



Υπηρεσία Απόδοσης Ονομάτων Χρήστη uLookup

Οδηγός για Προγραμματιστές

GUNET IDENTITY MANAGEMENT

Ημερομηνία: **07/04/2025**

Έκδοση: 2.1

Εκδόσεις Κειμένου		
Έκδοση	Ημερομηνία	Σχόλιο
1.0	01-06-2024	Αρχική Έκδοση
2.0	10-01-2025	Προσθήκη του Checker Endpoint
2.1	07-04-2025	Προσθήκη του Tracer Endpoint

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. ΑΝΤΙΚΕΙΜΕΝΟ	4
2. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΑΡΙ	4
3. ΠΡΟΣΒΑΣΗ ΣΤΟ ΑΡΙ	4
4. ΤΟ ENDPOINT ΤΟΥ FINDER	5
5. ΤΟ ENDPOINT ΤΟΥ VALIDATOR	7
6. ΤΟ ENDPOINT ΤΟΥ PROPOSER	8
7. ΤΟ ENDPOINT ΤΟΥ CHECKER	11
8. ΤΟ ENDPOINT ΤΟΥ TRACER	14
ΠΑΡΑΡΤΗΜΑ	17

1. ΑΝΤΙΚΕΙΜΕΝΟ

Το αντικείμενο της Υπηρεσίας uLookup είναι η εύρεση και η παρουσίαση των ρόλων των φυσικών προσώπων εντός ενός ιδρύματος, όπως και η επιβεβαίωση απόδοσης ενός όνομα χρήστη και η πρόταση νέων.

Πρόκειται για μία Υπηρεσία η οποία έρχεται να συμπληρώσει την Identity Management Υπηρεσία και να την κάνει πιο προσπελάσιμη, τόσο από το προσωπικό των Ιδρυμάτων που χρησιμοποιούν την Identity Management Υπηρεσία, όσο και υπηρεσιών που είναι διασυνδεδεμένες με αυτή.

Ο σκοπός του παρόντος είναι η περιγραφή του API που παρέχει η Υπηρεσία uLookup και η απόδοση οδηγιών για τη σωστή και αποδοτικότερη χρήση του.

2. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ API

Endpoint	Method	Περιγραφή
<code>/api/v2/finder</code>	POST	Αναζήτηση Ρόλων ενός προσώπου στο Ίδρυμα
<code>/api/v2/validator</code>	POST	Επιβεβαίωση ύπαρξης και εγκυρότητας ενός username στο Ίδρυμα
<code>/api/v2/proposer</code>	POST	Πρόταση έγκυρων usernames με βάση τα στοιχεία του χρήστη στο Ίδρυμα
<code>/api/v2/checker</code>	POST	Έλεγχος ορθότητας στοιχείων εγγραφών στο Ίδρυμα
<code>/api/v2/tracer</code>	POST	Παρουσίαση στοιχείων ενός προσώπου με βάση το loginName

3. ΠΡΟΣΒΑΣΗ ΣΤΟ API

Η πρόσβαση στο API γίνεται μέσω API key authentication. Για κάθε Ίδρυμα έχει εκδοθεί ένα API key το οποίο θα πρέπει να συμπεριληφθεί στο Header του HTTP POST REQUEST με όνομα ApiKey.

Για παράδειγμα, ένα Request με curl προς το endpoint του **validator** προς τον server `ulookup-test.gunet.gr` για την επιβεβαίωση του ονόματος χρήστη “gunetdemo”, με ApiKey το String “FMOSNZHOFXccSrMDQXnaVYyJy1mhqFOKe” θα είναι όπως φαίνεται παρακάτω.

Τα παραδείγματα σε αυτό το έγγραφο είναι ενδεικτικά. Για την πρόσβαση στο endpoint του Ιδρύματος θα σας δοθούν τα αντίστοιχα endpoints και API Keys.

```
curl -X POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/validator' \  
--header 'Content-Type: application/json' \  
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJy1mhqFOKe' \  
--data '{
```

```
"loginName": "gunetdemo"
}'
```

4. TO ENDPOINT ΤΟΥ FINDER

Endpoint	Method	Περιγραφή
/api/v2/finder	POST	Αναζήτηση Ρόλων ενός προσώπου στο Ίδρυμα

Το Request Body συμπεριλαμβάνει τα στοιχεία **ssn**, **ssnCountry**, **tin**, **tinCountry** τα οποία αποτελούν τα ταυτοποιητικά στοιχεία ενός χρήστη. Η περιγραφή των στοιχείων αυτών φαίνεται στο ΠΑΡΑΡΤΗΜΑ στο τέλος του εγγράφου.

Για την αναζήτηση του δοκιμαστικού χρήστη “**gunetdemo**”, όπου **ssn**=”12312312312”, **tin**=”0111111111” και **ssnCountry**=**tinCountry**=”GR”, το JSON Body θα είναι ως εξής:

```
{
  "ssn": "12312312312",
  "ssnCountry": "GR",
  "tin": "0111111111",
  "tinCountry": "GR"
}
```

Το uLookup αναζητά με βάση τα ζεύγη των στοιχείων (**ssn,ssnCountry** και **tin,tinCountry**) που έχουν δοθεί. Αν ένα ζεύγος υπάρχει στο αντίστοιχο ΠΣ του Ιδρύματος, τότε το uLookup θα επιστρέψει τις αντίστοιχες εγγραφές. Αν κάποιο ζεύγος στοιχείων δεν είναι γνωστό, τότε μπορεί είτε να μην συμπληρωθεί, είτε να δοθεί η τιμή null. Κάθε μία από τις δύο περιπτώσεις, όπου το ΑΦΜ δεν είναι γνωστό, είναι αποδεκτή.

```
{
  "ssn": "12312312312",
  "ssnCountry": "GR",
  "tin": null,
  "tinCountry": null
}
```

```
{
  "ssn": "12312312312",
  "ssnCountry": "GR"
}
```

Το παρακάτω Request Body **ΔΕΝ ΕΙΝΑΙ ΑΠΟΔΕΚΤΟ**, καθώς δεν είναι συμπληρωμένο το ζεύγος των στοιχείων.

```
{
  "ssn": "12312312312",
  "tin": null,
  "tinCountry": null
}
```

Οπότε, το request στο ενδεικτικό endpoint θα είναι ως εξής. Το uLookup επιστρέφει τους ρόλους του προσώπου στο Ίδρυμα.

```
curl -X POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/finder' \
--header 'Content-Type: application/json' \
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJylmhqFOKe' \
--data '{ \
  "ssn": "12312312312", \
  "ssnCountry": "GR", \
  "tin": "011111111", \
  "tinCountry": "GR" \
}'
```

Ενδεικτικά, το Response Body για το παραπάνω Request θα είναι ως εξής:

```
{
  "identities":
  [
    {
      "registrationId": "76607235",
      "systemId": "76607235",
      "loginName": "gunetdemo",
      "userStatus": "active",
      "viewType": "sis",
      "userStatusDate": "20200917",
      "activationStatus": "activated"
    },
    {
      "registrationId": "ΔΠ-128",
      "systemId": "99999",
      "loginName": "gunetdemo",
      "userStatus": "active",
      "viewType": "hrms",
      "userStatusDate": "19900101",
      "activationStatus": "activated"
    },
    {
      "registrationId": "ΔΠ-129",
      "systemId": "99998",
      "loginName": "demofinance",
      "userStatus": "active",
      "viewType": "hrms",
      "userStatusDate": "19890101",
      "activationStatus": "pending"
    },
    {
      "registrationId": "0000003",
      "systemId": "0000003",
      "loginName": "gunetdemo",
      "userStatus": "inactive",
      "viewType": "elke",
      "userStatusDate": "20180101",
      "activationStatus": "inactive"
    }
  ],
  "Message": "The user has identities",
  "responseCode": "2310"
}
```

5. TO ENDPOINT ΤΟΥ VALIDATOR

Endpoint	Method	Περιγραφή
/api/v2/validator	POST	Επιβεβαίωση ύπαρξης και εγκυρότητας ενός username στο Ίδρυμα

Το Request Body συμπεριλαμβάνει τα στοιχεία **ssn**, **ssnCountry**, **tin**, **tinCountry**, **loginName** τα οποία αποτελούν τα ταυτοποιητικά στοιχεία ενός χρήστη. Η περιγραφή των στοιχείων αυτών φαίνεται στο ΠΑΡΑΡΤΗΜΑ στο τέλος του εγγράφου.

Για την αναζήτηση του δοκιμαστικού χρήστη “**gunetdemo**”, όπου **ssn**=“12312312312”, **tin**=“0111111111”, **ssnCountry**=**tinCountry**=“GR” και **loginName**=“gunetdemo”, το JSON Body θα είναι ως εξής:

```
{
  "ssn": "12312312312",
  "ssnCountry": "GR",
  "tin": "0111111111",
  "tinCountry": "GR",
  "loginName": "gunetdemo"
}
```

Οπότε, το request στο ενδεικτικό endpoint θα είναι ως εξής. Το uLookup θα αποφανθεί για το αν το username είναι διαθέσιμο στο Ίδρυμα.

```
curl -X POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/validator' \
--header 'Content-Type: application/json' \
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJylmhqFOKe' \
--data '{ \
  "ssn": "12312312312", \
  "ssnCountry": "GR", \
  "tin": "0111111111", \
  "tinCountry": "GR", \
  "loginName": "gunetdemo" \
}'
```

Αν το πρόσωπο έχει ενεργό λογαριασμό στο Ίδρυμα με το loginName που υπάρχει στο Request, τότε θα λάβει το παρακάτω Response:

```
{
  "Message": "The loginName gunetdemo is already owned by this user",
  "registeredLoginNames":
  [
    "gunetdemo",
    "gunetdemo.p"
  ],
  "responseCode": "2111",
  "responseStatus": "available"
}
```

Αν το πρόσωπο δεν έχει ενεργό λογαριασμό στο Ίδρυμα με το loginName που δίνεται στο Request, αλλά υπάρχει αντίστοιχη εγγραφή γι' αυτό στο view του ιδρύματος, τότε θα λάβει το παρακάτω Response:

```
{
  "Message": "The loginName is available, this is the first account for the user",
  "registeredLoginNames": null,
  "responseCode": "2111",
  "responseStatus": "available"
}
```

Για το ζεύγος των ταυτοποιητικών στοιχείων ισχύουν ότι και στην περίπτωση του finder.

Υπάρχει η δυνατότητα αναζήτησης ενός username, χωρίς τα ταυτοποιητικά στοιχεία ενός χρήστη. Σε αυτήν την περίπτωση το uLookup ελέγχει αν απλά το username υπάρχει ή όχι (σε ενεργό ή μη λογαριασμό).

Ενδεικτικά, για την αναζήτηση του username "gunetdemo", το request body θα είναι ως εξής:

```
{
  "loginName": "gunetdemo"
}
```

Επειδή υπάρχει το username αυτό, θα λάβουμε το παρακάτω response:

```
{
  "Message": "The loginName gunetdemo is already owned by another user in IDM",
  "registeredLoginNames": null,
  "responseCode": "2141",
  "responseStatus": "owned"
}
```

Τα responseCodes και τα Messages που επιστρέφονται περιγράφονται στους πίνακες στο ΠΑΡΑΡΤΗΜΑ.

6. TO ENDPOINT ΤΟΥ PROPOSER

Το Request Body συμπεριλαμβάνει τα στοιχεία **ssn**, **ssnCountry**, **tin**, **tinCountry**, **loginName** τα οποία αποτελούν τα ταυτοποιητικά στοιχεία ενός χρήστη. Η περιγραφή των στοιχείων αυτών φαίνεται στο ΠΑΡΑΡΤΗΜΑ στο τέλος του εγγράφου.

Endpoint	Method	Περιγραφή
----------	--------	-----------

<code>/api/v2/proposer</code>	POST	Πρόταση έγκυρων usernames με βάση τα στοιχεία του χρήστη στο Ίδρυμα
-------------------------------	-------------	---

Για την αναζήτηση του δοκιμαστικού χρήστη “**gunetdemo**”, όπου **ssn**="12312312312", **tin**="011111111" και **ssnCountry**=**tinCountry**="GR" τότε το JSON Body θα είναι ως εξής:

```
{
  "ssn": "12312312312",
  "ssnCountry": "GR",
  "tin": "011111111",
  "tinCountry": "GR"
}
```

Οπότε, το request στο ενδεικτικό endpoint θα είναι ως εξής. Το uLookup θα επιστρέψει ένα ή περισσότερα διαθέσιμα και έγκυρα loginNames.

```
curl -x POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/proposer \
--header 'Content-Type: application/json' \
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJyImhqFOKe' \
--data '{ \
  "ssn": "12312312312", \
  "ssnCountry": "GR", \
  "tin": "011111111", \
  "tinCountry": "GR" \
}'
```

Αν το πρόσωπο έχει ήδη λογαριασμό στο Ίδρυμα, τότε θα επιστραφεί το παρακάτω response:

Τα **proposedLoginNames** είναι διαθέσιμο προς απόδοση, όπως επίσης και τα **registeredLoginNames**.

```
{
  "proposedLoginNames": [
    "dokimastikosct"
  ],
  "Message": "The user already has an account",
  "registeredLoginNames": [
    "gunetdemo",
    "gunetdemo.p"
  ],
  "responseCode": "2110"
}
```

Τα usernames που προτείνει ο proposer στο παραπάνω σενάριο προκύπτει από τα πεδία **firstNameEn**, **lastNameEn**, **firstNameEl**, **lastNameEl**, **middleNameEn**, **middleNameEl**, **fatherFirstNameEn**, **fatherLastNameEl**. Αυτά που είναι υποχρεωτικά είναι τουλάχιστον ένα από τα **firstName** και τουλάχιστον ένα από τα **lastName** σε κάποιο view. Αν κάποιο από τα υπόλοιπα στοιχεία δεν υπάρχει, τότε δε λαμβάνεται υπόψη. Επίσης, τα παραγόμενα usernames ελέγχονται για την εγκυρότητα και τη διαθεσιμότητά τους. Σε κάθε call επιλέγεται ένα διαφορετικό loginName, οπότε τα προτεινόμενα loginNames ενδέχεται να διαφέρουν από call σε call.

Υπάρχει και η δυνατότητα πρότασης username με βάση firstName και lastName που δίνονται στο request. Ένα παράδειγμα request body είναι το παρακάτω:

```
{
  "firstName": "Christos",
  "lastName": "Dokimastikos"
}
```

Τα πεδία αυτά μπορεί να είναι και στα Ελληνικά. Η πρόταση loginName γίνεται με βάση τα στοιχεία αυτά. Αν δοθούν τα ταυτοποιητικά ζεύγη, τότε αυτά κάνουν override τα firstName και lastName (δηλαδή τα firstName και lastName αγνοούνται).

Ένα ενδεικτικό response body είναι το παρακάτω:

```
{
  "proposedLoginNames": [
    "chridokimas"
  ],
  "Message": "This is the first account for the user",
  "registeredLoginNames": null,
  "responseCode": "2200"
}
```

Υπάρχει η δυνατότητα ενός empty request. Σε αυτήν την περίπτωση επιστρέφεται ένα username με το format που έχει καθοριστεί στο configuration της Υπηρεσίας.

Το request body θα είναι ως εξής:

```
{
}
```

Ένα ενδεικτικό response body για empty request είναι το παρακάτω:

```
{
  "proposedLoginNames": [
    "user4634"
  ],
  "Message": "This is the first account for the user",
  "registeredLoginNames": null,
  "responseCode": "2200"
}
```

Στο configuration έχουμε ορίσει ότι το proposed loginName θα έχει μήκος 8 χαρακτήρων, το prefix θα είναι το "user" και το suffix θα είναι 4 τυχαίοι αριθμοί. Αυτό το loginName, όπως και τις υπόλοιπες περιπτώσεις ελέγχεται για τον έλεγχο εγκυρότητας και μοναδικότητάς του.

Τα responseCodes και τα Messages που επιστρέφονται περιγράφονται στους πίνακες στο ΠΑΡΑΡΤΗΜΑ.

7. TO ENDPOINT ΤΟΥ CHECKER

Το Request Body συμπεριλαμβάνει τα στοιχεία **registrationId**, **systemId**, **viewType** τα οποία αποτελούν τα στοιχεία μία ιδιότητας ενός χρήστη σε ένα view. Η περιγραφή των στοιχείων αυτών φαίνεται στο ΠΑΡΑΡΤΗΜΑ στο τέλος του εγγράφου.

Endpoint	Method	Περιγραφή
/api/v2/checker	POST	Έλεγχος εγκυρότητας των πεδίων μία συγκεκριμένης ιδιότητας ενός χρήστη στο view

Για την αναζήτηση πιθανών προβλημάτων στην εγγραφή ενός χρήστη με registrationId="1" και systemId="1" στο SIS view το JSON Body θα είναι ως εξής:

```
{
  "registrationId": "1",
  "systemId": "1",
  "viewType": "sis"
}
```

Οπότε, το request στο ενδεικτικό endpoint θα είναι ως εξής:

```
curl -x POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/checker \
--header 'Content-Type: application/json' \
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJylmhqFOKe' \
--data '{ \
  "registrationId": "1", \
  "systemId": "1", \
  "viewType": "sis", \
}'
```

Αν δεν υπάρχει πρόβλημα με την εγγραφή του χρήστη, τότε θα επιστραφεί το παρακάτω response:

```
{
  "checkEntities": [],
  "message": "User entities do not have errors",
  "responseCode": "2400"
}
```

Σε περίπτωση που εντοπιστεί κάποιο πρόβλημα, τότε στο πεδίο **checkEntities** θα εμφανιστούν τα μηνύματα λάθους. Για παράδειγμα, στην περίπτωση που το ssn δεν είναι έγκυρο στο view, το response θα είναι ως εξής:

```
{
  "checkEntities": [
    "errorMessages": [
```

```

    "Το πεδίο ssn έχει μη έγκυρο checksum"
  ],
  "registrationId": "1",
  "systemId": "1",
  "viewType": "sis"
],
"message": "User entities have errors",
"responseCode": "2410"
}

```

Το πεδίο **errorMessages** επιστρέφει τα μηνύματα λάθους που αφορούν την τρέχουσα εγγραφή στο view, δηλαδή την εγγραφή με registrationId="1" και systemId="1" στο SIS view. Το πεδίο **crossCheckMessages** επιστρέφει τα μηνύματα λάθους της τρέχουσας εγγραφής με τις αντίστοιχες για το ίδιο φυσικό πρόσωπο στο ίδιο view και στα υπόλοιπα views του Ιδρύματος.

Το πεδίο **userDetails** περιέχει τις τιμές των πεδίων που βρίσκονται στο view για την ιδιότητα με το **registrationId** ή το **systemId** που αναφέρονται στο response. Κάθε ένα πεδίο έχει την τιμή(value) που έχει στο view και μία τιμή **score**. Η τιμή **score** αν έχει την τιμή 1, τότε το πεδίο έχει λάθος και αν έχει την τιμή 0 έχει περάσει κάθε έλεγχο.

Η Υπηρεσία καθορίζει αν μία εγγραφή ανήκει στο ίδιο πρόσωπο με βάση τα στοιχεία ssn ή tin. Αν δεν υπάρχει κανένα από τα δύο, τότε ο έλεγχος γίνεται με βάση τα firstName{En,EI}, lastName{En,EI} και birthDate.

Στην παρακάτω περίπτωση το ssn δεν έχει έγκυρη τιμή στο sis view και στο hrms view έχει την έγκυρη τιμή, αλλά η τιμή του mobilePhone είναι διαφορετική στα δύο views.

```

{
  "checkEntities": [
    {
      "errorMessages": [
        "Το πεδίο ssn έχει μη έγκυρο checksum"
      ],
      "userDetails": {
        "ssn": {
          "value": "12312312311",
          "score": 1
        }
        "ssnCountry": {
          "value": "GR",
          "score": 0
        }
        "tin": {
          "value": "011111111",
          "score": 0
        }
        "tinCountry": {
          "value": "GR",
          "score": 0
        }
        .
        .
        .
        "enrollmentStatus": {
          "value": "active",
          "score": 0
        }
      }
    }
  ]
}

```

```

        "enrollmentStatusDate": {
            "value": "19900101",
            "score": 0
        }
    },
    "registrationId": "1",
    "systemId": "1",
    "viewType": "sis"
},
{
    "errorMessages": [
        "Το πεδίο ssn έχει μη έγκυρο checksum"
    ],
    "userDetails": {
        "ssn": {
            "value": "12312312311",
            "score": 1
        }
        "ssnCountry": {
            "value": "GR",
            "score": 0
        }
        "tin": {
            "value": "011111111",
            "score": 0
        }
        "tinCountry": {
            "value": "GR",
            "score": 0
        }
        .
        .
        .
        "enrollmentStatus": {
            "value": "active",
            "score": 0
        }
        "enrollmentStatusDate": {
            "value": "19900101",
            "score": 0
        }
    }
},
    "registrationId": "2",
    "systemId": "2",
    "viewType": "hrms"
}
],
"crossCheckMessages": [
    "Το πεδίο ssn έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων",
    "Το πεδίο mobilePhone έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων"
],
"message": "User entities have errors",
"responseCode": "2410",
}

```

8. TO ENDPOINT ΤΟΥ TRACER

Το Request Body συμπεριλαμβάνει μόνο το στοιχείο **loginName**, το οποίο αντιστοιχεί σε μία ιδιότητα σε κάποιο db view.

Endpoint	Method	Περιγραφή
/api/v2/tracer	POST	Παρουσίαση στοιχείων ενός προσώπου με βάση το loginName

Ενδεικτικά, το requestBody για την αναζήτηση των στοιχείων ενός χρήστη που σε κάποιο db view έχει στο πεδίο loginName την τιμή “gunetdemo” είναι το παρακάτω:

```
{
  "loginName": "gunetdemo"
}
```

Οπότε, το request στο ενδεικτικό endpoint θα είναι ως εξής:

```
curl -x POST --location 'http://ulookup-test.gunet.gr:8080/api/v2/tracer \
--header 'Content-Type: application/json' \
--header 'ApiKey: FMOSNZHOFXccSrMDQXnaVYyJylmhqFOKe' \
--data '{ \
  "loginName": "gunetdemo"
}'
```

Σε περίπτωση που δεν υπάρχει χρήστης με το ζητούμενο loginName επιστρέφεται το παρακάτω:

```
{
  "message": "User not found",
  "attributes": null,
  "responseCode": "2500"
}
```

Τα πεδία message και attributes παίρνουν τιμές που φαίνονται στον Πίνακα 9 στο ΠΑΡΑΡΤΗΜΑ.

Σε περίπτωση που το loginName στο request περιέχει μη αποδεκτούς χαρακτήρες ή μη αποδεκτό μήκος, τότε το request δε στέλνεται στο backend και επιστρέφεται το παρακάτω response:

```
{
  "message": "The user has the following attributes",
  "attributes": {
    "uid": [
      "gunetdemo"
    ],
    "cn": [
      "CHRISTOS DOKIMASTIKOS",
      "ΧΡΗΣΤΟΣ ΔΟΚΙΜΑΣΤΙΚΟΣ"
    ]
  }
}
```

```

],
"displayName": [
  "CHRISTOS DOKIMASTIKOS"
],
"dn": [
  "schGrAcPersonID=E2Q4KXGLNSPLB25T8TLLT5,ou=People,dc=<inst>,dc=gr"
],
"eduPersonAffiliation": [
  "student",
  "staff"
],
"eduPersonEntitlement": [
  "urn:mace:terena.org:tcs:personal-user",
  "urn:mace:dir:entitlement:common-lib-terms",
  "arcanum_admin",
  "urn:mace:gunet.gr:uauth:role:admin",
  "urn:mace:gunet.gr:grafana:role:admin",
  "urn:mace:gunet.gr:ulookup:role:admin"
],
"eduPersonOrgDN": [
  "dc=aueb,dc=gr"
],
"eduPersonPrimaryAffiliation": [
  "staff"
],
"givenName": [
  "ΧΡΗΣΤΟΣ",
  "CHRISTOS"
],
"mail": [
  "gunetdemo@<inst>.gr"
],
"mobile": [
  "+306932933048"
],
"schacCountryOfCitizenship": [
  "GR"
],
"schacDateOfBirth": [
  "19770823"
],
"schacGender": [
  "1"
],
"schacHomeOrganization": [
  "<inst>.gr"
],
"schacPersonalPosition": [
  "urn:mace:terena.org:schac:personalPosition:gr:<inst>.gr:student:Undergraduate:313",
  "urn:mace:terena.org:schac:personalPosition:gr:<inst>.gr:staff:scientific-assistant:0"
],
"schacPersonalUniqueCode": [
  "urn:mace:terena.org:schac:personalUniqueCode:gr:<inst>.gr:0:7016",
  "urn:mace:terena.org:schac:personalUniqueCode:gr:<inst>.gr:313:AM123",
  "urn:schac:personalUniqueCode:int:esi:aueb.gr:E2Q4KXGLNSPLB25T8TLLT5"
],
"schacPersonalUniqueID": [
  "urn:mace:terena.org:schac:personalUniqueID:gr:SSN:12312312312",
  "urn:mace:terena.org:schac:personalUniqueID:gr:TIN:011111111",\
],

```

```
"schacYearOfBirth": [
  "1977"
],
"schGrAcEmployment": [
  "urn:mace:gunet.gr:schgrac:employment:gr:<inst>.gr:permanent::0"
],
"schGrAcEnrollment": [
  "urn:mace:gunet.gr:schgrac:enrollment:gr:<inst>.gr:12:full-time:313"
],
"schGrAcInscription": [
  "urn:mace:gunet.gr:schgrac:inscription:gr:<inst>.gr:1:2009:313"
],
"schGrAcPersonID": [
  "E2Q4KXGLNSPLB25T8TLLT5"
],
"schGrAcPersonSSN": [
  "12312312312"
],
"schGrAcPersonTIN": [
  "0111111111"
],
"sn": [
  "ΔΟΚΙΜΑΣΤΙΚΟΣ",
  "DOKIMASTIKOS"
],
"cn-lang-el": [
  "ΧΡΗΣΤΟΣ ΔΟΚΙΜΑΣΤΙΚΟΣ"
],
"cn-lang-en": [
  "CHRISTOS DOKIMASTIKOS"
],
"givenName-lang-el": [
  "ΧΡΗΣΤΟΣ"
],
"givenName-lang-en": [
  "CHRISTOS"
],
"sn-lang-el": [
  "ΔΟΚΙΜΑΣΤΙΚΟΣ"
],
"sn-lang-en": [
  "DOKIMASTIKOS"
],
},
"responseCode": "2510"
}
```

ΠΑΡΑΡΤΗΜΑ

Πίνακας 1: Πεδία των Requests και η περιγραφή τους.

Πεδίο Request	Περιγραφή	Endpoint
ssn	Ο Αριθμός Μητρώου Κοινωνικής Ασφάλισης (ΑΜΚΑ) του χρήστη.	finder validator proposer
ssnCountry	Η χώρα έκδοσης του ΑΜΚΑ. Θα πρέπει να ακολουθεί το ISO 3166-1 alpha-2. Πχ. Αν η χώρα έκδοσης είναι η Ελλάδα, τότε το πεδίο ssnCountry θα έχει την τιμή "GR".	finder validator proposer
tin	Ο Αριθμός Φορολογικού Μητρώου (ΑΦΜ) του χρήστη	finder validator proposer
tinCountry	Η χώρα έκδοσης του ΑΦΜ. Θα πρέπει να ακολουθεί το ISO 3166-1 alpha-2.	finder validator proposer
loginName	Το username του χρήστη προς αναζήτηση	validator
firstName	Το μικρό όνομα για τη δημιουργία username από τον proposer	proposer
lastName	Το επώνυμο για τη δημιουργία username από τον proposer	proposer
registrationId	Το registrationId της ιδιότητας του χρήστη στο view	checker
systemId	Το systemId της ιδιότητας του χρήστη στο view	checker
viewType	Το view στο οποίο γίνεται το query του checker	checker

Πίνακας 2: Πεδία των Responses και η περιγραφή τους.

Πεδίο Response	Περιγραφή	Endpoint
responseCode	Κωδικός λειτουργίας του uLookup. Οι πίνακες 3-5 δείχνουν τα response codes για το κάθε endpoint.	finder validator proposer
Message	Ένα μήνυμα που περιγράφει το αντίστοιχο responseCode.	finder validator proposer
registrationId	Το registrationId της Ιδιότητας του χρήστη στο view του Ιδρύματος	finder
systemId	Το systemId της Ιδιότητας του χρήστη στο view του Ιδρύματος	finder

loginName	Το username της ιδιότητας του χρήστη στο view του Ιδρύματος	finder
viewType	Το view στο οποίο βρίσκεται η ιδιότητα. Μπορεί να έχει τις τιμές “sis”, “hrms” και “elke”	finder
userStatusDate	Το πεδίο enrollmentStatusDate/employeeStatusDate από το view.	
activationStatus	Μπορεί να έχει τις τιμές “active”, “pending”, “inactive” <ul style="list-style-type: none"> • “active”: Πρόκειται για μία ιδιότητα που έχει enrollmentStatus/employeeStatus=“active” ή “interim” στο view και ο χρήστης έχει ενεργοποιήσει το λογαριασμό του μέσω uRegister. • “pending”: Πρόκειται για μία ιδιότητα που έχει enrollmentStatus/employeeStatus=“active” ή “interim” στο view και ο χρήστης ΔΕΝ έχει ενεργοποιήσει το λογαριασμό του μέσω uRegister. • “inactive”: Πρόκειται για μία ιδιότητα που έχει enrollmentStatus/employeeStatus διάφορο του “active” ή “interim” 	finder
registredLoginNames	Τα usernames με τα οποία το πρόσωπο στο request έχει ενεργό λογαριασμό και είναι διαθέσιμα προς απόδοση .	validator proposer
proposedLoginNames	Τα usernames τα οποία προτείνονται από την Υπηρεσία και είναι διαθέσιμα προς απόδοση .	proposer
responseStatus	Παίρνει τις τιμές “available”, “owned” και “reserved” <ul style="list-style-type: none"> • “available”: Το loginName στο request είναι διαθέσιμο προς απόδοση. • “owned”: Το loginName στο request ανήκει ήδη σε ενεργό λογαριασμό που ανήκει σε άλλο χρήστη. • “reserved”: Το loginName στο request ανήκει ήδη σε εγγραφή στο view, για την οποία δεν έχει ενεργοποιηθεί ο λογαριασμός ακόμα. • “invalid”: Προέκυψε κάποιο σφάλμα. 	validator
checkEntities	Πρόκειται για μία λίστα αντικειμένων, όπου το κάθε αντικείμενο περιέχει τα μηνύματα λάθους της κάθε ιδιότητας για τον συγκεκριμένο χρήστη. Περιέχει τα πεδία: <ul style="list-style-type: none"> • errorMessagees: Μηνύματα λάθους της συγκεκριμένης εγγραφής στο view. (Οι τιμές του πεδίου περιγράφονται στο ΠΑΡΑΡΤΗΜΑ) • crossCheckMessages: Μηνύματα λάθους της εγγραφής που αφορούν ασυμφωνίες με αντίστοιχες εγγραφές του χρήστη στο ίδιο ή σε άλλο view. (Οι τιμές του πεδίου περιγράφονται στο ΠΑΡΑΡΤΗΜΑ) • registrationId: Το registrationId του χρήστη στο view που ορίζεται viewType. • systemId: Το systemId του χρήστη στο view που ορίζεται στο viewType. • viewType: Το view για το οποίο γίνεται το ερώτημα. 	

Θα έχει μία από τις τιμές “sis”, “hrms” και “elke”.

Πίνακας 3: responseCodes του finder/

Response Code	Response Message
2300	The user does not have Identities
2310	The user has the following Identities
2320	SSN or TIN belong to multiple users
2328	Problem with the connection to DB
2329	Undefined
2330	Invalid Request Data

Πίνακας 4: responseCodes του proposer/

Response Code	Response Message
2200	This is the first account for the user
2210	The user already has an account
2221	The user does not exist
2222	Found multiple firstName and lastName pairs for the user
2223	Can't produce proposed loginNames
2224	Proposer can't produce loginNames due to errors identified by Validator. You should run Validator first
2225	SSN or TIN belong to multiple users
2230	Invalid Request Data
2238	Problem with the connection to DB
2239	Undefined

Πίνακας 5: responseCodes του validator/

Response Code	Response Message	Status
2100	The loginName is available, no one using it on the ldap or views	available
2110	The loginName is available, this is the first account for the user	available
2111	The loginName <loginName> is already owned by this user	available
2112	The loginName <loginName> is already owned by another user	owned
2113	The loginName <loginName> is available. User already has an account	available
2114	The loginName <loginName> is already owned by another user, but this user already has an account	owned
2115	The loginName belonged to an inactive user that exceeded the retention period	available
2120	The loginName <loginName> is already owned by a manually created user in DS. URegister procedure will fail	owned
2130	Invalid Request Data	invalid
2131	The loginName <loginName> belongs to a different user in the SIS VIEW.	reserved
2132	The loginName <loginName> belongs to a different user in the HRMS VIEW	reserved
2133	The loginName <loginName> belongs to a different user in the ELKE VIEW	reserved
2134	All ssn and tin are null in the view for <loginName>	invalid
2135	The loginName <loginName> belongs to multiple users in the views	reserved
2136	Inconsistent ssn and tin in the request	invalid
2138	Problem with the connection to DB	invalid
2139	Undefined	invalid

2140	The loginName <loginName> is available	available
2141	The loginName <loginName> is already owned by another user in IDM	owned
2142	The loginName <loginName> is already owned by a user in DS	owned
2143	The loginName <loginName> belongs to a user in SIS	reserved
2144	The loginName <loginName> belongs to a user in HRMS	reserved
2145	The loginName <loginName> belongs to a user in ELKE	reserved

Πίνακας 6: responseCodes του checker/

Response Code	Response Message
2400	User entities do not have errors
2401	There is no user with the provided registrationId or systemId
2410	User entities have errors
2420	There are more than one user with the provided registrationId or systemId
2428	Problem with the connection to DB
2429	Undefined
2430	Invalid Request Data
2431	SIS DB View is not enabled
2432	HRMS DB View is not enabled
2433	ELKE DB View is not enabled

Πίνακας 7: errorMessages του checker/

code	Μήνυμα
301	Το πεδίο ssn παρουσιάζει μη έγκυρη τιμή (GR format : 9 συνεχόμενα ψηφία/digits)

302	Το πεδίο ssn έχει μη έγκυρο checksum
303	Το πεδίο ssn περιέχει κενά
304	Το πεδίο ssn είναι κενό
305	Το πεδίο ssn παρουσιάζει μη έγκυρη τιμή (CY format : 10 συνεχόμενα ψηφία/digits)
401	Το πεδίο ssnCountry εμφανίζει τιμή εκτός των προκαθορισμένων τιμών στον Κώδικα Χωρών
402	Το πεδίο ssnCountry είναι κενό
501	Το πεδίο tin παρουσιάζει μη έγκυρη τιμή (format : 9 συνεχόμενα ψηφία/digits)
502	Το πεδίο tin έχει μη έγκυρο checksum
503	Το πεδίο tin είναι κενό
601	Το πεδίο tinCountry εμφανίζει τιμή εκτός των προκαθορισμένων τιμών στον Κώδικα Χωρών
602	Το πεδίο tinCountry είναι κενό
003	Ένα από τα πεδία ssn, ssnCountry είναι NULL ενώ το άλλο είναι συμπληρωμένο
004	Ένα από τα πεδία tin, tinCountry είναι NULL ενώ το άλλο είναι συμπληρωμένο
100	Το πεδίο registrationId είναι NULL
101	Το πεδίο registrationId περιέχει κενά
102	Το πεδίο registrationId είναι κενό
110	Το πεδίο systemId είναι NULL
111	Το πεδίο systemId περιέχει κενά

112	Το πεδίο systemId είναι κενό
280	Το πεδίο academicID περιέχει μη αποδεκτούς χαρακτήρες (A-Z0-9) ή έχει μήκος εκτός των τιμών {22,24}
281	Το πεδίο academicID είναι κενό
282	Το πεδίο academicID περιέχει κενά
290	Το πεδίο departmentId είναι NULL
291	Το πεδίο departmentId δεν έχει αριθμητική τιμή
292	Το πεδίο departmentId είναι κενό
293	Το πεδίο departmentId περιέχει κενά
001	Τα πεδία mobilePhone, extEmail είναι ταυτόχρονα NULL
701	Το πεδίο mobilePhone δεν ακολουθεί το ελληνικό/κυπριακό format : (+3069xxxxxxxx/+3579xxxxxxxx)
702	Το πεδίο mobilePhone δεν ξεκινά με '+' ή δεν έχει 7 έως 15 ψηφία
703	Το πεδίο mobilePhone είναι κενό
704	Το πεδίο mobilePhone περιέχει κενά
801	Το πεδίο extEmail δεν είναι έγκυρη διεύθυνση mail
802	Το πεδίο extEmail είναι κενό
803	Το πεδίο extEmail περιέχει κενά
900	Το πεδίο birthDate είναι NULL
901	Το πεδίο birthDate δεν έχει έγκυρο format
902	Το πεδίο birthDate είναι κενό
903	Το πεδίο birthDate περιέχει κενά

910	Το πεδίο birthYear είναι NULL
911	Το πεδίο birthYear δεν έχει έγκυρη τιμή έτους
912	Το πεδίο birthYear δεν ταιριάζει με το πεδίο birthDate
913	Το πεδίο birthYear είναι κενό
914	Το πεδίο birthYear περιέχει κενά
005	Τα πεδία firstNameEi και firstNameEn είναι και τα δύο NULL
006	Τα πεδία lastNameEi και lastNameEn είναι και τα δύο NULL
1100	Το πεδίο loginName είναι NULL
1101	Το πεδίο loginName περιέχει μη αποδεκτούς χαρακτήρες ή δεν έχει το αποδεκτό μήκος
1102	Το πεδίο loginName είναι κενό
1103	Το πεδίο loginName περιέχει κενά
1201	Το πεδίο firstNameEi περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (Α-Ωα-ω -)
1202	Το πεδίο firstNameEi περιέχει κενούς χαρακτήρες
1203	Το πεδίο firstNameEi είναι κενό
1301	Το πεδίο firstNameEn περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (Α-Zα-z -)
1302	Το πεδίο firstNameEn περιέχει κενούς χαρακτήρες
1303	Το πεδίο firstNameEn είναι κενό
1401	Το πεδίο lastNameEi περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (Α-Ωα-ω -)
1402	Το πεδίο lastNameEi περιέχει κενούς χαρακτήρες
1403	Το πεδίο lastNameEi είναι κενό
1501	Το πεδίο lastNameEn περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (Α-Zα-z -)
1502	Το πεδίο lastNameEn περιέχει κενούς χαρακτήρες

1503	Το πεδίο lastNameEn είναι κενό
1600	Το πεδίο gender είναι NULL
1601	Το πεδίο gender εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('0','1','2','9')
1602	Το πεδίο gender είναι κενό
1603	Το πεδίο gender περιέχει κενά
1700	Το πεδίο citizenship είναι NULL
1701	Το πεδίο citizenship εμφανίζει τιμή εκτός των προκαθορισμένων τιμών στον Κώδικα Χωρών
1702	Το πεδίο citizenship περιέχει κενά
1703	Το πεδίο citizenship είναι κενό
1801	Το πεδίο middleNameEl περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Ωα-ω -.)
1802	Το πεδίο middleNameEl περιέχει κενούς χαρακτήρες
1803	Το πεδίο middleNameEl είναι κενό
1901	Το πεδίο middleNameEn περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Za-z -.)
1902	Το πεδίο middleNameEn περιέχει κενούς χαρακτήρες
1903	Το πεδίο middleNameEn είναι κενό
2001	Το πεδίο fatherFirstNameEl περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Ωα-ω -.)
2002	Το πεδίο fatherFirstNameEl περιέχει κενούς χαρακτήρες
2003	Το πεδίο fatherFirstNameEl είναι κενό
2101	Το πεδίο fatherFirstNameEn περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Za-z -.)
2102	Το πεδίο fatherFirstNameEn περιέχει κενούς χαρακτήρες
2103	Το πεδίο fatherFirstNameEn είναι κενό
2201	Το πεδίο motherFirstNameEl περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Ωα-ω -.)

2202	Το πεδίο motherFirstNameEi περιέχει κενούς χαρακτήρες
2203	Το πεδίο motherFirstNameEi είναι κενό
2301	Το πεδίο motherFirstNameEn περιέχει μη αποδεκτούς χαρακτήρες. Το Σύνολο των αποδεκτών χαρακτήρων είναι: (A-Za-z -)
2302	Το πεδίο motherFirstNameEn περιέχει κενούς χαρακτήρες
2303	Το πεδίο motherFirstNameEn είναι κενό
2700	Το πεδίο accountStatus είναι NULL
2701	Το πεδίο accountStatus εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('enabled', 'disabled')
3000	Το πεδίο inscriptionAcYear είναι NULL
3001	Το πεδίο inscriptionAcYear δεν έχει έγκυρη ημερομηνία
3002	Το πεδίο inscriptionAcYear είναι κενό
2020	Το πεδίο enrollmentStatus είναι NULL
2120	Το πεδίο enrollmentStatus εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('active', 'interim', 'inactive', 'discontinued', 'graduated')
2400	Το πεδίο enrollmentType είναι NULL
2401	Το πεδίο enrollmentType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('undergraduate', 'postgraduate', 'doctoral')
2500	Το πεδίο attendanceType είναι NULL
2501	Το πεδίο attendanceType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('full-time', 'part-time')
3100	Το πεδίο enrollmentTerm είναι NULL
3101	Το πεδίο enrollmentTerm δεν έχει έγκυρη τιμή
3102	Το πεδίο enrollmentTerm είναι κενό
2600	Το πεδίο enrollmentStatusDate είναι NULL
2610	Το πεδίο enrollmentStatusDate είναι κενό
2021	Το πεδίο employeeStatus είναι NULL

2121	Το πεδίο employeeStatus εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('active', 'interim', 'inactive', 'sabbatical', 'training-leave', 'leave-of-absence', 'state-leave', 'posted', 'seconded', 'visiting-professor', 'resigned', 'retired', 'transferred', 'dismissed', 'suspended', 'available')
2410	Το πεδίο employeeType είναι NULL
2411	Το πεδίο employeeType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('faculty', 'staff', 'associate')
3200	Το πεδίο facultyType είναι NULL
3201	Το πεδίο facultyType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('professor', 'associate', 'assistant', 'emeritus', 'lecturer', 'appointee', 'instructor')
3202	Το πεδίο facultyType είναι κενό ή περιέχει κενά
3300	Το πεδίο staffType είναι NULL
3301	Το πεδίο staffType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('personnel', 'laboratory', 'scientific-assistant', 'admin', 'fin', 'secr', 'med', 'lib', 'tech', 'art', 'it', 'inter', 'eng', 'env', 'geo', 'drv', 'tyro', 'supp', 'lab')
3302	Το πεδίο staffType είναι κενό ή περιέχει κενά
3400	Το πεδίο contractType είναι NULL
3401	Το πεδίο contractType εμφανίζει τιμή εκτός των προκαθορισμένων τιμών ('permanent', 'regular', 'temporary', 'self')
3402	Το πεδίο contractType είναι κενό ή περιέχει κενά
2601	Το πεδίο employeeStatusDate είναι NULL
2620	Το πεδίο employeeStatusDate είναι κενό
3501	Τα αμοιβαία αποκλειόμενα πεδία staffType και facultyType εμφανίζουν ταυτόχρονα τιμές
3502	Τα πεδία staffType και facultyType είναι ταυτόχρονα NULL
3601	Το πεδίο staffType έχει τιμή staff ή associate ενώ το πεδίο staffType είναι NULL
3701	Το πεδίο employeeType έχει την τιμή 'faculty' ενώ το πεδίο facultyType είναι NULL

code	Μήνυμα
3800	Το πεδίο ssn έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3801	Το πεδίο ssnCountry έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3802	Το πεδίο tin έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3803	Το πεδίο tinCountry έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3804	Το πεδίο academicId έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3805	Το πεδίο firstNameEn έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3806	Το πεδίο firstNameEl έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3807	Το πεδίο lastNameEn έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3808	Το πεδίο lastNameEl έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3809	Το πεδίο birthDate έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3810	Το πεδίο birthYear έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3811	Το πεδίο gender έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3812	Το πεδίο citizenship έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3813	Το πεδίο mobilePhone έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3814	Το πεδίο extEmail έχει διαφορετική τιμή μεταξύ των διαφορετικών ιδιοτήτων
3901	Το πεδίο ssn έχει διαφορετική τιμή στο view και στην IDM Υποδομή
3902	Το πεδίο ssnCountry έχει διαφορετική τιμή στο view και στην IDM Υποδομή

3903	Το πεδίο tin έχει διαφορετική τιμή στο view και στην IDM Υποδομή
3904	Το πεδίο tinCountry έχει διαφορετική τιμή στο view και στην IDM Υποδομή
3905	Το πεδίο academicId έχει διαφορετική τιμή στο view και στην IDM Υποδομή

Πίνακας 9: responseCodes tracer/

Response Code	Response Message
2500	User not found
2510	The user has the following attributes
2520	Problem with the connection to Backend
2530	Invalid Request Data
2529	Undefined