirth": "Ahrensburg",
    "agreement1": 1,
    "agreement2": 1,
    "agreement3": 0,
    "agreement4": 0
  }
}

```

Sample Request

AddUser (Birgit)

```

curl --request POST \
  --url 'http://Request: {{host}}/user' \
  --header 'content-type: application/json' \
  --data '{
    "email": "demo@insic.de",
    "password": "demopassword",
    "profile": {
      "firstName": "Birgit",
      "greeting": "Mrs",
      "lastName": "Mustermann",
      "birthday": "1970-12-01",
      "zip": "22926",
      "street": "Bei der Doppeleiche",
      "houseNumber": "3",
      "phone": "+491721799161",
      "iban": "1352426736",
      "bic": "20050550",
      "city": "Ahrensburg",
      "placeOfBirth": "Ahrensburg",
      "agreement1": 1,
      "agreement2": 1,
      "agreement3": 0,
      "agreement4": 0
    }
  }'

```

Sample Response user exists, otherwise profile incl. user UUDI gets returned

```

[
  {
    "message": "Emailadresse bereits vorhanden",
    "path": "email"
  },

```

```

{
  "message": "Identität bereits vorhanden",
  "path": "profile.firstName"
},
{
  "message": "Identität bereits vorhanden",
  "path": "profile.lastName"
},
{
  "message": "Die Identität ist bereits vorhanden. Bitte kontaktiere die
Kundenbetreuung.",
  "path": "profile.birthday"
}
]

```

1.4. GET User BY ID

Request: `{{host}}/user/{{user-id}}?withRelated[]=profile`

Description: Lists account data for a specific user. To get the user profile use the `withrelated = profile` parameter.

Sample Request

Get User Account by Id

```

curl --request GET \
  --url 'http://Request: {{host}}/user/911d5613-d86e-4565-b248-5a257d60675d?
withRelated[]=profile' \
  --header 'content-type: application/json' \
  --header 'token: {{token-admin}}'

```

Sample Response

```

{
  "id": "911d5613-d86e-4565-b248-5a257d60675d",
  "email": "gf@insic.de",
  "createdAt": "2016-05-27T04:02:00.199Z",
  "updatedAt": "2017-06-14T15:47:14.098Z",
  "partnerId": "8fc1f75f-1234-42af-88e1-6cfa3e078095",
  "status": 1,
  "role": "admin",
  "isBlocked": 0,
  "lang": "de",
  "profile": {
    "id": "ddcbb417-dd66-4217-b081-c2986f882e42",
    "userId": "911d5613-d86e-4565-b248-5a257d60675d",
    "greeting": "Mr",
    "firstName": "Erwin",
    "lastName": "Mustermann",

```

```
"birthday": "1964-11-22T00:00:00.000Z",
"zip": "22926",
"province": "Schleswig-Holstein",
"city": "Ahrensburg",
"street": "Bei der Doppeleiche",
"houseNumber": "3",
"phone": null,
"iban": "DE79200505501234567890",
"bic": "HASPDEHHXXX",
"agreement1": 1,
"agreement2": 0,
"agreement3": 0,
"agreement4": 0,
"placeOfBirth": "Ahrensburg",
"passportFrontId": null,
"passportBackId": null
}
}
```

1.5. PUT Edit User Profile

Request: `{{host}}/user/{{user-id}}`

Description: Change account and profile with a given uuid setting e.g. new email and names in a single step.

BODY

```
{
  "profile": {
    "firstName": "Changes"
  }
}
```

Sample Request

Change profile

```
curl --request PUT \
  --url 'http://Request: {{host}}/user/911d5613-1234-4565-b248-5a257d60675d' \
  --header 'content-type: application/json' \
  --header 'token: {{token-admin}}' \
  --data '{
  "profile": {
    "firstName": "Ute"
  }
}'
```

Sample Response

```

{
  "id": "911d5613-d86e-4565-b248-5a257d60675d",
  "email": "gf@insic.de",
  "createdAt": "2016-05-27T04:02:00.199Z",
  "updatedAt": "2017-09-25T14:00:25.226Z",
  "partnerId": "8fc1f75f-fb59-42af-88e1-6cfa3e078095",
  "status": 1,
  "role": "admin",
  "isBlocked": 0,
  "lang": "de",
  "profile": {
    "id": "ddcbb417-dd66-4217-b081-c2986f882e42",
    "userId": "911d5613-d86e-4565-b248-5a257d60675d",
    "greeting": "Mr",
    "firstName": "Ute",
    "lastName": "Mustermann",
    "birthday": "1964-11-22T00:00:00.000Z",
    "zip": "22926",
    "province": "Schleswig-Holstein",
    "city": "Ahrensburg",
    "street": "Bei der Doppeleiche",
    "houseNumber": "3",
    "phone": null,
    "iban": "DE79200505501234567890",
    "bic": "HASPDEHHXXX",
    "agreement1": 1,
    "agreement2": 0,
    "agreement3": 0,
    "agreement4": 0,
    "placeOfBirth": "Ahrensburg",
    "passportFrontId": null,
    "passportBackId": null
  }
}

```

1.6. GET Available Services for User

Description: List all services available for a specific user. This request is explicitly important to get a service id required to perform actions (start, set status, set confirmation pin etc.) on a user/service key combination.

Sample Request

Available Services For User

```

curl --request GET \
  --url 'http://Request: {{host}}/user/00083d84-1234-4c23-82a2-a46d62ae48f6/engine/services' \
  --header 'content-type: application/json' \
  --header 'token: {{token-admin}}'

```

Sample Response

```
[
  {
    "id": "RgJs6anpHF",
    "name": "scanner",
    "status": 1,
    "maxAttempts": 3,
    "amountLimit": null,
    "isAvailable": true,
    "data": {
      "attemptsCount": 0,
      "confirmAttemptsCount": 0,
      "requiredFields": []
    }
  },
  {
    "id": "d0PncAxu37",
    "name": "identt",
    "status": 2,
    "maxAttempts": 4,
    "amountLimit": null,
    "isAvailable": false,
    "data": {
      "attemptsCount": 0,
      "confirmAttemptsCount": 0,
      "requiredFields": [
        "profile.passportBack"
      ],
      "redirectUrl": null
    }
  },
  {
    "id": "jh6qiqpZQz",
    "name": "oasis",
    "status": 1,
    "maxAttempts": 3,
    "amountLimit": null,
    "isAvailable": false,
    "data": {
      "attemptsCount": 0,
      "confirmAttemptsCount": 0,
      "requiredFields": []
    }
  }
]
```

The service status may return following values:

ServiceStatus.NOT_STARTED = 1;

ServiceStatus.STARTED = 2;

ServiceStatus.SUCCESS = 3;

ServiceStatus.FAILURE = 4;

ServiceStatus.RECHECKED = 5;

1.7. POST Start any ID service

Request: `{{host}}/user/{{user-id}}/engine/{{service-id}}/check`

Start a specific ID service based on the serviceID e.g. retrieved from request "Available services for user" for the given user.

Sample Request

Start service with response that some data for this service is missing, in this case the passport backside for the passport recognition service.

```
curl --request POST \  
  --url 'http://Request: {{host}}/user/00083d84-143d-4c23-82a2-a46d62ae48f6/engine/  
d0PncAxu37/check' \  
  --header 'content-type: application/json' \  
  --header 'token: {{token-admin}}'
```

Sample Response

```
{  
  "success": false,  
  "fields": [  
    "profile.passportBack"  
  ]  
}
```