

# IoTFeds

Καταναεμημένες Αγορές Δεδομένων για IoT Ομοσπονδίες  
Distributed Data Marketplaces for IoT Federations

Π1.5 – Τελική Έκδοση Ολοκληρωμένου Συστήματος

ΔΡΑΣΗ ΕΘΝΙΚΗΣ ΕΜΒΕΛΕΙΑΣ:  
«ΕΡΕΥΝΩ-ΔΗΜΙΟΥΡΓΩ-ΚΑΙΝΟΤΟΜΩ Β' ΚΥΚΛΟΣ»  
«ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ, ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ & ΚΑΙΝΟΤΟΜΙΑ» (ΕΠΑνΕΚ)  
Ειδική Υπηρεσία Διαχείρισης Επιχειρησιακού Προγράμματος Ανταγωνιστικότητα  
Επιχειρηματικότητα και Καινοτομία (ΕΥΔ ΕΠΑνΕΚ)  
Ειδική Υπηρεσία Διαχείρισης και Εφαρμογής Δράσεων στους τομείς  
Έρευνας, Τεχνολογικής Ανάπτυξης και Καινοτομίας  
(ΕΥΔΕ ΕΤΑΚ)



<b>Έργο</b>	IoTFeds
<b>Κωδικός Έργου</b>	T2EΔK-02178
<b>Ενότητα Εργασίας</b>	EE2 (Δ2.1)
<b>Υπεύθυνος Ενότητας</b>	ICOM
<b>Υπεύθυνος Παραδοτέου</b>	ICOM
<b>Έκδοση Παραδοτέου</b>	1.0
<b>Ημερομηνία</b>	10 Δεκεμβρίου, 2023

**Συντονιστής Παραδοτέου**

Απόστολος Νάσιου, ICOM

Έλvia Πιέτρη, ICOM

Κωνσταντίνος Τσάκαλος, ICOM

Μαρία Τσαρουχά, ICOM

Μάριος Χαραλαμπίδης, ICOM

**Υπεύθυνοι Συγγραφής**

Στέφανος Γεωργιάδης, EKETA

Ευάγγελος Αθανασάκης, EKETA

Στέλιος Γκούσκος, TERRACOM

Αναστάσιος Πεχλιβανίδης, EKETA

Σοφία Πολυμένη, EKETA

Ζήσης Σακελλαρίου, EKETA

Γεώργιος Σπανός, EKETA

## Ακρωνύμια

Ακρωνύμιο	Ορισμός
<b>API</b>	Application Programming Interface
<b>UI</b>	User Interface
<b>IoT</b>	Internet of Things
<b>BaaS</b>	Blockchain as a Service
<b>RAP</b>	Resource Access Proxy
<b>RDF</b>	Resource Description Framework
<b>RDFS</b>	Resource Description Framework Schema
<b>JSON</b>	JavaScript Object Notation
<b>URI</b>	Uniform Resource Identifier
<b>BIM</b>	Best-practice Information Model
<b>PIM</b>	Platform-specific Information Model
<b>AAM</b>	Authentication and Authorization Manager
<b>SM</b>	Semantic Manager
<b>RAP</b>	Resource Access Proxy
<b>RH</b>	Registration Handler

## Σύνοψη Παραδοτέου

Το παραδοτέο αυτό περιγράφει την τελική έκδοση του ολοκληρωμένου συστήματος της πλατφόρμας IoTFeds η οποία προέκυψε ως αποτέλεσμα της ενσωμάτωσης των υπηρεσιών των στοιχείων λογισμικού που αναπτύχθηκαν στα πλαίσια των τεχνικών δράσεων των ΕΕ2 και ΕΕ3.

Το παραδοτέο τεκμηριώνει και συνοδεύει τον κώδικα Ανοιχτής Πηγής (open source) του λογισμικού της τελικής έκδοσης του ολοκληρωμένου συστήματος. Η τελική αυτή έκδοση ενσωματώνει όλες τις λειτουργικότητες και μηχανισμούς τους συστήματος που αναπτύχθηκαν στις επιμέρους δράσεις και συνθέτουν την τελική πλατφόρμα IoTFeds, ακολουθώντας την αρχιτεκτονική που παρουσιάστηκε στο Π1.3 «Τελική Αρχιτεκτονική Συστήματος και Επιλεγμένες Τεχνολογίες». Η αλληλεπίδραση του χρήστη με το σύστημα IoTFeds γίνεται μέσω της γραφικής διεπαφής χρήστη του στοιχείου λογισμικού Administration για τη διαχείριση ομοσπονδιών, χρηστών και πλατφορμών ενώ το στοιχείο λογισμικού Marketplace είναι κυρίως υπεύθυνο για την πρόσβαση του χρήστη σε υπηρεσίες αγοράς όπως είναι η σύνθεση και αγορά ενός προϊόντος IoT δεδομένων. Αιτήματα χρήστη μέσω των διεπαφών των στοιχείων αυτών εκκινούν τις απαραίτητες ροές επικοινωνίας στο σύστημα με άλλα εμπλεκόμενα στοιχεία του λογισμικού του symbIoTe και του BaaS.

Στην αναφορά αυτή περιγράφονται οι διεπαφές/APIs (Application Programming Interfaces) που απαιτούνται για την υποστήριξη των λειτουργικοτήτων της πλατφόρμας IoTFeds και η λειτουργική δοκιμή των προσφερόμενων υπηρεσιών του συστήματος. Η έκδοση αυτή αποτελεί την επέκταση της ενδιάμεσης έκδοσης του ολοκληρωμένου συστήματος που περιγράφηκε στο Π1.4 «Ενδιάμεση Έκδοση Ολοκληρωμένου Συστήματος» και καλύπτει πλήρως και επιτυχώς όλες τις απαιτήσεις και τους μηχανισμούς που είχαν ήδη ορισθεί από τις τεχνικές δράσεις της ΕΕ2 και ΕΕ3.

## Πίνακας Περιεχομένων

1	Αντικείμενο και Δομή του Παραδοτέου.....	11
2	Ενσωμάτωση του ολοκληρωμένου συστήματος IoTFeds.....	13
3	Σχεδιασμός και δοκιμή του ολοκληρωμένου συστήματος IoTFeds – Ενδιάμεση έκδοση 16	
3.1	Υποσύστημα symbIoTe.....	16
3.1.1	Στοιχείο λογισμικού για τη διαχείριση ομοσπονδιών, χρηστών και πλατφορμών (Administration) .....	16
3.1.1.1	Περιγραφή.....	16
3.1.1.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	17
3.1.1.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	17
3.1.1.3.1	Διεπαφές για τη διαχείριση IoT ομοσπονδίας.....	17
3.1.1.4	Δοκιμή και επαλήθευση (Test cases) .....	18
3.1.1.4.1	Διεπαφές για τη διαχείριση IoT ομοσπονδίας.....	18
3.1.2	Στοιχείο λογισμικού για τη διαχείριση IoT πόρων (symbIoTe API).....	31
3.1.2.1	Περιγραφή.....	31
3.1.2.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	32
3.1.2.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	32
3.1.2.4	Δοκιμή και επαλήθευση (Test cases) .....	33
3.2	Στοιχείο λογισμικού Αγοράς.....	39
3.2.1	Στοιχείο λογισμικού για τη διαχείριση IoT προϊόντων και πόρων (symbIoTe Marketplace).....	39
3.2.1.1	Περιγραφή.....	39
3.2.1.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	39
3.2.1.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	39
3.3	<b>Υποσύστημα BaaS</b> .....	63
3.3.1	Στοιχείο λογισμικού για τη διαχείριση χρηστών.....	63
3.3.1.1	Περιγραφή.....	63
3.3.1.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	64
3.3.1.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	64
3.3.1.4	Δοκιμή και επαλήθευση (Test cases) .....	65
3.3.2	Στοιχείο λογισμικού για τη διαχείριση ομοσπονδιών.....	70
3.3.2.1	Περιγραφή.....	70
3.3.2.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	70

3.3.2.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	70
3.3.2.4	Δοκιμή και επαληθευση (Test cases) .....	71
3.3.3	Στοιχείο λογισμικού αγοράς .....	80
3.3.3.1	Περιγραφή .....	80
3.3.3.2	Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	81
3.3.3.3	Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος.....	81
3.3.3.4	Δοκιμή και επαληθευση (Test cases) .....	83
3.3.3.4.1	Δοκιμή διεπαφών αγοράς.....	83
3.3.3.4.2	Δοκιμή διεπαφών Ασφαλούς Πρόσβασης.....	91
3.3.3.4.3	Δοκιμή διεπαφών Ανταλλαγής Προϊόντων .....	94
3.3.3.4.4	Δοκιμή διεπαφών Αναζήτησης και Διαφήμισης προϊόντων .....	96
3.3.3.4.5	Δοκιμή διεπαφών Καταγραφής και Ελέγχου Συναλλαγών.....	99
3.3.3.4.6	Δοκιμή διεπαφών Φήμης.....	104
4	Λειτουργική δοκιμή και επίδειξη συστήματος.....	107
4.1	Λειτουργική δοκιμή και επίδειξη πύλης αγοράς .....	107
4.2	Λειτουργική δοκιμή και επίδειξη διαχείρισης ομοσπονδιών .....	133
5	Συμπεράσματα.....	158
	Αναφορές.....	159

Εικόνα 1: Αρχιτεκτονική IoTFeds.....	14
Εικόνα 2:Αποτέλεσμα διαγραφής L1 πόρου από το symbIoTe. ....	36
Εικόνα 3: Διαγραφή L2 πόρου από το symbIoTe. ....	37
Εικόνα 4: Παράμετροι για την αγορά ενός προϊόντος.....	41
Εικόνα 5 Κλήση της υπηρεσίας του marketplace για την αγορά ενός προϊόντος. ....	41
Εικόνα 6: Κλήση της υπηρεσίας του marketplace για την βαθμολόγηση ενός προϊόντος. ....	41
Εικόνα 7: Απάντηση επιτυχούς βαθμολόγησης ενός προϊόντος. ....	42
Εικόνα 8: Απάντηση ερωτήματος για πρόσβαση σε δεδομένα ροής. ....	42
Εικόνα 9: Κλήση της υπηρεσίας του marketplace για την ανάκτηση των προϊόντων της καθολικής αγοράς.....	42
Εικόνα 10: Ανάκτηση λίστας των προϊόντων της καθολικής αγοράς. ....	42
Εικόνα 11 Κλήση της υπηρεσίας του marketplace για ανάκτηση των προϊόντων της ομόσπονδης αγοράς «fedtsktsak». ....	43
Εικόνα 12: Απάντηση marketplace για ανάκτηση των προϊόντων της ομόσπονδης αγοράς «fedtsktsak».....	43
Εικόνα 13: Παράδειγμα κλήσης της υπηρεσίας του marketplace για αναζήτηση προϊόντων με βάση κάποια κριτήρια. ....	43
Εικόνα 14: Παράδειγμα κλήσης της υπηρεσίας του marketplace για αναζήτηση προϊόντων με βάση κάποια κριτήρια. ....	44
Εικόνα 15: Παράδειγμα κλήσης της υπηρεσίας του marketplace για την προσπέλαση ενός προϊόντος.....	44
Εικόνα 16: Παράδειγμα απάντησης της υπηρεσίας του marketplace για την προσπέλαση ενός προϊόντος.....	44
Εικόνα 17: Κλήση της υπηρεσίας του marketplace για την δημιουργία προϊόντος.....	44
Εικόνα 18 Απάντηση της υπηρεσίας του marketplace για την δημιουργία προϊόντος. ....	45
Εικόνα 19: Κλήση της υπηρεσίας του marketplace για την τιμή ενός προϊόντος. ....	45
Εικόνα 20: Απάντηση της υπηρεσίας του marketplace για την τιμή ενός προϊόντος. ....	45
Εικόνα 21: Κλήση της υπηρεσίας του marketplace για την διαγραφή προϊόντος.....	46
Εικόνα 22: Κλήση της υπηρεσίας του marketplace για την διαγραφή προϊόντος.....	46
Εικόνα 23: Αναζήτηση πόρων(L1) στην υπηρεσία Marketplace.....	48
Εικόνα 24 Απάντηση στην αναζήτηση πόρων(L1) στην υπηρεσία Marketplace.....	48
Εικόνα 25 Κλήση για αναζήτηση ομόσπονδων πόρων στην υπηρεσία Marketplace. ....	49
Εικόνα 26: Αποτέλεσμα κλήσης για την αναζήτηση ομόσπονδων πόρων(L2) στην υπηρεσία Marketplace.....	49
Εικόνα 27 Κλήση για την προσπέλαση δεδομένων ενός IoT πόρου στην υπηρεσία Marketplace. ....	49
Εικόνα 28: Απάντηση προσπέλασης του IoT πόρου στην υπηρεσία Marketplace.....	50
Εικόνα 29 Κλήση της υπηρεσίας του marketplace για την προσπέλαση IoT πόρου. ....	50
Εικόνα 30: Δεδομένα προσπέλασης του IoT πόρου «isen1».....	51
Εικόνα 31 Κλήση υπηρεσίας του marketplace για καταχώρηση IoT πόρου.....	51
Εικόνα 32:Απάντηση καταχώρησης IoT πόρου μέσω του marketplace. ....	52
Εικόνα 33 Κλήση της υπηρεσίας Marketplace για τον διαμοιρασμό ενός πόρου σε μια ομοσπονδία.....	52
Εικόνα 34: Αποτέλεσμα επιτυχούς διαμοιρασμού ενός πόρου σε μια ομοσπονδία.....	52
Εικόνα 35 Κλήση της υπηρεσίας Marketplace για τον διαμοιρασμό ενός πόρου σε μια ομοσπονδία.....	53
Εικόνα 36: Παράδειγμα κλήσης στο marketplace για υποβολή πρότασης ανταλλαγής κουπονιών προϊόντων. ....	54
Εικόνα 37:Απάντηση υποβολής πρότασης ανταλλαγής στο marketplace. ....	54
Εικόνα 38: Απάντηση ανάκτησης πρότασης ανταλλαγής κουπονιών.....	55
Εικόνα 39: Απάντηση ακύρωσης πρότασης ανταλλαγής.....	55
Εικόνα 40:Απάντηση αποδοχής πρότασης ανταλλαγής.....	55
Εικόνα 41: Απάντηση απόρριψης πρότασης ανταλλαγής. ....	56
Εικόνα 42:Απάντηση αιτήματος ανάκτησης με τις υποβληθείσες προτάσεις του χρήστη. ....	56

Εικόνα 43: Απάντηση αιτήματος ανάκτησης με τις υποβληθείσες προτάσεις του χρήστη και κατάσταση completed. ....	57
Εικόνα 44: Ανάκτηση λίστας κουπονιών του χρήστη.....	57
Εικόνα 45:Ανάκτηση λίστας κουπονιών σε μία αγορά.....	58
Εικόνα 46: Απάντηση αιτήματος για την τοποθέτηση του κουπονιού ως ανταλλάξιμο.....	58
Εικόνα 47:Απάντηση αιτήματος απόσυρσης κουπονιού από ανταλλάξιμο.....	59
Εικόνα 48: Παράμετροι στο σώμα του ελέγχου BaaSAPI-1.A.....	65
Εικόνα 49: Παράμετροι στο σώμα του ελέγχου BaaSAPI-2.A.....	65
Εικόνα 50:Παράμετροι στο σώμα του ελέγχου BaaSAPI-3.A.....	66
Εικόνα 51:Επιστροφή πληροφοριών χρήστη από το BaaS.....	66
Εικόνα 52 Επιστροφή πληροφοριών όλων των χρηστών του BaaS.....	67
Εικόνα 53 Παράμετροι στο σώμα του ελέγχου BaaSAPI-6.A.....	67
Εικόνα 54 Παράμετροι στο σώμα του ελέγχου BaaSAPI-6.B.....	67
Εικόνα 55 Παράμετροι στο σώμα του ελέγχου BaaSAPI-7.A.....	68
Εικόνα 56 Παράμετροι στο σώμα του ελέγχου BaaSAPI-7.B.....	68
Εικόνα 57 Παράμετροι στο σώμα του ελέγχου BaaSAPI-8.A.....	68
Εικόνα 58 Αντικείμενο διαγραμμένου χρήστη στο BaaS.....	69
Εικόνα 59 Παράμετροι στο σώμα του ελέγχου BaaSAPI-8.B.....	69
Εικόνα 60: Παράμετροι στο σώμα ελέγχου BaaSAPI-9.A.....	72
Εικόνα 61: Επιστροφή πληροφοριών ομοσπονδίας απο το BaaS.....	72
Εικόνα 62: Επιστροφή πληροφοριών όλων των ομοσπονδιών απο το BaaS.....	73
Εικόνα 63: Παράμετροι στο σώμα του ελέγχου BaaSAPI-12.A.....	73
Εικόνα 64: Παράμετροι στο σώμα του ελέγχου BaaSAPI-12.Γ.....	73
Εικόνα 65: Παράμετροι στο σώμα του ελέγχου BaaSAPI-13.A.....	74
Εικόνα 66: Παράμετροι στο σώμα του ελέγχου BaaSAPI-13.B.....	74
Εικόνα 67: Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.A.....	74
Εικόνα 68 Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.B.....	75
Εικόνα 69: Αντικείμενο που επιστρέφεται απο τον έλεγχο BaaSAPI-14.B.....	75
Εικόνα 70: Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.Γ.....	75
Εικόνα 71 Παράμετροι στο σώμα του ελέγχου BaaSAPI-15.A.....	75
Εικόνα 72: Παράμετροι στο σώμα του ελέγχου BaaSAPI-16.A.....	76
Εικόνα 73: Αντικείμενο που επιστρέφεται απο τον έλεγχο BaaSAPI-16.A.....	76
Εικόνα 74: Παράμετροι στο σώμα του ελέγχου BaaSAPI-16.B.....	76
Εικόνα 75: Αρχικό σετ κανόνων της ομοσπονδίας που χρησιμοποιείται.....	77
Εικόνα 76: Αναμενόμενο αποτέλεσμα επιστροφής του ελέγχου BaaSAPI-17.B.....	78
Εικόνα 77: Αντικείμενο που περιγράφει την ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-17.B.....	79
Εικόνα 78: Παράμετροι στο σώμα του ελέγχου BaaSAPI-20.A για καταχώρηση ψήφου στην ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-18.B.....	79
Εικόνα 79: Επιστροφή πληροφοριών προϊόντος απο το BaaS.....	83
Εικόνα 80: Επιστροφή πληροφοριών πόρου δεδομένων απο το BaaS.....	84
Εικόνα 81: Επιστροφή πληροφοριών όλων των προϊόντων από το BaaS.....	84
Εικόνα 82 Επιστροφή πληροφοριών όλων των πόρων δεδομένων από το BaaS.....	85
Εικόνα 83 Παράμετροι στο σώμα του ελέγχου BaaSAPI -26.A.....	85
Εικόνα 84: Παράμετροι στο σώμα του ελέγχου BaaSAPI -27.A.....	86
Εικόνα 85 Επιστροφή πληροφοριών όλων των πόρων δεδομένων μίας ομοσπονδίας από το BaaS.....	86
Εικόνα 86 Επιστροφή πληροφοριών όλων των προϊόντων μίας ομοσπονδίας από το BaaS.....	87
Εικόνα 87: Επιστροφή πληροφοριών όλων των προϊόντων της καθολικής αγοράς από το BaaS.....	87
Εικόνα 88: Παράμετροι στο σώμα του ελέγχου BaaSAPI-32.A.....	87
Εικόνα 89 Επιστροφή πληροφοριών δημιουργίας προϊόντος από το BaaS.....	88
Εικόνα 90 Παράμετροι στο σώμα του ελέγχου BaaSAPI-33.A.....	88
Εικόνα 91 Παράμετροι στο σώμα του ελέγχου BaaSAPI-34.A.....	88
Εικόνα 92: Επιστροφή πληροφοριών υπολογισμού προϊόντος από το BaaS (Περίπτωση 1).....	89



Εικόνα 93 Επιστροφή πληροφοριών υπολογισμού προϊόντος από το BaaS (Περίπτωση 2).	89
Εικόνα 94 Παράμετροι στο σώμα του ελέγχου BaaSAPI-35.A	89
Εικόνα 95 Επιστροφή πληροφοριών αγοράς προϊόντος από το BaaS.	90
Εικόνα 96: Τα κουπόνια πρόσβασης ενός χρήστη μαζί με τις πληροφορίες τους, μετά από επιτυχημένη κλήση της αντίστοιχης διεπαφής.	92
Εικόνα 97: Σώμα κλήσης του ελέγχου BaaSAPI-36.A.	92
Εικόνα 98: Το ανανεωμένο (αν χρειάζεται) κουπόνι πρόσβασης του χρήστη μετά από επιτυχημένη πρόσβαση στο προϊόν.	93
Εικόνα 99: Μη έγκυρο σώμα στην διεπαφή BaaSAPI-37	94
Εικόνα 100 Έγκυρο σώμα στην διεπαφή BaaSAPI-37	94
Εικόνα 101: Αντικείμενο μετά από έγκυρη κλήση της διεπαφής BaaSAPI-37.	94
Εικόνα 102: Παράδειγμα επιστροφής αντικειμένου της διεπαφής BaaSAPI-25.	95
Εικόνα 103: Μη έγκυρο σώμα στην διεπαφή BaaSAPI-38.	95
Εικόνα 104 Έγκυρο σώμα για κλήση της διεπαφής BaaSAPI-38	95
Εικόνα 105: Μη έγκυρο σώμα κατά την κλήση της διεπαφής BaaSAPI-39.	96
Εικόνα 106: Έγκυρο σώμα κλήσης της διεπαφής BaaSAPI-39	96
Εικόνα 107: Σώμα κλήσης του ελέγχου BaaSAPI-30.A.	97
Εικόνα 108: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τους πόρους δεδομένων, μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-30.A.	97
Εικόνα 109: Σώμα κλήσης του ελέγχου BaaSAPI-31.A.	98
Εικόνα 110: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τα προϊόντα, μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-31.A.	98
Εικόνα 111: Σώμα κλήσης του ελέγχου BaaSAPI-43.A.	99
Εικόνα 112: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τις αποδείξεις συναλλαγών, μετά από επιτυχημένη κλήση του αντίστοιχου ελέγχου.	99
Εικόνα 113: Κενή λίστα αποδείξεων όταν δεν υπάρχουν διαθέσιμες αποδείξεις μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-43.B.	99
Εικόνα 114: Σώμα κλήσης του ελέγχου BaaSAPI-44.A.	100
Εικόνα 115: Σώμα κλήσης του ελέγχου BaaSAPI-46.A.	101
Εικόνα 116: Επιστρεφόμενο αντικείμενο μετά από επιτυχημένο έλεγχο εγκυρότητας μιας απόδειξης.	101
Εικόνα 117: Επιστρεφόμενο αντικείμενο μετά από μη επιτυχημένο έλεγχο εγκυρότητας μιας απόδειξης.	101
Εικόνα 118: Σώμα κλήσης του ελέγχου BaaSAPI-47.A.	101
Εικόνα 119: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τις καταγραφές χρήσεων προϊόντος, μετά από επιτυχημένη κλήση του αντίστοιχου ελέγχου.	102
Εικόνα 120: Κενή λίστα αποδείξεων όταν δεν υπάρχουν διαθέσιμες καταγραφές χρήσεων ενός προϊόντος, μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-47.B.	102
Εικόνα 121: Σώμα κλήσης του ελέγχου BaaSAPI-48.A.	102
Εικόνα 122: Επιστρεφόμενο αντικείμενο μετά από επιτυχημένο έλεγχο εγκυρότητας μιας καταγραφής χρήσης προϊόντος.	103
Εικόνα 123: Επιστρεφόμενο αντικείμενο μετά από μη επιτυχημένο έλεγχο εγκυρότητας μιας καταγραφής χρήσης προϊόντος.	103
Εικόνα 124 Σώμα κλήσης του ελέγχου BaaSAPI-42.A.	104
Εικόνα 125 Σώμα κλήσης ελέγχου BaaSAPI-40.A.	105
Εικόνα 126 Σώμα ελέγχου BaaSAPI-40.B	105
Εικόνα 127 Προϊόν μετά την αξιολόγηση από χρήστη.	106
Εικόνα 128: Προϊόν μετά από ενημέρωση όλων των πεδίων φήμης.	106

Πίνακας 1:Διεπαφές στοιχείου symbIoTe Administration για διαχείριση ομοσπονδίας. ....	17
Πίνακας 2:Περίπτωσης δοκιμής στοιχείου symbIoTe's Administration για διαχείριση ομοσπονδίας. ....	30
Πίνακας 3:Διεπαφές στοιχείου Administration για τη διαχείριση IoT πόρων. ....	32
Πίνακας 4:Περίπτωσης δοκιμής στοιχείου symbIoTe-API. ....	38
Πίνακας 5: Δοκιμή διεπαφών για τη διαχείριση προϊόντος.....	46
Πίνακας 6: Διεπαφές marketplace για την αγορά ανταλλαγής προϊόντων.....	53
Πίνακας 7: Διεπαφές στοιχείου BaaS για την διαχείριση χρηστών. ....	64
Πίνακας 8:Περίπτωσης δοκιμής στοιχείου BaaS για την διαχείριση χρηστών. ....	69
Πίνακας 9:Διεπαφές στοιχείου BaaS.....	70
Πίνακας 10:Περίπτωσης δοκιμής στοιχείου BaaS για τη διαχείριση ομοσπονδιών.....	79
Πίνακας 11: Διεπαφές στοιχείου BaaS για την αγορά. ....	82
Πίνακας 12:Περίπτωσης δοκιμής στοιχείου BaaS για την αγορά προϊόντων. ....	90
Πίνακας 13: Περιπτώσεις δοκιμής στοιχείου BaaS για την ασφαλή πρόσβαση προϊόντων. ....	93
Πίνακας 14: Περιπτώσεις δοκιμής στοιχείου BaaS για την αναζήτηση και διαφήμιση προϊόντων. ....	98
Πίνακας 15: Περιπτώσεις δοκιμής στοιχείου BaaS για την καταγραφή και τον έλεγχο συναλλαγών.....	103
Πίνακας 16: Σενάρια Ελέγχου για τις διεπαφές φήμης. ....	106

## 1 Αντικείμενο και Δομή του Παραδοτέου

Στο παραδοτέο αυτό περιγράφεται η τελική έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας. Η έκδοση αυτή ενσωματώνει όλους τους μηχανισμούς που αναπτύχθηκαν στις δράσεις των ενοτήτων EE2 και EE3 στο ολοκληρωμένο σύστημα του IoTFeds επεκτείνοντας το ενδιαμέσο λογισμικό symbIoTe και ενσωματώνοντας μηχανισμούς blockchain. Η τελική αυτή έκδοση του ολοκληρωμένου συστήματος βασίζεται στην αρχιτεκτονική και τα σενάρια που περιγράφηκαν στο Π1.3 και αποτελεί συνέχεια της ενδιαμέσης έκδοσης που παρουσιάστηκε στο Π1.4. Οι λειτουργικότητες που περιγράφηκαν στο Π1.4 αποτέλεσαν τη βάση για την περαιτέρω ανάπτυξη του συστήματος και την υλοποίηση μίας τελικής έκδοσης που ενσωματώνει πλήρως και επιτυχώς όλες τις υπηρεσίες και μηχανισμούς που αναπτύχθηκαν στα πλαίσια του έργου. Παρακάτω συνοψίζεται το σύνολο των λειτουργικοτήτων που υποστηρίζονται στην τελική έκδοση του ολοκληρωμένου συστήματος:

- Εγγραφή χρήστη.
- Σύνδεση χρήστη.
- Διαγραφή χρήστη.
- Εγγραφή IoT πλατφόρμας.
- Διαγραφή IoT πλατφόρμας.
- Εγκαθίδρυση ομοσπονδίας.
- Προσθήκη μέλους στην ομοσπονδία.
- Απομάκρυνση μέλους από ομοσπονδία.
- Επιλογή και τροποποίηση κανόνων ομοσπονδίας.
- Διαγραφή ομοσπονδίας και των μελών της.
- Εγγραφή ή τροποποίηση IoT συσκευών/πόρων με βάση το μοντέλο πληροφορίας IoT-Feds για την Έξυπνη Πόλη.
- Διαμοιρασμός των IoT συσκευών/πόρων σε μία ομοσπονδία.
- Δημιουργία προϊόντος IoT δεδομένων.
- Διαγραφή μιας IoT συσκευής/πόρου από μία ομοσπονδία ή το σύστημα.
- Διαγραφή προϊόντος από ομόσπονδη αγορά.
- Διαφήμιση προϊόντων δεδομένων στην ομόσπονδη ή καθολική αγορά και (ομόσπονδων ή δημόσιων) IoT συσκευών/πόρων.
- Υπολογισμός τιμής και αγορά προϊόντος IoT δεδομένων.
- Πρόσβαση στα δεδομένα (ομόσπονδων ή δημόσιων) IoT πόρων σε επίπεδο προϊόντων και πηγών δεδομένων.
- Αξιολόγηση προϊόντος που χρησιμοποιήθηκε και υπολογισμός φήμης.
- Ανταλλαγή κουπονιών πρόσβασης σε IoT δεδομένα.
- Έλεγχος συναλλαγών.

Στη συνέχεια στο παραδοτέο αυτό, οι διεπαφές που τροποποιήθηκαν ή υλοποιήθηκαν στην τελική έκδοση του ολοκληρωμένου συστήματος της IoTFeds επισημαίνονται με την ένδειξη “v2.0”, ενώ οι διεπαφές που επαναχρησιμοποιήθηκαν από την ενδιαμέση έκδοση επισημαίνονται με την ένδειξη “v1.0”.

Η συνέχεια του παραδοτέου έχει την εξής δομή:

Στο κεφάλαιο 2 συνοψίζεται η μεθοδολογία που ακολουθείται για την ενσωμάτωση και δοκιμή του ολοκληρωμένου συστήματος, ενώ συνοψίζεται και η αρχιτεκτονική του συστήματος και η διασύνδεση των υποσυστημάτων που το αποτελούν.

Στο κεφάλαιο 3 παρουσιάζονται τα στοιχεία λογισμικού των υποσυστημάτων που συμμετέχουν στους νέους μηχανισμούς και την τελική έκδοση του ολοκληρωμένου συστήματος καθώς και τις σχετικές διεπαφές που υλοποιήθηκαν και συνοδεύουν τον κώδικα Ανοιχτής Πηγής.

Στο κεφάλαιο 4 παρουσιάζεται η λειτουργική δοκιμή του ολοκληρωμένου συστήματος με την επίδειξη της χρήσης της πλατφόρμας IoTFeds από τη μεριά του τελικού χρήστη εστιάζοντας στην αλληλεπίδραση με τις παρεχόμενες γραφικές διεπαφές χρήστη του συστήματος.

Τέλος, το κεφάλαιο 5 ολοκληρώνει το παραδοτέο.

## 2 Ενσωμάτωση του ολοκληρωμένου συστήματος IoTFeds

Όπως περιγράφεται στο παραδοτέο Π1.3, η πλατφόρμα IoTFeds παρέχει όλους τους απαραίτητους μηχανισμούς: α) για την δημιουργία και διαχείριση ομοσπονδιών παρόχων και καταναλωτών δεδομένων IoT, οι οποίοι μπορούν εκεί να διαφημίσουν και να ανακαλύψουν διαθέσιμους IoT πόρους, καθώς και β) των ομοσπονδιακών αγορών που προκύπτουν από την δημιουργία τους για τη σύνθεση και διάθεση υπηρεσιών δεδομένων IoT (προϊόντων) που μπορεί να προέρχονται από διαφορετικές πηγές δεδομένων. Πέραν των ομοσπονδιακών αγορών, η πλατφόρμα IoTFeds παρέχει μια καθολική αγορά για την προώθηση υπηρεσιών δεδομένων που δημιουργούνται από τις ομοσπονδιακές αγορές (προϊόντων) σε τρίτους. Στο επίπεδο της αγοράς, η πλατφόρμα IoTFeds έχει το δικό της κρυπτο-νόμισμα (FedCoin) μέσω της χρήσης του οποίου πραγματοποιούνται οι συναλλαγές στο σύστημα, ενώ η πρόσβαση στα δεδομένα γίνεται μέσω της χρήσης κουπονιών. Σε αγορές προϊόντων, η πρόσβαση σε IoT δεδομένα επιτρέπεται μέσω της δημιουργίας προϊόντων

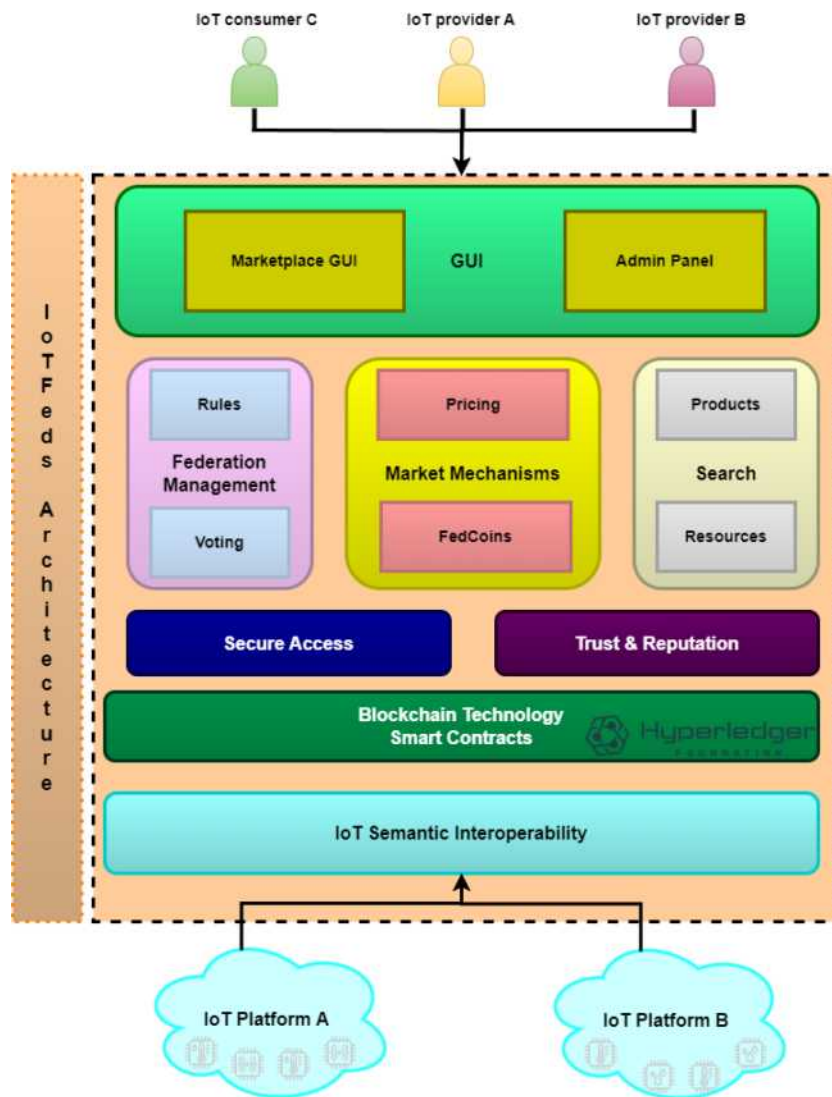
Στην ενότητα αυτή παρουσιάζεται η αρχιτεκτονική του τελικού συστήματος του IoTFeds και η μεθοδολογία για την ενσωμάτωση και δοκιμή της τελικής έκδοσης του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας. Η τελική αυτή έκδοση ενσωματώνει όλες τις λειτουργίες/μηχανισμούς που αναπτύχθηκαν ή τροποποιήθηκαν στις τεχνικές δράσεις του έργου και οι οποίες μπορούν να διαχωριστούν στις ακόλουθες κατηγορίες υπηρεσιών:

- διαχείριση ομοσπονδιών IoT (IoT παρόχων ή και καταναλωτών)
- υπηρεσίες ελέγχου πρόσβασης (σε πηγές και προϊόντα δεδομένων)
- κατανεμημένες υπηρεσίες αναζήτησης και ανακάλυψης (πηγών και προϊόντων δεδομένων)
- υπηρεσίες φήμης και εμπιστοσύνης (παρόχων, προϊόντων, πηγών και ομοσπονδιών)
- και τέλος υπηρεσίες αγοράς (έλεγχος, διεκπεραίωση συναλλαγών κ.α.).

Για την παροχή των παραπάνω υπηρεσιών, τα βασικά υποσυστήματα από τα οποία αποτελείται η IoTFeds πλατφόρμα (το ενδιάμεσο λογισμικό symbIoTe, το λογισμικό BaaS και το στοιχείο λογισμικού Marketplace) λειτουργούν συλλογικά στο ολοκληρωμένο σύστημα για να διευκολύνουν τη δημιουργία, λειτουργία και διαχείριση των ομοσπονδιών καθώς και των αγορών τους από τους χρήστες της πλατφόρμας. Συνοπτικά, οι κύριες λειτουργικότητες των υποσυστημάτων που συνδυάζονται είναι οι παρακάτω:

- Το λογισμικό symbIoTe επιτρέπει στις πλατφόρμες IoT να εκθέτουν επιλεγμένες συσκευές IoT και τα δεδομένα τους σε συστήματα τρίτων καθώς και να αλληλεπιδρούν μεταξύ τους για να δημιουργήσουν οριζόντιες λύσεις (λύσεις που βασίζονται σε δεδομένα από διαφορετικούς IoT παρόχους ή/και διαφορετικούς τομείς) παρέχοντας μια προσέγγιση σημασιολογικής διαλειτουργικότητας των δεδομένων και μια ενοποιημένη και ασφαλή διεπαφή για πρόσβαση σε πόρους IoT.
- Από την άλλη πλευρά, το υποσύστημα BaaS επιτρέπει τη δημιουργία και τη διαχείριση ταυτοτήτων χρηστών και πλατφορμών IoT για τη διασφάλιση της ασφάλειας και την παρακολούθηση των δεδομένων της πλατφόρμας IoTFeds, ενσωματώνοντας κατανεμημένους μηχανισμούς για την τιμολόγηση κάθε συναλλαγής και τον υπολογισμό των επιπέδων εμπιστοσύνης και φήμης κάθε παρόχου δεδομένων και τους πόρους τους.
- Τέλος, η αγορά (marketplace) λειτουργεί ως ένας διαδικτυακός χώρος συναλλαγών για τους χρήστες της πλατφόρμας, μέσω του οποίου πάροχοι IoT μπορούν συλλογικά να παρέχουν και να πωλούν δεδομένα IoT που μπορεί να προέρχονται από διάφορες πηγές δεδομένων σε δυνητικούς καταναλωτές IoT που επιθυμούν να έχουν πρόσβαση ή να τα αγοράσουν.

Η αρχιτεκτονική της τελικής έκδοσης του ολοκληρωμένου συστήματος που περιγράφηκε παραπάνω παρουσιάζεται αναλυτικά στην Εικόνα 1.



Εικόνα 1: Αρχιτεκτονική IoTFeds.

Συνοψίζοντας, οι βασικές υπηρεσίες που περιλαμβάνονται στην τελική έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας είναι οι παρακάτω:

- Εγγραφή χρήστη.
- Σύνδεση χρήστη.
- Διαγραφή χρήστη.
- Εγγραφή IoT πλατφόρμας.
- Διαγραφή IoT πλατφόρμας.
- Εγκαθίδρυση ομοσπονδίας.
- Προσθήκη μέλους στην ομοσπονδία.
- Απομάκρυνση μέλους από ομοσπονδία.
- Επιλογή και τροποποίηση κανόνων ομοσπονδίας.
- Διαγραφή ομοσπονδίας και των μελών της.
- Εγγραφή ή τροποποίηση IoT συσκευών/πόρων με βάση το μοντέλο πληροφορίας IoT-Feds για την Έξυπνη Πόλη.
- Διαμοιρασμός των IoT συσκευών/πόρων σε μία ομοσπονδία.
- Δημιουργία προϊόντος IoT δεδομένων.
- Διαγραφή μιας IoT συσκευής/πόρου από μία ομοσπονδία ή το σύστημα.

- Διαγραφή προϊόντος από ομόσπονδη αγορά.
- Διαφήμιση προϊόντων δεδομένων στην ομόσπονδη ή καθολική αγορά και (ομόσπονδων ή δημόσιων) IoT συσκευών/πόρων.
- Υπολογισμός τιμής και αγορά προϊόντος IoT δεδομένων.
- Πρόσβαση στα δεδομένα (ομόσπονδων ή δημόσιων) IoT πόρων σε επίπεδο προϊόντων και πηγών δεδομένων.
- Αξιολόγηση προϊόντος που χρησιμοποιήθηκε και υπολογισμός φήμης.
- Ανταλλαγή κουπονιών πρόσβασης σε IoT δεδομένα.
- Έλεγχος συναλλαγών.

Η διαδικασία που ακολουθήθηκε για την κατασκευή και ανάπτυξη του πλήρους συστήματος του έργου έχει περιγραφεί αναλυτικά στο Π1.4. Παρακάτω συνοψίζονται οι βασικές υποδομές που αξιοποιήθηκαν για την εγκατάσταση, ενσωμάτωση και δοκιμή του τελικού πρωτοτύπου:

- Το δημόσιο (public) αποθετήριο <https://github.com/iotfedsproject> χρησιμοποιήθηκε για την διάθεση του κώδικα ανοιχτής πηγής προς τρίτους και περιέχει οδηγίες για την εγκατάσταση κάθε απαραίτητου στοιχείου, όπως τα αρχεία yaml που αξιοποιούνται στην ανάπτυξη και ενσωμάτωση του ολοκληρωμένου συστήματος της πλατφόρμας IoTFeds.
- Το αποθετήριο εικόνων (image repository) <https://hub.docker.com/repository/docker/iotfedsproject/> στο DockerHub περιέχει τις εικόνες (container images) που ενσωματώνουν τον κώδικα κάθε στοιχείου λογισμικού, όπου είναι δυνατόν (εκτός δικτύου blockchain).
- Για την ανάπτυξη της τελικής έκδοσης του ολοκληρωμένου συστήματος και τις δοκιμές κατά τη διάρκεια της ενσωμάτωσης των νέων υπηρεσιών, τα υποσυστήματα εγκαταστάθηκαν στην υποδομή του εκάστοτε φορέα του υποσυστήματος (ICOM, EKETA): (α) το υποσύστημα symbIoTe (symbIoTe core services) με τη γραφική διεπαφή χρήστη στο url: <https://symbiote-integration.iotfeds.intracom-telecom.com/administration/> και το marketplace στο: <https://symbiote-integration.iotfeds.intracom-telecom.com/marketplace/> εγκαταστάθηκαν σε περιβάλλον Kubernetes στις υποδομές της ICOM, ενώ το υποσύστημα BaaS εγκαταστάθηκε στις υποδομές του EKETA στο παρακάτω url: [http:// 160.40.54.146:3009/baas/](http://160.40.54.146:3009/baas/).

Για την δοκιμή και εξασφάλιση της επιτυχούς διασύνδεσης των διαφορετικών στοιχείων λογισμικού μεταξύ τους και της ομαλής λειτουργίας του ενσωματωμένου συστήματος, ορίστηκαν και εκτελέστηκαν υποθέσεις και σενάρια δοκιμών (test cases) για τις υπηρεσίες κάθε υποσυστήματος που αποτελεί μέρος της πλατφόρμας IoTFeds και τις λειτουργικότητες του συστήματος. Για RESTful APIs, εκτελέστηκαν κατάλληλα αιτήματα HTTP(S) που καλύπτουν το πλήρες σύνολο δοκιμών για κάθε τελικό σημείο που θα δοκιμαστεί και η απάντηση που λαμβάνεται επικυρώνεται με βάση το αναμενόμενο αποτέλεσμα (απάντηση). Οι επιτυχείς δοκιμές αναφέρονται ως PASS και οι αποτυχημένες ως FAIL. Ομοίως, για τις γραφικές διεπαφές ορίστηκαν υποθέσεις και σενάρια δοκιμών και ελέγχθηκαν τα επιμέρους βήματα για τον έλεγχο της επιτυχούς λειτουργίας. Σε περίπτωση αποτυχίας οι προγραμματιστές που εμπλέκονται με το υπό δοκιμή λογισμικό ενημερώθηκαν για τη διόρθωση τυχόν σφαλμάτων που εντοπίστηκαν. Το αποτέλεσμα της δοκιμαστικής διαδικασίας θεωρείται αποδεκτό μετά την επιτυχή ολοκλήρωση όλων των υποθέσεων και σεναρίων δοκιμών και της επικύρωσης της αναμενόμενης συμπεριφοράς του συστήματος. Ο πηγαίος κώδικας με τις σχετικές λειτουργικότητες περιλαμβάνεται στην νέα έκδοση του ολοκληρωμένου συστήματος εφόσον όλες οι υποθέσεις και σενάρια δοκιμών κάθε διεπαφής έχουν ολοκληρωθεί με επιτυχία.



### 3 Σχεδιασμός και δοκιμή του ολοκληρωμένου συστήματος IoTFeds – Ενδιάμεση έκδοση

Στην ενότητα αυτή παρουσιάζονται οι διεπαφές που χρησιμοποιούνται για την διαχείριση της ομοσπονδίας στα επιμέρους υποσυστήματα.

#### 3.1 Υποσύστημα symbIoTe

Το υποσύστημα symbIoTe<sup>1</sup> επιτρέπει την καταγραφή, ανακάλυψη και διαμοιρασμό IoT συσκευών των IoT πλατφορμών που επιθυμούν να διαθέσουν IoT δεδομένα τους σε τρίτα συστήματα ή μεταξύ τους, καθώς επίσης προσφέρει μία ενοποιημένη διεπαφή για την πρόσβαση στα δεδομένα με ασφαλή τρόπο και μηχανισμούς για σημασιολογική διαλειτουργικότητα. Παρέχει λειτουργικότητες για τη διαχείριση χρηστών, ομοσπονδιών, πλατφορμών και των δεδομένων τους σε συνδυασμό με την τεχνολογία blockchain και την ενσωμάτωση των υπηρεσιών του BaaS υποσυστήματος.

Το symbIoTe υποσύστημα αποτελείται από ένα σύνολο από κεντριοποιημένες υπηρεσίες (symbIoTe core services) όπου οι εγγεγραμμένες IoT πλατφόρμες μπορούν να διαφημίσουν και να διαθέσουν τις IoT συσκευές τους σε άλλες πλατφόρμες του οικοσυστήματος IoTFeds ή προς τρίτους, παρόχους εφαρμογών.. Στα πλαίσια του έργου οι κεντριοποιημένες υπηρεσίες <symbIoTe-core> είναι διαθέσιμες στα παρακάτω urls:

<https://iotfeds-symbiotecore.intracom-telecom.com/administration/>, για την πιλοτική χρήση από τους φορείς του έργου,

<https://symbiote-core.iotfeds.intracom-telecom.com/administration/>, για τη δοκιμή του ενσωματωμένου συστήματος κατά την ανάπτυξη.

Τέλος, η πρόσβαση στα πραγματικά IoT δεδομένα των IoT συσκευών πραγματοποιείται από καταναεμημένες υπηρεσίες στη μεριά της IoT πλατφόρμας που τα προσφέρει (symbIoTe cloud services), καθώς και μηχανισμούς για τη διαχείριση των ομοσπονδιών της πλατφόρμας.

URL	
GitLab repository	<a href="https://github.com/iotfedsproject/">https://github.com/iotfedsproject/</a>
Container registry	<a href="https://hub.docker.com/repository/docker/iotfedsproject/">https://hub.docker.com/repository/docker/iotfedsproject/</a>
Deployment yaml file	<a href="https://github.com/iotfedsproject/IoTFeds-K8s-Deployment">https://github.com/iotfedsproject/IoTFeds-K8s-Deployment</a> , <a href="https://github.com/iotfedsproject/IoTFedsDockerSwarmDeployment">https://github.com/iotfedsproject/IoTFedsDockerSwarmDeployment</a>

##### 3.1.1 Στοιχείο λογισμικού για τη διαχείριση ομοσπονδιών, χρηστών και πλατφορμών (Administration)

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι δοκιμές τους για το στοιχείο λογισμικού Administration του symbIoTe για τη διαχείριση των IoT ομοσπονδιών, των χρηστών και των IoT πλατφορμών.

###### 3.1.1.1 Περιγραφή

Η διαχείριση των ομοσπονδιών, των χρηστών και των πλατφορμών τους στο οικοσύστημα του symbIoTe γίνεται μέσα από τη γραφική διεπαφή χρήστη (Administration GUI). Τα εκάστοτε αιτήματα του χρήστη αποστέλλονται στο στοιχείο λογισμικού Administration του symbIoTe μέσω των αντίστοιχων endpoints του στοιχείου αυτού από το οποίο και ενημερώνεται το υποσύστημα BaaS.

<sup>1</sup> <https://www.symbiote-h2020.eu/>



### 3.1.1.2 Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Τα παρακάτω χαρακτηριστικά προστέθηκαν στην ενδιάμεση έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- Διαχείριση χρηστών – για την διαχείριση των χρηστών στο σύστημα.
- Διαχείριση πλατφορμών – για την καταχώρηση των IoT πλατφορμών.
- Διαχείριση ομοσπονδιών – για τη διαχείριση των ομοσπονδιών, των μελών και των κανόνων τους.

### 3.1.1.3 Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Το Administration παρέχει τη γραφική διεπαφή χρήστη που είναι προσβάσιμη στο παρακάτω url: <symbIoTe-core>/administration. Το symbIoTe Administration backend είναι προσβάσιμο μέσω HTTPS GET/POST REST αιτήματα από το γραφικό περιβάλλον.

#### 3.1.1.3.1 Διεπαφές για τη διαχείριση IoT ομοσπονδίας

Στην ενότητα αυτή περιγράφονται οι διεπαφές του στοιχείου Administration του υποσυστήματος symbIoTe για τη διαχείριση των IoT ομοσπονδιών.

Πίνακας 1: Διεπαφές στοιχείου symbIoTe Administration για διαχείριση ομοσπονδίας.

Inter Face ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeAdminFed-1	REST	Request to the administration component to create a core user. Effectively like registering a user from the administration GUI	/administration/register	IoTFeds User (Data consumer)	Administration	v1.0
SymbIoTeAdminFed-2	REST	Request for an organization/platform to be added to a federation	/administration/federation_vote_request /joinFederation	IoTFeds User (Data consumer)	Administration	v1.0
SymbIoTeAdminFed-3	REST	Returns the list of the join requests for a specific user	/administration/federation_vote_request/all-ForUser	IoTFeds User (Data consumer)	Administration	v1.0
SymbIoTeAdminFed-4	REST	Request for an organization/platform to be deleted from a federation	<iotfeds-url>/administration/federation_vote_request/removeUserFromFed	IoTFeds User (Data consumer)	Administration	v1.0

SymbIoTeAdminFed-5	REST	Request to update the Rules/Smart Contract of a federation	/administration/federation_vote_request/updateFederationRules	IoTFeds User (Data consumer)	Administration	v1.0, v2.0
SymbIoTeAdminFed-6	REST	Receiving the result of the requests	/administration/federation_vote_request/all	Baas	Administration	v1.0
SymbIoTeAdminUser-1	REST	Invite a user to an existing federation	/administration/user/cpanel/federation_invite	IoTFeds User (Data producer)	Administration	v1.0, v2.0
SymbIoTeAdminUser-2	REST	Leave from a federation you are currently in	/administration/user/cpanel/leave_federation	IoTFeds User	Administration	v1.0, v2.0
SymbIoTeAdminUser-3	REST	Delete a federation	/administration/user/cpanel/delete_federation	IoTFeds User	Administration	v1.0, v2.0
SymbIoTeAdminUser-4	REST	Request for token	/administration/user/login	IoTFeds User	Administration	v1.0, v2.0
SymbIoTeAdminUser-5	REST	Register a platform to symbIoTe	/administration/user/cpanel/register_platform	IoTFeds User (Data Producer)	Administration	v1.0, v2.0
SymbIoTeAdminUser-6	REST	Register a new federation to symbiote	/administration/user/cpanel/create_federation	IoTFeds User	Administration	v1.0, v2.0
SymbIoTeAdminUser-7	REST	Return the list of federations registered in the system	/administration/user/cpanel/list_federations	IoTFeds User	Administration	v1.0, v2.0

Οι διεπαφές SymbIoTeAdminFed-5, SymbIoTeAdminUser-1, SymbIoTeAdminUser-2, SymbIoTeAdminUser-3, SymbIoTeAdminUser-4, SymbIoTeAdminUser-5, SymbIoTeAdminUser-6, SymbIoTeAdminUser-7 ενημερώθηκαν στην τελική έκδοση του λογισμικού όσον αφορά την επιστροφή της κλήσης τους για την ενσωμάτωση των τελικών κανόνων ομοσπονδίας που σχετίζονται με την τιμολόγηση και την αξιολόγηση.

#### 3.1.1.4 Δοκιμή και επαλήθευση (Test cases)

##### 3.1.1.4.1 Διεπαφές για τη διαχείριση IoT ομοσπονδίας

Στην ενότητα αυτή γίνεται η δοκιμή για τις διεπαφές του στοιχείου Administration του υποσυστήματος symbIoTe για τη διαχείριση των IoT ομοσπονδιών.

**Διεπαφή SymbIoTeAdminFed -1.** Η διεπαφή αυτή χρησιμοποιείται για το αίτημα δημιουργίας ενός χρήστη.

**SymbIoTeAdminFed -1.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "accountNonExpired": true,
  "accountNonLocked": true,
  "analyticsAndResearchConsent": true,
  "authorities": [
    {
      "authority": "string"
    }
  ],
  "conditionsAccepted": true,
  "credentialsNonExpired": true,
  "enabled": true,
  "password": "string",
  "recoveryMail": "string",
  "role": "USER",
  "termsAccepted": true,
  "username": "user1",
  "validPassword": "string",
  "validUsername": "string"
}
```

Η επιστροφή του μηνύματος είναι η αντίστοιχη σελίδα του Administration GUI σε περίπτωση επιτυχούς ή μη εγγραφής του χρήστη.

**Διεπαφή SymbIoTeAdminFed -2.** Η διεπαφή αυτή χρησιμοποιείται για το αίτημα ένταξης ενός οργανισμού σε μία ομοσπονδία.

**SymbIoTeAdminFed -2.A.** Οι ακόλουθοι παράμετροι θα πρέπει να οριστούν για την κλήση:

```
"federationId": "Smart City"
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "federationId": "Smart City",
  "handledDate": "2022-10-17T13:11:30.556Z",
  "requestDate": "2022-10-17T13:11:30.556Z",
  "requestingUser": "user1",
  "status": "PENDING",
  "username": "string",
  "voteAction": "ADD_MEMBER",
  "votingId": "string",
  "SmartContract": null
}
```

**Διεπαφή SymbIoTeAdminFed -3.** Η διεπαφή αυτή χρησιμοποιείται για το αίτημα της λίστας των αιτημάτων ενός χρήστη.

**SymbIoTeAdminFed -3.A.** Δεν απαιτούνται παράμετροι για την κλήση.

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
[{
  "federationId": "tstfed20231109a",
```

```

    "requestingUser": "icom1",
    "votingId": "sjbj4uy8rnhl9wh47ajar",
    "status": "PENDING",
    "voteAction": "Add Member",
    "requestDate": 1699537071604,
    "handledDate": null,
    "username": "icom2"
  }}

```

**Διεπαφή SymbIoTeAdminFed -4.** Η διεπαφή αυτή χρησιμοποιείται για το αίτημα διαγραφής ενός οργανισμού από μία ομοσπονδία.

**SymbIoTeAdminFed -4.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "federationId": "Smart City",
  "memberToAdd": "user1"
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```

{
  "federationId": "Smart City",
  "handledDate": "2022-10-17T13:11:30.556Z",
  "requestDate": "2022-10-17T13:11:30.556Z",
  "requestingUser": "user1",
  "status": "PENDING",
  "username": "string",
  "voteAction": "REMOVE_MEMBER",
  "votingId": "string",
  "SmartContract": null
}

```

**Διεπαφή SymbIoTeAdminFed -5.** Η διεπαφή αυτή χρησιμοποιείται για το αίτημα ενημέρωσης των κανόνων μίας ομοσπονδίας.

**SymbIoTeAdminFed -5.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "id": "fed_test",
  "name": "fed_test",
  "public": "true",
  "informationModel": {
    "id": "BIM",
    "uri": null,
    "name": null,
    "owner": null,
    "rdf": null,
    "rdfFormat": null
  },
  "members": [
    {
      "platformId": "icom-platform",
      "interworkingServiceURL": null
    }
  ],
  "SmartContract": {
    "IoTFedsRules": {

```

```

    "FedGov": {
      "BoardGov": [],
      "Proposals": [
        "ChangeRule",
        "DeleteFed",
        "InviteMember",
        "JoinRequest",
        "RequestRemove"
      ],
      "VoteRules": {
        "Tokens": 50,
        "Type": {
          "ApprovalPercentage": 75,
          "Base": "Board"
        }
      }
    },
    "FedMarketplace": {
      "ChargePolicy": "PerProduct",
      "FedMarketCharge": 5,
      "GlobalMarketCharge": 12,
      "Coin": "FedCoin",
      "FedProduct": "Packaging",
      "ProfitPolicy": "PerSource"
    },
    "FedTypeRules": {
      "DataAvailability": "Closed",
      "ServiceType": "Energy, Health, Environment",
      "SupportedOntologies": "Smart Environment, Smart City",
      "Type": "Providers"
    },
    "QualityAssuranceMetrics": {
      "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
      },
      "QoSPercentage": 60,
      "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
      },
      "Quality": {
        "MinValueFed": 6
      },
      "ReputationPercentage": 40,
      "UnderPerformance": "RequestRemove"
    }
  }
}

```

```
}
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "message", "Your change federation rules request is forwarded to voting procedure"
}
```

**Διεπαφή SymbIoTeAdminFed-6.** Η διεπαφή αυτή χρησιμοποιείται για τη λήψη της απάντησης ενός αιτήματος προς ψήφιση.

**SymbIoTeAdminFed-6.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAdminFed-6. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{ "votingId": "notAvalidVoteId", "status": "ACCEPTED" }
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "message": "Vote successfully registered"
}
```

**Διεπαφή SymbIoTeAdminUser-1.** Η διεπαφή αυτή χρησιμοποιείται για την πρόσκληση ενός χρήστη σε μία υπάρχουσα ομοσπονδία.

**SymbIoTeAdminUser -1.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "federationId": "fed1",
  "memberToAdd": "user1"
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "fed1": {
    "id": " fed1",
    "name": " fed1",
    "public": "true",
    "informationModel": {
      "id": "BIM",
      "uri": null,
      "name": null,
      "owner": null,
      "rdf": null,
      "rdfFormat": null
    },
    "members": [
      {
        "platformId": "icom-platform",
        "interworkingServiceURL": null
      }
    ],
    "SmartContract": {
      "IoTFedsRules": {
        "FedGov": {
          "BoardGov": [],
          "Proposals": [
            "ChangeRule",

```

```

        "DeleteFed",
        "InviteMember",
        "JoinRequest",
        "RequestRemove"
    ],
    "VoteRules": {
        "Tokens": 50,
        "Type": {
            "ApprovalPercentage": 75,
            "Base": "Board"
        }
    }
},
"FedMarketplace": {
    "ChargePolicy": "PerProduct",
    "FedMarketCharge": 5,
    "GlobalMarketCharge": 12,
    "Coin": "FedCoin",
    "FedProduct": "Packaging",
    "ProfitPolicy": "PerSource"
},
"FedTypeRules": {
    "DataAvailability": "Closed",
    "ServiceType": "Energy, Health, Environment",
    "SupportedOntologies": "Smart Environment, Smart City",
    "Type": "Providers"
},
"QualityAssuranceMetrics": {
    "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
    },
    "QoSPercentage": 60,
    "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
    },
    "Quality": {
        "MinValueFed": 6
    },
    "ReputationPercentage": 40,
    "UnderPerformance": "RequestRemove"
}
}
}
}
}

```

**Διεπαφή SymbIoTeAdminUser-2.** Η διεπαφή αυτή χρησιμοποιείται για την αποχώρηση ενός οργανισμού από μια ομοσπονδία.

**SymbIoTeAdminUser -2.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "federationId": "fed1",
  "organization": "user1"
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "fed1": {
    "id": " fed1",
    "name": " fed1",
    "public": "true",
    "informationModel": {
      "id": "BIM",
      "uri": null,
      "name": null,
      "owner": null,
      "rdf": null,
      "rdfFormat": null
    },
    "members": [
      {
        "platformId": "icom-platform",
        "interworkingServiceURL": null
      }
    ],
    "SmartContract": {
      "IoTFedsRules": {
        "FedGov": {
          "BoardGov": [],
          "Proposals": [
            "ChangeRule",
            "DeleteFed",
            "InviteMember",
            "JoinRequest",
            "RequestRemove"
          ],
          "VoteRules": {
            "Tokens": 50,
            "Type": {
              "ApprovalPercentage": 75,
              "Base": "Board"
            }
          }
        }
      },
      "FedMarketplace": {
        "ChargePolicy": "PerProduct",
        "FedMarketCharge": 5,
        "GlobalMarketCharge": 12,
        "Coin": "FedCoin",
        "FedProduct": "Packaging",
        "ProfitPolicy": "PerSource"
      }
    }
  }
}
```



```

    },
    "FedTypeRules": {
      "DataAvailability": "Closed",
      "ServiceType": "Energy, Health, Environment",
      "SupportedOntologies": "Smart Environment, Smart City",
      "Type": "Providers"
    },
    "QualityAssuranceMetrics": {
      "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
      },
      "QoSPercentage": 60,
      "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
      },
      "Quality": {
        "MinValueFed": 6
      },
      "ReputationPercentage": 40,
      "UnderPerformance": "RequestRemove"
    }
  }
}
}
}

```

**Διεπαφή SymbIoTeAdminUser-3.** Η διεπαφή αυτή χρησιμοποιείται για τη διαγραφή μίας ομοσπονδίας.

**SymbIoTeAdminUser -3.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "federationIdToDelete": "fed1",
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```

{
  "fed1": {
    "id": " fed1",
    "name": " fed1",
    "public": "true",
    "informationModel": {
      "id": "BIM",
      "uri": null,
      "name": null,
      "owner": null,
    }
  }
}

```

```

    "rdf": null,
    "rdfFormat": null
  },
  "members": [
    {
      "platformId": "icom-platform",
      "interworkingServiceURL": null
    }
  ],
  "SmartContract": {
    "IoTFedsRules": {
      "FedGov": {
        "BoardGov": [],
        "Proposals": [
          "ChangeRule",
          "DeleteFed",
          "InviteMember",
          "JoinRequest",
          "RequestRemove"
        ],
      },
      "VoteRules": {
        "Tokens": 50,
        "Type": {
          "ApprovalPercentage": 75,
          "Base": "Board"
        }
      }
    },
    "FedMarketplace": {
      "ChargePolicy": "PerProduct",
      "FedMarketCharge": 5,
      "GlobalMarketCharge": 12,
      "Coin": "FedCoin",
      "FedProduct": "Packaging",
      "ProfitPolicy": "PerSource"
    },
    "FedTypeRules": {
      "DataAvailability": "Closed",
      "ServiceType": "Energy, Health, Environment",
      "SupportedOntologies": "Smart Environment, Smart City",
      "Type": "Providers"
    },
    "QualityAssuranceMetrics": {
      "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
      },
      "QoSPercentage": 60,
      "QoSWeights": {

```

```

        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
    },
    "Quality": {
        "MinValueFed": 6
    },
    "ReputationPercentage": 40,
    "UnderPerformance": "RequestRemove"
}
}
}
}
}

```

**Διεπαφή SymbIoTeAdminUser-4.** Η διεπαφή αυτή χρησιμοποιείται για την είσοδο ενός χρήστη στο γραφικό περιβάλλον.

**SymbIoTeAdminUser -4.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{username, password }
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{Redirect to main page}
```

**Διεπαφή SymbIoTeAdminUser-5.** Η διεπαφή αυτή χρησιμοποιείται για την εγγραφή μίας πλατφόρμας.

**SymbIoTeAdminUser -5.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{ PlatformDetails platformDetails }
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{ }
```

**Διεπαφή SymbIoTeAdminUser-6.** Η διεπαφή αυτή χρησιμοποιείται για την εγγραφή μιας ομοσπονδίας.

**SymbIoTeAdminUser -6.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "id": "fed_test",
  "name": "fed_test",
  "public": "true",
  "informationModel": {
    "id": "BIM",
    "uri": null,
    "name": null,
    "owner": null,
    "rdf": null,
    "rdfFormat": null
  },
  "members": [
    {
      "platformId": "icom-platform",
      "interworkingServiceURL": null
    }
  ]
},

```

```

"SmartContract": {
  "IoTFedsRules": {
    "FedGov": {
      "BoardGov": [],
      "Proposals": [
        "ChangeRule",
        "DeleteFed",
        "InviteMember",
        "JoinRequest",
        "RequestRemove"
      ],
      "VoteRules": {
        "Tokens": 50,
        "Type": {
          "ApprovalPercentage": 75,
          "Base": "Board"
        }
      }
    },
    "FedMarketplace": {
      "ChargePolicy": "PerProduct",
      "FedMarketCharge": 5,
      "GlobalMarketCharge": 12,
      "Coin": "FedCoin",
      "FedProduct": "Packaging",
      "ProfitPolicy": "PerSource"
    },
    "FedTypeRules": {
      "DataAvailability": "Closed",
      "ServiceType": "Energy, Health, Environment",
      "SupportedOntologies": "Smart Environment, Smart City",
      "Type": "Providers"
    },
    "QualityAssuranceMetrics": {
      "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
      },
      "QoSPercentage": 60,
      "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
      },
      "Quality": {
        "MinValueFed": 6
      },
      "ReputationPercentage": 40,
      "UnderPerformance": "RequestRemove"
    }
  }
}

```

```

    }
  }
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{ "message" : "Federation Registration was successful!" }
```

**Διεπαφή SymbIoTeAdminUser-7.** Η διεπαφή αυτή χρησιμοποιείται για την απόκτηση των εγγεγραμμένων ομοσπονδιών στο σύστημα.

**SymbIoTeAdminUser -7.A.** Δεν απαιτούνται παράμετροι για την κλήση.

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
[{
  "id": "fed_test",
  "name": "fed_test",
  "public": "true",
  "informationModel": {
    "id": "BIM",
    "uri": null,
    "name": null,
    "owner": null,
    "rdf": null,
    "rdfFormat": null
  },
  "members": [
    {
      "platformId": "icom-platform",
      "interworkingServiceURL": null
    }
  ],
  "SmartContract": {
    "IoTFedsRules": {
      "FedGov": {
        "BoardGov": [],
        "Proposals": [
          "ChangeRule",
          "DeleteFed",
          "InviteMember",
          "JoinRequest",
          "RequestRemove"
        ],
        "VoteRules": {
          "Tokens": 50,
          "Type": {
            "ApprovalPercentage": 75,
            "Base": "Board"
          }
        }
      }
    },
    "FedMarketplace": {
      "ChargePolicy": "PerProduct",
      "FedMarketCharge": 5,

```

```
"GlobalMarketCharge":12,
"Coin": "FedCoin",
"FedProduct": "Packaging",
"ProfitPolicy": "PerSource"
},
"FedTypeRules": {
    "DataAvailability": "Closed",
    "ServiceType": "Energy, Health, Environment",
    "SupportedOntologies": "Smart Environment, Smart City",
    "Type": "Providers"
},
"QualityAssuranceMetrics": {
    "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
    },
    "QoSPercentage": 60,
    "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
    },
    "Quality": {
        "MinValueFed": 6
    },
    "ReputationPercentage": 40,
    "UnderPerformance": "RequestRemove"
}
}
```

Στον παρακάτω πίνακα συνοψίζονται οι δοκιμές και το αποτέλεσμα τους (Επιτυχές ή μη επιτυχές).

Πίνακας 2: Περίπτωσης δοκιμής στοιχείου *symbolTe's Administration* για διαχείριση ομοσπονδίας.

Interface	Test case	Method	Call	Status
SymbloTeAdminFed-1	SymbloTeAdminFed-1.A	POST	<iotfeds-url>/administration/register	PASS
SymbloTeAdminFed-2	SymbloTeAdminFed-2.A	POST	<iotfeds-url>/administration/federation_vote_request/joinFederation	PASS
SymbloTeAdminFed-3	SymbloTeAdminFed-3.A	GET	<iotfeds-url>/administration/federation_vote_request/allForUser	PASS

SymbloTeAdminFed-4	SymbloTeAdminFed-4.A	DELETE	<iotfeds-url>/administration/federation_vote_request/removeUserFromFed	PASS
SymbloTeAdminFed-5	SymbloTeAdminFed-5.A	POST	<iotfeds-url>/administration/federation_vote_request/updateFederationRules	PASS
SymbloTeAdminFed-6	SymbloTeAdminFed-6.A	POST	<iotfeds-url>/administration/federation_vote_request/result	
SymbloTeAdminUser-1	SymbloTeAdminUser-1.A	POST	<iotfeds-url>/administration/user/cpanel/federation_invite	PASS
SymbloTeAdminUser-2	SymbloTeAdminUser-2.A	POST	<iotfeds-url>/administration/user/cpanel/leave_federation	PASS
SymbloTeAdminUser-3	SymbloTeAdminUser-3.A	POST	<iotfeds-url>/administration/user/cpanel/delete_federation	PASS
SymbloTeAdminUser-4	SymbloTeAdminUser-4.A	POST	<iotfeds-url>/administration/user/login	PASS
SymbloTeAdminUser-5	SymbloTeAdminUser-5.A	POST	<iotfeds-url>/administration/user/cpanel/register_platform	PASS
SymbloTeAdminUser-6	SymbloTeAdminUser-6.A	POST	<iotfeds-url>/administration/user/cpanel/create_federation	PASS
SymbloTeAdminUser-7	SymbloTeAdminUser-7.A	GET	<iotfeds-url>/administration/user/cpanel/list_federations	PASS

### 3.1.2 Στοιχείο λογισμικού για τη διαχείριση IoT πόρων (symbloTe API)

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι δοκιμές του στοιχείου λογισμικού symbloTeAPI για τη διαχείριση IoT πόρων.

#### 3.1.2.1 Περιγραφή

Το στοιχείο λογισμικού symbloTe API παρέχει τη δυνατότητα στους χρήστες να αλληλεπιδρούν με το IoTFeds σύστημα μέσω RESTful API για την εγγραφή, τροποποίηση/διαγραφή και διαμοιρασμό των IoT πόρων των πλατφορμών καθώς και την πρόσβαση στα δεδομένα τους.

### 3.1.2.2 Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Τα παρακάτω χαρακτηριστικά προστέθηκαν στην ενδιάμεση έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- Εγγραφή και τροποποίηση IoT πόρων – καταγράφει έναν (καθολικό ή ομόσπονδο) IoT πόρο στην IoTFeds πλατφόρμα.
- Διαγραφή IoT πόρων – διαγράφει έναν IoT πόρο από μία ομοσπονδία ή την IoTFeds πλατφόρμα.
- Πρόσβαση στα IoT δεδομένα – επιστρέφει τα δεδομένα ενός IoT πόρου (καθολικού ή ομόσπονδου).
- Αναζήτηση IoT πόρων – αναζητάει τους IoT πόρους με βάση τα κριτήρια αναζήτησης.

### 3.1.2.3 Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Το symbIoTeAPI είναι προσβάσιμο μέσω Rest Get/Post HTTP αιτήματα στο παρακάτω uri:

symbIoTeAPI-url: <symbIoTe-core>/symbioteapi.

Πίνακας 3: Διεπαφές στοιχείου Administration για τη διαχείριση IoT πόρων.

Inter Face ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeAPI-1	REST	Register a resource to symbiote	/symbioteapi/resources/add /11	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-2	REST	Register a resource to symbiote	/symbioteapi/resources/add /12	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-3	REST	Share a resource to a federation	/symbioteapi/resources/shareResource	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-4	REST	Unshare (remove) a resource from a federation	/symbioteapi/resources/unshareResource	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-5	REST	Delete L1 resource	/symbioteapi/resources/deleteL1	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-6	REST	Delete L2 resource	/symbioteapi/resources/deleteL2	IoTFeds User (Data Producer)	symbIoTe API	v1.0
SymbIoTeAPI-7	REST	Get resource id from the internal id of a resource	/symbioteapi/resources/get ResourceId-FromInternalID	IoTFeds User	symbIoTe API	v1.0



SymbIoT eAPI-8	REST	Access re- source	/sympi- oteapi/re- sources/ac- cess/11/Id/	IoTFeds User (Data Consumer)	symbIoTe API	v1.0
-------------------	------	----------------------	---	------------------------------------	-----------------	------

### 3.1.2.4 Δοκιμή και επαλήθευση (Test cases)

Στην ενότητα αυτή παρουσιάζονται οι δοκιμές για τις διεπαφές του στοιχείου λογισμικού symbIoTeAPI για τη διαχείριση IoT πόρων.

**Διεπαφή symbIoTeAPI-1.** Η διεπαφή αυτή χρησιμοποιείται για την εγγραφή ενός L1 πόρου.

**symbIoTeAPI -1.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "credentials": {
    "platformId": "test_platform",
    "username": "test_user",
    "password": "password",
    "clientId": "client"
  },
  "cloudResource": {
    "internalId": "vds",
    "pluginId": "vds",
    "accessPolicy": "CAP",
    "filteringPolicy": "CAP",
    "resource": {
      "id": "resourceId",
      "name": "resourceName",
      "description": [
        "This is",
        "a description"
      ]
    },
    "interworkingServiceURL": "thisisa.url"
  }
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "key": [
    {
      "credentials": {
        "platformId": "test_platform",
        "username": "test_user",
        "password": "password",
        "clientId": "client"
      },
      "cloudResource": {
        "internalId": "cloud1",
        "pluginId": "plugin1",
        "accessPolicy": "CAP",
        "filteringPolicy": "CAP",
        "resource": {
          "id": "resourceId",
          "name": "resourceName",
          "description": [
```

```

        "This is",
        "a description"
    ],
    "interworkingServiceURL": "thisisa.url"
  }
}
]
}

```

**Διεπαφή symbIoTeAPI-2.** Η διεπαφή αυτή χρησιμοποιείται για την εγγραφή ενός L2 πόρου.

**symbIoTeAPI -2.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "credentials": {
    "platformId": "test_platform",
    "username": "test_user",
    "password": "password",
    "clientId": "client"
  },
  "cloudResource": {
    "internalId": "vsd",
    "pluginId": "vds",
    "accessPolicy": "CAP",
    "filteringPolicy": "CAP",
    "resource": {
      "id": "resourceId",
      "name": "resourceName",
      "description": [
        "This is",
        "a description"
      ],
    },
    "interworkingServiceURL": "thisisa.url"
  }
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```

[
  {
    "cloudResource": {
      "internalId": " cloud1",
      "pluginId": " plugin1",
      "accessPolicy": "CAP",
      "filteringPolicy": "CAP",
      "resource": {
        "id": "resourceId",
        "name": "resourceName",
        "description": [
          "This is",
          "a description"
        ],
      },
      "interworkingServiceURL": "thisisa.url"
    }
  }
]

```

]

**Διεπαφή symbIoTeAPI-3.** Η διεπαφή αυτή χρησιμοποιείται για κοινοποίηση ενός πόρου σε μία ομοσπονδία.

**symbIoTeAPI -3.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "credentials": {
    "platformId": "test_platform",
    "username": "test_user",
    "password": "password",
    "clientId": "client"
  },
  "resourceInternalId": "resourceId",
  "bartered": true,
  "federationId": "fed1"
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
[
  {
    "cloudResource": {
      "internalId": "cloud1",
      "pluginId": "plugin1",
      "accessPolicy": "CAP",
      "filteringPolicy": "CAP",
      "resource": {
        "id": "resourceId",
        "name": "resourceName",
        "description": [
          "This is",
          "a description"
        ],
        "interworkingServiceURL": "thisisa.url"
      }
    }
  }
]
```

**Διεπαφή symbIoTeAPI-4.** Η διεπαφή αυτή χρησιμοποιείται για την άρση κοινοποίησης ενός πόρου σε ομοσπονδία.

**symbIoTeAPI -4.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "credentials": {
    "platformId": "test_platform",
    "username": "test_user",
    "password": "password",
    "clientId": "client"
  },
  "resourceInternalId": "resourceId",
  "bartered": false,
  "federationId": "fed1"
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

[

```

{
  "cloudResource": {
    "internalId": " cloud1",
    "pluginId": " plugin1",
    "accessPolicy": "CAP",
    "filteringPolicy": "CAP",
    "resource": {
      "id": "resourceId",
      "name": "resourceName",
      "description": [
        "This is",
        "a description"
      ],
      "interworkingServiceURL": "thisisa.url"
    }
  }
}
]

```

**Διεπαφή symbIoTeAPI-5.** Η διεπαφή αυτή χρησιμοποιείται για τη διαγραφή ενός L1 πόρου.

**symbIoTeAPI -5.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-5. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "resourceInternalId": "testRes1",
  "credentials": {
    "platformId": "test_platform",
    "username": "test_user",
    "password": "password",
    "clientId": "client"
  }
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:



```

{
  "cloudResources": [
    {
      "internalId": "testRes1",
      "pluginId": "RapPluginExample",
      "accessPolicy": {
        "policyType": "SLHTIBAP",
        "requiredClaims": {
          "sub": "marketplace",
          "iss": "SymbIoTe_Core_AAM"
        }
      },
      "filteringPolicy": {
        "policyType": "PUBLIC",
        "requiredClaims": {}
      },
      "resource": {
        "@c": ".StationarySensor",
        "observesProperty": [

```

Εικόνα 2:Αποτέλεσμα διαγραφής L1 πόρου από το symbIoTe.

**Διεπαφή symbIoTeAPI-6.** Η διεπαφή αυτή χρησιμοποιείται για τη διαγραφή ενός L2 πόρου.

**symbIoTeAPI -6.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "resourceInternalId": "resourceId",

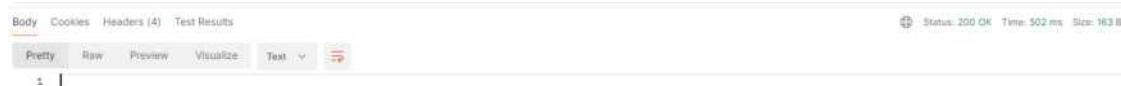
```

```

"credentials": {
  "platformId": "test_platform",
  "username": "test_user",
  "password": "password",
  "clientId": "client"
}
}

```

Το επιθυμητό αποτέλεσμα είναι ο κωδικός επιτυχίας (HTTPSTATUS=200) όπως φαίνεται παρακάτω. Το κείμενο της επιστροφής του συγκεκριμένου αιτήματος είναι κενό.



Εικόνα 3: Διαγραφή L2 πόρου από το symbIoTe.

**Διεπαφή symbIoTeAPI-7.** Η διεπαφή αυτή χρησιμοποιείται για την απόκτηση του id ενός πόρου χρησιμοποιώντας το internal id.

**symbIoTeAPI -7.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "internalId": "657872154211470001832b7b",
  "platformId": "paltformA"
}

```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```

{"resourceId"}

```

**Διεπαφή symbIoTeAPI-8.** Η διεπαφή αυτή χρησιμοποιείται για την πρόσβαση σε έναν πόρο.

**symbIoTeAPI -8.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```

{
  "platformCredentials": {
    "clientId": "Test_Client",
    "password": "password",
    "platformId": "icom-platform",
    "username": "username"
  },
  "remotePlatformId": "icom-platform"
}

```

Ακόμη προσθέτουμε ως μεταβλητή στο url το resourceId π.χ 62c591280f160e00013b7154

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```

[
  {
    "resourceId": "65bcffcca6d0200001ab2d8b",
    "location": {
      "@c": ".WGS84Location",
      "longitude": 48.2088475,
      "latitude": 16.3734492,
      "altitude": 158.0,
      "name": "Stephansdome",
      "description": [
        "City of Wien"
      ]
    }
  }
]

```

```
"resultTime": "2024-02-15T14:53:11",
"samplingTime": "2024-02-15T14:53:10",
"obsValues": [
  {
    "value": "39",
    "obsProperty": {
      "name": "Temperature",
      "iri": "TempIRI",
      "description": [
        "Air temperature"
      ]
    },
    "uom": {
      "symbol": "C",
      "name": "degree Celsius",
      "iri": "C_IRI",
      "description": null
    },
    "featureOfInterest": null
  }
]
}
```

Στον παρακάτω πίνακα συνοψίζονται οι περιπτώσεις δοκιμών και το αποτέλεσμα τους (επιτυχές ή μη) για το στοιχείο λογισμικού symbIoTeAPI.

Πίνακας 4:Περίπτωσης δοκιμής στοιχείου symbIoTe-API.

Interface	Test case	Method	Call	Status
SymbIoTeAPI-1	SymbIoTeAPI-1.A	POST	<iotfeds-url>/symbioteapi/resources/add/11	PASS
SymbIoTeAPI-2	SymbIoTeAPI-2.A	POST	<iotfeds-url>/symbioteapi/resources/add/12	PASS
SymbIoTeAPI-3	SymbIoTeAPI-3.A	POST	<iotfeds-url>/symbioteapi/resources/shareResource	PASS
SymbIoTeAPI-4	SymbIoTeAPI-4.A	POST	<iotfeds-url>/symbioteapi/resources/unshareResource	PASS
SymbIoTeAPI-5	SymbIoTeAPI-5.A	DELETE	<iotfeds-url>/symbioteapi/resources/deleteL1	PASS
SymbIoTeAPI-6	SymbIoTeAPI-6.A	DELETE	<iotfeds-url>/symbioteapi/resources/deleteL2	PASS
SymbIoTeAPI-7	SymbIoTeAPI-7.A	POST	/symbioteapi/resources/getResourceByIdFromInternalID	PASS

SymbIoTeA PI-8	SymbIoTeA PI-8.A	POST	/symbioteapi/re- sources/ac- cess/11/Id/	PASS
-------------------	---------------------	------	--	------

## 3.2 Στοιχείο λογισμικού Αγοράς

### 3.2.1 Στοιχείο λογισμικού για τη διαχείριση IoT προϊόντων και πόρων (symbIoTe Marketplace)

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι δοκιμές του στοιχείου λογισμικού symbIoTe Marketplace για τη διαχείριση IoT πόρων και προϊόντων.

#### 3.2.1.1 Περιγραφή

Το στοιχείο λογισμικού symbIoTe Marketplace παρέχει τη δυνατότητα στους χρήστες να αλληλεπιδρούν με το σύστημα IoTFeds μέσω UI και RESTful API για την εγγραφή, τροποποίηση/διαγραφή και διαμοιρασμό των IoT πόρων των πλατφορμών, την πρόσβαση στα δεδομένα τους καθώς και τη διαχείριση των προϊόντων.

#### 3.2.1.2 Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Τα παρακάτω χαρακτηριστικά συμπεριλαμβάνονται στην τελική έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- Εγγραφή και τροποποίηση IoT πόρων από έναν IoT πάροχο στην IoTFeds πλατφόρμα.
- Διαμοιρασμός ενός καταχωρημένου IoT πόρου από τον IoT πάροχο σε μία ομοσπονδία.
- Διαγραφή IoT πόρων από μία ομοσπονδία ή την IoTFeds πλατφόρμα από τον IoT πάροχο.
- Πρόσβαση στα IoT δεδομένα – επιστρέφει τα δεδομένα ενός IoT πόρου στο marketplace.
- Αναζήτηση IoT πόρων με βάση τα επιθυμητά κριτήρια αναζήτησης στο marketplace.
- Σύνθεση προϊόντος στην ομόσπονδη ή και την καθολική αγορά.
- Διαγραφή προϊόντος από την αγορά.
- Αναζήτηση προϊόντων από έναν IoTFeds χρήστη (IoT πάροχο ή καταναλωτή) με βάση τα επιθυμητά κριτήρια αναζήτησης.
- Αγορά ενός προϊόντος από έναν καταναλωτή IoT δεδομένων και ενημέρωση χρέωσης.
- Υποβολή αξιολόγησης ενός προϊόντος από έναν IoT καταναλωτή IoT δεδομένων.
- Προσθήκη κουπονιού προϊόντος προς ανταλλαγή από έναν IoT χρήστη.
- Διαχείριση πρότασης ανταλλαγής από έναν IoT χρήστη.
- Ανταλλαγή κουπονιών για πρόσβαση σε IoT δεδομένα μεταξύ δύο IoTFeds χρηστών.
- Προσπέλαση στα δεδομένα ενός προϊόντος από έναν IoT καταναλωτή IoT δεδομένων.
- Πρόσβαση στο πορτοφόλι του IoTFeds χρήστη (Fedcoins, tokens etc)
- Αναζήτηση και έλεγχο συναλλαγών του IoTFeds χρήστη.
- Αναζήτηση και έλεγχο των αρχείων καταγραφής πρόσβασης του IoTFeds χρήστη.

#### 3.2.1.3 Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Αρχικά περιγράφονται οι διεπαφές του στοιχείου λογισμικού αγοράς σχετικά με τη διαχείριση των προϊόντων δεδομένων και η επαλήθευσή τους.

Inter Face ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeMarketplace-1	REST	Buy product	/SymbIoTeMarketplace/product/buy	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-2	REST	Rate product	/SymbIoTeMarketplace/product/rate	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-3	REST	Subscribe to access streaming data	/SymbIoTeMarketplace/product/subscribe	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-4	REST	Get information of global products.	/SymbIoTeMarketplace/product/global	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-5	REST	Get information of products in a federation.	/SymbIoTeMarketplace/product/federation	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-6	REST	Search for products that match specific criteria.	/SymbIoTeMarketplace/product/search	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-7	REST	Access a product	/SymbIoTeMarketplace/product/access	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-8	REST	Create product	/SymbIoTeMarketplace/product/	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-9	REST	Calculate product price	/SymbIoTeMarketplace/product/get-price	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-10	REST	Delete product by ID	/SymbIoTeMarketplace/product/{id}	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0

**Διεπαφή symbIoTeMarketplace-1.** Η διεπαφή αυτή χρησιμοποιείται για την αγορά ενός προϊόντος από το marketplace.

**symbIoTeMarketplace-1.A.** Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

Οι παρακάτω παράμετροι πρέπει να οριστούν κατά την κλήση όπως φαίνεται στο παράδειγμα αγοράς ενός προϊόντος στην Εικόνα 5. Στο παράδειγμα αυτό ο «icom2» αγοράζει από τον «icom1» το προϊόν με αναγνωριστικό «65a791431adf7e68a78e728d». Το προϊόν έχει τιμή 8,6 FedCoins και διατίθεται στην αγορά με αναγνωριστικό «fedstktak». Ο «icom2» θέλει να το



αγοράσει για 10 προσπελάσεις μέσα στο χρονικό διάστημα που καθορίζεται από τα πεδία lp και hp, ενώ επιθυμεί την υπηρεσία streaming και με συχνότητα χρήσης που να είναι 2.

```
{
  "product_id": "65a791431adf7e68a78e728d",
  "buyer": "icom2",
  "seller": "icom1",
  "price": 8.6,
  "marketplace": "fedsttktsak",
  "streaming": false,
  "access": 10,
  "lp": "2023-08-29T17:30:50.100Z",
  "hp": "2024-01-30T11:30:50.100Z",
  "fp": "2"
}
```

Εικόνα 4: Παράμετροι για την αγορά ενός προϊόντος.

Το αποτέλεσμα μια επιτυχούς αγοράς είναι ένα αντικείμενο με πληροφορίες για την αγορά: το αναγνωριστικό του αποδεικτικού της αγοράς, το αναγνωριστικό του αγοραστή του προϊόντος, το αναγνωριστικό του πωλητή του προϊόντος, το αναγνωριστικό του προϊόντος και η ημ/νία αγοράς, όπως φαίνεται στο παρακάτω παράδειγμα. Ο χρήστης ενημερώνεται με ένα μήνυμα λάθους, εάν η αγορά απέτυχε.

```
{
  "TransactionID": "6e8ed285c4da561e49fdf0c5ff9c1436c1d9d3f9cad58f6bb11e3a64f9766f08",
  "Buyer": "icom2",
  "Seller": "icom1",
  "ProductID": "65a791431adf7e68a78e728d",
  "DateBought": "Wed, 17 Jan 2024 11:23:05 GMT"
}
```

Εικόνα 5 Κλήση της υπηρεσίας του marketplace για την αγορά ενός προϊόντος.

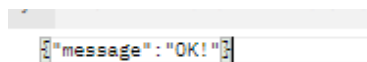
**Διεπαφή symbIoTeMarketplace-2.** Η διεπαφή αυτή χρησιμοποιείται από έναν χρήστη που επιθυμεί να βαθμολογήσει ένα προϊόν. Υπάρχει ο περιορισμός ότι ο αγοραστής ενός προϊόντος μπορεί να βαθμολογήσει το προϊόν του και μόνο αφού το έχει προσπελάσει τουλάχιστον μια φορά.

**symbIoTeMarketplace-2.A.** Το τεστ αυτό εκτελείται για τον έλεγχο της επιτυχούς υποβολής βαθμολογίας για ένα προϊόν προσδιορίζοντας στις παραμέτρους τον αγοραστή «icom2» του προϊόντος «65c64be15a685555270f131c»:

```
{
  "productId": "65c64be15a685555270f131c",
  "rating": {
    "businessEnablement": 1.0,
    "completeness": 1.0,
    "correctness": 1.0,
    "easeOfUse": 1.0,
    "precision": 1.0,
    "relevance": 1.0,
    "responseTime": 1.0,
    "valueForMoney": 1.0
  },
  "userId": "icom2"
}
```

Εικόνα 6: Κλήση της υπηρεσίας του marketplace για την βαθμολόγηση ενός προϊόντος.

Το επιθυμητό αποτέλεσμα πειραμβάνει το μήνυμα OK για την επιτυχή υποβολή της βαθμολογίας.



Εικόνα 7: Απάντηση επιτυχούς βαθμολόγησης ενός προϊόντος.

**Διεπαφή symbIoTeMarketplace-3.** Η διεπαφή χρησιμοποιείται για την ανάκτηση των πληροφοριών για την πρόσβαση των προϊόντων ροής (streaming).

**Διεπαφή symbIoTeMarketplace-3.A.** Για την εκτέλεση του τεστ είναι απαραίτητος ο καθορισμός του χρήστη στον οποίο ανήκει το προϊόν, το αναγνωριστικό του προϊόντος και η συχνότητα της ροής στο url του αιτήματος, όπως φαίνεται στο παρακάτω παράδειγμα:

```
subscribe?product_id=65bbaf21267b611e3d9ded9c&frequency=5&userid=icom2
```

Το επιθυμητό αποτέλεσμα περιλαμβάνει τις πληροφορίες που απαιτούνται για την πρόσβαση του προϊόντος ροής δεδομένων, όπως φαίνεται στην παρακάτω εικόνα:

```
[
  {
    "resourceId": "65a90863a6d0200001ab2d66",
    "platformId": "marios-cloud",
    "serviceUrl": "https://marios-cloud.iotfeds.intracom-telecom.com/rap/Sensors('65a90863a6d0200001ab2d66')"
  }
]
```

Εικόνα 8: Απάντηση ερωτήματος για πρόσβαση σε δεδομένα ροής.

**Διεπαφή symbIoTeMarketplace-4.** Η διεπαφή αυτή χρησιμοποιείται για να ανατρέξει ένας χρήστης στα προϊόντα που διατίθενται στην καθολική αγορά.

**Διεπαφή symbIoTeMarketplace-4.A.** Το τεστ αυτό εκτελείται για την ανάκτηση των προϊόντων του χρήστη στην καθολική αγορά και απαιτείται στις παραμέτρους το αναγνωριστικό χρήστη όπως φαίνεται στην Εικόνα 9 (και το οποίο λαμβάνεται από την πληροφορία του χρήστη που είναι συνδεδεμένος στην καθολική αγορά).

```
/marketplace/product/global?user_id=icom1
```

Εικόνα 9: Κλήση της υπηρεσίας του marketplace για την ανάκτηση των προϊόντων της καθολικής αγοράς.

Το επιθυμητό αποτέλεσμα περιλαμβάνει μια λίστα από προϊόντα όπως φαίνεται παρακάτω για το παράδειγμα αυτό:

```
[
  {
    "fedMarketplaceId": "fedststksak",
    "globalMarketplace": true,
    "productDetails": "",
    "productId": "65a791431adf7e68a78e728d",
    "seller": "icom1",
    "vertical": "{Humidity}",
    "name": "productExample",
    "description": "{Humidity data}",
    "location": "{Europe/Athens}",
    "observationProperties": ""
  }
]
```

Εικόνα 10: Ανάκτηση λίστας των προϊόντων της καθολικής αγοράς.

**Διεπαφή symbIoTeMarketplace-5.** Η διεπαφή αυτή χρησιμοποιείται για να ανατρέξει ένας χρήστης στα προϊόντα που διατίθενται σε μια ομόσπονδη αγορά.

**Διεπαφή symbIoTeMarketplace-5.A.** Ομοίως, το τεστ που εκτελείται απαιτεί στις παραμέτρους (ή url) τον χρήστη «icom1» και την ομόσπονδη αγορά «fedststksak» όπως φαίνεται στην Εικόνα 11.

Query Params		
<input checked="" type="checkbox"/>	Key	Value
<input checked="" type="checkbox"/>	user_id	icom1
<input checked="" type="checkbox"/>	fed_id	fedsttksak

Εικόνα 11 Κλήση της υπηρεσίας του marketplace για ανάκτηση των προϊόντων της ομόσπονδης αγοράς «fedsttksak».

Η διεπαφή απαντά και πάλι με μια λίστα από προϊόντα.

```
[
  {
    "fedMarketplaceId": "fedsttksak",
    "globalMarketplace": true,
    "productDetails": "",
    "productId": "65a791431adf7e68a78e728d",
    "seller": "icom1",
    "vertical": "{Humidity}",
    "name": "productExample",
    "description": "{Humidity data}",
    "location": "{Europe/Athens}",
    "observationProperties": ""
  }
]
```

Εικόνα 12: Απάντηση marketplace για ανάκτηση των προϊόντων της ομόσπονδης αγοράς «fedsttksak».

**Διεπαφή symbIoTeMarketplace-6.** Η διεπαφή αυτή χρησιμοποιείται για την αναζήτηση προϊόντων σε μια αγορά, με βάση τα κριτήρια που ορίζονται.

**Διεπαφή symbIoTeMarketplace-6.A** Το τεστ που εκτελείται απαιτεί να οριστούν οι παράμετροι που φαίνονται στην Εικόνα 14 όπου ο χρήστης αναζητά προϊόντα που πουλάει ο «icom1» στην ομόσπονδη αγορά «fedsttksak» και κοστίζουν από 0 έως και 50 FedCoins. Εάν γνωρίζει ότι τον ενδιαφέρουν το προϊόντα με όνομα «productExample», ο χρήστης μπορεί να εισάγει το κριτήριο "name" και επίσης μπορεί να ζητήσει να του επιστραφεί η λίστα προϊόντων σε φθίνουσα σειρά με βάση το όνομα προϊόντος.

```
{
  "sellerId": [
    "icom1"
  ],
  "priceMin": 0,
  "priceMax": 50,
  "name": [
    "productExample"
  ],
  "fedMarketplaceId": "fedsttksak",
  "sort": "descending",
  "credentials": {
    "clientId": "Test-client",
    "password": "***",
    "platformId": "SymbIoTe_Core_AAM",
    "username": "marketplace"
  }
}
```

Εικόνα 13: Παράδειγμα κλήσης της υπηρεσίας του marketplace για αναζήτηση προϊόντων με βάση κάποια κριτήρια.

Η διεπαφή απαντά με τη λίστα από προϊόντα, όπως φαίνεται στην απάντηση του παραδείγματος:

```
[
  {
    "fedMarketplaceId": "fedsttksak",
    "globalMarketplace": true,
    "productDetails": "",
    "productId": "65a791431adf7e68a78e728d",
    "seller": "icom1",
    "vertical": "{Humidity}",
    "name": "productExample",
    "description": "{Humidity data}",
    "location": "{Europe/Athens}",
    "observationProperties": ""
  }
]
```

Εικόνα 14: Παράδειγμα κλήσης της υπηρεσίας του marketplace για αναζήτηση προϊόντων με βάση κάποια κριτήρια.

**Διεπαφή symbIoTeMarketplace-7.** Η διεπαφή αυτή χρησιμοποιείται για να αποκτήσει ένας χρήστης πρόσβαση σε ένα προϊόν και στο περιεχόμενο των πόρων του. Η Εικόνα 15 δείχνει ένα παράδειγμα με τις παραμέτρους που πρέπει να οριστούν. Ο χρήστης «icom1», ο οποίος έχει αγοράσει ένα προϊόν με το αναγνωριστικό στην παράμετρο productId θέλει να το προσπελάσει για το καθορισμένο διάστημα και για μία προσπέλαση.

```
1 {
2   "productId": "65577d9b5b4d0906af3ca73d",
3   "dateFrom": "2023-11-10T00:00:00.000Z",
4   "dateTo": "2023-11-12T00:00:00.000Z",
5   "reqObservations": "1",
6   "credentials": {
7     "clientId": "Test_Client",
8     "password": "***",
9     "platformId": "SymbIoTe_Core_AAM",
10    "username": "icom1"
11  }
12 }
```

Εικόνα 15: Παράδειγμα κλήσης της υπηρεσίας του marketplace για την προσπέλαση ενός προϊόντος.

Παράδειγμα της επιθυμητής απάντησης είναι το παρακάτω:

```
1 {
2   "timestamp": 1705402369045,
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "",
6   "path": "/product/access"
7 }
```

Εικόνα 16: Παράδειγμα απάντησης της υπηρεσίας του marketplace για την προσπέλαση ενός προϊόντος.

#### Διεπαφή symbIoTeMarketplace-8.

Η διεπαφή αυτή χρησιμοποιείται για την δημιουργία ενός προϊόντος. Η Εικόνα 17 δείχνει ένα παράδειγμα δημιουργίας ενός προϊόντος που διατίθεται σε μια ομόσπονδη αγορά, καθώς επίσης και στην καθολική αγορά. Ο πάροχος «icom1» δημιουργεί το προϊόν με όνομα «productExample» που περιέχει τον πόρο «65a63782a6d0200001ab2d60». Ο πόρος αυτός έχει διαμοιραστεί στην αγορά «fedstkttsak» και περιέχει μετρικές υγρασίας(vertical:Humidity). Ο «icom1» επιλέγει να διαθέσει το προϊόν και στην καθολική αγορά ("globalMarketplaceId": true).

```
1 {
2   "productId": null,
3   "sellerId": "icom1",
4   "fedMarketplaceId": "fedstkttsak",
5   "globalMarketplaceId": true,
6   "name": "productExample",
7   "description": ["Humidity data"],
8   "productDetails": {},
9   "resourceIds": ["65a63782a6d0200001ab2d60"],
10  "vertical": ["Humidity"]
11 }
```

Εικόνα 17: Κλήση της υπηρεσίας του marketplace για την δημιουργία προϊόντος.

```

{
  "id": "65a791431adf7e68a78e728d",
  "sellerId": "icom1",
  "fedMarketplaceId": "fedtstkttsak",
  "globalMarketplaceId": true,
  "name": "productExample",
  "description": [
    "Humidity data"
  ],
  "productDetails": {},
  "observedProperties": null,
  "observationParameters": [
    {
      "resourceId": "65a63782a6d0200001ab2d60",
      "platformId": "marios-cloud",
      "interWorkingServiceUrl": "https://marios-cloud.iotfeds.intracom-telecom.com/rsp/Sensors('65a63782a6d0200001ab2d60')",
      "location": {
        "@c": ".WGS84Location",
        "longitude": 23.874748,
        "latitude": 37.940128,
        "altitude": 0.0,
        "name": "Europe/Athens",
        "description": [
          "humidity"
        ]
      }
    }
  ]
}

```

Εικόνα 18 Απάντηση της υπηρεσίας του marketplace για την δημιουργία προϊόντος.

### Διεπαφή symbIoTeMarketplace-9.

Η διεπαφή αυτή χρησιμοποιείται για να ζητήσει ένας χρήστης την τιμή ενός προϊόντος πριν το αγοράσει. Η Εικόνα 19 δείχνει ένα παράδειγμα στο οποίο ο «icom2» ζητάει από την υπηρεσία του marketplace μια τιμή για το προϊόν με αναγνωριστικό «65a791431adf7e68a78e728d». Δηλώνει ότι το προϊόν διατίθεται στην αγορά «fedtstkttsak» και ότι ενδιαφέρεται για 10 προσπελάσεις μέσα στο χρονικό διάστημα που καθορίζεται από τα πεδία lp και hp, ενώ επιθυμεί την υπηρεσία streaming και η συχνότητα χρήσης να είναι 2.

```

{
  "product_id": "65a791431adf7e68a78e728d",
  "user_id": "icom2",
  "marketplace": "fedtstkttsak",
  "streaming": false,
  "access": 10,
  "lp": "2023-08-29T17:30:50.100Z",
  "hp": "2024-01-30T11:30:50.100Z",
  "fp": "2"
}

```

Εικόνα 19: Κλήση της υπηρεσίας του marketplace για την τιμή ενός προϊόντος.

Η διεπαφή απαντάει με την τιμή του προϊόντος σε FedCoins.

```

{
  "price": 8.6
}

```

Εικόνα 20: Απάντηση της υπηρεσίας του marketplace για την τιμή ενός προϊόντος.

**Διεπαφή symbIoTeMarketplace-10.** Η διεπαφή αυτή χρησιμοποιείται για να διαγραφεί ένα προϊόν από την υπηρεσία του marketplace.

**symbIoTeMarketplace-10.A.** Το τεστ αυτό εκτελείται για τη διαγραφή ενός προϊόντος και απαιτεί να οριστούν οι παράμετροι όπως φαίνεται στην Εικόνα 21.

```

{
  "clientId": "Test_Client",
  "password": "***",
  "platformId": "marios-cloud",
  "username": "icom1"
}

```

Εικόνα 21: Κλήση της υπηρεσίας του marketplace για την διαγραφή προϊόντος.

Το επιθυμητό αποτέλεσμα ενημερώνει το χρήστη σχετικά με την επιτυχή διαγραφή του προϊόντος όπως φαίνεται στην παρακάτω εικόνα:

```

Product 65a791431adf7e68a78e728d successfully deleted.

```

Εικόνα 22: Κλήση της υπηρεσίας του marketplace για την διαγραφή προϊόντος.

Στον παρακάτω πίνακα συνοψίζονται οι δοκιμές και το αποτέλεσμα τους (Επιτυχές ή μη επιτυχές).

Πίνακας 5: Δοκιμή διεπαφών για τη διαχείριση προϊόντος.

Interface	Test case	Method	Call	Status
SymbIoTeMarketplace-1	SymbIoTeMarketplace-1.A	POST	/SymbIoTeMarketplace/product/buy	PASS
SymbIoTeMarketplace-2	SymbIoTeMarketplace-2.A	POST	/SymbIoTeMarketplace/product/rate	PASS
SymbIoTeMarketplace-3	SymbIoTeMarketplace-3.A	POST	/SymbIoTeMarketplace/product/subscribe	PASS
SymbIoTeMarketplace-4	SymbIoTeMarketplace-4.A	GET	/SymbIoTeMarketplace/product/global	PASS
SymbIoTeMarketplace-5	SymbIoTeMarketplace-5.A	GET	/SymbIoTeMarketplace/product/federation	PASS
SymbIoTeMarketplace-6	SymbIoTeMarketplace-6.A	POST	/SymbIoTeMarketplace/product/search	PASS
SymbIoTeMarketplace-7	SymbIoTeMarketplace-7.A	POST	/SymbIoTeMarketplace/product/access	PASS

SymbIoTeMarketplace-8	SymbIoTeMarketplace-8.A	POST	/SymbIoTeMarketplace/product/	PASS
SymbIoTeMarketplace-9	SymbIoTeMarketplace-9.A	POST	/SymbIoTeMarketplace/product/get-price	PASS
SymbIoTeMarketplace-10	SymbIoTeMarketplace-10.A	DELETE	/SymbIoTeMarketplace/product/{id}	PASS

Στη συνέχεια ακολουθούν οι διεπαφές σχετικά με τη διαχείριση των IoT πόρων (πηγών δεδομένων) και η επαλήθευσή τους.

Inter Face ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeMarketplace-11	REST	End-point to search for L1 resources that match specific criteria.	/SymbIoTeMarketplace/resource/search/11	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-12	REST	End-point to search for L2 resources that match specific criteria.	/SymbIoTeMarketplace/resource/search/L2	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-13	REST	End-point to access an L1 resource with a resource id.	/SymbIoTeMarketplace/resource/access/11/Id/{resourceId}	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-14	REST	End-point to access an L1 resource with an internal resource id.	/SymbIoTeMarketplace/resource/access/11/Id/{resourceId}	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-15	REST	End-point to access an L2 resource with an internal resource id.	/SymbIoTeMarketplace/resource/access/12/{resourceInternalId}	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0

SymbIoTeMarketplace-16	REST	End-point to register a new L1 resource.	/SymbIoTeMarketplace/resource/add/1	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-17	REST	End-point to share a resource to a federation.	/SymbIoTeMarketplace/share	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0

#### Διεπαφή symbIoTeMarketplace-11.

Η διεπαφή αυτή χρησιμοποιείται για να αναζητήσει ένας χρήστης IoT πόρους καταχωρημένους στο σύστημα.

**symbIoTeMarketplace-11.A.** Το τεστ αυτό εκτελείται για την αναζήτηση των καταχωρημένων στο σύστημα IoT πόρων καθορίζοντας στα κριτήρια αναζήτησης την πλατφόρμα «marios-cloud»:



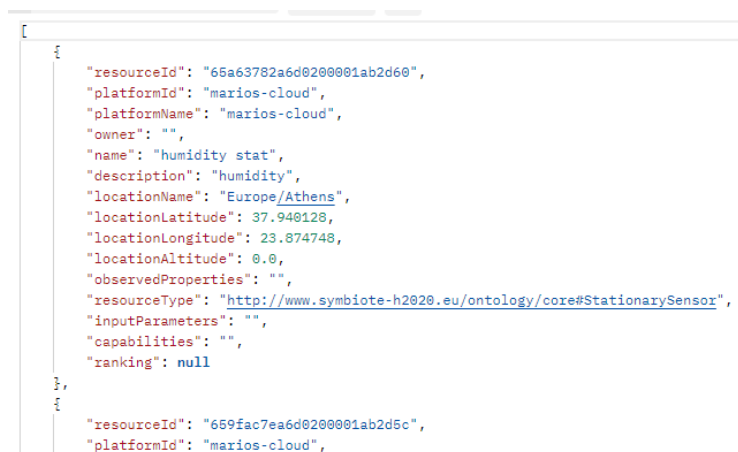
```

{
  "platform_id": "marios-cloud",
  "credentials": {
    "clientId": "test-client",
    "password": "***",
    "platformId": "SymbIoTe_Core_AAM",
    "username": "marketplace"
  }
}

```

Εικόνα 23: Αναζήτηση πόρων(L1) στην υπηρεσία Marketplace..

Η διεπαφή απαντάει με τη λίστα από τους πόρους της πλατφόρμας όπως φαίνεται παρακάτω:



```

[
  {
    "resourceId": "65a63782a6d0200001ab2d60",
    "platformId": "marios-cloud",
    "platformName": "marios-cloud",
    "owner": "",
    "name": "humidity stat",
    "description": "humidity",
    "locationName": "Europe/Athens",
    "locationLatitude": 37.940128,
    "locationLongitude": 23.874748,
    "locationAltitude": 0.0,
    "observedProperties": "",
    "resourceType": "http://www.symbiote-h2020.eu/ontology/core#StationarySensor",
    "inputParameters": "",
    "capabilities": "",
    "ranking": null
  },
  {
    "resourceId": "659fac7ea6d0200001ab2d5c",
    "platformId": "marios-cloud",
  }
]

```

Εικόνα 24 Απάντηση στην αναζήτηση πόρων(L1) στην υπηρεσία Marketplace..

**Διεπαφή symbIoTeMarketplace-12.** Η διεπαφή αυτή χρησιμοποιείται για να μπορεί ένας χρήστης να αναζητήσει ομόσπονδους πόρους που έχουν διαμοιραστεί στις ομόσπονδίες που συμμετέχει.

**symbIoTeMarketplace-12.A.** Το τεστ αυτό εκτελείται για την αναζήτηση διαθέσιμων IoT πόρων σε κάποια ομόσπονδη αγορά. Στα κριτήρια αναζήτησης του παραδείγματος καθορίζονται στα παραμέτρους το αναγνωριστικό της πλατφόρμας των IoT πόρων «marios-cloud» και το αναγνωριστικό της ομοσπονδίας «fedtsktsak» το οποίο καθορίζεται από την επισκεπτόμενη αγορά για κλήσεις μέσα από τη γραφική διεπαφή.



```

{
  "platformName": "marios-cloud",
  "resourceId": "659fac7ea6d020001ab2d5c",
  "maxDistance": 50,
  "observed_property": [],
  "federationId": "fedststksak",
  "priceMin": 6,
  "priceMax": 7,
  "repMin": 0,
  "repMax": 20,
  "credentials": {
    "clientId": "test-client",
    "password": "***",
    "platformId": "SymbIoTe_Core_AAM",
    "username": "marketplace"
  }
}

```

Εικόνα 25 Κλήση για αναζήτηση ομόσπονδων πόρων στην υπηρεσία Marketplace.

Το επιθυμητό αποτέλεσμα περιλαμβάνει τη λίστα με τους πόρους της πλατφόρμας στην ομοσπονδία όπως φαίνεται παρακάτω:

```

{
  "resourceId": "659fac7ea6d020001ab2d5c",
  "platform": "marios-cloud",
  "federationId": "fedststksak",
  "price": "0.1",
  "owner": "icom1",
  "name": "humidity stat",
  "description": "humidity",
  "locationName": "Europe/Athens",
  "locationLatitude": 37.940128,
  "locationLongitude": 23.874748,
  "locationAltitude": 0.0,
  "observedProperties": "",
  "resourceType": "http://www.symbiote-h2020.eu/ontology/core#StationarySensor",
  "lr": "1692207050100",
}

```

Εικόνα 26: Αποτέλεσμα κλήσης για την αναζήτηση ομόσπονδων πόρων(L2) στην υπηρεσία Marketplace.

**Διεπαφή symbIoTeMarketplace-13.** Η διεπαφή αυτή χρησιμοποιείται για την προσπέλαση δεδομένων ενός IoT πόρου.

**symbIoTeMarketplace-13.A.** Το εστιασμένο αυτό εκτελείται για την δοκιμή προσπέλασης των δεδομένων ενός IoT πόρου παρέχοντας στο url της κλήσης το αναγνωριστικό του IoT πόρου, τον αριθμό των προσπελάσεων (top=1 στο παράδειγμα) και το χρονικό πλαίσιο μέσα στο οποίο γίνεται η προσπέλαση:

?top=1&fromDate=2023-08-16T17:30:50.100Z&toDate=2024-01-30T11:30:50.100Z

```

{
  "platformCredentials": {
    "clientId": "Test_Client",
    "password": "***",
    "platformId": "SymbIoTe_Core_AAM",
    "username": "marketplace"
  },
  "remotePlatformId": "marios-cloud2"
}

```

Εικόνα 27 Κλήση για την προσπέλαση δεδομένων ενός IoT πόρου στην υπηρεσία Marketplace.

Το επιθυμητό αποτέλεσμα με τις μετρήσεις του IoT πόρου φαίνεται παρακάτω:

```
[
  {
    "resourceId": "65bcffcca6d0200001ab2d8b",
    "location": {
      "@c": ".WGS84Location",
      "longitude": 48.2088475,
      "latitude": 16.3734492,
      "altitude": 158.0,
      "name": "Stephansdome",
      "description": [
        "City of Wien"
      ]
    },
    "resultTime": "2024-02-14T08:44:54",
    "samplingTime": "2024-02-14T08:44:53",
    "obsValues": [
      {
        "value": "37",
        "obsProperty": {
          "name": "Temperature",
          "iri": "TempIRI",
          "description": [
            "Air temperature"
          ]
        },
        "uom": {
          "symbol": "C",
          "name": "degree Celsius",
          "iri": "C_IRI",
          "description": null
        }
      }
    ]
  }
]
```

Εικόνα 28: Απάντηση προσπέλασης του IoT πόρου στην υπηρεσία Marketplace.

**Διεπαφή symbIoTeMarketplace-14.** Η διεπαφή αυτή χρησιμοποιείται για να μπορεί ένας χρήστης να προσπελάσει το περιεχόμενο ενός πόρου με το αναγνωριστικό που του έδωσε κατά την εγγραφή του (internalId).

**Διεπαφή symbIoTeMarketplace-14.B.** Το τεστ αυτό εκτελείται για την προσπέλαση δεδομένων καθορίζοντας τώρα το εσωτερικό αναγνωριστικό πόρου internalId (isen1):

```
/isen1?top=1&fromDate=2023-08-16T17:30:50.100Z&toDate=2024-01-30T11:30:50.100Z
```

```
{
  "platformCredentials": {
    "clientId": "Test_Client",
    "password": "***",
    "platformId": "SymbIoTe_Core_AAM",
    "username": "marketplace"
  },
  "remotePlatformId": "marios-cloud"
}
```

Εικόνα 29 Κλήση της υπηρεσίας του marketplace για την προσπέλαση IoT πόρου.

Το επιθυμητό αποτέλεσμα επιστρέφει και πάλι τις μετρήσεις του IoT πόρου όπως φαίνεται στο παράδειγμα:

```
[
  {
    "resourceId": "65a90863a6d0200001ab2d66",
    "location": {
      "@c": ".WGS84Location",
      "longitude": 48.2088475,
      "latitude": 16.3734492,
      "altitude": 158.0,
      "name": "Stephansdome",
      "description": [
        "City of Wien"
      ]
    },
    "resultTime": "2024-01-23T15:16:12",
    "samplingTime": "2024-01-23T15:16:11",
    "obsValues": [
      {
        "value": "32",
        "obsProperty": {
          "name": "Temperature",
          "iri": "TempIRI",
          "description": [
            "Air temperature"
          ]
        }
      }
    ]
  }
]
```

Εικόνα 30: Δεδομένα προσπέλασης του IoT πόρου «isenI»..

**Διεπαφή symbIoTeMarketplace-16.** Η διεπαφή αυτή χρησιμοποιείται για την καταχώρηση ενός IoT πόρου στο σύστημα

**Διεπαφή symbIoTeMarketplace-16.A.** Το τεστ αυτό εκτελείται για την καταχώρηση ενός IoT πόρου «resource-example» από τον χρήστη «icom1» της πλατφόρμας «marios-cloud».

```
{
  "credentials": {
    "clientId": "Test-client",
    "password": "icom",
    "platformId": "marios-cloud",
    "username": "icom1"
  },
  "cloudResource": {
    "internalId": "resource-example",
    "pluginId": "platform_01",
    "accessPolicy": {
      "policyType": "PUBLIC",
      "requiredClaims": {}
    }
  },
  "filteringPolicy": {
    "policyType": "PUBLIC",
    "requiredClaims": {}
  },
  "resource": {
```

Εικόνα 31 Κλήση υπηρεσίας του marketplace για καταχώρηση IoT πόρου.

Το επιθυμητό αποτέλεσμα περιέχει την περιγραφή του IoT πόρου (μεταδεδομένα) που καταχωρήθηκε επιτυχώς όπως φαίνεται παρακάτω:

```

{
  "cloudResources": [
    {
      "internalId": "resource-example",
      "pluginId": "platform_01",
      "accessPolicy": {
        "policyType": "SLHTIBAP",
        "requiredClaims": {
          "sub": "marketplace",
          "iss": "SymbIoTe_Core_AAM"
        }
      },
      "filteringPolicy": {
        "policyType": "PUBLIC",

```

Εικόνα 32: Απάντηση καταχώρησης IoT πόρου μέσω του marketplace.

**Διεπαφή symbIoTeMarketplace-17.** Η διεπαφή αυτή χρησιμοποιείται για να μπορεί ο ιδιοκτήτης ενός πόρου να τον διαμοιράσει σε κάποια ομόσπονδη αγορά.

**Διεπαφή symbIoTeMarketplace-17.A.** Στο τεστ που εκτελείται (Εικόνα 33) ο πάροχος «icom1» διαμοιράζει τον πόρο «resource-example3» (internalId) στην ομοσπονδία «fedststktsak» θέτοντας την τιμή βάσης 0.1 FedCoins και αρχικοποιώντας τις υπόλοιπες παραμέτρους που επηρεάζουν άμεσα την συνάρτηση τιμολόγησης όπως το χρονικό παράθυρο που υποστηρίζεται ο πόρος και συντελεστές βαρύτητας:

```

{
  "bartered": true,
  "federationId": "fedststktsak",
  "price": "0.1",
  "resourceInternalId": "resource-example3",
  "platformCredentials": {
    "clientId": "test-client",
    "password": "icom",
    "platformId": "marios-cloud",
    "username": "icom1"
  },
  "lr": "2023-08-16T17:30:50.100Z",
  "hr": "2024-01-30T11:30:50.100Z",
  "fr": 5,
  "wr": 1440,
  "aro": 1.05,
  "ard": 0.001,
  "brd": 2,
  "sr": 1,
  "arf": 0.05,
  "brr": 12,
  "arr": 1.2
}

```

Εικόνα 33 Κλήση της υπηρεσίας Marketplace για τον διαμοιρασμό ενός πόρου σε μια ομοσπονδία..

Το επιθυμητό αποτέλεσμα επιστρέφει την πληροφορία του IoT πόρου περιλαμβάνοντας τα στοιχεία του διαμοιρασμού του στην ομοσπονδία.

```

{
  "fedststktsak": [
    {
      "internalId": "resource-example3",
      "pluginId": "platform_01",
      "accessPolicy": {
        "policyType": "SLHTIBAP",
        "requiredClaims": {
          "sub": "marketplace",
          "iss": "SymbIoTe_Core_AAM"

```

Εικόνα 34: Αποτέλεσμα επιτυχούς διαμοιρασμού ενός πόρου σε μια ομοσπονδία..

Στον παρακάτω πίνακα συνοψίζονται οι δοκιμές και το αποτέλεσμα τους (Επιτυχές ή μη επιτυχές).

Εικόνα 35 Κλήση της υπηρεσίας Marketplace για τον διαμοιρασμό ενός πόρου σε μια ομοσπονδία..

Interface	Test case	Method	Call	Status
SymbIoTeMarketplace-11	SymbIoTeMarketplace-11.A	POST	/SymbIoTeMarketplace/resource/search/11	PASS
SymbIoTeMarketplace-12	SymbIoTeMarketplace-12.A	POST	/SymbIoTeMarketplace/resource/search/L2	PASS
SymbIoTeMarketplace-13	SymbIoTeMarketplace-13.A	POST	/SymbIoTeMarketplace/resource/access/11/Id/{resourceId}	PASS
SymbIoTeMarketplace-14	SymbIoTeMarketplace-14.A	POST	/SymbIoTeMarketplace/resource/access/11/{resourceInternalId}	PASS
SymbIoTeMarketplace-15	SymbIoTeMarketplace-15.A	POST	/SymbIoTeMarketplace/resource/delete/{resourceInternalId}	PASS
SymbIoTeMarketplace-16	SymbIoTeMarketplace-16.A	POST	/SymbIoTeMarketplace/resource/add/11	PASS
SymbIoTeMarketplace-17	SymbIoTeMarketplace-17.A	POST	/SymbIoTeMarketplace/resource/share	PASS

Στη συνέχεια περιγράφονται οι διεπαφές για την αγορά ανταλλαγής προϊόντων και η επαλήθευσή τους.

Πίνακας 6: Διεπαφές marketplace για την αγορά ανταλλαγής προϊόντων.

Interface ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeMarketplace-18	REST	Submit new exchange offer	/SymbIoTeMarketplace/exchange/	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-19	REST	Retrieve exchange offer	/SymbIoTeMarketplace/exchange/{id}	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-20	REST	Cancel exchange offer	/SymbIoTeMarketplace/exchange/{id}/cancel	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-21	REST	Accept exchange offer	/SymbIoTeMarketplace/exchange/{id}/accept	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-22	REST	Decline exchange offer	/SymbIoTeMarketplace/{id}/decline	IoTFeds User	symbIoTeMarketplace	v2.0

SymbIoTeMarketplace-23	REST	Retrieve user's own exchange offers	/SymbIoTeMarketplace/exchange/	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-24	REST	Get exchangeable tokens	/SymbIoTeMarketplace/exchange/tokens	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-25	REST	Put token for exchange	/SymbIoTeMarketplace/exchange/tokens	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-26	REST	Remove token from exchange	/SymbIoTeMarketplace/exchange/tokens	IoTFeds User	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-27	REST	Exchange tokens	/SymbIoTeMarketplace/exchange/tokens	IoTFeds User	symbIoTeMarketplace	v2.0

**Διεπαφή symbIoTeMarketplace-18.** Η διεπαφή αυτή χρησιμοποιείται για την υποβολή πρότασης ανταλλαγής κουπονιών προϊόντων.

**symbIoTeMarketplace-18.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-18. Το ακόλουθο σώμα για την κλήση πρέπει να οριστεί:

```
{
  "userId": "icom1",
  "productId": "658563c17bd5de6c3ca17e4a",
  "targetProductId": "65855a1c7bd5de6c3ca17e49",
  "targetUser": "icom2"
}
```

Εικόνα 36: Παράδειγμα κλήσης στο marketplace για υποβολή πρότασης ανταλλαγής κουπονιών προϊόντων.

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω. Μία νέα πρόταση ανταλλαγής έχει καταγραφεί με αναγνωριστικό «65a796821adf7e68a78e728e»:

```
{
  "id": "65a796821adf7e68a78e728e",
  "userId": "icom1",
  "productId": "658563c17bd5de6c3ca17e4a",
  "targetUserId": "icom2",
  "targetProductId": "65855a1c7bd5de6c3ca17e49",
  "status": "pending",
  "accepted": false,
  "dateSubmitted": null
}
```

Εικόνα 37: Απάντηση υποβολής πρότασης ανταλλαγής στο marketplace.

**Διεπαφή symbIoTeMarketplace-19.** Η διεπαφή αυτή χρησιμοποιείται για την ανάκτηση συγκεκριμένης πρότασης ανταλλαγής κουπονιών βάσει του αναγνωριστικού (ID) της.

**symbIoTeMarketplace-19.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-19. Το αναγνωριστικό της πρότασης πρέπει να οριστεί ως παράμετρος στον σύνδεσμο (url) του αιτήματος όπως φαίνεται στο παράδειγμα:  
/marketplace/exchange/65a796821adf7e68a78e728e.

Το επιθυμητό αποτέλεσμα με τις λεπτομέρειες της πρότασης που ανακτήθηκαν επιτυχώς φαίνεται παρακάτω:

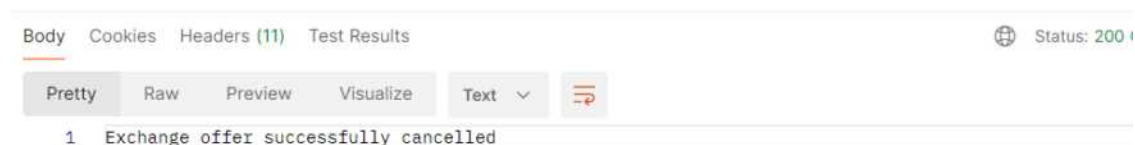
```
1  {  
2    "id": "65a796821adf7e68a78e728e",  
3    "userId": "icom1",  
4    "productId": "658563c17bd5de6c3ca17e4a",  
5    "targetUserId": "icom2",  
6    "targetProductId": "65855a1c7bd5de6c3ca17e49",  
7    "status": "pending",  
8    "accepted": false,  
9    "dateSubmitted": null  
10 }
```

Εικόνα 38: Απάντηση ανάκτησης πρότασης ανταλλαγής κουπονιών.

**Διεπαφή symbIoTeMarketplace-20.:** Η διεπαφή χρησιμοποιείται για την ακύρωση μια πρότασης ανταλλαγής που έχει υποβάλει κάποιος χρήστης.

**symbIoTeMarketplace-20.A:** Το τεστ εκτελείται για την διεπαφή symbIoTeMarketplace-20 παρέχοντας το αναγνωριστικό της πρότασης ακολουθούμενου από την κατάληξη «/cancel».  
/marketplace/exchange/65a796821adf7e68a78e728e/cancel.

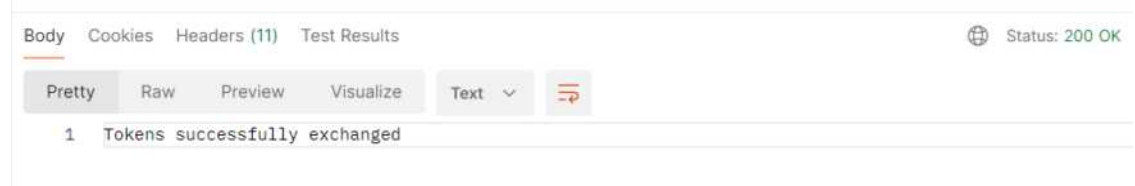
Το επιθυμητό αποτέλεσμα φαίνεται στην παρακάτω εικόνα και επιβεβαιώνει ότι η πρόταση ακυρώθηκε επιτυχώς:



Εικόνα 39: Απάντηση ακύρωσης πρότασης ανταλλαγής.

**Διεπαφή symbIoTeMarketplace-21.** Η διεπαφή ενημερώνει την πύλη αγοράς όταν ο αποδέκτης μιας πρότασης ανταλλαγής την αποδέχτηκε.

**symbIoTeMarketplace-21.A:** Το ακόλουθο τεστ εκτελεί το αίτημα αποδοχής της πρότασης που απαιτεί το αναγνωριστικό της πρότασης ακολουθούμενου από την κατάληξη «/accept»:  
/marketplace/exchange/65ae79ba987c3308dcb4d452/accept. Με την υποβολή του αιτήματος ενημερώνεται αυτομάτως το BaaS ώστε να εκτελεστεί και η ανταλλαγή των κουπονιών. Το επιθυμητό αποτέλεσμα φαίνεται στην παρακάτω εικόνα.



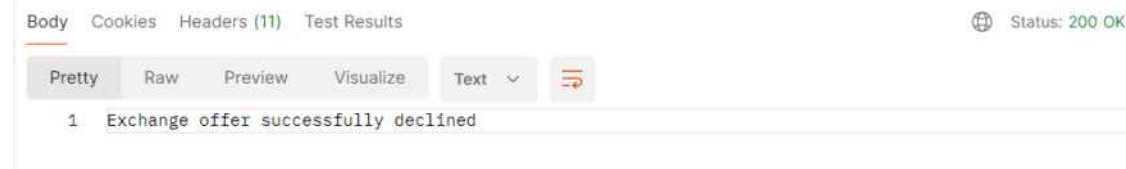
Εικόνα 40: Απάντηση αποδοχής πρότασης ανταλλαγής.

**Διεπαφή symbIoTeMarketplace-22.** Η διεπαφή ενημερώνει την πύλη αγοράς όταν ο αποδέκτης μιας πρότασης ανταλλαγής την αρνήθηκε.

**symbIoTeMarketplace-22.A:** Το ακόλουθο τεστ εκτελεί το αίτημα άρνησης της πρότασης με την παροχή του αναγνωριστικού της πρότασης ακολουθούμενου από την κατάληξη «/decline» στο url του αιτήματος.

/marketplace/exchange/65ae79ba987c3308dcb4d452/decline.

Το αναμενόμενο αποτέλεσμα φαίνεται και στην παρακάτω εικόνα:



Εικόνα 41: Απάντηση απόρριψης πρότασης ανταλλαγής.

**Διεπαφή symbIoTeMarketplace-23:** Η διεπαφή ανακτά τις προτάσεις που έχουν αποσταλεί από ένα χρήστη.

**symbIoTeMarketplace-23.A:** Το αναγνωριστικό του χρήστη παρέχεται στο url του αιτήματος, ενώ προαιρετικά μπορεί να δοθεί ως φίλτρο και η κατάσταση της πρότασης (ακυρωμένη, ολοκληρωμένη ή εκκρεμής). Το ακόλουθο τεστ εκτελεί το αίτημα ανάκτησης όλων των προτάσεων που έχει αποστείλει ο χρήστης «icom1» χωρίς την παροχή κάποιου φίλτρου: /marketplace/exchange?userId=icom1

Το επιτυχές αποτέλεσμα φαίνεται στην παρακάτω εικόνα:

```

1  [
2    {
3      "id": "65a796821adf7e68a78e728e",
4      "userId": "icom1",
5      "productId": "658563c17bd5de6c3ca17e4a",
6      "targetUserId": "icom2",
7      "targetProductId": "65855a1c7bd5de6c3ca17e49",
8      "status": "cancelled",
9      "accepted": false,
10     "dateSubmitted": null
11   },
12   {
13     "id": "65ae79ba987c3308dcb4d452",
14     "userId": "icom1",
15     "productId": "658563c17bd5de6c3ca17e4a",
16     "targetUserId": "icom2",
17     "targetProductId": "65855a1c7bd5de6c3ca17e49",
18     "status": "completed",
19     "accepted": true,
20     "dateSubmitted": null
21   }
22 ]

```

Εικόνα 42: Απάντηση αιτήματος ανάκτησης με τις υποβληθείσες προτάσεις του χρήστη.

**symbIoTeMarketplace-23.B:** Το ακόλουθο τεστ εκτελεί το αίτημα ανάκτησης όλων των προτάσεων που έχει αποστείλει ο χρήστης «icom1» φιλτράροντας ταυτόχρονα την κατάσταση της πρότασης ως «completed» (ολοκληρωμένη). Απαιτούνται να εισαχθούν στις παραμέτρους το αναγνωριστικό του χρήστη και η κατάσταση της πρότασης για το φίλτρο:

marketplace/exchange?userId=icom1&status=completed

Το επιτυχές αποτέλεσμα φαίνεται και στην παρακάτω εικόνα:



```

1  [
2    {
3      "id": "65ae79ba987c3308dcb4d452",
4      "userId": "icom1",
5      "productId": "658563c17bd5de6c3ca17e4a",
6      "targetUserId": "icom2",
7      "targetProductId": "65855a1c7bd5de6c3ca17e49",
8      "status": "completed",
9      "accepted": true,
10     "dateSubmitted": null
11   }
12 ]

```

Εικόνα 43: Απάντηση αιτήματος ανάκτησης με τις υποβληθείσες προτάσεις του χρήστη και κατάσταση *completed*.

**Διεπαφή symbIoTeMarketplace-24:** Η διεπαφή ανακτά όλα τα κουπόνια που ανήκουν στον χρήστη.

**symbIoTeMarketplace-24.A:** Το τεστ που ακολουθεί ανακτά από το BaaS όλα τα κουπόνια του χρήστη. Το αίτημα πρέπει να περιέχει ως παράμετρο το αναγνωριστικό του χρήστη στο url όπως φαίνεται στο παραδειγμα:

/marketplace/exchange/user-tokens?userId=icom2

Το επιθυμητό αποτέλεσμα φαίνεται στην παρακάτω εικόνα:

```

{
  "code": "OK",
  "body": [
    {
      "productId": "65a791431adf7e68a78e728d",
      "owner": "icom2",
      "accessTimes": 10,
      "dataAvailableFrom": "2023-08-29T17:30:50.100Z",
      "dataAvailableUntil": "2024-01-30T11:30:50.100Z",
      "marketplaceId": "fedtstktsak",
      "toBeExchanged": false,
      "userHasRated": false
    },
    {
      "productId": "anasiouProd",
      "owner": "icom2",
      "accessTimes": 8,
      "dataAvailableFrom": "2023-08-29T17:30:50.100Z",
      "dataAvailableUntil": "2024-01-30T11:30:50.100Z",
      "marketplaceId": "fedtstktsak",
      "toBeExchanged": false,
      "userHasRated": false
    }
  ]
}

```

Εικόνα 44: Ανάκτηση λίστας κουπονιών του χρήστη.

**Διεπαφή symbIoTeMarketplace-25:** Η διεπαφή αυτή ανακτά τα ανταλλάξιμα κουπόνια μιας αγοράς.

**symbIoTeMarketplace-25.A:** Στο ακόλουθο τεστ απαιτείται να οριστούν το αναγνωριστικό της ομοσπονδίας στο url του αιτήματος όπως φαίνεται παρακάτω για την ανάκτηση της λίστας των κουπονιών που έχουν σημανθεί ως ανταλλάξιμα:

/marketplace/exchange/tokens?marketplace=fedtstktsak.

Το επιθυμητό αποτέλεσμα με την λίστα των ανταλλάξιμων κουπονιών στην αγορά φαίνεται παρακάτω:

```
[
  {
    "productId": "65a791431adf7e68a78e728d",
    "owner": "icom2",
    "accessTimes": 10,
    "dataAvailableFrom": "2023-08-29T17:30:50.100Z",
    "dataAvailableUntil": "2024-01-30T11:30:50.100Z",
    "marketplaceId": "fedtstksak",
    "toBeExchanged": true,
    "userHasRated": false
  }
]
```

Εικόνα 45: Ανάκτηση λίστας κουπονιών σε μία αγορά.

**Διεπαφή symbIoTeMarketplace-26:** Η διεπαφή επιτρέπει την τοποθέτηση ενός κουπονιού του χρήστη ως διαθέσιμο για ανταλλαγή. Στην περίπτωση αυτή, η τιμή του πεδίου του κουπονιού «toBeExchanged» αλλάζει από false σε true.

**symbIoTeMarketplace-26.A:** Στο ακόλουθο τεστ πρέπει να οριστούν οι παρακάτω υποχρεωτικές παράμετροι στο url του αιτήματος (αναγνωριστικά χρήστη, προϊόντος και ομοσπονδίας):

/marketplace/exchange/tokens?userId=icom2&productId=65a791431adf7e68a78e728d&marketplace=fedtstksak

Το επιθυμητό αποτέλεσμα φαίνεται στην παρακάτω εικόνα, όπου η τιμή «toBeExchanged» αλλάζει σε “true” που δηλώνει το κουπόνι προϊόντος ως ανταλλάξιμο ώστε να εμφανίζεται στη λίστα ανταλλάξιμων κουπονιών στην πύλη αγοράς.

```
{
  "code": "OK",
  "body": [
    {
      "productId": "65a791431adf7e68a78e728d",
      "owner": "icom2",
      "accessTimes": 10,
      "dataAvailableFrom": "2023-08-29T17:30:50.100Z",
      "dataAvailableUntil": "2024-01-30T11:30:50.100Z",
      "marketplaceId": "fedtstksak",
      "toBeExchanged": true,
      "userHasRated": false
    }
  ]
}
```

Εικόνα 46: Απάντηση αιτήματος για την τοποθέτηση του κουπονιού ως ανταλλάξιμο.

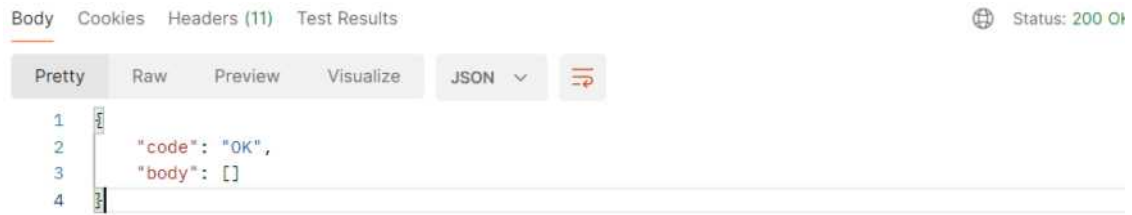
**Διεπαφή symbIoTeMarketplace-27:** Η διεπαφή αυτή επισημαίνει το κουπόνι προϊόντος ως μη ανταλλάξιμο στην πύλη αγοράς.

**symbIoTeMarketplace-27.A:** Στο ακόλουθο τεστ απαιτούνται να οριστούν στο url του αιτήματος τα αναγνωριστικά χρήστη, προϊόντος και ομοσπονδίας όπως φαίνονται παρακάτω για να επισημανθεί το κουπόνι προϊόντος με αναγνωριστικό «65a791431adf7e68a78e728d» ως μη ανταλλάξιμο:

/marketplace/exchange/tokens?userId=icom2&productId=65a791431adf7e68a78e728d&marketplace=fedtstksak

Το επιθυμητό αποτέλεσμα περιλαμβάνει τη λίστα των ανταλλάξιμων κουπονιών στην αγορά η οποία είναι κενή καθώς, για επαλήθευση, εκτελέστηκε αμέσως μετά αίτημα προς τη διεπαφή symbIoTeMarketplace-25 ώστε να ανακτηθούν όλα τα ανταλλάξιμα κουπόνια της ίδιας αγοράς.

Εφόσον το κουπόνι ήταν και το μοναδικό ανταλλάξιμο στην ομοσπονδία η λίστα που επιστρέφεται είναι κενή όπως φαίνεται στην εικόνα.



Εικόνα 47: Απάντηση αιτήματος απόσυρσης κουπονιού από ανταλλάξιμο.

Στον παρακάτω πίνακα συνοψίζονται οι δοκιμές των διεπαφών και το αποτέλεσμα τους (επιτυχές ή μη).

Interface	Test case	Method	Call	Status
SymbIoTeMarketplace-19	SymbIoTeMarketplace-19.A	POST	/SymbIoTeMarketplace/exchange/	PASS
SymbIoTeMarketplace-20	SymbIoTeMarketplace-20.A	GET	/SymbIoTeMarketplace/exchange/{id}	PASS
SymbIoTeMarketplace-21	SymbIoTeMarketplace-21.A	POST	/SymbIoTeMarketplace/exchange/{id}/cancel	PASS
SymbIoTeMarketplace-22	SymbIoTeMarketplace-22.A	POST	/SymbIoTeMarketplace/exchange/{id}/accept	PASS
SymbIoTeMarketplace-23	SymbIoTeMarketplace-23.A	POST	/SymbIoTeMarketplace/{id}/decline	PASS
SymbIoTeMarketplace-24	SymbIoTeMarketplace-24.A	GET	/SymbIoTeMarketplace/exchange/	PASS
SymbIoTeMarketplace-25	SymbIoTeMarketplace-25.A	GET	/SymbIoTeMarketplace/exchange/tokens	PASS
SymbIoTeMarketplace-26	SymbIoTeMarketplace-26.A	POST	/SymbIoTeMarketplace/exchange/tokens	PASS

arket-place-26				
SymbIoTeMarketplace-27	SymbIoTeMarketplace-27.A	DELETE	/SymbIoTeMarketplace/exchange/tokens	PASS

Παρακάτω παρουσιάζονται οι διεπαφές σχετικά με τον έλεγχο εγκυρότητας των συναλλαγών αγοράς και αποδεικτικών χρήσης των κουπονιών.

Inter Face ID	Type	Feature	Interface	Request from	Request to	Release
SymbIoTeMarketplace-28	REST	Get Receipt for buyer	/SymbIoTeMarketplace/auditing/receipt_buyer	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-29	REST	Get Receipt for seller	/SymbIoTeMarketplace/auditing/receipt_seller	IoTFeds User (Data Producer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-30	REST	Get logs for transactions	/SymbIoTeMarketplace/auditing/logs	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-31	REST	Verify receipt of transaction	/SymbIoTeMarketplace/auditing/verify_receipt	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0
SymbIoTeMarketplace-32	REST	Verify access	/SymbIoTeMarketplace/auditing/verify_access_log	IoTFeds User (Data Consumer)	symbIoTeMarketplace	v2.0

**Διεπαφή symbIoTeMarketplace-28.** Η διεπαφή αυτή χρησιμοποιείται για τη λήψη αποδείξεων συναλλαγής από τον αγοραστή.

**symbIoTeMarketplace-28.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-28. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "product_id":"testProd",
  "fromDate":"2024-01-11T09:22:52.000Z",
  "toDate":"2024-01-17T09:22:52.000Z"
}.
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
[
  {
```

```
    "TransactionID":  
    "d3ac04fb58680171d798e299c330b84aa603afd8e830ebd9207c634c580c5455",  
    "Buyer": "icom2",  
    "Seller": "icom1",  
    "ProductID": "anasiouProd",  
    "DateBought": "2024-01-16T13:49:30.000Z"  
  }  
]
```

**Διεπαφή symbIoTeMarketplace-29.** Η διεπαφή αυτή χρησιμοποιείται για τη λήψη αποδείξεων συναλλαγής από τον πωλητή.

**symbIoTeMarketplace-29.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-29. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{  
  "product_id": "testProd",  
  "fromDate": "2024-01-11T09:22:52.000Z",  
  "toDate": "2024-01-17T09:22:52.000Z"  
}.  
Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:
```

```
[  
  {  
    "TransactionID":  
    "d3ac04fb58680171d798e299c330b84aa603afd8e830ebd9207c634c580c5455",  
    "Buyer": "icom2",  
    "Seller": "icom1",  
    "ProductID": "anasiouProd",  
    "DateBought": "2024-01-16T13:49:30.000Z"  
  }  
].
```

**Διεπαφή symbIoTeMarketplace-30.** Η διεπαφή αυτή χρησιμοποιείται για τη λήψη αποδείξεων πρόσβασης ενός προϊόντος.

**symbIoTeMarketplace-30.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-30. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{  
  "product_id": "testProd",  
  "fromDate": "2024-01-11T09:22:52.000Z",  
  "toDate": "2024-01-17T09:22:52.000Z"  
}  
Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:  
[  
  {  
    "TransactionID":  
    "72ab20803c6e88dc75965a513be01b3799e3eebda9ec5f30f570c8f550c2b953",  
    "User": "icom2",  
    "ProductID": "anasiouProd",  
    "DateAccessed": "2024-01-16T14:03:28.000Z"  
  }  
].
```

**Διεπαφή symbIoTeMarketplace-31.** Η διεπαφή αυτή χρησιμοποιείται για την επαλήθευση απόδειξης συναλλαγής.

**symbIoTeMarketplace-31.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-31. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "receipt": {
    "TransactionID":
"d3ac04fb58680171d798e299c330b84aa603afd8e830ebd9207c634c580c5455",
    "Buyer": "icom2",
    "Seller": "icom1",
    "ProductID": "anasiouProd",
    "DateBought": "2024-01-16T13:49:30.000Z"
  },
  "user_id": "icom1"
}.
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "Verified": true
}
```

**Διεπαφή symbIoTeMarketplace-32.** Η διεπαφή αυτή χρησιμοποιείται για την επαλήθευση απόδειξης πρόσβασης ενός προϊόντος.

**symbIoTeMarketplace-32.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-32. Το ακόλουθο σώμα για την κλήση θα πρέπει να οριστεί:

```
{
  "log": {
    "TransactionID":
"72ab20803c6e88dc75965a513be01b3799e3eebda9ec5f30f570c8f550c2b953",
    "User": "icom2",
    "ProductID": "anasiouProd",
    "DateAccessed": "2024-01-16T14:03:28.000Z"
  },
  "user_id": "iotFedsAdmin"
}
```

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
  "Verified": true
}
```

**Διεπαφή symbIoTeMarketplace-33.** Η διεπαφή αυτή χρησιμοποιείται για την ενημέρωση της πληροφορίας (balance ) των κρυπτονομισμάτων (FedCoins) του χρήστη που έχει εισέλθει στο marketplace.

**symbIoTeMarketplace-33.A.** Το τεστ εκτελείται για τη διεπαφή SymbIoTeAPI-33. Δε χρειάζεται κάποιο σώμα, μόνο το token που έχει λάβει ο χρήστης κατά την είσοδο του στο marketplace.

Το επιθυμητό αποτέλεσμα φαίνεται παρακάτω:

```
{
```

```
"balance": 99
}
```

Interface	Test case	Method	Call	Status
SymbIoTeMarketplace-28	SymbIoTeMarketplace-28.A	POST	/SymbIoTeMarketplace/auditing/receipt_buyer	PASS
SymbIoTeMarketplace-29	SymbIoTeMarketplace-29.A	POST	/SymbIoTeMarketplace/auditing/receipt_seller	PASS
SymbIoTeMarketplace-30	SymbIoTeMarketplace-30.A	POST	/SymbIoTeMarketplace/auditing/logs	PASS
SymbIoTeMarketplace-31	SymbIoTeMarketplace-31.A	POST	/SymbIoTeMarketplace/auditing/verify_receipt	PASS
SymbIoTeMarketplace-32	SymbIoTeMarketplace-32.A	POST	/SymbIoTeMarketplace/auditing/verify_access_log	PASS
SymbIoTeMarketplace-33	SymbIoTeMarketplace-33.A	GET	/SymbIoTeMarketplace/user/cpanel/get_balance	PASS

### 3.3 Υποσύστημα BaaS

URL	
GitHub repository	<a href="https://github.com/iotfedsproject/BaaS">https://github.com/iotfedsproject/BaaS</a>
Container registry	fabric-ca, fabric-peer, fabric-orderer
Deployment yaml file	docker-compose-ca.yaml, docker-compose-test-net.yaml

#### 3.3.1 Στοιχείο λογισμικού για τη διαχείριση χρηστών

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι περιπτώσεις δοκιμών για το στοιχείο λογισμικού BaaS για τη διαχείριση χρηστών.

##### 3.3.1.1 Περιγραφή

Το στοιχείο λογισμικού BaaS για τη διαχείριση χρηστών παρέχει τη δυνατότητα στους χρήστες να εγγράφονται στο blockchain και να καταχωρούν τις πλατφόρμες και συσκευές που έχουν εγγράψει στο symbIoTe. Για τις παραπάνω διεργασίες προσφέρονται και οι αντίστροφες

λειτουργίες της αντίστοιχης διαγραφής. Επιπλέον προσφέρεται η δυνατότητα ενημέρωσης του υπολοίπου ενός χρήστη, και πρόσβασης σε πληροφορίες του χρήστη.

### 3.3.1.2 Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Τα παρακάτω χαρακτηριστικά προστέθηκαν στην ενδιάμεση έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- **registerUserToBc** – εγγραφή ενός χρήστη στο BaaS
- **regiterPlatforms** – εγγραφή μιας πλατφόρμας σε έναν χρήστη στο BaaS
- **registerDevice** - εγγραφή μιας συσκευής σε μια πλατφόρμα χρήστη στο BaaS
- **getUserInfo** – πρόσβαση σε στοιχεία ενός χρήστη στο BaaS
- **getAllUsers** – πρόσβαση σε στοιχεία όλων των χρηστών στο BaaS
- **updateUserBalance** – ενημέρωση υπολοίπου χρήστη στο BaaS
- **removePlatform** – Αφαίρεση πλατφόρμας χρήστη από το BaaS
- **removePlatformResources** – Αφαίρεση πόρων πλατφόρμας χρήστη από το BaaS
- **removeDevice** – Αφαίρεση συσκευής χρήστη από το BaaS
- **deleteUser** – Αφαίρεση χρήστη από το BaaS

### 3.3.1.3 Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Το BaaS API για διαχείριση χρηστών είναι προσβάσιμο μέσω Rest Get/Post/Patch/Delete HTTP αιτήματα στο παρακάτω url:

BaaS-API-url-user: <http://160.40.54.146:3009/baas/user>.

Πίνακας 7: Διεπαφές στοιχείου BaaS για την διαχείριση χρηστών.

Inter Face ID	Type	Feature	Endpoint /Queue	Request from	Request to	Release
BaaSAPI-1	REST	registerUser ToBc	POST <BaaS-API-url-user>/register_user	sybmIoTe	BaaS	v1.0
BaaSAPI-2	REST	regiterPlatform	POST <BaaS-API-url-user>/register_platform	sybmIoTe	BaaS	v1.0
BaaSAPI-3	REST	registerDevice	POST <BaaS-API-url-user>/register_device	sybmIoTe	BaaS	v2.0
BaaSAPI-4	REST	getUserInfo	GET < BaaS-API-url-user>/get_user_info	sybmIoTe	BaaS	v2.0
BaaSAPI-5	REST	getAllUsers	GET < BaaS-API-url-user>/get_all_users	sybmIoTe	BaaS	v2.0
BaaSAPI-6	REST	removePlatform	DELETE < BaaS-API-url-user>/remove_platform	sybmIoTe	BaaS	v1.0
BaaSAPI-7	REST	removeDevice	DELETE < BaaS-API-url-user>/remove_device	sybmIoTe	BaaS	v1.0
BaaSAPI-8	REST	deleteUser	DELETE < BaaS-API-url-user>/delete_user	sybmIoTe	BaaS	v1.0



### 3.3.1.4 Δοκιμή και επαλήθευση (Test cases)

Στην ενότητα αυτή περιγράφονται οι περιπτώσεις δοκιμών του στοιχείου λογισμικού BaaS για τη διαχείριση χρηστών.

**Διεπαφή BaaSAPI-1.** Χρησιμοποιείται για την εγγραφή ενός χρήστη στο BaaS.

**BaaSAPI-1.A.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης εγγράφεται επιτυχώς. Οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος όπως φαίνεται στην Εικόνα 48.



```
{
  "id": "gpapadop",
  "role": "researcher",
  "mail": "gpapadop@iti.gr",
  "organization": "CERTH"
}
```

Εικόνα 48: Παράμετροι στο σώμα του ελέγχου BaaSAPI-1.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!»

**BaaSAPI-1.B.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης δε μπορεί να διπλοεγγραφεί στο σύστημα. Οι παράμετροι που πρέπει να οριστούν στο σώμα του αιτήματος είναι ίδιοι με τον προηγούμενο έλεγχο. Το αναμενόμενο αποτέλεσμα είναι status: 403 με μήνυμα που περιέχει «The user gpapadop already exists...».

**Διεπαφή BaaSAPI-2.** Χρησιμοποιείται για την εγγραφή μιας πλατφόρμας ενός χρήστη στο BaaS.

**BaaSAPI-2.A.** Αυτός ο έλεγχος επαληθεύει ότι μια πλατφόρμα εγγράφεται επιτυχώς. Οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 49.



```
{
  "platform_id": "vicinity",
  "device_id": "device1",
  "user": "gpapadop"
}
```

Εικόνα 49: Παράμετροι στο σώμα του ελέγχου BaaSAPI-2.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!»

**BaaSAPI-2.B.** Αυτός ο έλεγχος επαληθεύει ότι μια πλατφόρμα δε μπορεί να δίπλο-εγγραφεί στον ίδιο χρήστη, οι παράμετροι που πρέπει να οριστούν στο σώμα του αιτήματος είναι ίδιο με τον προηγούμενο έλεγχο. Το αναμενόμενο αποτέλεσμα είναι status: 400 με μήνυμα που περιέχει «Update failed...» και «The platform vicinity is already registered»

**Διεπαφή BaaSAPI-3.** Χρησιμοποιείται για την εγγραφή μιας συσκευής σε μια πλατφόρμα ενός χρήστη στο BaaS.

**BaaSAPI-3.A.** Αυτός ο έλεγχος επαληθεύει ότι μια συσκευή γράφεται με επιτυχία, οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 50.

```

{
  ... "platform_id": "vicinity",
  ... "device_id": "device1"
}

```

Εικόνα 50: Παράμετροι στο σώμα του ελέγχου BaaSAPI-3.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!»

**BaaSAPI-3.B.** Αυτός ο έλεγχος επαληθεύει ότι μια συσκευή δε μπορεί να διπλοεγγραφεί σε μια πλατφόρμα, οι παράμετροι που πρέπει να οριστούν στο σώμα του αιτήματος είναι ίδιες με τον προηγούμενο έλεγχο.

Το αναμενόμενο αποτέλεσμα για διπλοεγγραφή είναι status: 400 με μήνυμα που περιέχει «Registration failed» και «The device device1 is already registered...»

**Διεπαφή BaaSAPI-4.** Χρησιμοποιείται για την πρόσβαση σε πληροφορίες ενός χρήστη στο BaaS.

**BaaSAPI-4.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες ενός χρήστη επιστρέφονται με επιτυχία, εφόσον το αναγνωριστικό του χρήστη περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 51.

```

{
  "ActiveProducts": [],
  "AssociatedPlatforms": {
    "vicinity": []
  },
  "Balance": 1000,
  "Fed_owner": false,
  "Feds": [],
  "ID": "gpapadop",
  "Mail": "gpapadop@iti.gr",
  "Organization": "CERTH1",
  "Reputation": 0,
  "Role": "researcher",
  "docType": "user"
}

```

Εικόνα 51: Επιστροφή πληροφοριών χρήστη από το BaaS.

**BaaSAPI-4.B.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφεται το κατάλληλο μήνυμα για χρήστη που δεν υπάρχει, εφόσον το αναγνωριστικό του χρήστη περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 403 και μήνυμα που περιέχει «Access Failed...» και «The user does not exist».

**Διεπαφή BaaSAPI-5.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των χρηστών στο BaaS.

**BaaSAPI-5.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των χρηστών επιστρέφονται με επιτυχία. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 52.

```
[{"ActiveProducts": [], "AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat", "Mail": "hmmyforeveryahoo.gr", "Organization": "Admiral", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat10", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral10", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat2", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral2", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat4", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral4", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat6", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral6", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat8", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral8", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "Mat9", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Admiral9", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "alex", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Adidas", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"platformAvramt": {"resourceAval": "resourceAval2", "resourceAVal": "platformAvramz2"}, {"Balance": 920.75, "Fed_omer": true, "Feds": [{"ID": "avram", "Mail": "hmmyforeveryyahoo.gr", "Organization": "CERTH", "Reputation": null, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "george", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Intra", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"vicinity": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ID": "gspadot", "Mail": "gspadodottit.gr", "Organization": "CERTH", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "john", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Nike", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "katerina", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Asia", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "mitsos", "Mail": "hmmyforeveryyahoo.gr", "Organization": "File", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "naoso", "Mail": "hmmyforeveryyahoo.gr", "Organization": "Puma", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"productK1": "productK2"}, {"AssociatedPlatforms": [{"platformVag1": "resourceVag1", "resourceV2": "resourceV3"}, {"Balance": 1062.703839517996, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "vaggelis", "Mail": "hmmyforeveryyahoo.gr", "Organization": "IPTIL", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}], "ActiveProducts": [{"AssociatedPlatforms": [{"platformZ": [{"Balance": 1000, "Fed_omer": false, "Feds": [{"ektat FED": "ID": "zisla", "Mail": "hmmyforeveryyahoo.gr", "Organization": "INET", "Reputation": 0, "Role": "researcher"}, {"doctype": "user"}]}]}}
```

Εικόνα 52 Επιστροφή πληροφοριών όλων των χρηστών του BaaS

**Διεπαφή BaaSAPI-6.** Χρησιμοποιείται για την διαγραφή πλατφόρμας ενός χρήστη στο BaaS

**BaaSAPI-6.A.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή πλατφόρμας ενός χρήστη γίνεται με επιτυχία, οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 53.

```

    ... "platform_id": "vicinity",
    ... "assoc_user_id": "gpapadop"

```

Εικόνα 53 Παράμετροι στο σώμα του ελέγχου BaaSAPI-6.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!» και σε εκτέλεση του ελέγχου BaaAPI-4 για τον χρήστη αυτόν δεν πρέπει να φαίνεται η διαγραμμένη πλατφόρμα.

**BaaSAPI-6.B.** Αυτός ο έλεγχος επαληθεύει εάν η προς διαγραφή πλατφόρμα ενός χρήστη δεν υπάρχει, το σύστημα ενημερώνει κατάλληλα, οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 54.

```
1 {
2   -- "platform_id": "false_platform",
3   -- "assoc_user_id": "gpapadop"
4 }
```

Εικόνα 54 Παράμετροι στο σώμα του ελέγχου BaaSAPI-6.B.

Το αναμενόμενο αποτέλεσμα είναι status: 404 με μήνυμα «Deletion failed, platform does not exist...».

**Διεπαφή BaaSAPI-7.** Χρησιμοποιείται για την διαγραφή όλων των πόρων μιας πλατφόρμας ενός χρήστη στο BaaS

**Διεπαφή BaaSAPI-7.** Χρησιμοποιείται για την διαγραφή ενός πόρου από μια πλατφόρμα ενός χρήστη στο BaaS

**BaaSAPI-7.A.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή πόρου από πλατφόρμα ενός χρήστη γίνεται με επιτυχία, οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 55.

```
{
  ... "platform_id": "vicinity",
  ... "device_id": "device1"
}
```

Εικόνα 55 Παράμετροι στο σώμα του ελέγχου BaaSAPI-7.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!» και σε εκτέλεση του ελέγχου BaaSAPI-4 για τον χρήστη αυτόν δεν πρέπει να φαίνεται η συσκευή μέσα στην αντίστοιχη πλατφόρμα.

**BaaSAPI-7.B.** Αυτός ο έλεγχος επαληθεύει ότι εάν η προς διαγραφή συσκευή δεν υπάρχει στην αντίστοιχη πλατφόρμα, το σύστημα ενημερώνει κατάλληλα. Οι παράμετροι φαίνονται στην Εικόνα 56.

```
1 {
2   ... "platform_id": "vicinity",
3   ... "device_id": "device2"
4 }
```

Εικόνα 56 Παράμετροι στο σώμα του ελέγχου BaaSAPI-7.B.

Το αναμενόμενο αποτέλεσμα είναι status: 404 με μήνυμα «Deletion failed, device not found in platform».

**Διεπαφή BaaSAPI-8.** Χρησιμοποιείται για την διαγραφή ενός χρήστη στο BaaS

**BaaSAPI-8.A.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή ενός χρήστη γίνεται με επιτυχία, οι παρακάτω παράμετροι πρέπει να οριστούν στο σώμα του αιτήματος, όπως φαίνεται στην Εικόνα 57:

```
{
  ... "user_id": "gpadop"
}
```

Εικόνα 57 Παράμετροι στο σώμα του ελέγχου BaaSAPI-8.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!» καθώς και το αντικείμενο που περιγράφει τον χρήστη που διαγράφηκε (Εικόνα 58). Σε εκτέλεση του ελέγχου BaaSAPI-5 δεν πρέπει να φαίνεται ο χρήστης αυτός.

```

{
  "AssociatedPlatforms": {
    "vicinity": []
  },
  "Balance": 0,
  "ID": "gpapadop",
  "Mail": "gpapadop@iti.gr",
  "Organization": "CERTH",
  "Role": "researcher",
  "docType": "user"
}

```

Εικόνα 58 Αντικείμενο διαγραμμένου χρήστη στο BaaS.

**BaaSAPI-8.B.** Αυτός ο έλεγχος επαληθεύει ότι εάν ο προς διαγραφή χρήστης δεν υπάρχει τότε το σύστημα ενημερώνει κατάλληλα. Οι παράμετροι φαίνονται παρακάτω στην Εικόνα 59.

```

{
  "user_id": "unreal_user"
}

```

Εικόνα 59 Παράμετροι στο σώμα του ελέγχου BaaSAPI-8.B.

Το αναμενόμενο αποτέλεσμα είναι status: 400 με μήνυμα που περιέχει «Deletion failed ...» και «The user unreal\_user does not exist».

Πίνακας 8:Περίπτωσης δοκιμής στοιχείου BaaS για την διαχείριση χρηστών.

Interface	Test case	Method	Call	Status
BaaSAPI-1	BaaSAPI-1.A	POST	<BaaS-API-url-user> /register_user	PASS
	BaaSAPI-1.B	POST	<BaaS-API-url-user> / register_user	PASS
BaaSAPI-2	BaaSAPI-2.A	POST	<BaaS-API-url-user> /register_platform	PASS
	BaaSAPI-2.B	POST	<BaaS-API-url-user> /register_platform	PASS
BaaSAPI-3	BaaSAPI-3.A	POST	<BaaS-API-url-user> /register_device	PASS
	BaaSAPI-3.B	POST	<BaaS-API-url-user> /register_device	PASS
BaaSAPI-4	BaaSAPI-4.A.	GET	<BaaS-API-url-user> /get_user_info?user_id =gpapadop	PASS
	BaaSAPI-4.A.	GET	<BaaS-API-url-user> /get_user_info?user_id =gpapadop2	PASS
BaaSAPI-5	BaaSAPI-5.A	GET	<BaaS-API-url-user> /get_all_user_info	PASS
BaaSAPI-6	BaaSAPI-6.A	DELETE	<BaaS-API-url-user> /remove_platform	PASS
	BaaSAPI-6.B	DELETE	<BaaS-API-url-user> /remove_platform	PASS
BaaSAPI-7	BaaSAPI-7.A	DELETE	<BaaS-API-url-user> /remove_device	PASS
	BaaSAPI-7.B	DELETE	<BaaS-API-url-user> /remove_device	PASS
BaaSAPI-8	BaaSAPI-8.A	DELETE	<BaaS-API-url-user> /delete_user	PASS

	BaaS-API-8.B	DELETE	<BaaS-API-url-user> /delete_user	PASS
--	--------------	--------	-------------------------------------	------

### 3.3.2 Στοιχείο λογισμικού για τη διαχείριση ομοσπονδιών

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι περιπτώσεις δοκιμών για το στοιχείο λογισμικού BaaS για τη διαχείριση ομοσπονδιών.

#### 3.3.2.1 Περιγραφή

Το στοιχείο λογισμικού BaaS για τη διαχείριση ομοσπονδιών παρέχει τη δυνατότητα εγγραφής ομοσπονδιών, πρόσβασης σε πληροφορίες αυτών, καθώς και αποχώρησης μελών από αυτές ή εξ ολοκλήρου διαγραφής τους. Επίσης, δίνεται σε χρήστες η δυνατότητα αιτήματος για πρόσκληση ή αφαίρεση μέλους καθώς και αλλαγή κανόνων ομοσπονδίας. Τα αιτήματα αυτά εκκινούν διαδικασίες ψηφοφορίας στις ομοσπονδίες. Για να υποστηριχτούν αυτές οι ψηφοφορίες παρέχεται επιπλέον γραφικό περιβάλλον ώστε ο εκάστοτε ψηφοφόρος να έχει πρόσβαση στην ψηφοφορία και να μπορεί να ψηφίσει.

#### 3.3.2.2 Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Τα παρακάτω χαρακτηριστικά προστέθηκαν στην ενδιάμεση έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- **registerFedToBc** – δημιουργεί την εγγραφή μιας ομοσπονδίας στο BaaS
- **getFedInfo** – πρόσβαση σε πληροφορίες ομοσπονδίας στο BaaS
- **getAllFeds** – πρόσβαση σε πληροφορίες όλων των ομοσπονδιών στο BaaS
- **leaveFed** – αποχώρηση χρήστη από ομοσπονδία στο BaaS
- **deleteFederation** – διαγραφή ομοσπονδίας από το BaaS
- **addFedMemberRequest** – αίτημα προσθήκης μέλους σε ομοσπονδία στο BaaS (από κάτοχο ομοσπονδίας)
- **joinFedRequest** - αίτημα προσθήκης μέλους σε ομοσπονδία στο BaaS (από χρήστη του BaaS)
- **removeFedMemberRequest** – αίτημα αφαίρεσης μέλους από ομοσπονδία στο BaaS
- **updateFedRules** – αίτημα αλλαγής κανόνων σε ομοσπονδία στο BaaS
- **getVotingDescription** – πρόσβαση σε πληροφορίες ψηφοφορίας στο BaaS
- **registerVote** – καταγραφή ψήφου στο BaaS

#### 3.3.2.3 Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος

Το BaaS API για διαχείριση ομοσπονδιών είναι προσβάσιμο μέσω Rest Get/Post/Patch/Delete HTTP αιτήματα στο παρακάτω url:

BaaS-API-url-fed: http://160.40.54.146:3009/baas/federation.

Πίνακας 9: Διεπαφές στοιχείου BaaS.

Inter Face ID	Type	Feature	Endpoint /Queue	Request from	Request to	Release
BaaS-API-9	REST	registerFedToBc	POST <BaaS-API-url-fed>/register_fed	sybmIoTe	BaaS	v2.0
BaaS-API-10	REST	getFedInfo	GET <BaaS-API-url-fed>/get_fed_info	sybmIoTe	BaaS	v1.0
BaaS-API-11	REST	getAllFeds	GET <BaaS-API-url-fed>/get_all_fed_info	sybmIoTe	BaaS	v1.0

BaaSAPI-12	REST	leaveFed	PATCH <BaaS-API-url-fed>/leave_fed	sybmIoTe	BaaS	v1.0
BaaSAPI-13	REST	deleteFederation	DELETE <BaaS-API-url-fed>/delete_fed	sybmIoTe	BaaS	v1.0
BaaSAPI-14	REST	addFedMemberRequest	POST <BaaS-API-url-fed>/add_fed_member_request	sybmIoTe	BaaS	v1.0
BaaSAPI-15	REST	joinFedRequest	POST <BaaS-API-url-fed>/join_request	sybmIoTe	BaaS	v2.0
BaaSAPI-16	REST	removeFedMemberRequest	POST <BaaS-API-url-fed>/remove_fed_member_request	sybmIoTe	BaaS	v1.0
BaaSAPI-17	REST	updateFedRules	POST <BaaS-API-url-fed>/update_fed_rules_request	sybmIoTe	BaaS	v1.0
BaaSAPI-18	REST	getVotingDescription	GET <BaaS-API-url-fed>/get_voting_description	BaaS	ioTFeds user	v1.0
BaaSAPI-19	REST	registerVote	POST <BaaS-API-url-fed>/register_vote	IoTFeds user	BaaS	v1.0

### 3.3.2.4 Δοκιμή και επαλήθευση (Test cases)

Στην ενότητα αυτή περιγράφονται οι περιπτώσεις δοκιμών για τις διεπαφές που περιγράφηκαν παραπάνω.

**Διεπαφή BaaSAPI-11.** Χρησιμοποιείται για την εγγραφή μιας ομοσπονδίας στο BaaS.

**BaaSAPI -9.A.** Αυτός ο έλεγχος χρησιμοποιείται για την επαλήθευση της επιτυχούς εγγραφής μιας ομοσπονδίας στο BaaS. Οι παράμετροι που περνιούνται στο σώμα φαίνονται στην Εικόνα

```

{
  "fed_id": "fed4",
  "creator_id": "gpapadop",
  "inf_model": "BIM",
  "role": "federation",
  "mail": "gpapadop@iti.gr",
  "organization": "CERTH",
  "related_applications": ["platform1", "platform2"],
  "rules": {
    "IoTFedsRules": {
      // ...
    }
  }
}

```

60

αποτέλεσμα είναι status 200 με μήνυμα «OK...»

. Το αναμενόμενο



```

{
  "fed_id": "fed4",
  "creator_id": "gpapadop",
  "inf_model": "BIM",
  "role": "federation",
  "mail": "gpapadop@iti.gr",
  "organization": "CERTH",
  "related_applications": ["platform1", "platform2"],
  "rules": {
    "IoTFedsRules": {
      // ...
    }
  }
}

```

Εικόνα 60: Παράμετροι στο σώμα ελέγχου BaaSAPI-9.A.

**BaaSAPI -9.B.** Αυτός ο έλεγχος επαληθεύει ότι μια ομοσπονδία δε μπορεί να δίπλο-εγγραφεί στο BaaS, οι παράμετροι που πρέπει να οριστούν στο σώμα του αιτήματος είναι ίδιες με τον προηγούμενο έλεγχο. Το αναμενόμενο αποτέλεσμα είναι status: 403 με μήνυμα που περιέχει «Registration failed...» και «The federation with id fed4 already exists»

**Διεπαφή BaaSAPI-10.** Χρησιμοποιείται για την πρόσβαση σε πληροφορίες μίας ομοσπονδίας στο BaaS.

**BaaSAPI-10.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες μιας ομοσπονδίας επιστρέφονται με επιτυχία, εφόσον το αναγνωριστικό της ομοσπονδίας και του δημιουργού της περνούν ως παράμετροι στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 61.

```

{"ID": "fed4", "balance": 0, "creator_id": "gpapadop", "docType": "fed", "inf_model": "BIM", "members_ids": ["gpapadop"], "related_applications": ["platform1", "platform2"], "reputation": 0, "rules": {
  "IoTFedsRules": {
    "BoardGov": {
      "BoardGov": {
        "BoardGov": {
          "Proposals": {
            "ChangeRule": "DeleteFed",
            "InviteMember": "JoinRequest",
            "RequestRemove": "VoteRules",
            "Tokens": 50,
            "Type": "ApprovalPercentage": 75,
            "Base": "Board"
          }
        }
      }
    },
    "FedMarketPlace": {
      "ChargePolicy": "PerProduct",
      "Coin": "FedCoin",
      "FedMarketCharge": 1.05,
      "FedProduct": "Packaging",
      "GlobalMarketCharge": 1.1,
      "ProfitPolicy": "PerSource",
      "FedTypeRules": {
        "DataAvailability": "Hybrid",
        "ServiceType": "Energy, Health, Environment",
        "SupportedOntologies": "Smart Environment, Smart City",
        "Type": "Providers"
      },
      "QualityAssuranceMetrics": {
        "QoSWeights": {
          "business_enablement": 0.1,
          "completeness": 0.1,
          "correctness": 0.1,
          "ease_of_use": 0.1,
          "precision": 0.2,
          "relevance": 0.1,
          "response_time": 0.2,
          "value_for_money": 0.1,
          "QoSPercentage": 60,
          "QoSWeights": {
            "availability": 0.5,
            "precision": 0.2,
            "response_time": 0.3,
            "Quality": {
              "MinValueFed": 6,
              "ReputationPercentage": 40,
              "UnderPerformance": "RequestRemove"
            }
          }
        }
      }
    }
  }
}

```

Εικόνα 61: Επιστροφή πληροφοριών ομοσπονδίας απο το BaaS.

**BaaSAPI-10.B.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφεται το κατάλληλο μήνυμα για ομοσπονδία που δεν υπάρχει, εφόσον το αναγνωριστικό της ομοσπονδίας και του δημιουργού περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 403 και μήνυμα «Access Failed...» και «id does not exist».

**Διεπαφή BaaSAPI-11.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των ομοσπονδιών στο BaaS.

**BaaSAPI-11.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των ομοσπονδιών επιστρέφονται με επιτυχία. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 62.



```
[{"ID": "eketaFED", "balance": 9.50085328650707, "creator_id": "avram", "docType": "fed", "inf_model": "BIM", "members_ids": ["alex", "avram", "george", "john", "katerina", "mitos", "nanos", "vaggellis", "zisis"], "related_applications": [{"platform": "platform1", "platform2": ""}, {"platform": "platform2", "platform1": ""}], "reputation": null, "rules": [{"id": "IoIFedsRules", "name": "IoIFedsRules"}, {"id": "FedGov", "name": "BoardGov"}, {"id": "Proposals", "name": "ChangeRule", "delete": true, "inviteMember": false}, {"id": "JoinRequest", "name": "RequestRemove"}, {"id": "VoteRules", "name": "Tokens": 50, "type": "ApprovalPercentage": 75, "base": "Board"}], "FedMarketPlace": {"ChargePolicy": "PerProduct", "Coin": "FedCoin", "FedMarketCharge": 1.05, "FedProduct": "Packaging", "GlobalMarketCharge": 1.1, "ProfitPolicy": "PerSource", "FedTypeRules": {"DataAvailability": "Hybrid", "ServiceType": "Energy, Health, Environment", "used_on": "Smart Environment, Smart City, Type": "Providers", "QualityAssuranceMetrics": {"QoEWeights": {"business_enablement": 0.1, "completeness": 0.1, "correctness": 0.1, "ease_of_use": 0.1, "precision": 0.2, "relevance": 0.1, "response_time": 0.2, "value_for_money": 0.1}, "QoSPercentage": 60, "QoSWeights": {"availability": 0.5, "precision": 0.2, "response_time": 0.3}, "Quality": {"MinValueFed": 6, "ReputationPercentage": 40, "UnderPerformance": "RequestRemove"}}}}, {"ID": "fed4", "balance": 0.0, "creator_id": "gapadapop", "docType": "fed", "inf_model": "BIM", "members_ids": ["gapadapop"], "related_applications": [{"platform": "platform1", "platform2": ""}, {"platform": "platform2", "platform1": ""}], "reputation": 0, "rules": [{"id": "IoIFedsRules", "name": "IoIFedsRules"}, {"id": "FedGov", "name": "BoardGov"}, {"id": "Proposals", "name": "ChangeRule", "delete": true, "inviteMember": false}, {"id": "JoinRequest", "name": "RequestRemove"}, {"id": "VoteRules", "name": "Tokens": 50, "type": "ApprovalPercentage": 75, "base": "Board"}], "FedMarketPlace": {"ChargePolicy": "PerProduct", "Coin": "FedCoin", "FedMarketCharge": 1.05, "FedProduct": "Packaging", "GlobalMarketCharge": 1.1, "ProfitPolicy": "PerSource", "FedTypeRules": {"DataAvailability": "Hybrid", "ServiceType": "Energy, Health, Environment", "used_on": "Smart Environment, Smart City, Type": "Providers", "QualityAssuranceMetrics": {"QoEWeights": {"business_enablement": 0.1, "completeness": 0.1, "correctness": 0.1, "ease_of_use": 0.1, "precision": 0.2, "relevance": 0.1, "response_time": 0.2, "value_for_money": 0.1}, "QoSPercentage": 60, "QoSWeights": {"availability": 0.5, "precision": 0.2, "response_time": 0.3}, "Quality": {"MinValueFed": 6, "ReputationPercentage": 40, "UnderPerformance": "RequestRemove"}}}]}
```

Εικόνα 62: Επιστροφή πληροφοριών όλων των ομοσπονδιών απο το BaaS.

**Διεπαφή BaaSAPI-12.** Χρησιμοποιείται για την αποχώρηση μέλους από ομοσπονδία στο BaaS. Για τους ελέγχους αυτής της διεπαφής δημιουργείται ομοσπονδία με id fed4 από τον χρήστη gpadop στην οποία προσθέτουμε τον χρήστη evathana.

**BaaSAPI-12.A.** Αυτός ο έλεγχος επαληθεύει ότι μέλος μιας ομοσπονδίας μπορεί να αποχωρήσει με επιτυχία. Οι παράμετροι που περνιούνται στο σώμα είναι αυτοί στην Εικόνα 63 **Error! Reference source not found.** Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 61.

```

    ... "user_id": "evathana",
    ... "fed_id": "fed4"

```

Εικόνα 63: Παράμετροι στο σώμα του ελέγχου BaaSAPI-12.A.

**BaaSAPI-12.B.** Αυτός ο έλεγχος επαληθεύει ότι δε μπορεί να αφαιρεθεί από ομοσπονδία χρήστης που δεν υπάρχει σε αυτήν. Το σώμα του ελέγχου είναι ίδιο με αυτό του προηγούμενου ελέγχου, όπου αυτή τη φορά ο προς αφαίρεση χρήστης δεν υπάρχει στην ομοσπονδία ως μέλος. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «User leaving failed ...» και «The user is not registered in federation fed4»

**BaaSAPI-12.Γ.** Αυτός ο έλεγχος επαληθεύει ότι δε μπορεί να αποχωρήσει από μια ομοσπονδία ο δημιουργός της. Οι παράμετροι που περνιούνται στο σώμα είναι αυτοί στην Εικόνα 64. Το αναμενόμενο αποτέλεσμα είναι status 400 και μήνυμα που περιέχει «User leaving failed» και «The creator of the federation is not allowed to leave it».

```

-- "user_id": "gpapadop",
-- "fed_id": "fed4"

```

Εικόνα 64: Παράμετροι στο σώμα του ελέγχου BaaSAPI-12.Γ.

**Διεπαφή BaaSAPI-13.** Χρησιμοποιείται για την διαγραφή μιας ομοσπονδίας στο BaaS. Για τους ελέγχους αυτής της διεπαφής χρησιμοποιείται η ομοσπονδία με id fed4 η οποία δημιουργήθηκε σε προηγούμενο βήμα, από τον χρήστη gnapador

**BaaSAPI-13.A.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή μιας ομοσπονδίας δεν μπορεί να γίνει παρά μόνο από τον δημιουργό της. Οι παράμετροι είναι αυτές στην Εικόνα 65. Το αναμενόμενο αποτέλεσμα είναι status: 400 με μήνυμα που περιέχει «Deletion failed ...» και «Only the creator of the federation can delete it».

```

{
  ... "fed_id": "fed4",
  ... "request_user_id": "RandomText"
}

```

Εικόνα 65: Παράμετροι στο σώμα του ελέγχου BaaSAPI-13.A.

**BaaSAPI-13.B.** Αυτός ο έλεγχος επαληθεύει ότι εάν η ομοσπονδία περιέχει μέλη δε μπορεί να διαγραφεί. Εγγράφουμε στην ομοσπονδία fedtest2 τον χρήστη icom2 και καλούμε την διεπαφή με τις παραμέτρους στην Εικόνα 66 **Error! Reference source not found..** Το αναμενόμενο αποτέλεσμα είναι μήνυμα που περιέχει «Deletion failed» και «This federation still contains members and cannot be deleted».

```

{
  ... "fed_id": "fed4",
  ... "request_user_id": "gpapadop"
}

```

Εικόνα 66: Παράμετροι στο σώμα του ελέγχου BaaSAPI-13.B.

**BaaSAPI-13.Γ.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή μιας ομοσπονδίας γίνεται με επιτυχία, οι παράμετροι που πρέπει να οριστούν στο σώμα του αιτήματος είναι ίδιες με την στην Εικόνα 76. αφού διαγράψουμε τον χρήστη icom2 από την ομοσπονδία. Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!». Σε εκτέλεση του ελέγχου BaaSAPI-10 δεν πρέπει να φαίνεται η ομοσπονδία αυτή.

**BaaSAPI-13.Δ.** Αυτός ο έλεγχος επαληθεύει ότι εάν η προς διαγραφή ομοσπονδία δεν υπάρχει, το σύστημα ενημερώνει κατάλληλα. Οι παράμετροι είναι ξανά ίδιοι με αυτές στην Εικόνα 66. Το αναμενόμενο αποτέλεσμα είναι status: 400 με μήνυμα που περιέχει «Deletion failed ...» και «id is not defined».

**Διεπαφή BaaSAPI-14.** Χρησιμοποιείται για την έναρξη ψηφοφορίας για την προσθήκη νέου μέλους σε ομοσπονδία στο BaaS από τον δημιουργό της ομοσπονδίας. Η πραγματοποίηση των ελέγχων γίνεται στην ομοσπονδία «fed2» με μοναδικό χρήστη «spolymeni».

**BaaSAPI-14.A.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η έναρξη ψηφοφορίας για προσθήκη χρήστη που υπάρχει ήδη στην ομοσπονδία. Οι παράμετροι στο σώμα του ελέγχου είναι αυτοί στην Εικόνα 67. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «Add new federation member request failed ...» και «This member is already a member of this federation».

```

{
  ... "fed_id": "fed2",
  ... "requestor_id": "spolymeni",
  ... "member_id": "spolymeni"
}

```

Εικόνα 67: Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.A.

**BaaSAPI-14.B.** Αυτός ο έλεγχος επαληθεύει ότι η έναρξη ψηφοφορίας για προσθήκη χρήστη σε ομοσπονδία, ο οποίος είναι ήδη εγγεγραμμένος στο BaaS λειτουργεί κανονικά. Οι παράμετροι στο σώμα του ελέγχου είναι αυτοί στην Εικόνα 68 **Error! Reference source not found..** Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 69, με πληροφορίες της ψηφοφορίας.

```

{
  ... "fed_id": "fed2",
  ... "requestor_id": "spolymeni",
  ... "member_id": "gspanos"
}

```

Εικόνα 68 Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.B.

```

{
  "ID": "1igowujn4c33icchr38twu",
  "docType": "voting",
  "federation": "fed2",
  "descr": {
    "votingType": "addition",
    "requestorID": "Dimos Kalamarias",
    "memberID": "CERTH",
    "fedID": "fed2"
  },
  "status": true,
  "startDate": 1673274890022,
  "endDate": 1673361290022,
  "votes": {
    "spolymeni": "pending"
  }
}

```

Εικόνα 69: Αντικείμενο που επιστρέφεται από τον έλεγχο BaaSAPI-14.B.

**BaaSAPI-14.Γ.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η έναρξη δεύτερης ψηφοφορίας για μια ομοσπονδία πριν λήξει η πρώτη. Οι παράμετροι στο σώμα του ελέγχου είναι οι παρακάτω:

```

{
  ... "fed_id": "fed2",
  ... "requestor_id": "spolymeni",
  ... "member_id": "evathana"
}

```

Εικόνα 70: Παράμετροι στο σώμα του ελέγχου BaaSAPI-14.Γ.

Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «Add new federation member request failed ...» και «There is already an ongoing voting for this federation».

**Διεπαφή BaaSAPI-15.** Χρησιμοποιείται για την έναρξη ψηφοφορίας για την προσθήκη νέου μέλους σε ομοσπονδία στο BaaS από το υποψήφιο μέλος της ομοσπονδίας. Η πραγματοποίηση των ελέγχων γίνεται στην ομοσπονδία «fed2» με μοναδικό χρήστη «spolymeni».

**BaaSAPI-15.A.** Αυτός ο έλεγχος επαληθεύει ότι η έναρξη ψηφοφορίας για προσθήκη χρήστη σε ομοσπονδία, ο οποίος είναι ήδη εγγεγραμμένος στο BaaS λειτουργεί κανονικά. Οι παράμετροι στο σώμα του ελέγχου είναι αυτοί στην Εικόνα 71. Αν ο χρήστης υπάρχει ήδη στην ομοσπονδία, το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «Join federation member request failed ...» και «This member is already a member of this federation». Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 70, με πληροφορίες της ψηφοφορίας.

```

{
  ... "fed_id": "fed2",
  ... "member_id": "spolymeni"
}

```

Εικόνα 71 Παράμετροι στο σώμα του ελέγχου BaaSAPI-15.A.

**BaaSAPI-15.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η έναρξη δεύτερης ψηφοφορίας για μια ομοσπονδία πριν λήξει η πρώτη. Οι παράμετροι στο σώμα του ελέγχου είναι αυτοί στην *Εικόνα 71*. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «Join new federation member request failed ...» και «There is already an ongoing voting for this federation».

**Διεπαφή BaaSAPI-16.** Χρησιμοποιείται για την έναρξη ψηφοφορίας για την αφαίρεση μέλους από ομοσπονδία στο BaaS. Η πραγματοποίηση των ελέγχων γίνεται στην ομοσπονδία «fed1» με χρήστες «spolymeni», «evathana» και «gsrpanos».

**BaaSAPI-16.A.** Αυτός ο έλεγχος επαληθεύει ότι η έναρξη ψηφοφορίας προτεινόμενη από μέλος της, για αφαίρεση χρήστη, ο οποίος είναι ήδη μέλος της ομοσπονδίας λειτουργεί κανονικά. Οι παράμετροι στο σώμα του ελέγχου φαίνονται παρακάτω, στην *Εικόνα 72* όπου ο χρήστης «spolymeni» προτείνει την αφαίρεση του χρήστη «gsrpanos». Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 73*, με πληροφορίες της ψηφοφορίας, όπως φαίνεται το αντικείμενο αυτό που επιστρέφεται, δεν εκθέτει τα IDs των οργανισμών αλλά μόνο τα ονόματά τους.

```

{
  "fed_id": "fed1",
  "requestor_id": "spolymeni",
  "user_id": "gsrpanos"
}

```

Εικόνα 72: Παράμετροι στο σώμα του ελέγχου BaaSAPI-16.A.

```

{
  "ID": "vksqymf8cbhvyq5wk9xosk",
  "docType": "voting",
  "federation": "fed1",
  "descr": {
    "votingType": "removal",
    "requestorID": "Dimos Kalamarias",
    "memberID": "CERTH",
    "fedID": "fed1"
  },
  "status": true,
  "startDate": 1673879446220,
  "endDate": 1673965846220,
  "votes": {
    "evathana": "pending",
    "spolymeni": "pending"
  }
}

```

Εικόνα 73: Αντικείμενο που επιστρέφεται από τον έλεγχο BaaSAPI-16.A.

**BaaSAPI-16.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η έναρξη ψηφοφορίας για αφαίρεση μέλους που δεν υπάρχει ήδη στην ομοσπονδία. Οι παράμετροι στο σώμα του ελέγχου φαίνονται παρακάτω. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα « Request to remove a member from the federation failed » και «This user is not a member of this federation».

```

{
  "fed_id": "fed1",
  "requestor_id": "spolymeni",
  "user_id": "gsrpadop"
}

```

Εικόνα 74: Παράμετροι στο σώμα του ελέγχου BaaSAPI-16.B.

**Διεπαφή BaaSAPI-17.** Χρησιμοποιείται για την έναρξη ψηφοφορίας για την αλλαγή κανόνων σε ομοσπονδία στο BaaS. Η πραγματοποίηση των ελέγχων γίνεται στην ομοσπονδία «fed1» με αρχικό σετ κανόνων αυτό στην *Εικόνα 75*.

```

"rules": {
  "IoTFedsRules": {
    "FedGov": {
      "BoardGov": [
        "gpapadop"
      ],
      "Proposals": [
        "ChangeRule",
        "DeleteFed",
        "InviteMember",
        "JoinRequest",
        "RequestRemove"
      ],
      "VoteRules": {
        "Tokens": 50,
        "Type": {
          "ApprovalPercentage": 75,
          "Base": "Board"
        }
      }
    },
    "FedMarketplace": {
      "ChargePolicy": "PerProduct",
      "Coin": "FedCoin",
      "FedMarketCharge": 1.05,
      "FedProduct": "Packaging",
      "GlobalMarketCharge": 1.1,
      "ProfitPolicy": "PerSource"
    },
    "FedTypeRules": {
      "DataAvailability": "Hybrid",
      "ServiceType": "Energy, Health, Environment",
      "SupportedOntologies": "Smart Environment, Smart City",
      "Type": "Providers"
    },
    "QualityAssuranceMetrics": {
      "QoEWeights": {
        "business_enablement": 0.1,
        "completeness": 0.1,
        "correctness": 0.1,
        "ease_of_use": 0.1,
        "precision": 0.2,
        "relevance": 0.1,
        "response_time": 0.2,
        "value_for_money": 0.1
      },
      "QoSPercentage": 60,
      "QoSWeights": {
        "availability": 0.5,
        "precision": 0.2,
        "response_time": 0.3
      },
      "Quality": {
        "MinValueFed": 6
      },
      "ReputationPercentage": 40,
      "UnderPerformance": "RequestRemove"
    }
  }
}

```

Εικόνα 75: Αρχικό σετ κανόνων της ομοσπονδίας που χρησιμοποιείται.

**BaaSAPI-17.A.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η έναρξη ψηφοφορίας εάν οι κανόνες που προτείνονται είναι ίδιοι με τους υπάρχοντες. Οι παράμετροι στο σώμα του ελέγχου είναι ακριβώς αυτοί στην *Εικόνα 75*. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει « Request to update federation rules failed ... » και «Proposed rules are same as existing ones».

**BaaSAPI-17.B.** Αυτός ο έλεγχος επαληθεύει ότι η έναρξη ψηφοφορίας για αλλαγή κανόνων σε ομοσπονδία, λειτουργεί κανονικά. Μπορούμε να το δοκιμάσουμε προτείνοντας αλλαγή στο πεδίο "BoardGov" αφαιρώντας τον χρήστη «spolymeni». Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο όπως το παρακάτω με πληροφορίες της ψηφοφορίας.

```

{
  "ID": "n2nfb8iaf7je3w73zadky",
  "docType": "voting",
  "federation": "fed1",
  "descr": {
    "votingType": "rule change",
    "requestorID": "Dimos Thermis",
    "proposedRules": {
      "changedField": "BoardGov",
      "proposed": [
        "evathana",
        "gspanos"
      ],
      "current": [
        "evathana",
        "gspanos",
        "spolymeni"
      ]
    }
  },
  "fedID": "fed1"
},
{
  "status": true,
  "startDate": 1673880787931,
  "endDate": 1673967187931,
  "votes": {
    "evathana": "pending",
    "gspanos": "pending",
    "spolymeni": "pending"
  }
}

```

*Εικόνα 76: Αναμενόμενο αποτέλεσμα επιστροφής του ελέγχου BaaSAPI-17.B.*

**Διεπαφή BaaSAPI-18.** Χρησιμοποιείται για την πρόσβαση σε πληροφορίες μιας ψηφοφορίας από έναν χρήστη.

**BaaSAPI-18.A.** Αυτός ο έλεγχος επαληθεύει ότι πληροφορίες για μια ψηφοφορία είναι προσβάσιμες από έναν χρήστη με το token που εκδίδεται για αυτόν από τα endpoints που εκκινούν την διαδικασία ψηφοφορίας. Το token αυτό παρέχεται σε κάθε μέλος της ομοσπονδίας που έχει δικαίωμα ψήφου (BoardGov) και είναι ένα Json Web Token (JWT) που περνιέται ως παράμετρος στο url του αιτήματος και εφόσον είναι έγκυρο επιστρέφεται αντικείμενο με περιγραφή της ψηφοφορίας. Μετά την ψήφο του χρήστη το token αυτό εξαργυρώνεται. Για παράδειγμα για την ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-18.B. το αντικείμενο θα είναι αυτό στην παρακάτω εικόνα. Αξίζει να σημειωθεί πως στην κανονική ροή του συστήματος, η κλήση αυτής της διεπαφής γίνεται αυτόματα κατά το κλικ του συνδέσμου που λαμβάνει ο χρήστης στο e-mail του προκειμένου να ψηφίσει.



```

{
  "fedID": "fed1",
  "proposedRules": {
    "changedField": "BoardGov",
    "current": [
      "evathana",
      "gspanos",
      "spolymeni"
    ],
    "proposed": [
      "evathana",
      "gspanos"
    ]
  },
  "requestorID": "Dimos Thermis",
  "votingType": "rule change"
}

```

Εικόνα 77: Αντικείμενο που περιγράφει την ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-17.B.

**BaaSAPI-18.B.** Αυτός ο έλεγχος επαληθεύει ότι η διεπαφή δε μπορεί να δώσει αποτέλεσμα αν το token δεν είναι έγκυρο. Καλούμε τη διεπαφή χωρίς να θέσουμε τιμή στην παράμετρο url του token. Το αναμενόμενο αποτέλεσμα είναι status 403 με μήνυμα «Invalid token».

**Διεπαφή BaaSAPI- 19.** Χρησιμοποιείται για την καταχώρηση ψήφου ενός χρήστη σε μια ψηφοφορία

**BaaSAPI-19.A.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης έχει δικαίωμα ψήφου σε μια ψηφοφορία μέσω του token του. Για παράδειγμα για την ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-17.B. για θετική ψήφο του χρήστη «evathana», οι παράμετροι θα είναι αυτοί στην **Εικόνα 78****Error! Reference source not found..** Αξίζει να σημειωθεί πως στην κανονική ροή του συστήματος, η κλήση αυτής της διεπαφής γίνεται αυτόματα κατά το κλικ του πλήκτρου submit (μετά την επιλογή ψήφου) στην σελίδα που βλέπει ο χρήστης εφόσον επαληθευτεί το token του.

```

{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ3RHZvdGluZyI6Im4ybWZ1OG1hZjdgZTN3NzN6YWRreSIsIklEdm90ZXIiOiJldmF0aGFuYSIsIm1hdCI6MTY3Mzg4MDc5MCwiZXhwIjoxNjc2OTY3MTkwfQ.NdU6Wimr1gKuk00noDX-nzm7iFb8Uad2PW6qLL8AdU4",
  "vote": "yes"
}

```

Εικόνα 78: Παράμετροι στο σώμα του ελέγχου BaaSAPI-20.A για καταχώρηση ψήφου στην ψηφοφορία που εκκίνησε στον έλεγχο BaaSAPI-18.B.

**BaaSAPI-19.B.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης δεν μπορεί να ψηφίσει δυο φορές για την ίδια ψηφοφορία. Οι παράμετροι στο σώμα είναι ίδιες με αυτές του προηγούμενου ελέγχου. Το αναμενόμενο αποτέλεσμα είναι status 400 με μήνυμα που περιέχει «Register vote of user failed» και «Voter evathana has already voted and cannot vote again».

Πίνακας 10: Περίπτωσης δοκιμής στοιχείου BaaS για τη διαχείριση ομοσπονδιών.

Interface	Test case	Method	Call	Status
BaaSAPI-9	BaaSAPI-9.A	POST	<BaaS-API-url-fed>/register_fed	PASS
	BaaSAPI-9.B	POST	<BaaS-API-url-fed>/register_fed	PASS
BaaSAPI-10	BaaSAPI-10.A	GET	<BaaS-API-url-fed>/get_fed_info?creator_id=evathana&fed_id=fed4	PASS
	BaaSAPI-10.B	GET	<BaaS-API-url-fed>/get_fed_info?creat	PASS

			or_id=evathana&fed_id=fed6	
BaaSAPI-11	BaaSAPI-11.A	GET	<BaaS-API-url-fed>/get_all_fed_info	PASS
BaaSAPI-12	BaaSAPI-12.A	PATCH	<BaaS-API-url-fed>/leave_fed	PASS
	BaaSAPI-12.B	PATCH	<BaaS-API-url-fed>/leave_fed	PASS
	BaaSAPI-12.Γ	PATCH	<BaaS-API-url-fed>/leave_fed	PASS
BaaSAPI-13	BaaSAPI-13.A	DELETE	<BaaS-API-url-fed>/delete_fed/	PASS
	BaaSAPI-13.B	DELETE	<BaaS-API-url-fed>/delete_fed/	PASS
	BaaSAPI-13.Γ	DELETE	<BaaS-API-url-fed>/delete_fed/	PASS
	BaaSAPI-13.Δ	DELETE	<BaaS-API-url-fed>/delete_fed/	PASS
BaaSAPI-14	BaaSAPI-14.A	POST	<BaaS-API-url-fed>/new_member_request	PASS
	BaaSAPI-14.B	POST	<BaaS-API-url-fed>/new_member_request	PASS
	BaaSAPI-14.B	POST	<BaaS-API-url-fed>/new_member_request	PASS
BaaSAPI-15	BaaSAPI-15.A	POST	<BaaS-API-url-fed>/join_member_request	PASS
	BaaSAPI-15.B	POST	<BaaS-API-url-fed>/join_member_request	PASS
BaaSAPI-16	BaaSAPI-16.A	POST	<BaaS-API-url-fed>/remove_member_request	PASS
	BaaSAPI-16.B	POST	<BaaS-API-url-fed>/remove_member_request	PASS
BaaSAPI-17	BaaSAPI-17.A	POST	<BaaS-API-url-fed>/update_fed_rules_request	PASS
	BaaSAPI-17.B	POST	<BaaS-API-url-fed>/update_fed_rules_request	PASS
BaaSAPI-18	BaaSAPI-18.A	GET	<BaaS-API-url-fed>/get_voting_description	PASS
	BaaSAPI-18.B	GET	<BaaS-API-url-fed>/get_voting_description	PASS
BaaSAPI-19	BaaSAPI-19.A	POST	<BaaS-API-url-fed>/voting/vote	PASS
	BaaSAPI-19.B	POST	<BaaS-API-url-fed>/voting/vote	PASS

### 3.3.3 Στοιχείο λογισμικού αγοράς

Στην ενότητα αυτή περιγράφονται τα χαρακτηριστικά, οι διεπαφές και οι περιπτώσεις δοκιμών για το στοιχείο λογισμικού BaaS για την αγορά προϊόντων.

#### 3.3.3.1 Περιγραφή

Το στοιχείο λογισμικού BaaS για την αγορά προϊόντων παρέχει την δυνατότητα σε χρήστες να αναζητήσουν και να αγοράσουν προϊόντα είτε στα πλαίσια ομόσπονδης είτε καθολικής αγοράς,



επίσης παρέχεται η δυνατότητα ανταλλαγής προϊόντων και αξιολόγησης μέσω της οποίας υπολογίζονται βαθμοί φήμης που εξυπηρετούν τις διαδικασίες αναζήτησης.

### 3.3.3.2 *Χαρακτηριστικά που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος*

Τα παρακάτω χαρακτηριστικά προστέθηκαν στην ενδιάμεση έκδοση του ολοκληρωμένου συστήματος της IoTFeds πλατφόρμας:

- **getProduct** – πρόσβαση σε στοιχεία ενός προϊόντος στο BaaS.
- **getResource** – πρόσβαση σε στοιχεία ενός πόρου δεδομένων στο BaaS.
- **getAllProducts** – πρόσβαση σε στοιχεία όλων των προϊόντων στο BaaS.
- **getAllResources** – πρόσβαση σε στοιχεία όλων των πόρων δεδομένων στο BaaS.
- **getUserTokens** – πρόσβαση σε όλα τα κουπόνια πρόσβασης ενός χρήστη από τον ίδιο
- **getAllExchangableTokens** – πρόσβαση σε όλα τα κουπόνια προϊόντων που έχουν τεθεί προς ανταλλαγή
- **AddResourceToFed** – κοινοποίηση ενός πόρου σε μία ομοσπονδία στο BaaS.
- **getFederatedResources** – πρόσβαση σε στοιχεία όλων των πόρων δεδομένων σε μία ομοσπονδία στο BaaS.
- **getFederatedProducts** – πρόσβαση σε στοιχεία όλων των προϊόντων σε μία ομοσπονδία στο BaaS.
- **getGlobalProducts** – πρόσβαση σε στοιχεία όλων των προϊόντων της καθολικής αγοράς στο BaaS.
- **searchResources** – αναζήτηση και διαφήμιση πόρων δεδομένων
- **searchProducts** – αναζήτηση και διαφήμιση προϊόντων δεδομένων
- **createProduct** – δημιουργία προϊόντος στο BaaS.
- **deleteProduct** – διαγραφή προϊόντος στο BaaS.
- **calculateProductPrice** – δυναμικός υπολογισμός τιμής ενός προϊόντος στο BaaS.
- **buyProduct** – αγορά προϊόντος στο BaaS.
- **checkAccess** – έλεγχος πρόσβασης σε προϊόν δεδομένων
- **putTokenForExchange** – προσφορά ενός κουπονιού πρόσβασης για ανταλλαγή του προϊόντος
- **removeTokenFromExchange** – αφαίρεση κουπονιού πρόσβασης από την προσφορά για ανταλλαγή προϊόντος
- **exchangeTokens** – ανταλλαγή κουπονιών πρόσβασης για ανταλλαγή προϊόντων
- **rateProduct** – αξιολόγηση ενός προϊόντος από τον χρήστη
- **updateAllReputations** – ενημέρωση της φήμης όλων των οντοτήτων
- **updateResourceQos** – ενημέρωση των μετρικών ποιότητας και αντικειμενικής φήμης μιας συσκευής (πόρου)
- **getReceiptsSeller** – ανάκτηση αποδείξεων συναλλαγών από τον πωλητή
- **getReceiptsBuyer** – ανάκτηση αποδείξεων συναλλαγών από τον αγοραστή
- **getReceiptsAdmin** – ανάκτηση αποδείξεων συναλλαγών από τον διαχειριστή του συστήματος
- **verifyReceipt** – έλεγχος εγκυρότητας απόδειξης συναλλαγής
- **getAccessLogs** – ανάκτηση καταγραφών χρήσης προϊόντων από τον διαχειριστή του συστήματος
- **verifyAccessLog** – έλεγχος εγκυρότητας καταγραφής χρήσης προϊόντος από τον διαχειριστή του συστήματος

### 3.3.3.3 *Διεπαφές που συμπεριλαμβάνονται στην τρέχουσα έκδοση του ολοκληρωμένου συστήματος*

Το BaaS API για αγορά προϊόντων είναι προσβάσιμο μέσω των Rest Get/Post/Patch/Delete HTTP αιτήματα στο παρακάτω uri:

BaaS-API-url-products: <http://160.40.54.146:3009/baas/products>.

Πίνακας 11: Διεπαφές στοιχείου BaaS για την αγορά.

Inter Face ID	Type	Feature	Endpoint /Queue	Request from	Request to	Release
BaaS- PI-20	RES T	getProduct	GET < BaaS-API-url- products>/get_product_info	Marketpl ace	BaaS	v2.0
BaaS- PI-21	RES T	getResource	GET < BaaS-API-url- products>/get_resource_info	Marketpl ace	BaaS	v2.0
BaaS- PI-22	RES T	getAllProducts	GET < BaaS-API-url- products>/get_all_product_info	Marketpl ace	BaaS	v2.0
BaaS- PI-23	RES T	getAllResources	GET < BaaS-API-url- products>/get_all_resource_info	SymbIoT e	BaaS	v2.0
BaaS- PI-24	RES T	getUserTokens	GET < BaaS-API-url- products>/get_user_tokens	Marketpl ace	BaaS	v2.0
BaaS- PI-25	RES T	getAllExchangableTo kens	GET < BaaS-API-url- products>/get_all_exchangable_ tokens	Marketpl ace	BaaS	v2.0
BaaS- PI-26	RES T	AddResourceToFed	PATCH < BaaS-API-url- products>/add_resource_fed	symbIoT e	BaaS	v2.0
BaaS- PI-27	RES T	getFederatedResource s	POST < BaaS-API-url- products>/get_federated_resourc es	Marketpl ace	BaaS	v2.0
BaaS- PI-28	RES T	getFederatedProducts	GET < BaaS-API-url- products>/get_federated_produc ts	Marketpl ace	BaaS	v2.0
BaaS- PI-29	RES T	getGlobalProducts	GET < BaaS-API-url- products>/get_global_products	Marketpl ace	BaaS	v2.0
BaaS- PI-30	RES T	searchResources	POST < BaaS-API-url- products>/search_resources	Marketpl ace	BaaS	v2.0
BaaS- PI-31	RES T	searchProducts	POST < BaaS-API-url- products>/search_products	Marketpl ace	BaaS	v2.0
BaaS- PI-32	RES T	createProduct	POST < BaaS-API-url- products>/create_product	Marketpl ace	BaaS	v2.0
BaaS- PI-33	RES T	deleteProduct	DELETE < BaaS-API-url- products>/delete_product	Marketpl ace	BaaS	v2.0
BaaS- PI-34	RES T	calculateProductPrice	POST < BaaS-API-url- products>/calc_product_price	Marketpl ace	BaaS	v2.0
BaaS- PI-35	RES T	buyProduct	POST < BaaS-API-url- products>/buy_product	Marketpl ace	BaaS	v2.0
BaaS- PI-36	RES T	checkAccess	POST < BaaS-API-url- products>/check_access	Marketpl ace	BaaS	v2.0
BaaS- PI-37	RES T	putTokenForExchange	PATCH < BaaS-API-url- products>/put_token_for_excha nge	Marketpl ace	BaaS	v2.0
BaaS- PI-38	RES T	removeTokenFromEx change	PATCH < BaaS-API-url- products>/remove_token_from_ exchange	Marketpl ace	BaaS	v2.0
BaaS- PI-39	RES T	exchangeTokens	PATCH < BaaS-API-url- products>/exchange_tokens	Marketpl ace	BaaS	v2.0
BaaS- PI-40	RES T	rateProduct	PATCH < BaaS-API-url- products>/rate_product	Marketpl ace	BaaS	v2.0
BaaS- PI-41	RES T	updateAllReputations	PATCH < BaaS-API-url- products>/update_all_reputation s	Marketpl ace	BaaS	v2.0
BaaS- PI-42	RES T	updateResourceQos	PATCH < BaaS-API-url- products>/update_resource_qos	Marketpl ace	BaaS	v2.0
BaaS- PI-43	RES T	getReceiptsSeller	POST < BaaS-API-url- products>/get_receipts_seller	Marketpl ace	BaaS	v2.0
BaaS- PI-44	RES T	getReceiptsBuyer	POST < BaaS-API-url- products>/get_receipts_buyer	Marketpl ace	BaaS	v2.0
BaaS- PI-45	RES T	getReceiptsAdmin	POST < BaaS-API-url- products>/get_receipts_admin	Marketpl ace	BaaS	v2.0
BaaS- PI-46	RES T	verifyReceipts	POST < BaaS-API-url- products>/verify_receipts	Marketpl ace	BaaS	v2.0

BaaS PI-47	RES T	getLogs	POST < BaaS-API-url- products>/get_logs	Marketpl ace	BaaS	v2.0
BaaS PI-48	RES T	verifyAccessLogs	POST < BaaS-API-url- products>/verify_access_logs	Marketpl ace	BaaS	v2.0

### 3.3.3.4 Δοκιμή και επαληθευση (Test cases)

Στην ενότητα αυτή περιγράφονται οι περιπτώσεις δοκιμών για τις διεπαφές που περιγράφηκαν παραπάνω.

#### 3.3.3.4.1 Δοκιμή διεπαφών αγοράς

**Διεπαφή BaaSAPI-20.** Χρησιμοποιείται για την πρόσβαση σε πληροφορίες ενός προϊόντος στο BaaS.

**BaaSAPI-20.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες ενός προϊόντος επιστρέφονται με επιτυχία, εφόσον το αναγνωριστικό του προϊόντος περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 79*.

```

{
  "FedMarketplace_id": "eketaFED",
  "GlobalMarketplace_id": true,
  "Price": 0.8,
  "Product_details": {},
  "Product_id": "productX2",
  "Reputation": 0,
  "Resource_ids": {
    "resourceAv1": "platformAvraam1",
    "resourceV1": "platformVag1"
  },
  "Seller": "vaggelis",
  "docType": "Product",
  "subjReputation": [],
  "transactionCounter": 1
}

```

Εικόνα 79: Επιστροφή πληροφοριών προϊόντος απο το BaaS.

**BaaSAPI-20.B.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφεται το κατάλληλο μήνυμα για προϊόν που δεν υπάρχει, εφόσον το αναγνωριστικό του προϊόντος περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 403 και μήνυμα που περιέχει «Get product information failed» και «The asset with id product\_id does not exist».

**Διεπαφή BaaSAPI-21.** Χρησιμοποιείται για την πρόσβαση σε πληροφορίες ενός πόρου δεδομένων στο BaaS.

**BaaSAPI-21.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες ενός πόρου δεδομένων επιστρέφονται με επιτυχία, εφόσον το αναγνωριστικό του πόρου περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 80*.

```

    "Brd": 2,
    "Brr": 12,
    "Fr": 5,
    "Hr": 1701257450100,
    "Lr": 1692207050100,
    "Platform": "platformAvraam1",
    "Price": 0.3,
    "Resource_id": "resourceAv1",
    "Wr": 1440,
    "ard": 0.001,
    "arf": 0.05,
    "aro": 1.05,
    "arr": 1.2,
    "docType": "Resource",
    "fed_id": [
      "eketaFED"
    ],
    "fr": 1,
    "objReputation": 0,
    "overallReputation": null,
    "owner": "avraam",
    "relatedFeds": [
      "eketaFED"
    ],
    "subjReputation": 0,
    "transactionCounter": 1
  }
}

```

Εικόνα 80: Επιστροφή πληροφοριών πόρου δεδομένων απο το BaaS.

**BaaSAPI-21.B.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφεται το κατάλληλο μήνυμα για πόρο δεδομένων που δεν υπάρχει, εφόσον το αναγνωριστικό του πόρου περνιέται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 403 και μήνυμα που περιέχει «Get resource information failed» και «The asset with id resource\_id does not exist».

**Διεπαφή BaaSAPI-22.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των προϊόντων στο BaaS.

**BaaSAPI-22.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των προϊόντων επιστρέφονται με επιτυχία. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 81.

```

[{"FedMarketplace_id": "eketaFED", "GlobalMarketplace_id": true, "Price": 0.9, "Product_details": [{"Product_id": "productX1", "Reputation": null, "Resource_ids": [{"resourceId": "platformAvraam1", "resourceId": "platformVagi1", "Seller": "vaggelis", "docType": "Product", "subjReputation": 0, "transactionCounter": 1}], "FedMarketplace_id": "eketaFED", "GlobalMarketplace_id": true, "Price": 0.8, "Product_details": [{"Product_id": "productX2", "Reputation": 0, "Resource_ids": [{"resourceId": "platformAvraam1", "resourceId": "platformVagi1", "Seller": "vaggelis", "docType": "Product", "subjReputation": 0, "transactionCounter": 1}]}]}]

```

Εικόνα 81: Επιστροφή πληροφοριών όλων των προϊόντων από το BaaS.

**Διεπαφή BaaSAPI-23.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των πόρων δεδομένων στο BaaS.

**BaaSAPI-23.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των πόρων δεδομένων επιστρέφονται με επιτυχία. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 82.

```
[{"Brd":12,"Brr":12,"Fr":5,"Hr":1701257450100,"Lr":1692207050100,"Platform":"platformAvraan1","Price":0.3,"Resource_id":"resourceAv1","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":"eketaFED"},"fr":1,"objReputation":0,"overallReputation":null,"owner":"Avraan","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":1}, {"Brd":12,"Brr":12,"Fr":5,"Hr":1701257450100,"Lr":1692207050100,"Platform":"platformAvraan1","Price":0.4,"Resource_id":"resourceAv2","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":"eketaFED"},"fr":1,"objReputation":0,"overallReputation":0,"owner":"Avraan","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":1}, {"Brd":12,"Brr":12,"Fr":3,"Hr":1703417450100,"Lr":1695144650100,"Platform":"platformAvraan2","Price":0.5,"Resource_id":"resourceAv3","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":"eketaFED"},"fr":1,"objReputation":0,"overallReputation":0,"owner":"Avraan","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":0}, {"Brd":2,"Brr":12,"Fr":5,"Hr":1701257450100,"Lr":1692207050100,"Platform":"platformVagi","Price":0.5,"Resource_id":"resourceV1","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":["eketaFED"],"fr":1,"objReputation":0,"overallReputation":0,"owner":"vaggelis","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":2}, {"Brd":2,"Brr":12,"Fr":5,"Hr":1701257450100,"Lr":1692207050100,"Platform":"platformVagi","Price":0.5,"Resource_id":"resourceV2","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":["eketaFED"],"fr":1,"objReputation":0,"overallReputation":0,"owner":"vaggelis","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":0}, {"Brd":2,"Brr":12,"Fr":4,"Hr":1703417450100,"Lr":1694805450100,"Platform":"platformVagi","Price":0.5,"Resource_id":"resourceV3","Wr":1440,"ard":0.001,"arf":0.05,"aro":1.05,"arr":1.2,"docType":"Resource","fed_id":["eketaFED"],"fr":1,"objReputation":0,"overallReputation":0,"owner":"vaggelis","relatedFeds":["eketaFED"],"subReputation":0,"transactionCounter":0}]
```

Εικόνα 82 Επιστροφή πληροφοριών όλων των πόρων δεδομένων από το BaaS

**Διεπαφή BaaSAPI-26.** Χρησιμοποιείται για κοινοποίηση ενός πόρου σε μία ομοσπονδία στο BaaS.

**BaaSAPI -26.A.** Αυτός ο έλεγχος χρησιμοποιείται για την επαλήθευση της επιτυχούς κοινοποίησης ενός πόρου σε μία ομοσπονδία στο BaaS. Οι παράμετροι που περνιούνται στο σώμα φαίνονται στην Εικόνα 83. Το αναμενόμενο αποτέλεσμα είναι status 200 με μήνυμα «Resource resourceV4 added to federation eketaFED with price 0.9 FedCoins!»

```
{
  "user": "vaggelis",
  "device_id": "resourceV4",
  "price": "0.9",
  "fed_id": "eketaFED",
  "Lr": "2023-09-19T17:30:50.100Z",
  "Hr": "2024-02-25T11:30:50.100Z",
  "Fr": 6,
  "Wr": 1440,
  "aro": 1.05,
  "ard": 0.001,
  "Brd": 2,
  "fr": 1,
  "arf": 0.05,
  "Brr": 12,
  "arr": 1.2
}
```

Εικόνα 83 Παράμετροι στο σώμα του ελέγχου BaaSAPI -26.A.

**BaaSAPI-26.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η κοινοποίηση πόρου σε μία ομοσπονδία για πόρο που δεν ανήκει στον χρήστη που εκτελεί την διεπαφή. Οι παράμετροι στο σώμα του ελέγχου είναι αυτοί στην Εικόνα 83, χωρίς να έχουμε εγγράψει τον εν λόγω πόρο στον αντίστοιχο χρήστη. Το αναμενόμενο αποτέλεσμα είναι status 500 με μήνυμα « Update failed » και «The resource resource\_id not belongs to user user\_id».

**BaaSAPI-26.Γ.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η κοινοποίηση πόρου σε μία ομοσπονδία για πόρο που ανήκει ήδη στην συγκεκριμένη ομοσπονδία. Οι παράμετροι στο σώμα του ελέγχου είναι και πάλι αυτοί στην Εικόνα 83 αμέσως μετά τον έλεγχο BaaSAPI-26.A. Το αναμενόμενο αποτέλεσμα είναι status 500 με μήνυμα « Update failed » και «The resource resource\_id already belongs to federation federation\_id».

**Διεπαφή BaaSAPI-27.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των πόρων δεδομένων μίας ομοσπονδίας στο BaaS.

**BaaSAPI-27.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των πόρων δεδομένων μίας ομοσπονδίας επιστρέφονται με επιτυχία. Οι παράμετροι που περνιούνται στο σώμα φαίνονται στην *Εικόνα 84*.

```
{
  "user_id": "vaggelis",
  "fed_id": "eketaFED",
  "Hr": "2024-02-25T11:30:50.100Z",
  "Lr": "2023-07-10T17:30:50.100Z",
  "Fr": 7,
  "price_min": "0.8",
  "price_max": "0.9",
  "rep_min": "0",
  "rep_max": "2"
}
```

*Εικόνα 84: Παράμετροι στο σώμα του ελέγχου BaaSAPI -27.A.*

Το αναμενόμενο αποτέλεσμα είναι status 200 και επιστροφή αντικειμένου σαν αυτό στην *Εικόνα 85*.

```
{
  "Resources": [
    {
      "Brd": 2,
      "Brr": 12,
      "Fr": 6,
      "Hr": 1708860650100,
      "Lr": 1695144650100,
      "Platform": "platformVag1",
      "Price": 0.9,
      "Resource_id": "resourceV4",
      "Wr": 1440,
      "ard": 0.001,
      "arf": 0.05,
      "aro": 1.05,
      "arr": 1.2,
      "docType": "Resource",
      "fed_id": [
        "eketaFED"
      ],
      "fr": 1,
      "objReputation": 0,
      "overallReputation": 0,
      "owner": "vaggelis",
      "relatedFeds": [],
      "subjReputation": 0,
      "transactionCounter": 0
    }
  ]
}
```

*Εικόνα 85 Επιστροφή πληροφοριών όλων των πόρων δεδομένων μίας ομοσπονδίας από το BaaS*

**BaaSAPI-27.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η ανάκτηση των πόρων μίας ομοσπονδίας. Αυτό γίνεται όταν καλούμε την διεπαφή με χρήστη που δεν υπάρχει στο BaaS. Το αναμενόμενο αποτέλεσμα είναι status 500 με μήνυμα « Get federated resources failed » και «Identity not found in wallet: user1».

**Διεπαφή BaaSAPI-28.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των προϊόντων μίας ομοσπονδίας στο BaaS.



**BaaSAPI-28.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των προϊόντων μίας ομοσπονδίας επιστρέφονται με επιτυχία, εφόσον τα αναγνωριστικά του χρήστη και της ομοσπονδίας καταχωρούνται ως παράμετροι στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 86*.

```
{
  "Products": [
    {
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.9,
      "Product_details": {},
      "Product_id": "productX1",
      "Reputation": null,
      "Resource_ids": [
        {
          "resourceAv2": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ],
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.8,
      "Product_details": {},
      "Product_id": "productX2",
      "Reputation": 0,
      "Resource_ids": [
        {
          "resourceAv1": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ],
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.8,
      "Product_details": {},
      "Product_id": "productX5",
      "Reputation": 0,
      "Resource_ids": [
        {
          "resourceAv1": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ]
    }
  ]
}
```

*Εικόνα 86 Επιστροφή πληροφοριών όλων των προϊόντων μίας ομοσπονδίας από το BaaS*

**Διεπαφή BaaSAPI-29.** Χρησιμοποιείται για την πρόσβαση πληροφοριών όλων των προϊόντων της καθολικής αγοράς στο BaaS.

**BaaSAPI-29.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες όλων των προϊόντων της καθολικής αγοράς επιστρέφονται με επιτυχία, εφόσον το αναγνωριστικό του χρήστη καταχωρείται ως παράμετρος στο «url» του αιτήματος. Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 87*.

```
{
  "Products": [
    {
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.9,
      "Product_details": {},
      "Product_id": "productX1",
      "Reputation": null,
      "Resource_ids": [
        {
          "resourceAv2": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ],
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.8,
      "Product_details": {},
      "Product_id": "productX2",
      "Reputation": 0,
      "Resource_ids": [
        {
          "resourceAv1": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ],
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Price": 0.8,
      "Product_details": {},
      "Product_id": "productX5",
      "Reputation": 0,
      "Resource_ids": [
        {
          "resourceAv1": "platformAvraam1",
          "resourceV1": "platformVag1",
          "Seller": "vaggelis",
          "docType": "Product",
          "subjReputation": [],
          "transactionCounter": 1
        }
      ]
    }
  ]
}
```

*Εικόνα 87: Επιστροφή πληροφοριών όλων των προϊόντων της καθολικής αγοράς από το BaaS.*

**Διεπαφή BaaSAPI-32.** Χρησιμοποιείται για την δημιουργία ενός προϊόντος σε μια ομοσπονδία στο BaaS.

**BaaSAPI-32.A.** Αυτός ο έλεγχος επαληθεύει ότι ένα προϊόν δημιουργείται με επιτυχία, οι παρακάτω παράμετροι (*Εικόνα 88*) πρέπει να οριστούν στο σώμα του αιτήματος:

```
{
  "product_id": "productX5",
  "seller_id": "vaggelis",
  "resource_ids": [
    "resourceV1",
    "resourceAv1"
  ],
  "fedMarketplace_id": "eketaFED",
  "product_details": {}
}
```

*Εικόνα 88: Παράμετροι στο σώμα του ελέγχου BaaSAPI-32.A.*

Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 89*.



```

{
  "Product_id": "productX5",
  "docType": "Product",
  "Price": 0.8,
  "Resource_ids": {
    "resourceV1": "platformVag1",
    "resourceAv1": "platformAvraam1"
  },
  "Product_details": {},
  "FedMarketplace_id": "eketaFED",
  "GlobalMarketplace_id": true,
  "transactionCounter": 0,
  "subjReputation": [],
  "Reputation": 0,
  "Seller": "vaggelis"
}

```

Εικόνα 89 Επιστροφή πληροφοριών δημιουργίας προϊόντος από το BaaS

**BaaSAPI-32.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η δημιουργία προϊόντος, όταν καλούμε την διεπαφή με πόρο που δεν υπάρχει στο BaaS. Το αναμενόμενο αποτέλεσμα είναι status 500 με μήνυμα « Product creation failed » και «The resource with id resource\_id does not exist».

**Διεπαφή BaaSAPI-33.** Χρησιμοποιείται για την διαγραφή ενός προϊόντος ενός χρήστη στο BaaS.

**BaaSAPI-33.A.** Αυτός ο έλεγχος επαληθεύει ότι η διαγραφή προϊόντος γίνεται με επιτυχία, οι παρακάτω παράμετροι (Εικόνα 90) πρέπει να οριστούν στο σώμα του αιτήματος:



```

{
  "product_id": "product1",
  "user_id": "avraam"
}

```

Εικόνα 90 Παράμετροι στο σώμα του ελέγχου BaaSAPI-33.A.

Το αναμενόμενο αποτέλεσμα είναι status: 200 με μήνυμα «OK!» και σε εκτέλεση του ελέγχου BaaSAPI-4 για τον χρήστη αυτόν δεν πρέπει να φαίνεται το προϊόν.

**BaaSAPI-33.B.** Αυτός ο έλεγχος επαληθεύει ότι εάν το προς διαγραφή προϊόν δεν υπάρχει στο BaaS, το αναμενόμενο αποτέλεσμα είναι status: 404 με μήνυμα «Deletion failed, product not found».

**Διεπαφή BaaSAPI-34.** Χρησιμοποιείται για τον δυναμικό υπολογισμό της τιμής ενός προϊόντος στο BaaS.

**BaaSAPI-34.A.** Αυτός ο έλεγχος επαληθεύει ότι η τιμή του προϊόντος υπολογίζεται με επιτυχία, οι παρακάτω παράμετροι (Εικόνα 91) πρέπει να οριστούν στο σώμα του αιτήματος:



```

{
  "product_id": "productX5",
  "user_id": "avraam",
  "marketplace": "eketaFED",
  "streaming": false,
  "access": 50,
  "lp": "2023-08-29T17:30:50.100Z",
  "hp": "2023-11-28T17:30:50.100Z",
  "fp": "2"
}

```

Εικόνα 91 Παράμετροι στο σώμα του ελέγχου BaaSAPI-34.A.



Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 92*.



*Εικόνα 92: Επιστροφή πληροφοριών υπολογισμού προϊόντος από το BaaS (Περίπτωση 1).*

Αν ο υποψήφιος αγοραστής του προϊόντος, είναι μέλος της ομοσπονδίας του πωλητή, και επιλέξει στις παραμέτρους τον υπολογισμό της τιμής στην καθολική αγορά, του εμφανίζει το παρακάτω μήνυμα (*Εικόνα 93*).

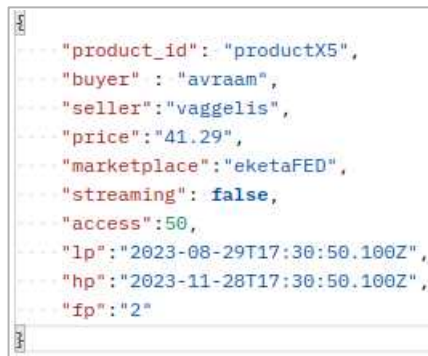


*Εικόνα 93 Επιστροφή πληροφοριών υπολογισμού προϊόντος από το BaaS (Περίπτωση 2).*

**BaaSAPI-34.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται ο υπολογισμός τιμής του προϊόντος όταν καλούμε την διεπαφή με τιμές στις παραμέτρους hp-lp-fp, που δεν συμφωνούν με τις αντίστοιχες τιμές των πόρων στο BaaS. Παράδειγμα τέτοιου λάθους: status 400 με μήνυμα « Not acceptable hp value» και «hp: 2024-01-28T17:30:50.100Z > min(Hr): 2023-11-29T11:30:50.100Z».

**Διεπαφή BaaSAPI-35.** Χρησιμοποιείται για την αγορά ενός προϊόντος στο BaaS.

**BaaSAPI-35.A.** Αυτός ο έλεγχος επαληθεύει ότι η αγορά του προϊόντος εκτελέστηκε με επιτυχία, οι παρακάτω παράμετροι (*Εικόνα 94*) πρέπει να οριστούν στο σώμα του αιτήματος:



*Εικόνα 94 Παράμετροι στο σώμα του ελέγχου BaaSAPI-35.A*

Το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην *Εικόνα 95*.

```

"Receipt": {
  "TransactionID": "30934506236dc35426ce471ac7a4fbf3dbfc9a09d234af42e6f7b686ad8c1214",
  "Buyer": "avraam",
  "Seller": "vaggelis",
  "ProductID": "productX5",
  "DateBought": "Fri, 12 Jan 2024 09:05:15 GMT"
},
"AccessToken": {
  "AccessTimes": 50,
  "userHasRated": false,
  "toBeExchanged": false,
  "inUse": false,
  "Marketplace": "eketaFED",
  "Frequency": "2",
  "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
  "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
  "ValidUntil": "2024-02-11T09:05:15.010Z"
}

```

Εικόνα 95 Επιστροφή πληροφοριών αγοράς προϊόντος από το BaaS.

**BaaSAPI-35.B.** Αυτός ο έλεγχος επαληθεύει ότι δεν επιτρέπεται η αγορά του προϊόντος, όταν η πραγματική τιμή του είναι διαφορετική από αυτήν που δίνεται στις παραμέτρους της κλήσης στην Εικόνα 94. Για τον υπολογισμό της πραγματικής τιμής του εκτελείται ο έλεγχος BaaSAPI-34. Το αναμενόμενο αποτέλεσμα είναι status: 400 με μήνυμα «Product price changed! New price is 100 FedCoins. Please repeat the procedure again if you want to buy this product.».

**BaaSAPI-35.Γ.** Αυτός ο έλεγχος επαληθεύει ότι δεν δύναται η αγορά του προϊόντος, όταν καλούμε την διεπαφή για προϊόν το οποίο ήδη έχει αγοράσει ο χρήστης. Το αναμενόμενο αποτέλεσμα είναι status: 500 με μήνυμα «Buying product failed» και «You already own this product. Please use your remaining time/number of access».

Πίνακας 12: Περιπτώσεις δοκιμής στοιχείου BaaS για την αγορά προϊόντων.

Interface	Test case	Method	Call	Status
BaaSAPI-20	BaaSAPI-20.A	GET	<BaaS-API-url-user>/get_product_info	PASS
	BaaSAPI-20.B	GET	<BaaS-API-url-user>/get_product_info	PASS
BaaSAPI-21	BaaSAPI-21.A	GET	<BaaS-API-url-products>/get_resource_info	PASS
	BaaSAPI-21.B	GET	<BaaS-API-url-products>/get_resource_info	PASS
BaaSAPI-22	BaaSAPI-22.A	GET	<BaaS-API-url-products>/get_all_product_info	PASS
BaaSAPI-23	BaaSAPI-23.A	GET	<BaaS-API-url-products>/get_all_resource_info	PASS
BaaSAPI-26	BaaSAPI-26.A	PATCH	<BaaS-API-url-products>/add_resource_fed	PASS
	BaaSAPI-26.B	PATCH	<BaaS-API-url-products>/add_resource_fed	PASS
	BaaSAPI-26.Γ	PATCH	<BaaS-API-url-products>/add_resource_fed	PASS
BaaSAPI-27	BaaSAPI-27.A	POST	<BaaS-API-url-products>/get_federate	PASS

			d_resources	
	BaaSAPI-27.B	POST	<BaaS-API-url-products>/get_federated_resources	PASS
BaaSAPI-28	BaaSAPI-28.A	GET	<BaaS-API-url-products>/get_federated_products	PASS
BaaSAPI-29	BaaSAPI-29.A	GET	<BaaS-API-url-products>/get_global_products	PASS
BaaSAPI-32	BaaSAPI-32.A	POST	<BaaS-API-url-products>/create_product	PASS
	BaaSAPI-32.B	POST	<BaaS-API-url-products>/create_product	PASS
BaaSAPI-33	BaaSAPI-33.A	DELETE	<BaaS-API-url-products>/delete_product	PASS
	BaaSAPI-33.B	DELETE	<BaaS-API-url-products>/delete_product	PASS
BaaSAPI-34	BaaSAPI-34.A	POST	<BaaS-API-url-products>/calc_product_price	PASS
	BaaSAPI-34.B	POST	<BaaS-API-url-products>/calc_product_price	PASS
BaaSAPI-35	BaaSAPI-35.A	POST	<BaaS-API-url-products>/buy_product	PASS
	BaaSAPI-35.B	POST	<BaaS-API-url-products>/buy_product	PASS
	BaaSAPI-35.Γ	POST	<BaaS-API-url-products>/buy_product	PASS

#### 3.3.3.4.2 Δοκιμή διεπαφών Ασφαλούς Πρόσβασης

Οι διεπαφές αυτής της υπό-ενότητας αφορούν την ασφαλή πρόσβαση προϊόντων και τον έλεγχο των κουπονιών πρόσβασης που ο χρήστης έχει στην κατοχή του. Το κουπόνι πρόσβασης αποκτάται από τον χρήστη κατά την επιτυχημένη αγορά του προϊόντος, η οποία περιεγράφηκε παραπάνω. Για τους σκοπούς των ελέγχων αυτής της υπό-ενότητας, δημιουργήθηκαν δύο προϊόντα και αγοράστηκαν από έναν χρήστη. Αυτές οι λειτουργίες δεν αναλύονται περαιτέρω εδώ, μιας και έχουν ήδη περιγραφεί.

**Διεπαφή BaaSAPI-24.** Χρησιμοποιείται για τον έλεγχο και ενημέρωση του χρήστη για τα κουπόνια πρόσβασης που έχει στην κατοχή του.

**BaaSAPI-24.A.** Αυτός ο έλεγχος επαληθεύει ότι οι πληροφορίες για όλα τα κουπόνια πρόσβασης του χρήστη επιστρέφονται με επιτυχία. Το αποτέλεσμα που αναμένεται είναι status 200 και το αντικείμενο που επιστρέφεται φαίνεται στην *Εικόνα 96*.

```

{
  "productX1": {
    "AccessTimes": 50,
    "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
    "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
    "Frequency": "2",
    "Marketplace": "global",
    "ValidUntil": "2024-02-11T09:22:52.035Z",
    "inUse": true,
    "toBeExchanged": false,
    "userHasRated": false
  },
  "productX2": {
    "AccessTimes": 50,
    "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
    "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
    "Frequency": "2",
    "Marketplace": "global",
    "ValidUntil": "2024-02-11T11:04:26.610Z",
    "inUse": true,
    "toBeExchanged": false,
    "userHasRated": false
  }
}

```

Εικόνα 96: Τα κουπόνια πρόσβασης ενός χρήστη μαζί με τις πληροφορίες τους, μετά από επιτυχημένη κλήση της αντίστοιχης διεπαφής.

**BaaSAPI-24.B.** Ο έλεγχος αυτός επαληθεύει ότι επιστρέφεται κατάλληλο μήνυμα λάθους στην περίπτωση που ο χρήστης δεν έχει κανένα κουπόνι πρόσβασης στην κατοχή του. Το αναμενόμενο αποτέλεσμα είναι status 403 που συνοδεύεται από το ακόλουθο μήνυμα «Get product tokens failed.» και «No tokens exist for user....»

**Διεπαφή BaaSAPI-36.** Η διεπαφή αυτή χρησιμοποιείται για τον έλεγχο ασφαλούς πρόσβασης σε ένα προϊόν από κάποιο χρήστη.

**BaaSAPI-36.A.** Αυτός ο έλεγχος επαληθεύει ότι ο χρήστης έχει το δικαίωμα πρόσβασης σε προϊόν που έχει αγοράσει και μπορεί να το χρησιμοποιήσει. Η ορθή και ασφαλής πρόσβαση προκύπτει όταν οι απαραίτητοι έλεγχοι που συμβαίνουν εσωτερικά στο BaaS είναι επιτυχημένοι και βασίζονται στα στοιχεία που βρίσκονται στο σώμα του ελέγχου (Εικόνα 97) και τις πληροφορίες του κουπονιού πρόσβασης. Το αποτέλεσμα της επιτυχημένης πρόσβασης είναι το ανανεωμένο, όπου χρειάζεται, κουπόνι πρόσβασης του χρήστη και φαίνεται στην Εικόνα 98.

```

{
  "product_id": "productX2",
  "user_id": "avraam",
  "data_from": "2023-08-29T17:30:50.100Z",
  "data_until": "2023-09-29T17:30:50.100Z",
  "req_observations": "2"
}

```

Εικόνα 97: Σώμα κλήσης του ελέγχου BaaSAPI-36.A.

```

{
  "AccessToken": {
    "AccessTimes": 50,
    "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
    "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
    "Frequency": "2",
    "Marketplace": "global",
    "ValidUntil": "2024-02-11T14:45:06.933Z",
    "inUse": true,
    "toBeExchanged": false,
    "userHasRated": false
  }
}

```

Εικόνα 98: Το ανανεωμένο (αν χρειάζεται) κουπόνι πρόσβασης του χρήστη μετά από επιτυχημένη πρόσβαση στο προϊόν.

**BaaSAPI-36.B.** Ο έλεγχος αυτός επιβεβαιώνει την αποτροπή χρήσης ενός προϊόντος για το οποίο ο χρήστης δεν έχει το αντίστοιχο κουπόνι πρόσβασης. Το σώμα ελέγχου παραμένει ίδιο, ενώ το αναμενόμενο αποτέλεσμα είναι status 403 και συνοδεύεται από το μήνυμα «Check access failed» και «You can't have access to product you haven't bought or no longer own.».

**BaaSAPI-36.Γ.** Ο έλεγχος αυτός επαληθεύει ότι το κουπόνι πρόσβασης του χρήστη δεν είναι πια έγκυρο (έχει λήξει ή έχει εξαντλήσει τον αριθμό των διαθέσιμων χρήσεων) και σωστά απαγορεύει την πρόσβαση και χρήση του προϊόντος. Το αναμενόμενο αποτέλεσμα είναι status 403 και συνοδεύεται από τα μηνύματα «Check access failed» και «You no longer have access to this product.»

**BaaSAPI-36.Δ.** Αυτός ο έλεγχος επιβεβαιώνει ότι ο χρήστης δεν μπορεί να κάνει χρήση του προϊόντος γιατί έχει θέσει το αντίστοιχο κουπόνι πρόσβασης προς ανταλλαγή. Το αποτέλεσμα είναι status 403 και τα αντίστοιχα μηνύματα είναι «Check access failed» και «You have set this product for exchange. You can't use it anymore.».

**BaaSAPI-36.E.** Αυτός ο έλεγχος επιβεβαιώνει ότι ο χρήστης δεν μπορεί να κάνει χρήση του προϊόντος και να πάρει τα δεδομένα της χρονικής περιόδου που επιθυμεί, καθώς η συγκεκριμένη χρονική περίοδος είναι εκτός των χρονικών ορίων για τα οποία ο χρήστης έχει αγοράσει το προϊόν. Το αποτέλεσμα είναι status 403 και τα αντίστοιχα μηνύματα είναι «Check access failed» και «Requested data are out of the data bounds you have the product for...».

**BaaSAPI-36.ΣΤ.** Αυτός ο έλεγχος επιβεβαιώνει ότι ο χρήστης δεν μπορεί να κάνει χρήση του προϊόντος γιατί ο αριθμός των ζητούμενων μετρήσεων (observations) είναι μεγαλύτερος των διαθέσιμων μετρήσεων που είναι δηλωμένες στο κουπόνι χρήσης του για αυτό το προϊόν. Το αποτέλεσμα είναι status 403 και τα αντίστοιχα μηνύματα είναι «Check access failed» και «Requested observations are more than what you have left.».

Πίνακας 13: Περιπτώσεις δοκιμής στοιχείου BaaS για την ασφαλή πρόσβαση προϊόντων.

Interface	Test case	Method	Call	Status
BaaSAPI-24	BaaSAPI-24.A	GET	<BaaS-API-url-product>/get_user_tokens?user_id="username"	PASS
	BaaSAPI-24.B	GET	<BaaS-API-url-product>/get_user_tokens?user_id="username"	PASS
BaaSAPI-36	BaaSAPI-36.A.	POST	<BaaS-API-url-product>/check_access	PASS
	BaaSAPI-36.B.	POST	<BaaS-API-url-product>/check_access	PASS

	BaaSAPI-36.Γ.	POST	<BaaS-API-url-product>/check_access	PASS
	BaaSAPI-36.Δ.	POST	<BaaS-API-url-product>/check_access	PASS
	BaaSAPI-36.Ε.	POST	<BaaS-API-url-product>/check_access	PASS
	BaaSAPI-36.ΣΤ.	POST	<BaaS-API-url-product>/check_access	PASS

### 3.3.3.4.3 Δοκιμή διεπαφών Ανταλλαγής Προϊόντων

Οι διεπαφές της ενότητας αυτής αφορούν την ανταλλαγή κουπονιών πρόσβασης προϊόντων μεταξύ χρηστών. Για την κλήση των ελέγχων που ακολουθούν δημιουργήθηκαν δυο χρήστες τους «exchange\_user» και «exchange\_user2» και στον καθένα από αυτούς δημιουργήθηκαν δυο προϊόντα τα «productX1» και «productX2» αντίστοιχα. Στη συνέχεια αγοράστηκε με κάθε χρήστη το προϊόν του άλλου ώστε να αποκτηθεί από ένα κουπόνι πρόσβασης στον καθένα. Οι διαδοχικές κλήσεις διεπαφών που απαιτούνται για να φτάσουμε σε αυτό το σημείο δεν αναλύονται καθώς έχουν εξηγηθεί σε προηγούμενα κεφάλαια.

**Διεπαφή BaaSAPI-37.** Χρησιμοποιείται για την προσθήκη ενός προϊόντος προς ανταλλαγή από έναν χρήστη.

**BaaSAPI-37.A.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης δε μπορεί να θέσει προς ανταλλαγή προϊόν που δεν έχει αγοράσει. Προσπάθεια κλήσης της διεπαφής BaaSAPI37 με παραμέτρους που δεν αντιστοιχούν σε κουπόνι πρόσβασης όπως στην Εικόνα 99, αναμένεται να επιστρέφουν status 403.

```

{
  "user_id": "exchange_user",
  "product_id": "productX1"
}

```

Εικόνα 99: Μη έγκυρο σώμα στην διεπαφή BaaSAPI-37

**BaaSAPI-37.B.** Αυτός ο έλεγχος επαληθεύει ότι ένας χρήστης μπορεί να προσθέσει προς ανταλλαγή χωρίς πρόβλημα προϊόν για το οποίο έχει κουπόνι πρόσβασης. Καλώντας την διεπαφή με το σώμα στην Εικόνα 100 το αναμενόμενο αποτέλεσμα είναι status 200 και αντικείμενο σαν αυτό στην Εικόνα 101.

```

{
  "user_id": "exchange_user2",
  "product_id": "productX1"
}

```

Εικόνα 100 Έγκυρο σώμα στην διεπαφή BaaSAPI-37

```

{
  "AccessTimes": 50,
  "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
  "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
  "Frequency": "2",
  "Marketplace": "global",
  "ValidUntil": "2024-02-11T11:17:20.155Z",
  "inUse": false,
  "toBeExchanged": true,
  "userHasRated": false
}

```

Εικόνα 101: Αντικείμενο μετά απο έγκυρη κλήση της διεπαφής BaaSAPI-37.

**Διεπαφή BaaSAPI-25.** Χρησιμοποιείται για αναζήτηση όλων των προς ανταλλαγή προϊόντων.

**BaaSAPI-25.A.** Αυτός ο έλεγχος επαληθεύει ότι όλα τα προς αναταλλαγή προϊόντα επιστρέφονται από την διεπαφή. Μετά τον έλεγχο BaaSAPI-37.B καλούμε την διεπαφή και παίρνουμε το αντικείμενο στην *Εικόνα 102*

```

{
  "allExchangableTokens": [
    {
      "exchange_user": []
    },
    {
      "exchange_user2": [
        {
          "AccessTimes": 50,
          "DataAvailableFrom": "2023-08-29T17:30:50.100Z",
          "DataAvailableUntil": "2023-11-28T17:30:50.100Z",
          "Frequency": "2",
          "Marketplace": "global",
          "ValidUntil": "2024-02-11T11:17:20.155Z",
          "inUse": false,
          "toBeExchanged": true,
          "userHasRated": false,
          "productID": "productX1"
        }
      ]
    }
  ]
}

```

*Εικόνα 102: Παράδειγμα επιστροφής αντικειμένου της διεπαφής BaaSAPI-25.*

**Διεπαφή BaaSAPI-38.** Χρησιμοποιείται για την αφαίρεση ενός προϊόντος από διαθεσιμότητα προς ανταλλαγή.

**BaaSAPI-38.A.** Αυτός ο έλεγχος επαληθεύει πως η κλήση του endpoint με σώμα που δεν αντιστοιχεί σε προϊόν που έχει τεθεί προς ανταλλαγή από έναν χρήστη απαγορεύεται με status 403. Η κλήση γίνεται με το σώμα στην *Εικόνα 103*

```

{
  "product_id": "productX1",
  "user_id": "exchange_user"
}

```

*Εικόνα 103: Μη έγκυρο σώμα στην διεπαφή BaaSAPI-38.*

**BaaSAPI-38.B.** Αυτός ο έλεγχος επαληθεύει πως η αφαίρεση προϊόντος από την προς ανταλλαγή επιλογή, όταν η κλήση της διεπαφής γίνεται για προϊόν που ο χρήστης έχει ήδη αγοράσει και προσθέσει προς ανταλλαγή λειτουργεί κανονικά. Καλώντας με σώμα σαν αυτό στην *Εικόνα 104*, το αναμενόμενο αποτέλεσμα είναι status 200 με μήνυμα OK.

```

{
  "product_id": "productX2",
  "user_id": "exchange_user"
}

```

*Εικόνα 104 Έγκυρο σώμα για κλήση της διεπαφής BaaSAPI-38*



**Διεπαφή BaaSAPI-39.** Χρησιμοποιείται για την πραγματοποίηση ανταλλαγής μεταξύ δυο προϊόντων

**BaaSAPI-39.A.** Αυτός ο έλεγχος επαληθεύει την απαγόρευση μιας ειδικής περίπτωσης, αυτής στην οποία σε μια ανταλλαγή προϊόντων ο ένας ή και οι δυο χρήστες είναι ιδιοκτήτες του προϊόντος που πρόκειται να λάβουν. Αυτό το σενάριο υφίσταται αν ορίσουμε ανταλλαγή των κουπονιών που δημιουργήθηκαν για τον έλεγχο BaaSAPI-37.A, οπότε το σώμα θα είναι αυτό στην *Εικόνα 105*. Στην περίπτωση αυτή αναμένεται status 403 και μήνυμα «exchange\_user2 already owns productX2».



```

{
  "product_id1": "productX1",
  "user_id1": "exchange_user2",
  "product_id2": "productX2",
  "user_id2": "exchange_user"
}

```

*Εικόνα 105: Μη έγκυρο σώμα κατά την κλήση της διεπαφής BaaSAPI-39*

**BaaSAPI-39.B.** Αυτός ο έλεγχος επαληθεύει ότι μια έγκυρη ανταλλαγή προϊόντων μπορεί να προχωρήσει κανονικά. Για επαλήθευσης θα χρειαστεί ένας τρίτος χρήστης «exchange\_user3» ο οποίος αγοράζει ένα νέο προϊόν «productX1b» του «exchange\_user». Έπειτα προτείνεται ανταλλαγή αυτού με το προϊόν που ο «exchange\_user2» είχε θέσει προς ανταλλαγή, οποιαδήποτε άλλη λογική θα αποτύχει. Το σώμα της κλήσης φαίνεται στην *Εικόνα 106*, όπου είναι σημαντικό να αναφέρονται πρώτα ο χρήστης και το προϊόν που έχουν τεθεί προς ανταλλαγή και να ακολουθούνται από τον χρήστη που προτείνει την ανταλλαγή. Μετά την κλήση αναμένεται status 200 και μήνυμα OK.



```

{
  "product_id1": "productX1",
  "user_id1": "exchange_user2",
  "product_id2": "productX1b",
  "user_id2": "exchange_user3"
}

```

*Εικόνα 106: Έγκυρο σώμα κλήσης της διεπαφής BaaSAPI-39*

#### 3.3.3.4.4 Δοκιμή διεπαφών Αναζήτησης και Διαφήμισης προϊόντων

Οι διεπαφές αυτής της υπό-ενότητας περιγράφουν την αναζήτηση και την διαφήμιση τόσο των πόρων δεδομένων, όσο και των προϊόντων. Η αναζήτηση τους γίνεται με βάση κάποια κριτήρια τα οποία περνούν ως παράμετροι στους ελέγχους που ακολουθούν, ενώ η διαφήμιση τους αποτελεί μια εσωτερική διεργασία του BaaS, και παρουσιάζει τα αποτελέσματα της αναζήτησης με τυχαίο τρόπο (περισσότερες λεπτομέρειες υπάρχουν και στο αντίστοιχο παραδοτέο, Π2.2). Για τους σκοπούς αυτής της ενότητας, δημιουργήθηκαν μερικοί πόροι και προϊόντα δεδομένων, και στη συνέχεια εφαρμόστηκε σε αυτά ο μηχανισμός της αναζήτησης και διαφήμισης. Η δημιουργία των παραπάνω δεν αναλύεται εδώ, αφού οι αντίστοιχες διαδικασίες έχουν περιγραφεί πιο πάνω.

**Διεπαφή BaaSAPI-30.** Αυτή η διεπαφή χρησιμοποιείται για την αναζήτηση και διαφήμιση των πόρων δεδομένων.

**BaaSAPI-30.A.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφονται οι σωστοί πόροι δεδομένων, μετά από επιτυχημένη αναζήτηση από τον χρήστη. Οι πόροι που επιστρέφονται είναι αυτοί που ικανοποιούν τα κριτήρια που τέθηκαν κατά την αναζήτηση. Οι παράμετροι στο σώμα του ελέγχου φαίνονται στην *Εικόνα 107*, ενώ το αναμενόμενο αποτέλεσμα είναι status 200 και το επιστρεφόμενο αποτέλεσμα φαίνεται στην *Εικόνα 108*.



```

1  {
2    "resources": ["resource2", "resourceAv1", "resourceV1"],
3    "price_min": 0.2,
4    "price_max": 60,
5    "rep_min": 0,
6    "rep_max": 1,
7    "user_id": "zisis"
8  }

```

Εικόνα 107: Σώμα κλήσης του ελέγχου BaaSAPI-30.A.

```

{
  "Resources": [
    {
      "Brd": 2,
      "Brr": 12,
      "Fr": 5,
      "Hr": 1701257450100,
      "Lr": 1692207050100,
      "Platform": "platform1",
      "Price": 0.5,
      "Resource_id": "resource2",
      "Wr": 1440,
      "ard": 0.001,
      "arf": 0.05,
      "aro": 1.05,
      "arr": 1.2,
      "docType": "Resource",
      "fed_id": [
        "eketaFED"
      ],
      "fr": 1,
      "objReputation": 0,
      "overallReputation": 0,
      "owner": "zisis",
      "relatedFeds": [],
      "subjReputation": 0,
      "transactionCounter": 0
    },
    {
      "Brd": 2,
      "Brr": 12,
      "Fr": 5,
      "Hr": 1701257450100,
      "Lr": 1692207050100,
      "Platform": "platform1",
      "Price": 0.5,
      "Resource_id": "resourceAv1",
      "Wr": 1440,
      "ard": 0.001,
      "arf": 0.05,
      "aro": 1.05,
      "arr": 1.2,
      "docType": "Resource",
      "fed_id": [
        "eketaFED"
      ],
      "fr": 1,
      "objReputation": 0,
      "overallReputation": 0,
      "owner": "zisis",
      "relatedFeds": [],
      "subjReputation": 0,
      "transactionCounter": 0
    }
  ]
}

```

Εικόνα 108: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τους πόρους δεδομένων, μετά απο επιτυχημένη κλήση του ελέγχου BaaSAPI-30.A.

**BaaSAPI-30.B.** Ο έλεγχος αυτός επαληθεύει ότι αν δεν βρεθεί κανένας πόρος δεδομένων που να ικανοποιεί τα κριτήρια αναζήτησης, επιστρέφει μια κενή λίστα («{}»).

**Διεπαφή BaaSAPI-31.** Αυτή η διεπαφή χρησιμοποιείται για την αναζήτηση και διαφήμιση των προϊόντων.

**BaaSAPI-31.A.** Αυτός ο έλεγχος επαληθεύει ότι επιστρέφονται τα σωστά προϊόντα, μετά από επιτυχημένη αναζήτηση από τον χρήστη. Τα προϊόντα που επιστρέφονται είναι εκείνα που ικανοποιούν τα κριτήρια που τέθηκαν κατά την αναζήτηση. Οι παράμετροι στο σώμα του ελέγχου φαίνονται στην Εικόνα 109, ενώ το αναμενόμενο αποτέλεσμα είναι status 200 και το

επιστρεφόμενο αποτέλεσμα φαίνεται στην *Εικόνα 110* και περιλαμβάνει τα αντικείμενα των προϊόντων.

```

1  {
2    ...."products": ["productX1", "productX2"],
3    ...."price_min": 0,
4    ...."price_max": 1,
5    ...."rep_min": 0,
6    ...."rep_max": 1,
7    ...."user_id": "zisis"
8  }

```

*Εικόνα 109: Σώμα κλήσης του ελέγχου BaaSAPI-31.A.*

```

{
  "Products": [
    {
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Product_details": {},
      "Product_id": "productX1",
      "Reputation": null,
      "Resource_ids": {
        "resourceAv1": "platform1"
      },
      "Seller": "zisis",
      "docType": "Product",
      "subjReputation": [],
      "transactionCounter": 0,
      "totalPrice": 0.5
    },
    {
      "FedMarketplace_id": "eketaFED",
      "GlobalMarketplace_id": true,
      "Product_details": {},
      "Product_id": "productX2",
      "Reputation": 0,
      "Resource_ids": {
        "resource2": "platform1",
        "resourceAv1": "platform1"
      },
      "Seller": "zisis",
      "docType": "Product",
      "subjReputation": [],
      "transactionCounter": 0,
      "totalPrice": 1
    }
  ]
}

```

*Εικόνα 110: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τα προϊόντα, μετά απο επιτυχημένη κλήση του ελέγχου BaaSAPI-31.A.*

**BaaSAPI-31.B.** Ο έλεγχος αυτός επαληθεύει ότι αν δεν βρεθεί κανένα προϊόν δεδομένων που να ικανοποιεί τα κριτήρια αναζήτησης, επιστρέφει μια κενή λίστα («{}»).

**Πίνακας 14:** Περιπτώσεις δοκιμής στοιχείου BaaS για την αναζήτηση και διαφήμιση προϊόντων.

Interface	Test case	Method	Call	Status
BaaSAPI-30	BaaSAPI-30.A	POST	<BaaS-API-url-product>/search_resources	PASS

	BaaSAPI-30.B	POST	<BaaS-API-url-product>/search_resources	PASS
BaaSAPI-31	BaaSAPI-36.A.	POST	<BaaS-API-url-product>/search_products	PASS
	BaaSAPI-36.B.	POST	<BaaS-API-url-product>/search_products	PASS

### 3.3.3.4.5 Δοκιμή διεπαφών Καταγραφής και Ελέγχου Συναλλαγών

Οι διεπαφές που αναλύονται σε αυτή την υπό-ενότητα περιγράφουν τις διαδικασίες καταγραφής και ελέγχου των συναλλαγών, είτε αυτές είναι αγορές προϊόντων, είτε καταγραφή χρήσης τους. Για τους σκοπούς των ελέγχων σε αυτήν την υπό-ενότητα, αξιοποιήθηκαν οι αγορές που πραγματοποιήθηκαν για την [Δοκιμή διεπαφών Ασφαλούς Πρόσβασης](#), καθώς και οι αντίστοιχες χρήσεις ενός προϊόντος που προήλθαν από την ίδια υπό-ενότητα.

**Διεπαφή BaaSAPI-43.** Αυτή η διεπαφή χρησιμοποιείται από έναν πωλητή προϊόντων για να ανασύρει όλες τις αποδείξεις των προϊόντων που έχει πουλήσει.

**BaaSAPI-43.A.** Ο έλεγχος αυτός επιβεβαιώνει την επιτυχημένη αναζήτηση και την ανάκτηση των αποδείξεων συναλλαγών από την πλευρά του πωλητή. Η αναζήτηση αυτή γίνεται με βάση τα κριτήρια που υπάρχουν στο σώμα ελέγχου, όπως φαίνεται στην *Εικόνα 111*. Το αναμενόμενο αποτέλεσμα είναι status 200 μαζί με μια λίστα όλων των αποδείξεων, όπως φαίνεται στην *Εικόνα 112*.

```
{
  "user": "avraam",
  "product_id": "productX1",
  "fromDate": "2024-01-10T09:22:52.000Z",
  "toDate": "2024-01-15T09:22:52.000Z"
}
```

Εικόνα 111: Σώμα κλήσης του ελέγχου BaaSAPI-43.A.

```
{
  "Receipts": [
    {
      "TransactionID": "2a5be3986c00a574b435f8ce0f914b32bec3bd80887759b9d45474302b038bae",
      "Buyer": "zisis",
      "Seller": "avraam",
      "ProductID": "productX1",
      "DateBought": "2024-01-12T09:22:52.000Z"
    }
  ]
}
```

Εικόνα 112: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τις αποδείξεις συναλλαγών, μετά από επιτυχημένη κλήση του αντίστοιχου ελέγχου.

**BaaSAPI-43.B.** Ο έλεγχος αυτός επιβεβαιώνει την ορθή αναζήτηση των αποδείξεων συναλλαγών, αλλά σε αυτήν την περίπτωση δεν υπάρχουν αποδείξεις που να ικανοποιούν τα κριτήρια αναζήτησης τους, οπότε και επιστρέφεται μια κενή λίστα., όπως φαίνεται στην *Εικόνα 113*.

```
{
  "Receipts": []
}
```

Εικόνα 113: Κενή λίστα αποδείξεων όταν δεν υπάρχουν διαθέσιμες αποδείξεις μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-43.B.

**BaaSAPI-43.Γ.** Ο έλεγχος αυτός επαληθεύει ότι αν τα κριτήρια αναζήτησης που έχουν να κάνουν με τις ημερομηνίες έχουν λανθασμένη μορφή, ενημερώνει τον χρήστη ότι έχει κάνει

κάποιο λάθος. Το αναμενόμενο αποτέλεσμα είναι status 400 και τα μηνύματα είναι «Retrieve receipts operation failed» και «Not a valid date».

**BaaSAPI-43.Δ.** Ο έλεγχος αυτός επαληθεύει ότι σε περίπτωση εσωτερικής αποτυχίας του συστήματος να ανταποκριθεί στο αίτημα του χρήστη, επιστρέφεται αποτέλεσμα με status 500 και μήνυμα «Retrieve receipts operation failed».

**Διεπαφή BaaSAPI-44.** Αυτή η διεπαφή χρησιμοποιείται από έναν αγοραστή προϊόντων για να ανασύρει όλες τις αποδείξεις των προϊόντων που έχει πουλήσει.

**BaaSAPI-44.A.** Ο έλεγχος αυτός επιβεβαιώνει την επιτυχημένη αναζήτηση και την ανάκτηση των αποδείξεων συναλλαγών από την πλευρά του αγοραστή. Η αναζήτηση αυτή γίνεται με βάση τα κριτήρια που υπάρχουν στο σώμα ελέγχου, όπως φαίνεται στην *Εικόνα 114*. Το αναμενόμενο αποτέλεσμα είναι status 200 μαζί με μια λίστα όλων των αποδείξεων, όπως φαίνεται στην *Εικόνα 112*.

```
{
  "user": "zisis",
  "product_id": "productX1",
  "fromDate": "2024-01-11T09:22:52.000Z",
  "toDate": "2024-01-15T09:22:52.000Z"
}
```

*Εικόνα 114: Σώμα κλήσης του ελέγχου BaaSAPI-44.A.*

**BaaSAPI-44.B.** Ο έλεγχος αυτός επιβεβαιώνει την ορθή αναζήτηση των αποδείξεων συναλλαγών, αλλά σε αυτήν την περίπτωση δεν υπάρχουν αποδείξεις που να ικανοποιούν τα κριτήρια αναζήτησης τους, οπότε και επιστρέφεται μια κενή λίστα, όπως φαίνεται στην *Εικόνα 113*.

**BaaSAPI-44.Γ.** Ο έλεγχος αυτός επαληθεύει ότι αν τα κριτήρια αναζήτησης που έχουν να κάνουν με τις ημερομηνίες έχουν λανθασμένη μορφή, ενημερώνει τον χρήστη ότι έχει κάνει κάποιο λάθος. Το αναμενόμενο αποτέλεσμα είναι status 400 και τα μηνύματα είναι «Retrieve receipts operation failed» και «Not a valid date».

**BaaSAPI-44.Δ.** Ο έλεγχος αυτός επαληθεύει ότι σε περίπτωση εσωτερικής αποτυχίας του συστήματος να ανταποκριθεί στο αίτημα του χρήστη, επιστρέφεται αποτέλεσμα με status 500 και μήνυμα «Retrieve receipts operation failed».

**Διεπαφή BaaSAPI-45.** Αυτή η διεπαφή χρησιμοποιείται από τον διαχειριστή του συστήματος, οποίος μπορεί να αναζητήσει τις αποδείξεις συναλλαγών, είτε αγορών, είτε πωλήσεων, για ένα συγκεκριμένο προϊόν.

**BaaSAPI-45.A.** Ο έλεγχος αυτός επιβεβαιώνει ότι ο διαχειριστής του συστήματος μπορεί να ανασύρει τις αποδείξεις, άλλων χρηστών είτε αυτοί ενεργούν ως πωλητές, είτε ως αγοραστές, ενός συγκεκριμένου προϊόντος. Παράδειγμα του σώματος ελέγχου φαίνεται στην *Εικόνα 114*, ενώ το αναμενόμενο αποτέλεσμα είναι status 200 μαζί με μια λίστα όλων των αποδείξεων, όπως φαίνεται στην *Εικόνα 112*.

**BaaSAPI-45.B., BaaSAPI-45.Γ., BaaSAPI-45.Δ.** Οι έλεγχοι αυτοί επιβεβαιώνουν τις ίδιες περιπτώσεις με τις διεπαφές **BaaSAPI-43.B., BaaSAPI-43.Γ., BaaSAPI-43.Δ** και **BaaSAPI-44.B., BaaSAPI-44.Γ., BaaSAPI-44.Δ.**

**Διεπαφή BaaSAPI-46.** Αυτή η διεπαφή είναι υπεύθυνη για τον έλεγχο εγκυρότητας μιας απόδειξης συναλλαγής και μπορεί να εκτελεστεί είτε από πωλητή, είτε από κάποιον αγοραστή.

**BaaSAPI-46.A.** Ο έλεγχος αυτός επαληθεύει τον επιτυχημένο έλεγχο εγκυρότητας μιας απόδειξης συναλλαγής. Το σώμα του ελέγχου φαίνεται στην *Εικόνα 115* και το αναμενόμενο αποτέλεσμα είναι status 200 και συνοδεύεται από ένα αντικείμενο που έχει περιλαμβάνει την λογική τιμή true, όπως φαίνεται στην *Εικόνα 116*.

```
{
  "receipt": {
    "TransactionID": "2a5be3986c00a574b435f8ce0f914b32bec3bd80887759b9d45474302b038bae",
    "Buyer": "zisis",
    "Seller": "avraam",
    "ProductID": "productX1",
    "DateBought": "2024-01-12T09:22:52.000Z"
  },
  "user_id": "zisis"
}
```

*Εικόνα 115: Σώμα κλήσης του ελέγχου BaaSAPI-46.A.*

```
{
  "Verified": true
}
```

*Εικόνα 116: Επιστρεφόμενο αντικείμενο μετά από επιτυχημένο έλεγχο εγκυρότητας μιας απόδειξης.*

**BaaSAPI-46.B.** Ο έλεγχος αυτός επαληθεύει την μη εγκυρότητα μιας απόδειξης συναλλαγής. Το αναμενόμενο αποτέλεσμα είναι status 200 και συνοδεύεται από ένα αντικείμενο που έχει περιλαμβάνει την λογική τιμή false, όπως φαίνεται στην *Εικόνα 117*.

```
{
  "Verified": false
}
```

*Εικόνα 117: Επιστρεφόμενο αντικείμενο μετά από μη επιτυχημένο έλεγχο εγκυρότητας μιας απόδειξης.*

**BaaSAPI-46.Γ.** Ο έλεγχος αυτός επιβεβαιώνει την αδυναμία να εκτελεστεί αυτή λειτουργία του συστήματος και το αναμενόμενο αποτέλεσμα είναι status 500 μαζί με το μήνυμα «Verify receipt operation failed».

**Διεπαφή BaaSAPI-47.** Αυτή η διεπαφή είναι υπεύθυνη για τον έλεγχο των καταγραφών χρήσεων των προϊόντων από χρήστες και μπορεί να εκτελεστεί μόνο από τον διαχειριστή του συστήματος.

**BaaSAPI-47.A.** Ο έλεγχος αυτός επαληθεύει την επιτυχημένη αναζήτηση των καταγραφών χρήσης ενός προϊόντος από ένα χρήστη. Το σώμα του ελέγχου φαίνεται στην *Εικόνα 118*. Το αναμενόμενο αποτέλεσμα είναι status 200 μαζί με μια λίστα όλων των σχετικών καταγραφών και φαίνονται στην *Εικόνα 119*.

```
{
  "user": "zisis",
  "product_id": "productX1",
  "fromDate": "2024-01-11T09:22:52.000Z",
  "toDate": "2024-01-15T09:22:52.000Z"
}
```

*Εικόνα 118: Σώμα κλήσης του ελέγχου BaaSAPI-47.A.*

```

{
  "Logs": [
    {
      "TransactionID": "586dd0a1e895bd4ee6bb668d73713ddcf164e8526c51d22b11be9506a0838c1a",
      "User": "zisis",
      "ProductID": "productX1",
      "DateAccessed": "2024-01-12T10:14:35.000Z"
    },
    {
      "TransactionID": "c24a663f09155659c8d27a6efeed8487984a74b2d47a1cc58b326d17818034f4",
      "User": "zisis",
      "ProductID": "productX1",
      "DateAccessed": "2024-01-12T10:14:55.000Z"
    },
    {
      "TransactionID": "0b3d9d654431232c90fed6aa860521096d43d58e8003a65d36e91d6ac29ea2f1",
      "User": "zisis",
      "ProductID": "productX1",
      "DateAccessed": "2024-01-12T10:53:16.000Z"
    }
  ]
}

```

Εικόνα 119: Επιστρεφόμενο αντικείμενο που περιλαμβάνει τις καταγραφές χρήσεων προϊόντος, μετά από επιτυχημένη κλήση του αντίστοιχου ελέγχου.

**BaaSAPI-47.B.** Ο έλεγχος αυτός επιβεβαιώνει την ορθή αναζήτηση των καταγραφών χρήσεων, αλλά σε αυτήν την περίπτωση δεν υπάρχουν καταγραφές που να ικανοποιούν τα κριτήρια αναζήτησης τους, οπότε και επιστρέφεται μια κενή λίστα., όπως φαίνεται στην Εικόνα 120.

```

{
  "Logs": []
}

```

Εικόνα 120: Κενή λίστα αποδείξεων όταν δεν υπάρχουν διαθέσιμες καταγραφές χρήσεων ενός προϊόντος, μετά από επιτυχημένη κλήση του ελέγχου BaaSAPI-47.B.

**BaaSAPI-47.Γ., BaaSAPI-47.Δ.** Οι έλεγχοι αυτοί επιβεβαιώνουν τις ίδιες περιπτώσεις με τις διεπαφές **BaaSAPI-44.Γ., BaaSAPI-44.Δ.**, με τη διαφορά ότι το μήνυμα είναι το ακόλουθο «Retrieve logs operation failed».

**Διεπαφή BaaSAPI-48.** Η διεπαφή αυτή είναι υπεύθυνη για τον έλεγχο εγκυρότητας μια καταγραφής πρόσβασης για ένα προϊόν και εκτελείται μόνο από τον διαχειριστή του συστήματος.

**BaaSAPI-48.A.** Ο έλεγχος αυτός επαληθεύει τον επιτυχημένο έλεγχο εγκυρότητας μιας καταγραφής χρήσης. Το σώμα του ελέγχου φαίνεται στην Εικόνα 121 και το αναμενόμενο αποτέλεσμα είναι status 200 και συνοδεύεται από ένα αντικείμενο που έχει περιλαμβάνει την λογική τιμή true, όπως φαίνεται στην Εικόνα 122.

```

{
  "log": {
    "TransactionID": "c24a663f09155659c8d27a6efeed8487984a74b2d47a1cc58b326d17818034f4",
    "User": "zisis",
    "ProductID": "productX1",
    "DateAccessed": "2024-01-12T10:14:55.000Z"
  },
  "user_id": "iotFedsAdmin"
}

```

Εικόνα 121: Σώμα κλήσης του ελέγχου BaaSAPI-48.A.

```

{
  "Verified": true
}

```

Εικόνα 122: Επιστρεφόμενο αντικείμενο μετά από επιτυχημένο έλεγχο εγκυρότητας μιας καταγραφής χρήσης προϊόντος.

**BaaSAPI-48.B.** Ο έλεγχος αυτός επαληθεύει την μη εγκυρότητα μιας καταγραφής χρήσης ενός προϊόντος. Το αναμενόμενο αποτέλεσμα είναι status 200 και συνοδεύεται από ένα αντικείμενο που έχει περιλαμβάνει την λογική τιμή false, όπως φαίνεται στην Εικόνα 123.

```

{
  "Verified": false
}

```

Εικόνα 123: Επιστρεφόμενο αντικείμενο μετά από μη επιτυχημένο έλεγχο εγκυρότητας μιας καταγραφής χρήσης προϊόντος..

**BaaSAPI-48.Γ.** Ο έλεγχος αυτός επιβεβαιώνει την αδυναμία να εκτελεστεί αυτή λειτουργία του συστήματος και το αναμενόμενο αποτέλεσμα είναι status 500 μαζί με το μήνυμα «Verify receipt operation failed».

Πίνακας 15: Περιπτώσεις δοκιμής στοιχείου BaaS για την καταγραφή και τον έλεγχο συναλλαγών

Interface	Test case	Method	Call	Status
BaaSAPI-43	BaaSAPI-43.A	POST	<BaaS-API-url-auditing>/get_receipts_seller	PASS
	BaaSAPI-43.B	POST	<BaaS-API-url-auditing>/get_receipts_seller	PASS
	BaaSAPI-43.Γ	POST	<BaaS-API-url-auditing>/get_receipts_seller	PASS
	BaaSAPI-43.Δ	POST	<BaaS-API-url-auditing>/get_receipts_seller	PASS
BaaSAPI-44	BaaSAPI-44.A	POST	<BaaS-API-url-auditing>/get_receipts_buyer	PASS
	BaaSAPI-44.B	POST	<BaaS-API-url-auditing>/get_receipts_buyer	PASS
	BaaSAPI-44.Γ	POST	<BaaS-API-url-auditing>/get_receipts_buyer	PASS
	BaaSAPI-44.Δ	POST	<BaaS-API-url-auditing>/get_receipts_buyer	PASS
BaaSAPI-45	BaaSAPI-45.A	POST	<BaaS-API-url-auditing>/get_receipts_admin	PASS
	BaaSAPI-45.B	POST	<BaaS-API-url-auditing>/get_receipts_admin	PASS
	BaaSAPI-45.Γ	POST	<BaaS-API-url-auditing>/get_receipts_admin	PASS
	BaaSAPI-45.Δ	POST	<BaaS-API-url-auditing>/get_receipts_admin	PASS

BaaSAPI-46	BaaSAPI-46.A	POST	<BaaS-API-url-auditing>/verify_receipt	PASS
	BaaSAPI-46.B	POST	<BaaS-API-url-auditing>/verify_receipt	PASS
	BaaSAPI-46.Γ	POST	<BaaS-API-url-auditing>/verify_receipt	PASS
BaaSAPI-47	BaaSAPI-47.A	POST	<BaaS-API-url-auditing>/get_logs	PASS
	BaaSAPI-47.B	POST	<BaaS-API-url-auditing>/get_logs	PASS
	BaaSAPI-47.Γ	POST	<BaaS-API-url-auditing>/get_logs	PASS
	BaaSAPI-47.A	POST	<BaaS-API-url-auditing>/get_logs	PASS
BaaSAPI-48	BaaSAPI-48.A	POST	<BaaS-API-url-auditing>/verify_access_log	PASS
	BaaSAPI-48.B	POST	<BaaS-API-url-auditing>/verify_access_log	PASS
	BaaSAPI-48.Γ	POST	<BaaS-API-url-auditing>/verify_access_log	PASS

#### 3.3.3.4.6 Δοκιμή διεπαφών Φήμης

Οι διεπαφές αυτής της υπο-ενότητας αφορούν τις διαδικασίες υπολογισμού φήμης όλων των οντοτήτων στο BaaS. Για την δοκιμή των διεπαφών αυτής της υπο-ενότητας, αξιοποιούνται οι χρήστες, συσκευές και προϊόντα που δημιουργήθηκαν στην προηγούμενη, συγκεκριμένα, το σύστημα βρίσκεται στην κατάσταση του τελευταίου ελέγχου.

**Διεπαφή BaaSAPI-42.** Χρησιμοποιείται για την ενημέρωση των μετρικών ποιότητας και συνεπώς της αντικειμενικής φήμης μιας συσκευής

**BaaSAPI-42.A.** Αυτός ο έλεγχος επαληθεύει ότι οι μετρικές ποιότητας μια πηγής δεδομένων μπορούν να ενημερωθούν με επιτυχία. Το σώμα είναι αυτό στην *Εικόνα 124*, και το αναμενόμενο αποτέλεσμα είναι status 200 με μήνυμα OK.

```

{
  "qos_list": [
    {
      "resource_id": "resource",
      "qos": {
        "availability": 0.93,
        "response_time": 0.72,
        "precision": 0.8
      }
    }
  ]
}

```

Εικόνα 124 Σώμα κλήσης του ελέγχου BaaSAPI-42.A.

**Διεπαφή BaaSAPI-40.** Χρησιμοποιείται για την αξιολόγηση ενός προϊόντος και συνεπώς ενημέρωσης της υποκειμενικής φήμης μιας συσκευής από έναν χρήστη

**BaaSAPI-40.A.** Ο έλεγχος αυτός επαληθεύει ότι ένας χρήστης δε μπορεί να αξιολογήσει προϊόν για το οποίο δεν έχει κουπόνι πρόσβασης. Μετά την ανταλλαγή της προηγούμενης ενότητας,



για το «productX1» διαθέτει κουπόνι μόνο ο χρήστης «exchange\_user3», οπότε μια κλήση με το σώμα στην *Εικόνα 125*, αναμένεται να δώσει status 403 με μήνυμα «A user cannot vote for a product he has no access to».

```
{
  "product_id": "productX1",
  "user_id": "exchange_user",
  "rating": {
    "ease_of_use": 0.9,
    "value_for_money": 0.9,
    "business_enablement": 0.8,
    "correctness": 0.8,
    "completeness": 0.7,
    "relevance": 0.4,
    "response_time": 0.7,
    "precision": 0.6
  }
}
```

*Εικόνα 125 Σώμα κλήσης ελέγχου BaaSAPI-40.A*

**BaaSAPI-40.B.** Αυτός ο έλεγχος επαληθεύει ότι η αξιολόγηση ενός χρήστη μπορεί να πραγματοποιηθεί κανονικά ενημερώνοντας τον υποκειμενικό βαθμό ενός προϊόντος. Το σώμα του ελέγχου είναι αυτό στην *Εικόνα 126*, και η κλήση του αναμένεται να δώσει status 200 με μήνυμα OK. Μετά τον έλεγχο επαληθεύεται μέσω της διεπαφής «getProduct» ότι το πεδίο υποκειμενικής φήμης «subjReputation» του προϊόντος έχει ενημερωθεί, Όπως φαίνεται στην *Εικόνα 127*.

```
{
  "product_id": "productX1",
  "user_id": "exchange_user3",
  "rating": {
    "ease_of_use": 0.9,
    "value_for_money": 0.9,
    "business_enablement": 0.8,
    "correctness": 0.8,
    "completeness": 0.7,
    "relevance": 0.4,
    "response_time": 0.7,
    "precision": 0.6
  }
}
```

*Εικόνα 126 Σώμα ελέγχου BaaSAPI-40.B*

```

{
  "FedMarketplace_id": "eketaFED",
  "GlobalMarketplace_id": true,
  "Price": 0.5,
  "Product_details": {},
  "Product_id": "productX1",
  "Reputation": null,
  "Resource_ids": {
    "resource": "exchange"
  },
  "Seller": "exchange_user",
  "docType": "Product",
  "subjReputation": [
    0.7100000000000001
  ],
  "transactionCounter": 2
}

```

Εικόνα 127 Προϊόν μετά την αξιολόγηση από χρήστη

**Διεπαφή BaaSAPI-41.** Χρησιμοποιείται για την ενημέρωση φήμης όλων των οντοτήτων στο σύστημα, συνδυάζοντας αντικειμενικές και υποκειμενικές μετρικές.

**BaaSAPI-41.A.** Αυτός ο έλεγχος επαληθεύει ότι η κλήση της διεπαφής BaaSAPI-41 ενημερώνει τα πεδία φήμης «reputation» των διαφόρων οντοτήτων. Η διεπαφή καλείται χωρίς σώμα και η αναμενόμενη απάντηση είναι status 200 με μήνυμα OK. Μετά τον έλεγχο επαληθεύεται μέσω της διεπαφής «getProduct» ότι το πεδίο φήμης του προϊόντος «productX1» έχει ενημερωθεί, Όπως φαίνεται στην Εικόνα 128.

```

{
  "FedMarketplace_id": "eketaFED",
  "GlobalMarketplace_id": true,
  "Price": 0.5,
  "Product_details": {},
  "Product_id": "productX1",
  "Reputation": 0.35500000000000004,
  "Resource_ids": {
    "resource": "exchange"
  },
  "Seller": "exchange_user",
  "docType": "Product",
  "subjReputation": [
    0.7100000000000001
  ],
  "transactionCounter": 2
}

```

Εικόνα 128: Προϊόν μετά από ενημέρωση όλων των πεδίων φήμης.

Πίνακας 16: Σενάρια Ελέγχου για τις διεπαφές φήμης.

Interface	Test case	Method	Call	Status
BaaSAPI-42	BaaSAPI-42.A	PATCH	<BaaS-API-url-products>/rate_product	PASS
BaaSAPI-40	BaaSAPI-40.A	PATCH	<BaaS-API-url-products>/update_resource_qos	PASS
	BaaSAPI-40.B	PATCH	<BaaS-API-url-products>/update_resource_qos	PASS
BaaSAPI-41	BaaSAPI-41.A	PATCH	<BaaS-API-url-products>/update_all_reputations	PASS

## 4 Λειτουργική δοκιμή και επίδειξη συστήματος

Στο κεφάλαιο αυτό παρουσιάζεται η αλληλεπίδραση του χρήστη με την IoTFeds πλατφόρμα μέσω των γραφικών διεπαφών χρήστη Administration GUI για τη διαχείριση ομοσπονδιών (καθώς και χρηστών και πλατφορμών) και την πύλη αγοράς Marketplace. Τα επιτυχή τεστς επισημαίνονται με την ένδειξη “PASS”.

### 4.1 Λειτουργική δοκιμή και επίδειξη πύλης αγοράς

Οι λειτουργικότητες που υποστηρίζονται στην πύλη της αγοράς και την γραφική διεπαφή της (marketplace GUI) είναι οι παρακάτω:

- Σύνδεση στην πύλη αγοράς ενός εγγεγραμμένου στην IoTFeds πλατφόρμα χρήστη.
- Πρόσβαση στη λίστα των αγορών που συμμετέχει ο χρήστης.
- Πρόσβαση στο πορτοφόλι (wallet) του IoTFeds χρήστη.
- Σύνθεση προϊόντων στην επισκεπτόμενη ομόσπονδη αγορά με αναζήτηση ομόσπονδων IoT πορων.
- Αναζήτηση προϊόντων από έναν IoTFeds χρήστη (IoT πάροχο ή καταναλωτή) με βάση τα επιθυμητά κριτήρια αναζήτησης.
- Έλεγχος τιμής και αγορά προϊόντος από έναν καταναλωτή IoT δεδομένων.
- Ανταλλαγή κουπονιών προϊόντων.
- Έλεγχος συναλλαγών και αποδεικτικών χρήσης.

#### MarketplaceGUI-1: Σύνδεση χρήστη στην IoT marketplace πλατφόρμα PASS

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να συνδεθεί επιτυχώς στην IoT marketplace πλατφόρμα.

Ο χρήστης έχει πρόσβαση στις υπηρεσίες του marketplace και τις πληροφορίες του.

##### Προαπαιτούμενο:

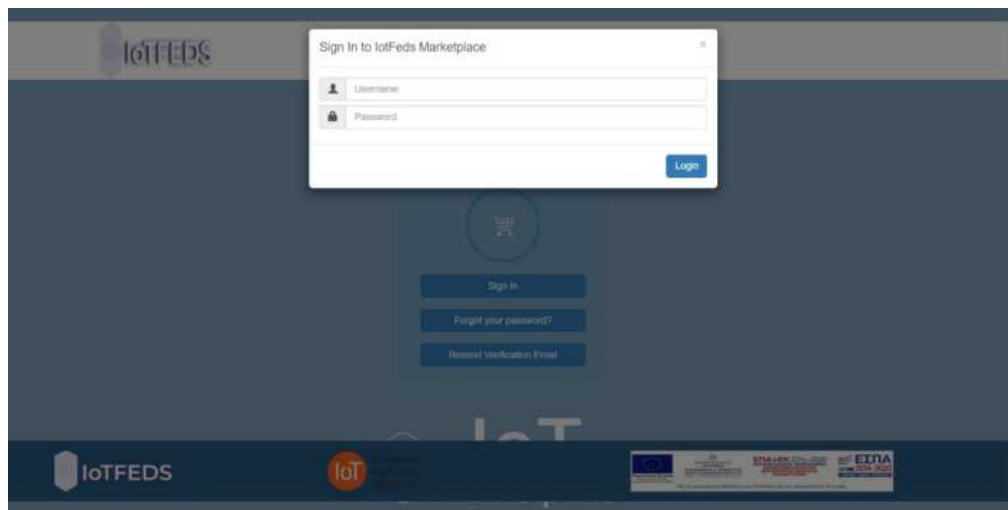
Ο χρήστης έχει εγγραφεί στην IoTFeds πλατφόρμα μέσω της γραφικής διεπαφής ομοσπονδιών Adminsitration GUI (δοκιμή AdminGUI-1 που περιγράφεται στην επόμενη ενότητα).

##### Αναμενόμενα αποτελέσματα:

Η φόρμα για τη συμπλήρωση των διαπιστευτηρίων του χρήστη εμφανίζεται όταν ο χρήστης πατήσει το κουμπί “Sign in” από το κεντρικό μενού.

##### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



Ο χρήστης εισάγει τα διαπιστευτήρια του στην φόρμα και πατάει το κουμπί “Sign In”.

**Αναμενόμενα αποτελέσματα:**

Η σύνδεση του χρήστη στην πύλη αγοράς είναι επιτυχής με τα διαπιστευτήρια του χρήστη με τα οποία έχει εγγραφεί στην IoTFeds πλατφόρμα.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



**MarketplaceGUI-2: Ο χρήστης έχει πρόσβαση στο πορτοφόλι του.**

**PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να ανακτήσει τις πληροφορίες σχετικά με το πορτοφόλι (wallet) του.

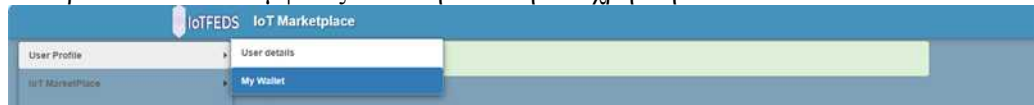
Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “User Profile”.

**Αναμενόμενα αποτελέσματα:**

Εμφανίζεται το μενού με τις επιλογές στον χρήστη.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



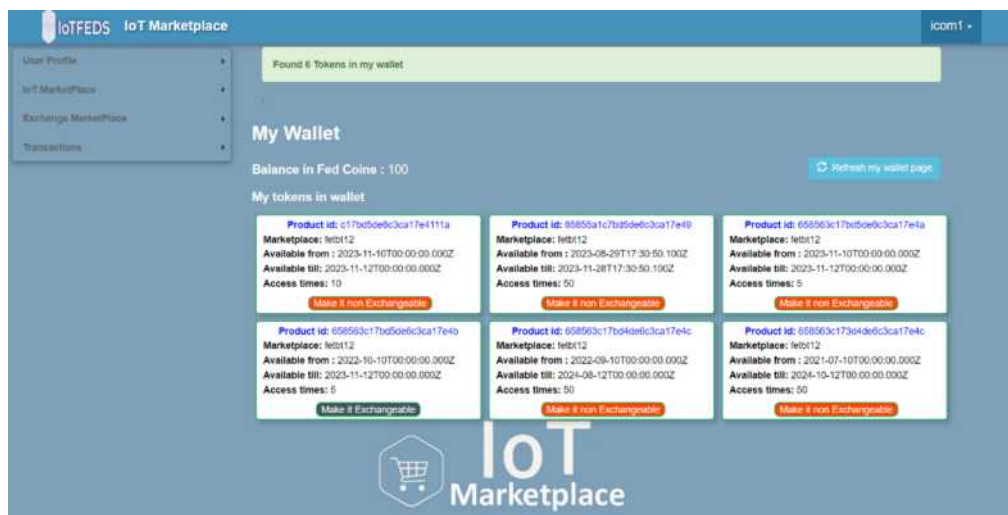
Ο χρήστης επιλέγει την επιλογή “My Wallet”.

**Αναμενόμενα αποτελέσματα:**

Στην σελίδα εμφανίζεται μια λίστα με τα αγορασμένα κουπόνια/προϊόντα και την αξία τους σε κρυπτονομίσματα FedCoins που έχει στην κατοχή του ο χρήστης.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



### MarketplaceGUI-3: Ο χρήστης επιλέγει να συνθέσει προϊόν σε μία ομοσπονδιακή αγορά

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να δημιουργήσει ένα προϊόν για τη διάθεση του στην αγορά επιλέγοντας από τους διαθέσιμους ομόσπονδους IoT πόρους.

Ο χρήστης επιλέγει το κεντρικό μενού την επιλογή “IoT Marketplace”.

**Αναμενόμενα αποτελέσματα:**

Το μενού με την επιλογή για τη σύνθεση προϊόντος “Create Product” εμφανίζεται στον χρήστη.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



Ο χρήστης επιλέγει τη σύνθεση προϊόντος “Create Product” και την επιθυμητή επισκεπτόμενη ομόσπονδη αγορά.

**Αναμενόμενα αποτελέσματα:**

Εμφανίζεται η φόρμα εισαγωγής κριτηρίων αναζήτησης IoT πόρων στην επιλεγμένη ομόσπονδη αγορά.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στον χρήστη όπου έχει τη δυνατότητα να επιλέξει τα επιθυμητά κριτήρια αναζήτησης IoT πόρων που τον ενδιαφέρουν για τη σύνθεση προϊόντος.

Ο χρήστης εισάγει στη φόρμα αναζήτησης τα κριτήρια που τον ενδιαφέρουν και επιλέγει “Search Resources”.

**Αναμενόμενα αποτελέσματα:**

Εφόσον υπάρχουν IoT πόροι που να πληρούν τα κριτήρια, εμφανίζεται η λίστα στην οθόνη του χρήστη.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στο χρήστη με τη λίστα από IoT πόρους που ικανοποιούν τα κριτήρια αναζήτησης.

The screenshot displays the 'IoT Marketplace' interface. On the left is a sidebar with navigation links: 'User Profile', 'IoT Marketplace', 'Exchange Marketplace', and 'Trends/Alerts'. The main content area is titled 'Found 4 L2 resources'. Below this is a search bar 'Q Search Federated Resources in "SmartTraffic" Marketplace'. The search filters are organized into sections: 'Observed properties' (with a dropdown showing 'humidity, temperature'), 'Platform Name' (with a text input), 'Resource Name' (with a text input), 'Resource Id' (with a text input), 'Resource Type' (with a dropdown showing 'stationary'), and 'Platform Id' (with a text input). There are also input fields for 'Min Price', 'Max Price', 'Min Reputation', and 'Max Reputation'. A 'Location search criteria' section includes a 'Location' input and a 'Searching area, specify coordinates and radius' section with 'Longitude', 'Latitude', and 'Maximum distance' inputs. A 'Show marketplaces' button is at the bottom of the search filters. Below the search filters is a section titled 'Created Product Specifications' which contains four product cards: 'Weather station sensor', 'Meteo station sensor', 'Traffic data sensor', and 'Air Quality sensor'. Each card has an 'Add to product' button.

Ο χρήστης ενημερώνεται σχετικά με τα χαρακτηριστικά ενός IoT πόρου επιλέγοντας τον και μπορεί να τον προσθέσει στο προϊόν πατώντας το κουμπί "Add to product", ενώ συμπληρώνει στη συνέχεια τις πληροφορίες του προϊόντος.

**Αναμενόμενα αποτελέσματα:**

Οι IoT πόροι που επέλεξε ο χρήστης εμφανίζονται στην περιοχή "List of product Resources" καθώς και οι σχετικές πληροφορίες του προϊόντος.

**Αποτελέσματα που παρατηρήθηκαν:**

Στη σελίδα εμφανίζονται τα χαρακτηριστικά του IoT προϊόντος που αναφέρονται στην τιμή του, το χρονικό διάστημα χρήσης του και τη συχνότητα παροχής ροών

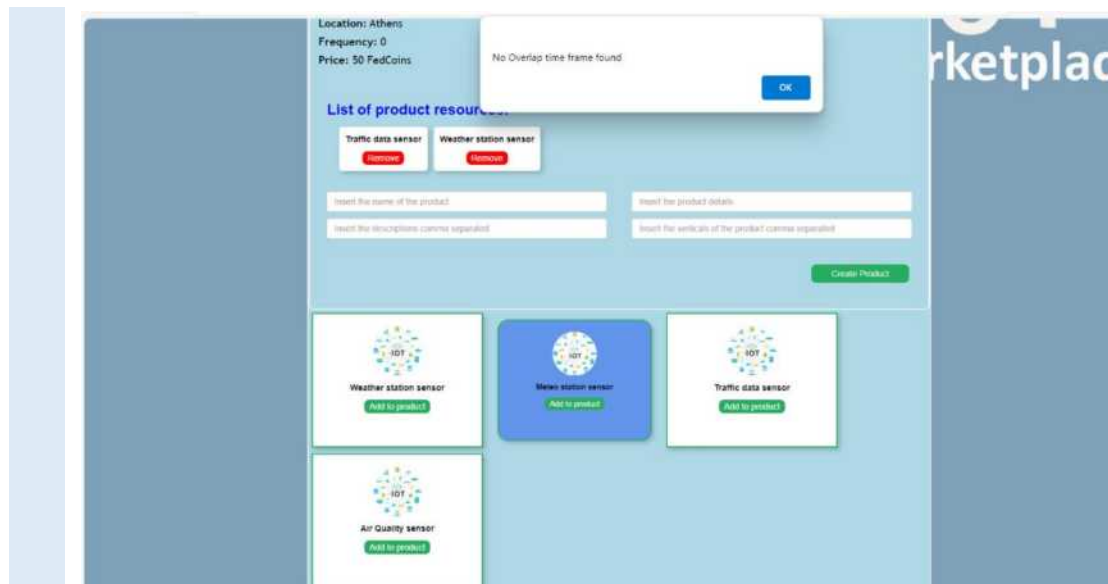


δεδομένων. Ο χρήστης έχει τη δυνατότητα να αφαιρέσει έναν IoT πόρο από τη λίστα σύνθεσης του IoT προϊόντος επιλέγοντας το κουμπί “Remove”.

The screenshot displays the IoT Marketplace interface. At the top, a navigation bar includes 'User Profile', 'IoT Marketplace', 'Exchange Marketplace', and 'Transactions'. A green banner indicates 'Found 4 L2 resources'. Below this is a search bar for 'SmartTraffic Marketplace'. The search filters are organized into sections: 'Observed properties' (humidity, temperature), 'Resource Name', 'Resource Id', 'Resource Type' (stationary), 'Platform Name', 'Platform Id', 'Min Price', 'Max Price', 'Min Reputation', 'Max Reputation', and 'Location search criteria' (Location, Searching area, Longitude, Latitude, Maximum distance). A 'Show marketplaces' button and a 'Search Resources' button are present. Below the search filters, a section titled 'Created Product Specifications' shows details for a product: 'Valid from: 12-04-2021', 'Valid till: 14-09-2023', 'Frequency: 5', and 'Price: 1070'. The 'Resource name' is 'Weather station sensor', with measures of humidity, temperature, and atmospheric pressure. The description states: 'Meteo station measures weather data 24h in Attica'. The valid date range is '12-08-2021 to: 01-09-2023', the location is 'Saronida', the frequency is '5', and the price is '1000 FedCoins'. A 'List of product resources' section shows two resources: 'Traffic data sensor' and 'Weather station sensor', each with a 'Remove' button. At the bottom, there are input fields for 'Insert the name of the product', 'Insert the product details', 'Insert the descriptions comma separated', and 'Insert the vertices of the product comma separated', followed by a 'Create Product' button.

Σημειώνεται ότι στην περίπτωση που ένας IoT πόρος που επιλέγει ο χρήστης δεν είναι συμβατός με τους IoT πόρους που έχουν ήδη προστεθεί στο προϊόν (π.χ. χρονικό διάστημα διαθεσιμότητας) τότε εμφανίζεται μήνυμα λάθους, όπως φαίνεται στην παρακάτω σελίδα:





Εφόσον ολοκληρωθεί η επιτυχής επιλογή όλων των απαραίτητων IoT πόρων και η συμπλήρωση των πληροφοριών του προϊόντος, ο χρήστης επιλέγει το κουμπί “Create Product”.

**Αναμενόμενα αποτελέσματα:**

Στην περίπτωση επιτυχούς σύνθεσης του IoT προϊόντος, τότε εμφανίζεται στην οθόνη του χρήστη μήνυμα επιβεβαίωσης της σύνθεσης του προϊόντος, διαφορετικά εμφανίζεται μήνυμα λάθους.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα είναι το αποτέλεσμα της επιτυχούς σύνθεσης προϊόντος:

IoTFEDS IoT Marketplace

User Profile

IoT Marketplace

Exchange Marketplace

Transactions

The product created with id: 655dbdd4104bf30ec2802ee9

Q Search Federated Resources in "SmartTraffic" Marketplace

Observed properties

Platform Name

humidity,temperature

Insert the platform name

Resource Name

Resource Id

Insert the resource name

Insert the resource id

Resource Type

Platform Id

stationary

smart-city

Min Price:

Max Price:

Min Reputation:

Max Reputation:

Location search criteria

Location:

Insert the location of the resource

Searching area, specify coordinates and radius

Longitude

Latitude

Maximum distance

longitude of point of interest

latitude of point of interest

max distance from point of interest

Show marketplaces

Q Search R

Created Product Specifications

Resource name: Traffic data sensor

Measures: humidity, number\_of\_bikes, number\_of\_cars

Description: Station collects traffic data 24h in Attica

Valid date from: 12-04-2021 to: 14-09-2023

Location: Syntagma

Frequency: 15

Price: 70 FedCoins

List of product resources:

Weather station sensor

Traffic data sensor

Remove

Remove

Smart Product

An IoT product about traffic and Weather Data in Attic area

traffic and weather data

Smart Environment

Create Prod

IoT

Weather station sensor

Add to product

IoT

Meteo station sensor

Add to product

IoT

Traffic data sensor

Add to product

IoT

Air Quality sensor

Add to product

#### MarketplaceGUI-4: Ο χρήστης επιλέγει την αναζήτηση προϊόντος και αγορά ή διαγραφή.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να ανακαλύψει τα διαθέσιμα προϊόντα και να επιλέξει την αγορά του επιθυμητού προϊόντος

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “IoT Marketplace” και στη συνέχεια “Search and Buy Products”.

##### Αναμενόμενα αποτελέσματα:

Εμφανίζεται το μενού αναζήτησης προϊόντος στον χρήστη για την επισκεπτόμενη αγορά.

##### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:

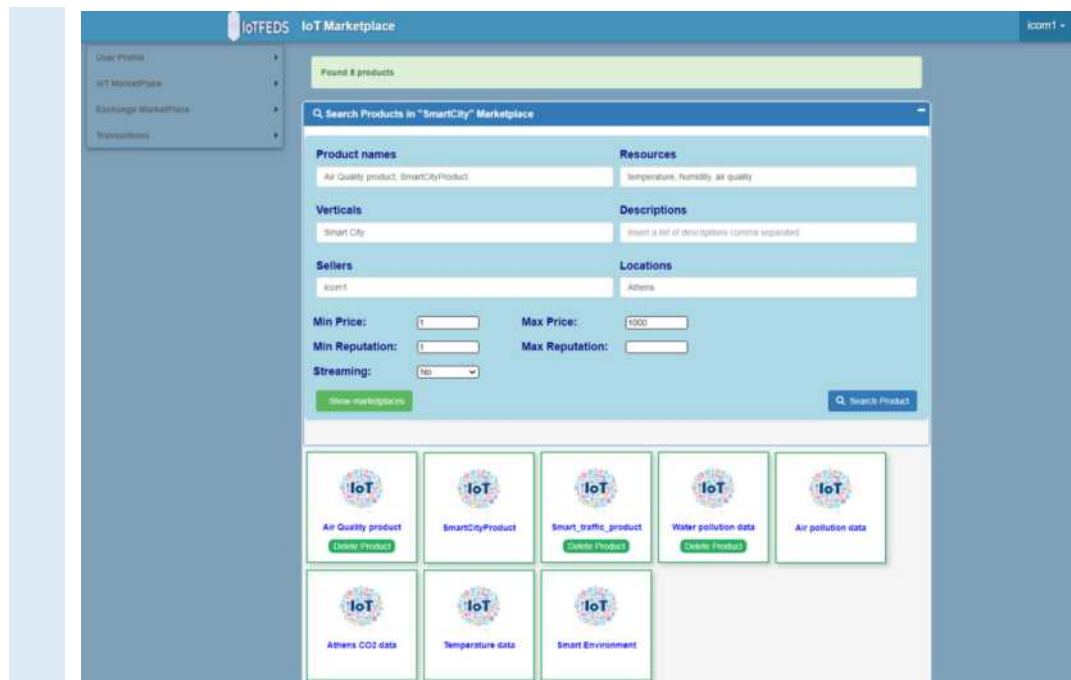
Ο χρήστης εισάγει στη φόρμα κριτήρια αναζήτησης προκειμένου να ανακαλύψει IoT προϊόντα που τον ενδιαφέρουν για αγορά ή διαγραφή (σε περίπτωση δικών του προϊόντων) και στο τέλος επιλέγει το button “Search Product”.

##### Αναμενόμενα αποτελέσματα:

Στην οθόνη του χρήστη εμφανίζονται IoT προϊόντα εφόσον ικανοποιούν τα κριτήρια αναζήτησης που έχει εισάγει ο χρήστης στη φόρμα.

##### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη μαζί με μια λίστα από IoT προϊόντα που ικανοποιούν τα κριτήρια αναζήτησης.



Ο χρήστης επιλέγει το IoT προϊόν που τον ενδιαφέρει και στη συνέχεια ενεργοποιεί τη διαδικασία διαμόρφωσης του προϊόντος.

**Αναμενόμενα αποτελέσματα:**

Επιλέγοντας ο χρήστης ένα προϊόν από τη λίστα, εμφανίζεται στη σελίδα η φόρμα διαμόρφωσης του προϊόντος.

**Αποτελέσματα που παρατηρήθηκαν:**

Η φόρμα διαμόρφωσης του προϊόντος εμφανίζεται στην οθόνη:

The screenshot displays the IoT Marketplace interface. On the left is a sidebar with navigation links: User Profile, IoT Marketplace, Exchange Marketplace, and Transactions. The main content area is titled 'IoT Marketplace' and shows a search bar with the text 'Found 8 products'. Below the search bar is a form titled 'Search Products in "SmartCity" Marketplace'. The form includes several input fields: 'Product names' (containing 'Air Quality product, SmartCityProduct'), 'Resources' (containing 'temperature, humidity, air quality'), 'Verticals' (containing 'Smart City'), 'Descriptions' (containing 'Insert a list of descriptions comma separated'), 'Sellers' (containing 'icom1'), and 'Locations' (containing 'Athens'). There are also fields for 'Min Price' (1), 'Max Price' (1000), 'Min Reputation' (1), 'Max Reputation' (empty), and a 'Streaming' dropdown set to 'No'. A 'Show marketplaces' button is at the bottom left of the search form, and a 'Search Product' button is at the bottom right. Below the search form, a product card for 'Air Quality product' is shown with details: 'Details: Measures PM2.5, PM1.5, dust, temperature, humidity', 'Streaming Availability: NO', and 'Description: Measures air quality parameters'. Below the product card is a section titled 'Configure Product for observations:' with fields for 'From (minimum date 10-2-2023):', 'To (maximum date 2-8-2023):', 'Number of Observations:', and 'Calculated Price: 0 FedCoins'. There are 'Check Price' and 'Buy Product' buttons at the bottom right of this section. At the bottom of the page, there is a grid of product cards, each with an IoT icon and a 'Delete Product' button. The cards are labeled: 'Air Quality product', 'SmartCityProduct', 'Smart\_traffic\_product', 'Water pollution data', 'Air pollution data', 'Athens CO2 data', 'Temperature data', and 'Smart Environment'.

Ο χρήστης επιλέγει τις παραμέτρους για την αγορά του προϊόντος και ελέγχει την τιμή του επιλέγοντας το κουμπί “Check Price”.

#### **Αναμενόμενα αποτελέσματα:**

Η τιμή που αντιστοιχεί στις επιλεγμένες παραμέτρους εμφανίζεται στην οθόνη.

#### **Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη, ενημερώνοντας για την τιμή του προϊόντος με βάση της παραμέτρους διαμόρφωσης που επέλεξε:

The screenshot displays the IoTMarketplace interface. On the left is a sidebar with navigation links: User Profile, IoT Marketplace, Exchange Marketplace, and Transactions. The main content area has a top navigation bar with the IoT FEDS logo and 'IoT Marketplace'. A green success message at the top states: 'Success, the price of product is calculated'. Below this is a search bar titled 'Search Products in "SmartCity" Marketplace'. The search filters include: Product names (Air Quality product, SmartCityProduct), Resources (temperature, humidity, air quality), Verticals (Smart City), Descriptions (Insert a list of descriptions comma separated), Sellers (icom1), Locations (Athens), Min Price (1), Max Price (1000), Min Reputation (1), Max Reputation (empty), and Streaming (No). A 'Show marketplaces' button and a 'Search Product' button are present. The product details section shows: Product name: Air Quality product, Details: Measures PM2.5, PM1.5, dust, temperature, humidity, Streaming Availability: NO, Description: Measures air quality parameters. Below this is a 'Configure Product for observations' section with fields for 'From (minimum date 10-2-2023):' (20-03-2023), 'To (maximum date 2-8-2023):' (01-06-2023), 'Number of Observations:' (100), and 'Calculated Price:' (316 FedCoins). 'Check Price' and 'Buy Product' buttons are at the bottom right. A grid of product cards is shown at the bottom, including 'Air Quality product', 'SmartCityProduct', 'Smart traffic product', 'Water pollution data', 'Air pollution data', 'Athens CO2 data', 'Temperature data', and 'Smart Environment'. Each card has an 'IoT' logo and a 'Delete Product' button.

Ο χρήστης προχωράει στην αγορά του προϊόντος επιλέγοντας το κουμπί “Buy Product”.

**Αναμενόμενα αποτελέσματα:**

Ο χρήστης ενημερώνεται με μήνυμα επιτυχίας ή αποτυχίας (όπως π.χ. αν δεν έχει αρκετά FedCoins) και αποκτά το κουπόνι πρόσβασης.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στο χρήστη, ενημερώνοντας για την επιτυχή αγορά του προϊόντος:

The screenshot displays the IoTMarketplace interface. At the top, a green notification bar states "The product purchased completely successfully". Below this is a search bar labeled "Search Products in 'SmartCity' Marketplace". The search filters include:

- Product names:** Air Quality product, SmartCityProduct
- Resources:** temperature, humidity, air quality
- Verticals:** Smart City
- Descriptions:** meet a list of descriptions comma separated
- Sellers:** kcm1
- Locations:** Athens
- Min Price:** 1
- Max Price:** 1000
- Min Reputation:** 1
- Max Reputation:**
- Streaming:** No

Buttons for "Show marketplaces" and "Search Product" are present. Below the filters, the details for the selected product "Air Quality product" are shown:

- Product name:** Air Quality product
- Details:** Measures PM2.5, PM1.5, dust, temperature, humidity
- Streaming Availability:** NO
- Description:** Measures air quality parameters

The "Configure Product for observations" section includes:

- From (minimum date 10-2-2023):** 20-03-2023
- To (maximum date 2-8-2023):** 01-06-2023
- Number of Observations:** 100
- Calculated Price:** 316 FedCoins

Buttons for "Check Price" and "Buy Product" are available. At the bottom, a grid of product cards is shown, each with an IoT icon and a "Delete Product" button. The first card is "Air Quality product", which is the product being deleted. Other cards include "SmartCityProduct", "Smart\_traffic\_product", "Water pollution data", "Air pollution data", "Athens CO2 data", "Temperature data", and "Smart Environment".

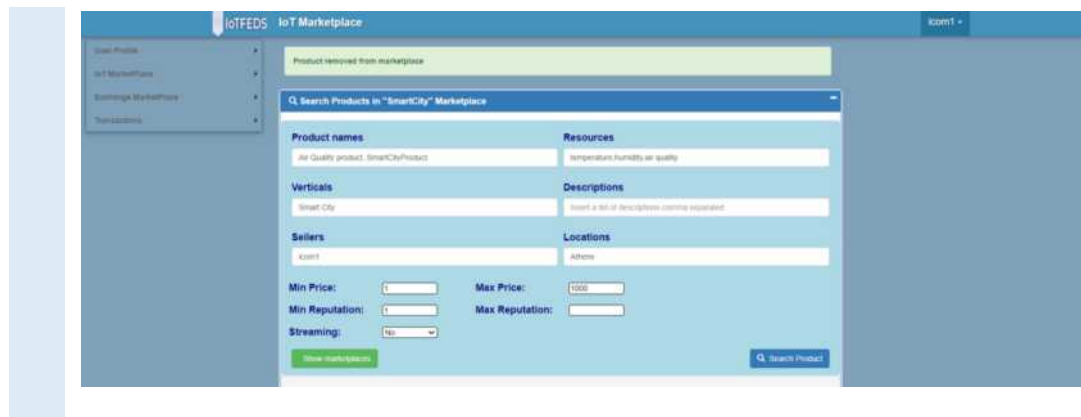
Ο χρήστης διαλέγει από τη λίστα αποτελεσμάτων αναζήτησης προϊόντων να διαγράψει ένα προϊόν που έχει δημιουργήσει επιλέγοντας το κουμπί "Delete".

#### Αναμενόμενα αποτελέσματα:

Στην οθόνη του χρήστη εμφανίζεται μήνυμα επιτυχίας ή αποτυχίας και το προϊόν διαγράφεται από τη λίστα.

#### Αποτελέσματα που παρατηρήθηκαν:

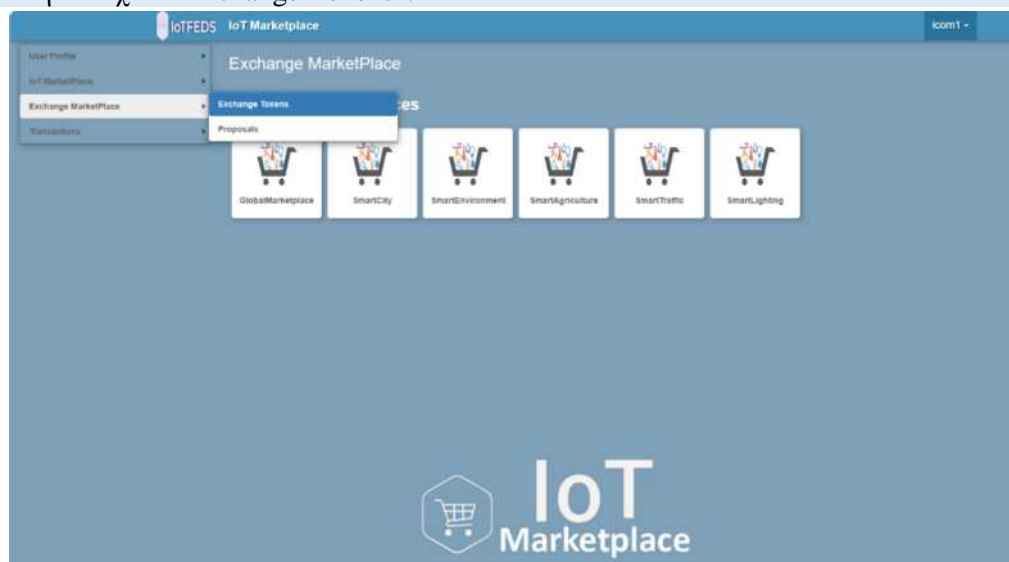
Η παρακάτω σελίδα εμφανίζεται στο χρήστη, ενημερώνοντας για την επιτυχή αφαίρεση του προϊόντος. Ο χρήστης μπορεί να ανανεώσει τη σελίδα για να ενημερωθεί με τη νέα λίστα των κουπονιών.



### MarketplaceGUI-5: Ο χρήστης κάνει πρόταση ανταλλαγής προϊόντων.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να χρησιμοποιήσει την υπηρεσία ανταλλαγής κουπονιών.

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “Exchange Marketplace” και στη συνέχεια “Exchange Tokens”.



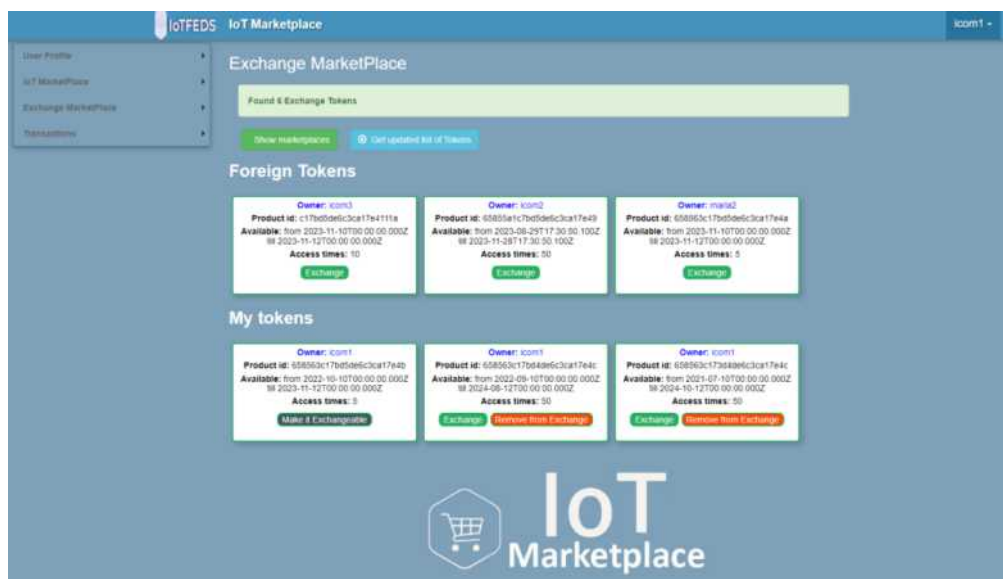
### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζονται τα κουπόνια του χρήστη καθώς και τα ανταλλάξιμα κουπόνια των άλλων IoT παρόχων που συμμετέχουν στην επισκεπτόμενη αγορά.

### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται ενημερώνοντας τον χρήστη για τον αριθμό των κουπονιών που ανακαλύφθηκαν στην επιλεγμένη αγορά.





Ο χρήστης επιλέγει για ανταλλαγή ένα δικό του ανταλλάξιμο κουπόνι (exchangeable token) με ένα ανταλλάξιμο κουπόνι (foreign) ενός άλλου IoT παρόχου επιλέγοντας το κουμπί “Exchange” κάθε κουπονιού. Στη συνέχεια υποβάλλει την πρόταση ανταλλαγής επιλέγοντας το κουμπί “Exchange” δεξιά πάνω.

#### Αναμενόμενα αποτελέσματα:

Πυροδοτείται η διαδικασία πρότασης ανταλλαγής κουπονιών και κατά συνέπεια η διαδικασία ανταλλαγής των αντίστοιχων προϊόντων (όπως περιγράφεται στην δοκιμή που ακολουθεί **MarketplaceGUI-7**).

#### Αποτελέσματα που παρατηρήθηκαν:

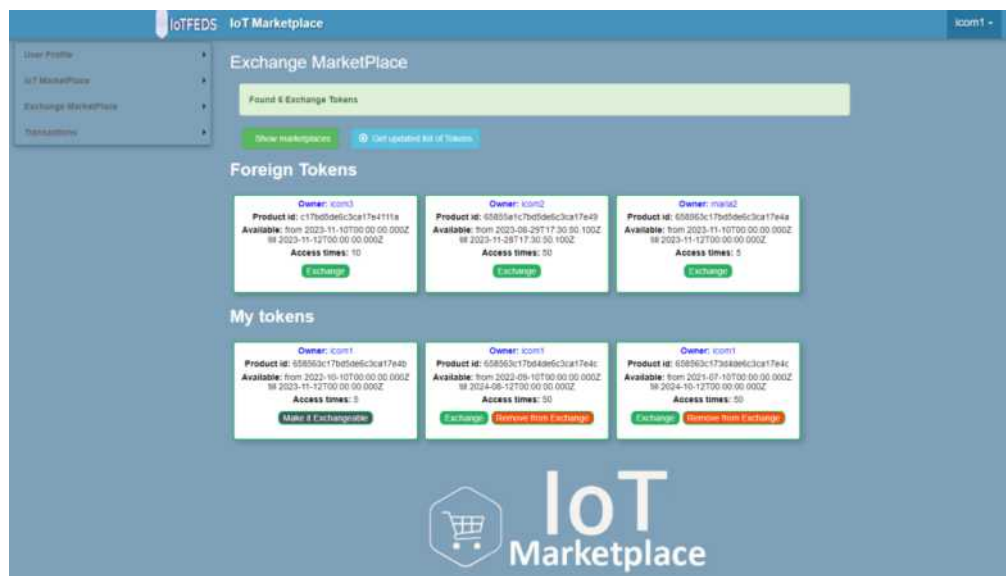
Η παρακάτω σελίδα εμφανίζεται ενημερώνοντας το χρήστη για την επιτυχή πρόταση ανταλλαγής του δικού του κουπονιού με ένα κουπόνι άλλου χρήστη της ίδιας αγοράς:



### MarketplaceGUI-6: Ο χρήστης τοποθετεί ή αποσύρει κουπόνι στην ανταλλαγή αγοράς (Exchange Market).

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να κάνει ένα κουπόνι του ανταλλάξιμο ή αντίστροφα.

Ο χρήστης μπορεί να τοποθετήσει ένα κουπόνι του στην αγορά ανταλλαγής επιλέγοντας το “Make it Exchangeable” που φαίνεται σε αυτό το βήμα.



#### Αναμενόμενα αποτελέσματα:

Το κουπόνι μετά από επιτυχή έλεγχο μετατρέπεται σε ανταλλάξιμο, εμφανίζοντας τις Επιλογές “Exchange” και “Remove from Exchange”.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται ενημερώνοντας τον χρήστη (με μήνυμα ACCEPTED) για την μετατροπή του κουπονιού του σε ανταλλάξιμο κουπόνι. Στο κουπόνι τώρα εμφανίζονται οι επιλογές “Exchange” και “Remove from Exchange”.



Ο χρήστης στη συνέχεια επιλέγει ένα δικό του ανταλλάξιμο κουπόνι (αυτό που εμφανίζεται κάτω στο κέντρο) να αποσυρθεί από την αγορά ανταλλαγής επιλέγοντας το κουμπί “Remove from Exchange”.

#### Αναμενόμενα αποτελέσματα:

Το κουπόνι μετά από επιτυχή έλεγχο μετατρέπεται σε μη ανταλλάξιμο, εμφανίζοντας τη μοναδική επιλογή “Make it Exchangeable”.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται ενημερώνοντας τον χρήστη για την επιτυχή (με μήνυμα ACCEPTED) μετατροπή του δικού του κουπονιού σε μη ανταλλάξιμο. Το κουπόνι τώρα έχει τη μοναδική επιλογή “Make it Exchangeable”.



#### MarketplaceGUI-7: Διαχείριση των προτάσεων ανταλλαγής κουπονιών.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να διαχειριστεί τις προτάσεις ανταλλαγής κουπονιών που συμμετέχει με δικά του κουπόνια.

Ο χρήστης επιλέγει να αποσύρει μια δική του πρόταση για ανταλλαγή κουπονιών επιλέγοντας το κουμπί “Cancel”.

**IoT MarketPlace**

**Proposals Page**

Proposal with id: 6346436436654398724 rejected

Proposal ID	Date submitted	Exchange offer id	Status	Foreign Token	My Token
6346436436654398724	2023-11-10T00:00:00.000Z	24242343424	pending	Valid From: 2023-11-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 64554564363465346 Marketplace: smart city Access Times: 50 Owner: kcom2	Valid From: 2023-24-10T00:00:00.000Z to: 2024-04-10T00:00:00.000Z Product id: 534545345 Marketplace: smart environment Access Times: 10 Owner: kcom1
6346436436654398724	2022-21-10T00:00:00.000Z	534534534534	pending	Valid From: 2022-09-02T00:20:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 34534534534532 Marketplace: smart agriculture Access Times: 10 Owner: kcom2	Valid From: 2023-11-10T00:00:00.000Z to: 2023-11-10T00:00:00.000Z Product id: 5235324532452 Marketplace: smart traffic Access Times: 100 Owner: kcom1
6346436436654398724	2023-11-10T00:00:00.000Z	634643643634	canceled	Valid From: 2023-11-10T00:00:00.000Z to: 2023-11-10T00:00:00.000Z Product id: 45235235325 Marketplace: smart environment Access Times: 10 Owner: ipiet	Valid From: 2023-06-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 435345345345 Marketplace: metro data Access Times: 10 Owner: kcom1
6346436436654398724	2021-11-10T00:00:00.000Z	634643643634	pending	Valid From: 2020-11-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 45235235344325 Marketplace: smart traffic Access Times: 10 Owner: ipiet	Valid From: 2021-06-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 9e09w9k2 Marketplace: smart city Access Times: 15 Owner: kcom1
6346436436654398724	2024-01-10T00:00:00.000Z	6346436436654398724	pending	Valid From: 2020-11-10T00:00:00.000Z to: 2024-01-10T00:00:00.000Z Product id: 45235235344325 Marketplace: smart traffic Access Times: 10 Owner: kcom1	Valid From: 2021-09-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 9e09w9k23445 Marketplace: smart city Access Times: 20 Owner: ipiet
6346436436654398724	2023-12-10T00:00:00.000Z	6346436436654398724	pending	Valid From: 2020-11-10T00:00:00.000Z to: 2024-06-10T00:00:00.000Z Product id: 45235235344325 Marketplace: smart agriculture Access Times: 200 Owner: kcom1	Valid From: 2021-09-10T00:00:00.000Z to: 2024-11-10T00:00:00.000Z Product id: 9e09w9k23445 Marketplace: smart environment Access Times: 20 Owner: kcom2

### Αναμενόμενα αποτελέσματα:

Το κουπόνι μετά από επιτυχή απόσυρση από το μηχανισμό ανταλλαγής κουπονιών εμφανίζεται με την ένδειξη: status:canceled, χωρίς την εμφάνιση του κουμπιού “Cancel”.

### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται ενημερώνοντας τον χρήστη για την επιτυχή απόσυρση του κουπονιού από το μηχανισμό ανταλλαγής κουπονιών:

The screenshot shows the 'Proposals Page' of the IoT MarketPlace. On the left, a sidebar menu includes 'User Profile', 'IoT MarketPlace', 'Exchange Marketplace' (selected), and 'Transactions'. The main content area displays a grid of six proposal cards. Each card is divided into two sections: 'My Proposal' and 'Foreign Proposal'. Each section contains the following information: 'Date submitted', 'Exchange offer id', 'Status', 'Valid From', 'Valid To', 'Product id', 'Marketplace', 'Access Times', and 'Owner'. At the bottom of each card, there are buttons for 'Cancel', 'Accept', and 'Reject'. The proposals are listed with various IDs and statuses, including one that is canceled and others that are pending.

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “Exchange Marketplace” και στη συνέχεια “Proposals”.

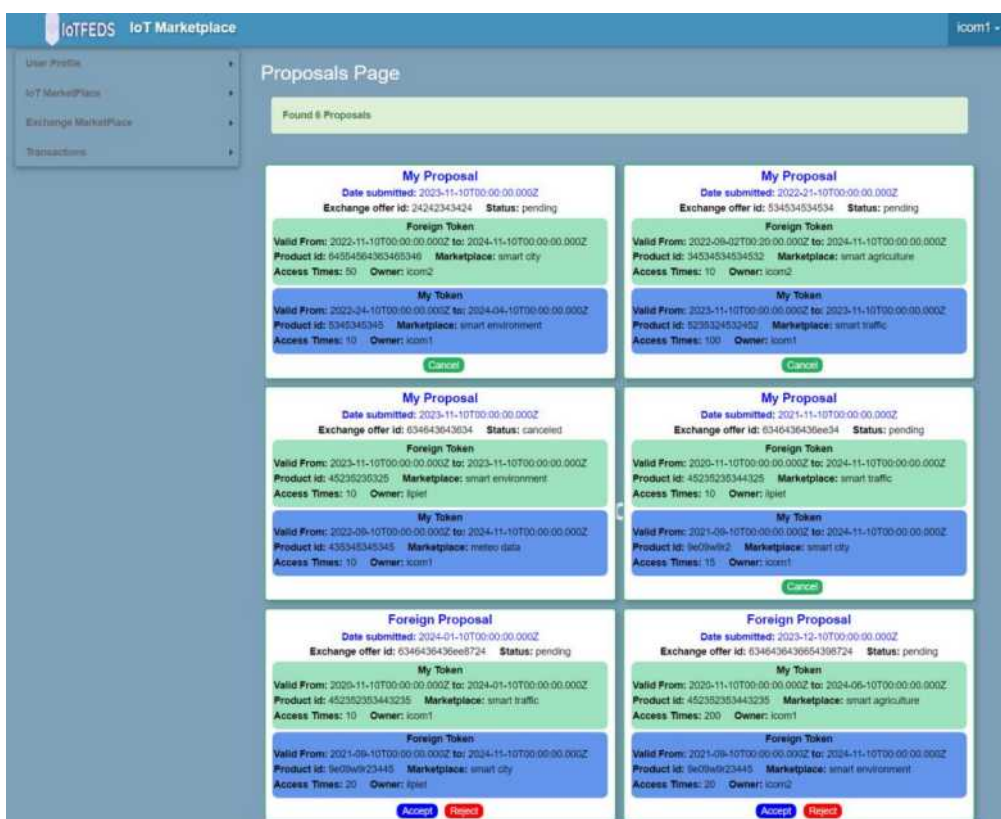
#### Αναμενόμενα αποτελέσματα:

Στη σελίδα εμφανίζεται επιτυχώς η λίστα των προτάσεων τις οποίες ο χρήστης έχει δημιουργήσει στο **MarketplaceGUI-5** μαζί με τις προτάσεις που έχουν δημιουργήσει άλλοι IoT πάροχοι για κουπόνια (προϊόντα) αυτού του χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη, ενημερώνοντας για τη λίστα των προτάσεων ανταλλαγής κουπονιών που αφορούν το χρήστη. Η σελίδα εμφανίζει τον αριθμό των προτάσεων ανταλλαγής κουπονιών. Οι προτάσεις διαφοροποιούνται σε “My Proposal” και αφορούν τις προτάσεις ανταλλαγής κουπονιών που έχουν δημιουργηθεί από τον ίδιο τον χρήστη (στο **MarketplaceGUI-5**) και από τις “Foreign Proposal” που έχουν δημιουργηθεί από άλλους IoT παρόχους και αφορούν προτάσεις ανταλλαγής δικών τους προϊόντων με τα προϊόντα του χρήστη.





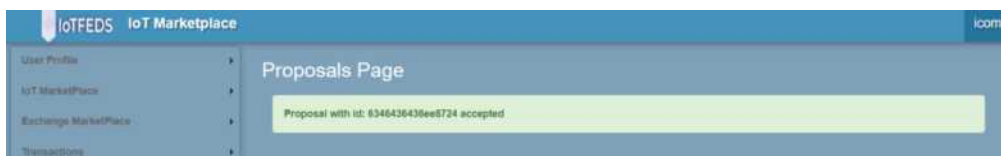
Ο χρήστης αποδέχεται την πρόταση ανταλλαγής κουπονιών προερχόμενη από άλλο μέλος της αγοράς πατώντας το κουμπί “Accept” της πρότασης που τον ενδιαφέρει.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται το μήνυμα αποδοχής της πρότασης στην οθόνη του χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη ενημερώνοντας τον χρήστη για την επιτυχία αποδοχής της πρότασης ανταλλαγής κουπονιών. Ο χρήστης πρέπει να ανανεώσει τη σελίδα του για να δει την αλλαγή της κατάστασης στο νέο κουπόνι. Αντίστοιχη ενημέρωση θα εμφανιστεί στον δεύτερο χρήστη που συμμετέχει στην πρόταση.



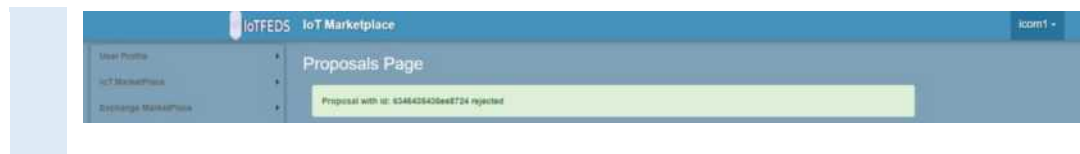
Εναλλακτικά, ο χρήστης μπορεί να επιλέξει να απορρίψει μια πρόταση ανταλλαγής κουπονιών που έχει γίνει από άλλο IoT πάροχο προϊόντων πατώντας το κουμπί “Reject” της πρότασης που επιθυμεί να απορρίψει.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται το μήνυμα επιβεβαίωσης της απόρριψης της πρότασης στην οθόνη του χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη ενημερώνοντας τον χρήστη για την επιβεβαίωση της απόρριψης της πρότασης ανταλλαγής κουπονιών και νέα κατάσταση στο κουπόνι θα εμφανιστεί μετά την ανανέωση (refresh). Ομοίως ο χρήστης που έκανε την προσφορά ενημερώνεται με την αντίστοιχη κατάσταση.



### MarketplaceGUI-8: Ο χρήστης ζητά έλεγχο αποδείξεων αγοράς προϊόντων.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να ανακτήσει το ιστορικό αγοράς προϊόντων.

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “Transactions” και στη συνέχεια “Audit Receipt Buyer” ως αγοραστής.



### Αναμενόμενα αποτελέσματα:

Στη σελίδα εμφανίζεται η φόρμα αναζήτησης αποδείξεων αγοράς IoT προϊόντων.

### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω φόρμα εμφανίζεται στην οθόνη του χρήστη.



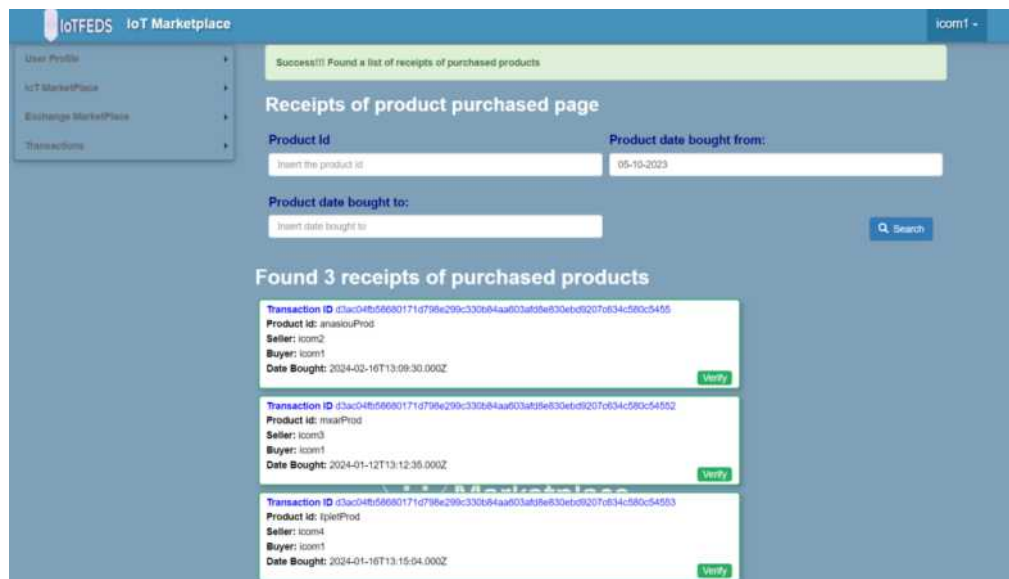
Ο χρήστης εισάγει στη φόρμα τα κριτήρια για την αναζήτηση ιστορικού αγοράς IoT προϊόντων και επιλέγει το κουμπί “Search”.

### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται με βάση τα κριτήρια αναζήτησης μια λίστα αποδείξεων αγορών προϊόντων που πραγματοποίησε ο χρήστης στο παρελθόν.

### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζει στην οθόνη μια λίστα αποδείξεων αγορών IoT προϊόντων που πραγματοποίησε ο χρήστης στο παρελθόν μαζί με το μήνυμα της επιτυχούς αναζήτησης.



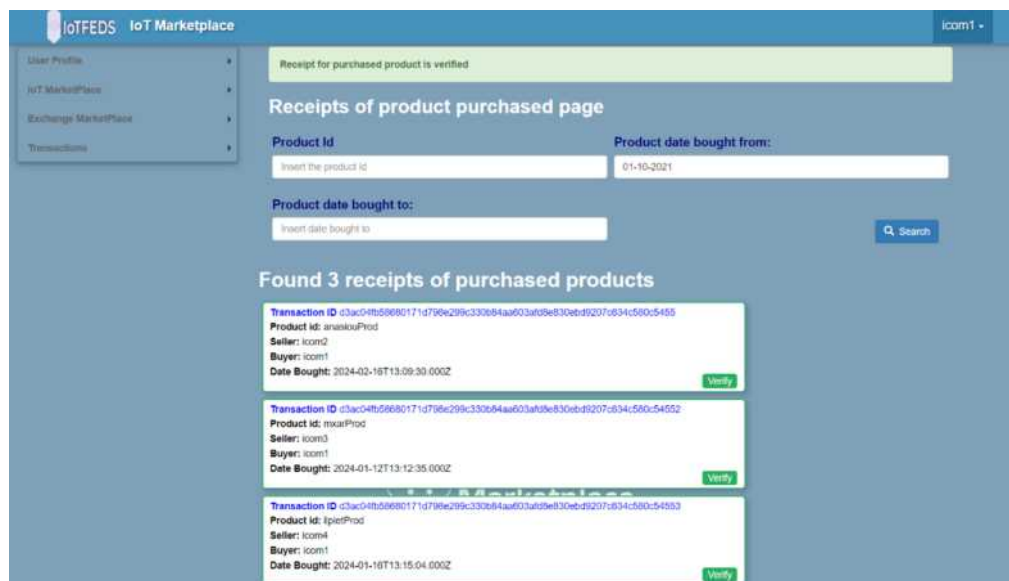
Ο χρήστης επιλέγει την επιβεβαίωση μιας απόδειξης αγοράς προϊόντος επιλέγοντας το κουμπί “Verify”.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται το μήνυμα επιβεβαίωσης ή όχι της αγοράς προϊόντος στο παρελθόν από τον χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη ενημερώνοντας τον χρήστη για την επιβεβαίωση της αγοράς του προϊόντος στο παρελθόν από τον ίδιο.





### MarketplaceGUI-9: Ο χρήστης επιλέγει την ανάκτηση αποδείξεων αγοράς προϊόντων που έχει πωλήσει.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να χρησιμοποιήσει την υπηρεσία εμφάνισης του ιστορικού αγοράς IoT προϊόντων.

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “Transactions” και στη συνέχεια “Audit Receipt Seller”.



#### Αναμενόμενα αποτελέσματα:

Στη σελίδα εμφανίζεται η φόρμα αναζήτησης αποδείξεων πώλησης IoT προϊόντων.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω φόρμα εμφανίζεται στην οθόνη του χρήστη.



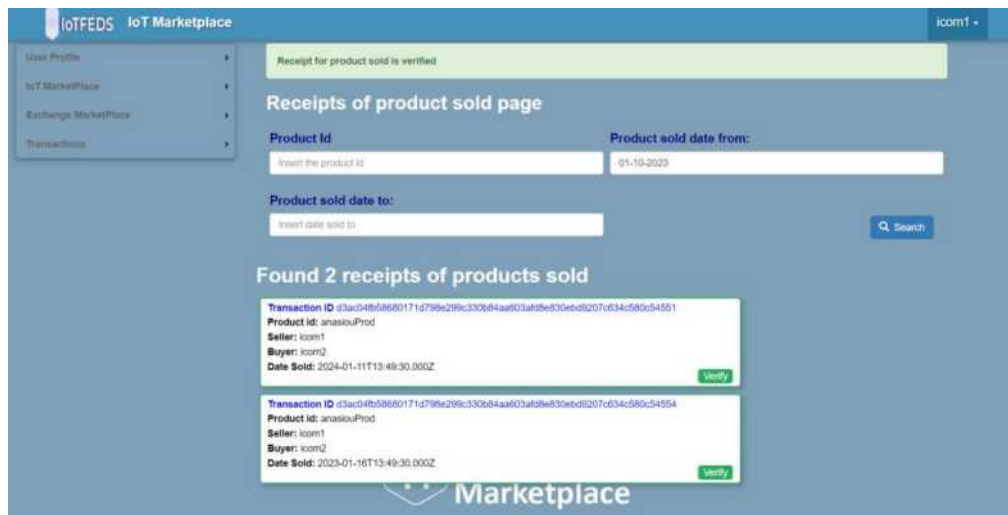
Ο χρήστης εισάγει στη φόρμα τα κριτήρια για την αναζήτηση ιστορικού πώλησης IoT προϊόντων και επιλέγει το κουμπί “Search”.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται με βάση τα κριτήρια αναζήτησης μια λίστα αποδείξεων πώλησης IoT προϊόντων που πραγματοποίησε ο χρήστης στο παρελθόν.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζει στην οθόνη μια λίστα αποδείξεων πώλησης IoT προϊόντων που πραγματοποίησε ο χρήστης στο παρελθόν μαζί με το μήνυμα της επιτυχούς αναζήτησης.



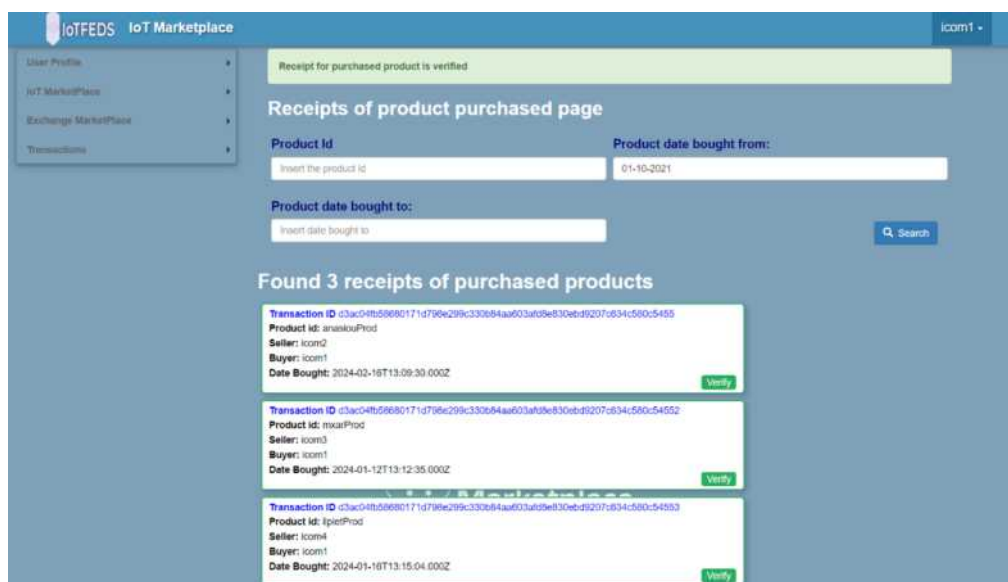
Ο χρήστης επιλέγει την επιβεβαίωση μιας απόδειξης πώλησης IoT προϊόντος επιλέγοντας το κουμπί “Verify”.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή επιβεβαιώνεται η πώληση IoT προϊόντος στο παρελθόν από τον χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη ενημερώνοντας τον χρήστη για την επιβεβαίωση της πώλησης του IoT προϊόντος στο παρελθόν από τον ίδιο.



#### MarketplaceGUI-10: Ο χρήστης επιλέγει τον έλεγχο αρχείων καταγραφής.

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να αποκτήσει πρόσβαση στα αρχεία καταγραφής.

Ο χρήστης επιλέγει από το κεντρικό μενού την επιλογή “Transactions” και στη συνέχεια “Audit Logs”.



**Αναμενόμενα αποτελέσματα:**

Στη σελίδα εμφανίζεται η φόρμα αναζήτησης αρχείων καταγραφής.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω φόρμα εμφανίζεται στην οθόνη του χρήστη.



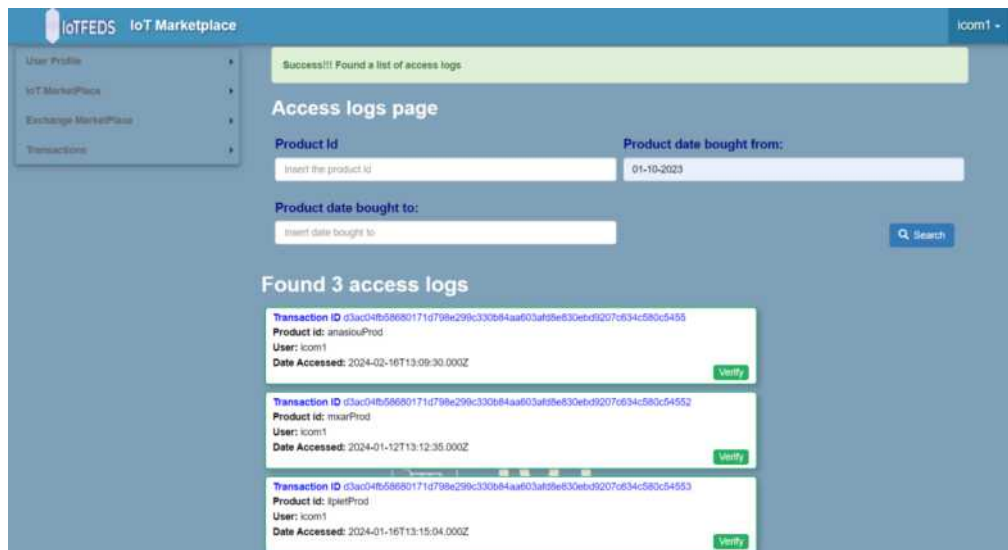
Ο χρήστης εισάγει στη φόρμα τα κριτήρια για την αναζήτηση αρχείων καταγραφής και επιλέγει το κουμπί “Search”.

**Αναμενόμενα αποτελέσματα:**

Με την επιλογή αυτή εμφανίζεται με βάση τα κριτήρια αναζήτησης μια λίστα αρχείων καταγραφής του χρήστη.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζει στην οθόνη μια λίστα αρχείων καταγραφής του χρήστη μαζί με το μήνυμα της επιτυχούς αναζήτησης.



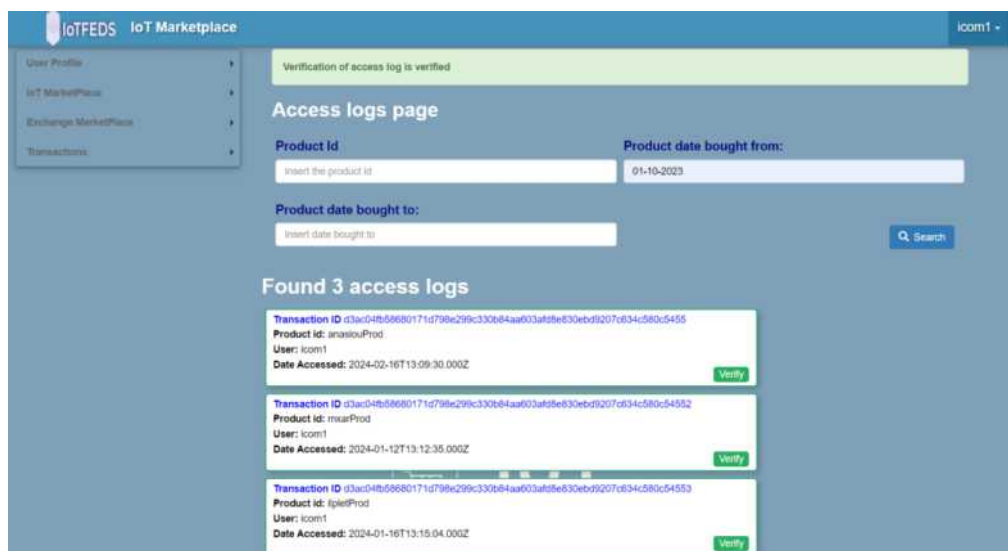
Ο χρήστης επιλέγει την επιβεβαίωση ενός αρχείου καταγραφής επιλέγοντας το κουμπί “Verify”.

#### Αναμενόμενα αποτελέσματα:

Με την επιλογή αυτή εμφανίζεται το μήνυμα επιβεβαίωσης του αρχείου καταγραφής.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στην οθόνη ενημερώνοντας τον χρήστη για την επιβεβαίωση του αρχείου καταγραφής.



## 4.2 Λειτουργική δοκιμή και επίδειξη διαχείρισης ομοσπονδιών

Στην ενότητα αυτή παρουσιάζεται η αλληλεπίδραση του χρήστη με τη γραφική διεπαφή χρήστη για τη διαχείριση ομοσπονδιών καθώς και των σχετικών οργανισμών (χρηστών) και πλατφορμών τους στο σύστημα. Οι παρακάτω δοκιμές έχουν περιγραφεί αναλυτικά στο Π1.4 ωστόσο περιλαμβάνονται για λόγους πληρότητας. Οι πληροφορίες της ομοσπονδίας έχουν τροποποιηθεί με την ενσωμάτωση των μηχανισμών τιμολόγησης και φήμης και εμπιστοσύνης. Τα αιτήματα που υποστηρίζονται από το Administration GUI αφορούν τις παρακάτω λειτουργικότητες:

- Εγγραφή ενός χρήστη στην IoTFeds πλατφόρμα.
- Σύνδεση ενός χρήστη στη γραφική διεπαφή χρήστη.
- Εγγραφή μίας IoT πλατφόρμας.
- Δημιουργία IoT ομοσπονδίας.
- Τροποποίηση κανόνων ομοσπονδίας.
- Προσθήκη νέου μέλους σε υπάρχουσα ομοσπονδία.
- Αίτηση για συμμετοχή σε υπάρχουσα ομοσπονδία.
- Απομάκρυνση μέλους από ομοσπονδία.
- Αποχώρηση από ομοσπονδία.
- Διαγραφή ομοσπονδίας.
- Διαγραφή IoT πλατφόρμας.
- Διαγραφή χρήστη.

Περίπτωση Δοκιμής	Κατάσταση Εκτέλεσης
-------------------	------------------------

### AdminGUI-1: Πρόσβαση στον ιστότοπο της IoTFeds πλατφόρμας.

**PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ο χρήστης έχει πρόσβαση στον ιστότοπο της IoTFeds πλατφόρμας.

Ο χρήστης επισκέπτεται την κεντρική σελίδα του ιστότοπου της IoTFeds πλατφόρμας.

#### Προαπαιτούμενο:

Γνώση της διεύθυνσης του ιστότοπου της IoTFeds πλατφόρμας:

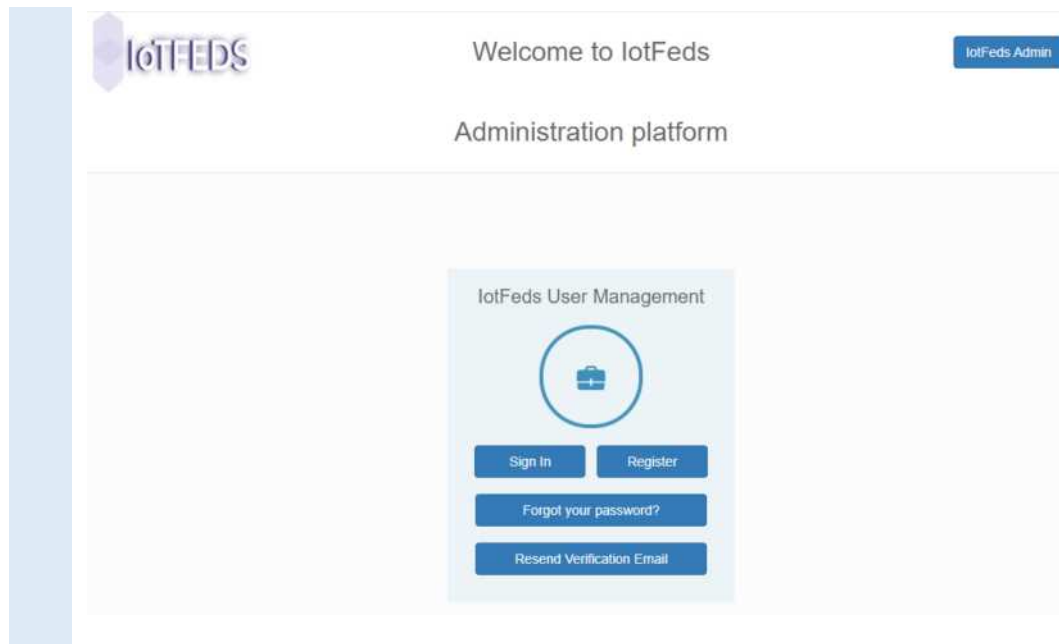
<https://iotfeds-symbiotecore.intracom-telecom.com/administration/>.

#### Αναμενόμενα αποτελέσματα:

Η αρχική σελίδα με το κεντρικό μενού της IoTFeds πλατφόρμας εμφανίζεται στην οθόνη του χρήστη.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στον χρήστη.

**AdminGUI-2: Εγγραφή χρήστη στην IoTFeds πλατφόρμα.****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να εγγραφεί επιτυχώς στην IoTFeds πλατφόρμα.

Ο χρήστης κάνει εγγραφή στην κεντρική σελίδα του ιστότοπου της IoTFeds πλατφόρμας.

**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-1

**Αναμενόμενα αποτελέσματα:**

Η φόρμα για τη συμπλήρωση των υποχρεωτικών στοιχείων του χρήστη εμφανίζεται όταν ο χρήστης πατήσει από το κεντρικό μενού της περίπτωσης δοκιμής AdminGUI-1 το κουμπί "Register".

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη πατώντας το κουμπί "Register".

Registration

Username

Password

Email

Choose your User Role

Organization

lot feds role

We use the username for uniquely identify each user in the system

We use the email address as part of allowing access to your account, and in order to contact you with important information about any changes to your account

We use the Organization for uniquely identify each user in the system

Terms and Conditions

Data protection is of a particularly high priority for the management of the symbloTe. The use of the Internet pages of the symbloTe is possible without any indication of personal data; however, if a data subject wants to use special enterprise services via our website, processing of personal data could become necessary. If the processing of personal data is necessary and there is no statutory basis for such processing, we generally obtain consent from the data subject.

The processing of personal data, such as the name, address, e-mail address, or telephone number of a data subject shall always be in line with the General Data Protection Regulation (GDPR), and in accordance with the country-specific data protection

☐ I understand the Terms and Conditions

☐ I accept the Term and Conditions

Permissions

Username

Email

Public Keys, bound to the user clients

JWT tokens, issued for clients

☐ Please, specify if the data above can be used for analysis and research purposes

Data Breach Policy

1. Introduction

1. This policy sets out the policies and procedure of symbloTe (the "consortium") with respect to GDPR Art. 33 Notification of a personal data breach to the supervisory authority and Art. 34 Communication of a personal data breach to the data subject.

2. When dealing with personal data breaches, the consortium and all consortium members must focus on protecting individuals and their personal data, as well as protecting the interests of the consortium.

2. Definition

Register

Ο χρήστης συμπληρώνει τα απαιτούμενα στοιχεία ορθώς.

### Αναμενόμενα αποτελέσματα:

Η εγγραφή του χρήστη ολοκληρώνεται με επιτυχία εφόσον συμπληρώνονται όλα τα απαιτούμενα στοιχεία. Στη δοκιμή αυτή το κουμπί “Register” της φόρμας συμπλήρωσης παραμένει ανενεργό μέχρι ο χρήστης να συμπληρώσει όλα τα υποχρεωτικά στοιχεία και αποδεχτεί τους όρους και προϋποθέσεις της IoTFeds πλατφόρμας.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη στην περίπτωση επιτυχούς εγγραφής του στην IoTFeds πλατφόρμα:

**Registration Successful!**

Please, login with your new credentials

Login


Ο χρήστης προσπαθεί να εγγραφεί στην IoTFeds πλατφόρμα επιλέγοντας ένα όνομα χρήστη (user name) που ήδη υπάρχει στην πλατφόρμα.


**Αναμενόμενα αποτελέσματα:**  
Ο χρήστης ενημερώνεται ότι το όνομα χρήστη (user name) που έχει εισάγει στην φόρμα εγγραφής έχει ήδη καταχωρηθεί από άλλον χρήστη της IoTFeds πλατφόρμας.

**Αποτελέσματα που παρατηρήθηκαν:**  
Το παρακάτω μήνυμα λάθους εμφανίζεται στην οθόνη του χρήστη.


Registration

An error occurred: Username exists!

 iotfedsc1 ✓


 ..... ✓


We use the username for uniquely identify each user in the system

 iotfedsc1@icom.com ✓

User x

We use your email address as part of allowing access to your account, and in order to contact you with important information about any changes to your account

 icom ✓

 organization ✓

We use the Organization for uniquely identify each user in the system

**AdminGUI-3: Σύνδεση χρήστη στην IoTFeds πλατφόρμα.****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να συνδεθεί επιτυχώς στην IoTFeds πλατφόρμα.

Ο χρήστης επιθυμεί πρόσβαση στην κεντρική σελίδα του ιστότοπου της IoTFeds πλατφόρμας.

**Προαπαιτούμενη περίπτωση δοκιμής:**  
AdminGUI-2

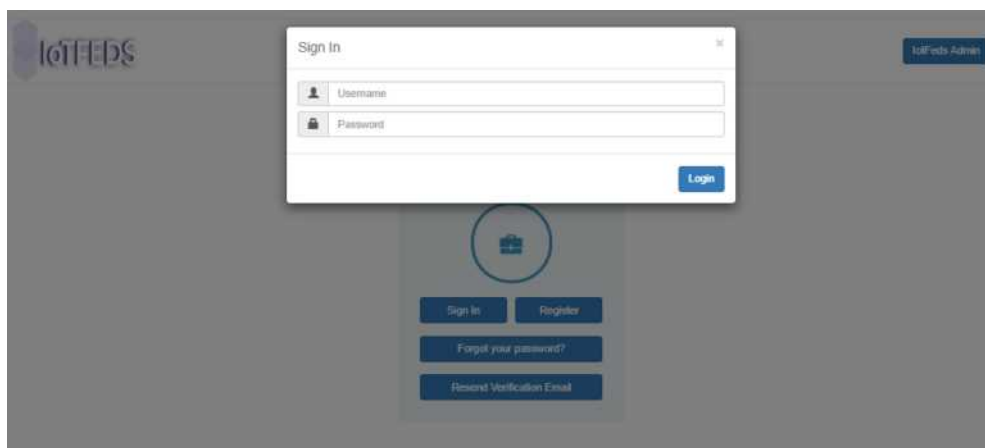
**Αναμενόμενα αποτελέσματα:**

Η φόρμα για τη συμπλήρωση των διαπιστευτηρίων του χρήστη εμφανίζεται όταν ο χρήστης πατήσει το κουμπί “Sign in” από το κεντρικό μενού.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη.





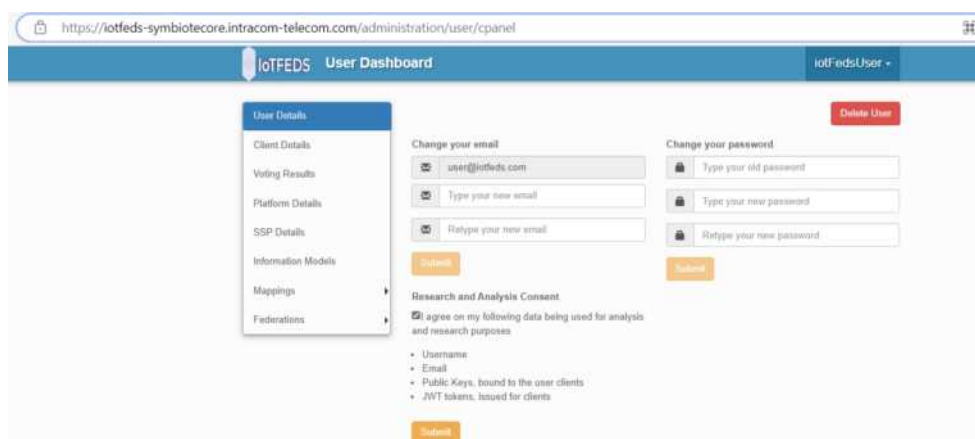
Ο χρήστης εισάγει τα διαπιστευτήρια του στην φόρμα και πατάει το κουμπί “Login”.

**Αναμενόμενα αποτελέσματα:**

Η σύνδεση του χρήστη ολοκληρώνεται με επιτυχία σε περίπτωση εισαγωγής σωστών διαπιστευτηρίων εμφανίζοντας στην οθόνη την ιστοσελίδα User Dashboard που αφορά αποκλειστικά τις δραστηριότητες του χρήστη στην IoTFeds πλατφόρμα.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στον χρήστη.



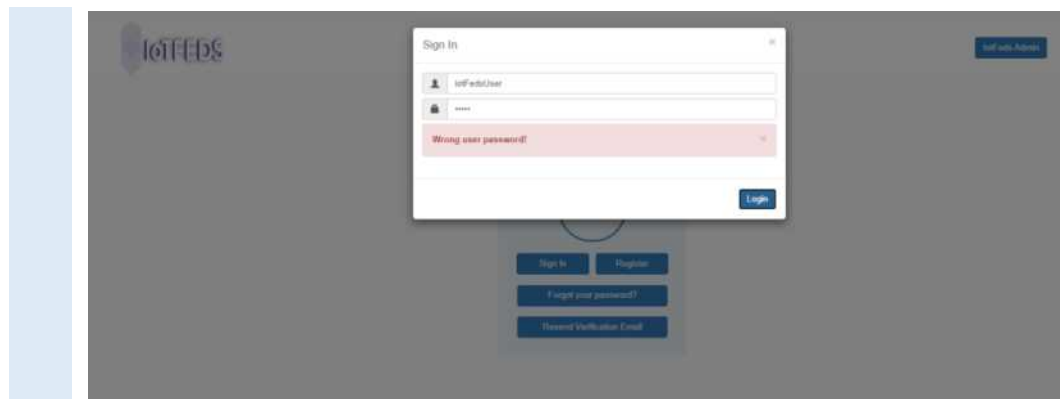
Ο χρήστης εισάγει λανθασμένα διαπιστευτήρια.

**Αναμενόμενα αποτελέσματα:**

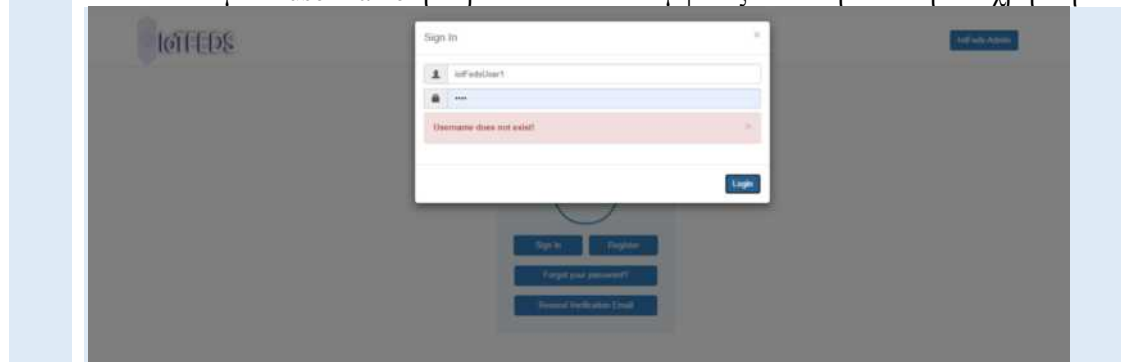
Σε περίπτωση λανθασμένων στοιχείων όπως κωδικό ο χρήστης ενημερώνεται με αντίστοιχο μήνυμα.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στον χρήστη.



Για λανθασμένο user name η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη:



#### AdminGUI-4: Εγγραφή IoT πλατφόρμας στο IoTFeds σύστημα.

PASS

Η δοκιμή αυτή εξασφαλίζει ότι ένας εγγεγραμμένος χρήστης μπορεί να εγγράψει μία IoT πλατφόρμα ιδιοκτησίας του στο IoTFeds σύστημα.

Ο χρήστης ζητά να κάνει εγγραφή της IoT πλατφόρμας μέσα από το μενού “Platform Details”.

#### Προαπαιτούμενη περίπτωση δοκιμής:

AdminGUI-3 (Σύνδεση χρήστη στην IoTFeds πλατφόρμα με ρόλο “Service Owner”)

#### Αναμενόμενα αποτελέσματα:

Η φόρμα για τη συμπλήρωση των στοιχείων της IoT πλατφόρμας εμφανίζεται στην οθόνη όταν ο χρήστης πατά το κουμπί “Register New Platform”.

#### Αποτελέσματα που παρατηρήθηκαν:

Η παρακάτω σελίδα εμφανίζεται στο χρήστη για να εισάγει τα στοιχεία της IoT πλατφόρμας του.

The screenshot shows the 'Platform Registration' modal form. It includes fields for 'Platform Id' (with a note: 'From 4 to 30 characters. Include only letters, digits, "." and "-". You can leave it empty for autogeneration'), 'Platform Name' (with a note: 'From 3 to 30 characters'), 'Platform Descriptions' (with a note: 'From 4 to 300 characters'), 'Interworking Services' (with a note: 'Enter a valid https url'), and a 'Type' dropdown menu (with a note: 'Select your Platform type'). There are also '+' and '-' buttons next to the description field and a 'Submit' button at the bottom right.

Η εγγραφή της IoT πλατφόρμας επιτυγχάνεται όταν τα στοιχεία είναι ορθά.

Τα στοιχεία που εισάγει ο εγγεγραμμένος χρήστης είναι το όνομα της IoT πλατφόρμας (platform name), μια μοναδική ταυτότητα της πλατφόρμας (platform id), μια σύντομη περιγραφή της πλατφόρμας (Platform Description), ο τύπος της IoT πλατφόρμας (Platform ή Enabler type), το μοντέλο οντολογίας το οποίο χρησιμοποιείται από την IoT πλατφόρμα (Information Model) και η μοναδική διεύθυνση προσπέλασης (http://, https://) της IoT πλατφόρμας (Interworking Services).

Η επιτυχημένη εγγραφή της IoT πλατφόρμας στο IoTfeds σύστημα εμφανίζει στην οθόνη του χρήστη το μήνυμα: "Registration of platform ... was successful!". Μετά την επιτυχή εγγραφή της πλατφόρμας, ο χρήστης μπορεί να κατεβάσει τα αρχεία παραμέτρων για την διαμόρφωση και την εγκατάσταση του symbIoTe-cloud που περιγράφηκε στο Π1.4 πατώντας το κουμπί «Get Configuration». Στη συνέχεια ο χρήστης μπορεί να εγγράψει τους IoT πόρους που επιθυμεί να διαμοιράσει στο σύστημα.

The screenshot shows the 'Platform list' interface. At the top, there is a green message box that says 'Registration of platform "test-platform" was successful!'. Below this, there is a table with one row for 'test-platform'. The table has buttons for 'Get Configuration', 'Update', and 'Delete'. There is also a 'Register New Platform' button at the top right of the table.

Σε περίπτωση λανθασμένων στοιχείων το σύστημα επιστρέφει το αντίστοιχο μήνυμα λάθους στον χρήστη.

Στο παρακάτω παράδειγμα αποτυγχάνει η εγγραφή IoT πλατφόρμας με τιμή Interworking Services ή platform Id που ήδη έχουν καταχωρηθεί στην IotFeds πλατφόρμα.

The image displays two screenshots of the IoT FEDS User Dashboard, specifically the Platform Registration form. The dashboard has a sidebar with navigation links: User Details, Client Details, Voting Results, Platform Details (selected), SSP Details, Information Models, Mappings, and Federations. The top right of the dashboard shows the user 'iotFedsUser'.

**Top Screenshot:** The Platform Registration form is shown with a red error message at the top: "AAM threw CommunicationException: Service gateway address already in use". The form fields are: Platform Id (test-platform1), Platform Name (test-platform), Platform Descriptions (test-platform), Interworking Services (https://iotfeds-symbiotecloud.intracom-telecom.cor), and Type (Platform). The Submit button is highlighted in blue.

**Bottom Screenshot:** The Platform Registration form is shown with a red error message at the top: "AAM threw CommunicationException: Service with this instance id already exists". The form fields are: Platform Id (test-platform), Platform Name (test-platform), Platform Descriptions (test-platform), Interworking Services (https://iotfeds-symbiotecloud.intracom-telecom.cor), and Type (Platform). The Submit button is highlighted in blue.

**AdminGUI-5: Δημιουργία ομοσπονδίας.****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης (με ρόλο Service Owner) μπορεί να δημιουργήσει μία ομοσπονδία στο IoTFeds σύστημα.

- 1 Ο χρήστης ζητά να κάνει εγγραφή μία ομοσπονδίας μέσα από την επιλογή "Federation List" του μενού "Federations".

**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-3 (Σύνδεση χρήστη με ρόλο “Service Owner”).

AdminGUI-4 (Εγγραφή IoT πλατφόρμας στο IoTFeds σύστημα).

**Αναμενόμενα αποτελέσματα:**

Η φόρμα για τη συμπλήρωση των στοιχείων της ομοσπονδίας εμφανίζεται όταν ο χρήστης πατάει το κουμπί “Register New Federation”.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στην οθόνη του χρήστη.

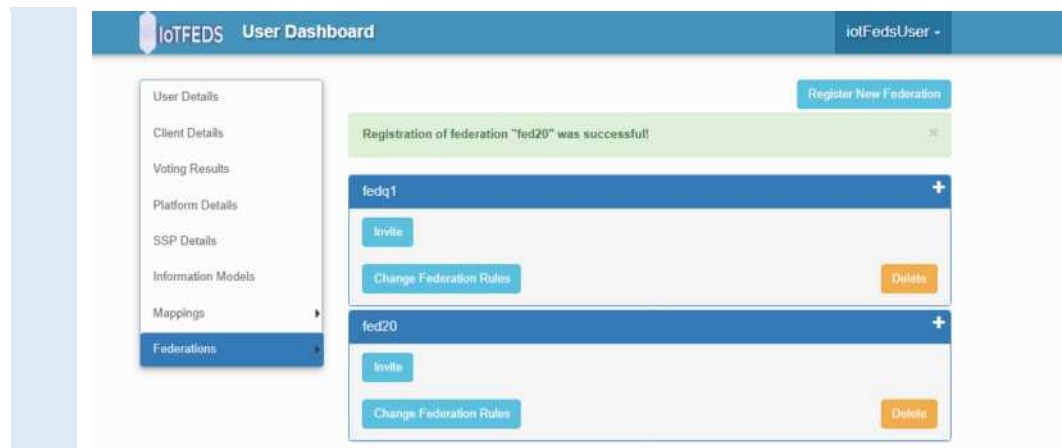
The screenshot displays the 'Federation Registration' form within the 'IoTFeds User Dashboard'. The form is divided into several sections:

- Federation Id:** A text input field with a placeholder 'Enter the federation id'. Below it, a note states: 'From 4 to 30 characters. Include only letters, digits, '-' and '\_'. You can leave it empty for autogeneration'.
- Federation Name:** A text input field with a placeholder 'Enter the federation name'. Below it, a note states: 'From 3 to 30 characters'.
- Information Model:** A dropdown menu currently showing 'Information Model'. Below it, a note says: 'Select the federation information model'.
- Public:** A dropdown menu currently showing 'Yes'. Below it, a note asks: 'Has the federation public visibility?'.
- Name of Organization:** A text input field with a placeholder 'Enter the organization'. To its right are '+' and '-' buttons. Below it, a note states: 'From 4 to 30 characters. Include only letters, digits, '-' and '\_' '.
- QoS Constraints:** A section with a '+' button.
- Federation type:** A blue-bordered section containing:
  - Members Type:** A dropdown menu with the placeholder 'Select the type of members'.
  - Products Visibility:** A dropdown menu with the placeholder 'Select the product visibility'.
  - Service Type:** A text input field with the placeholder 'Insert the type of service'.
  - Supported Ontologies:** A text input field with the placeholder 'Insert the ontology'.
- Federation Governance:** A blue-bordered section with the text 'Insert the supported proposals.' and a text input field with the placeholder 'Add the proposal'. To the right of the input field are '+' and '-' buttons.

Ο χρήστης συμπληρώνει και υποβάλλει τη φόρμα.

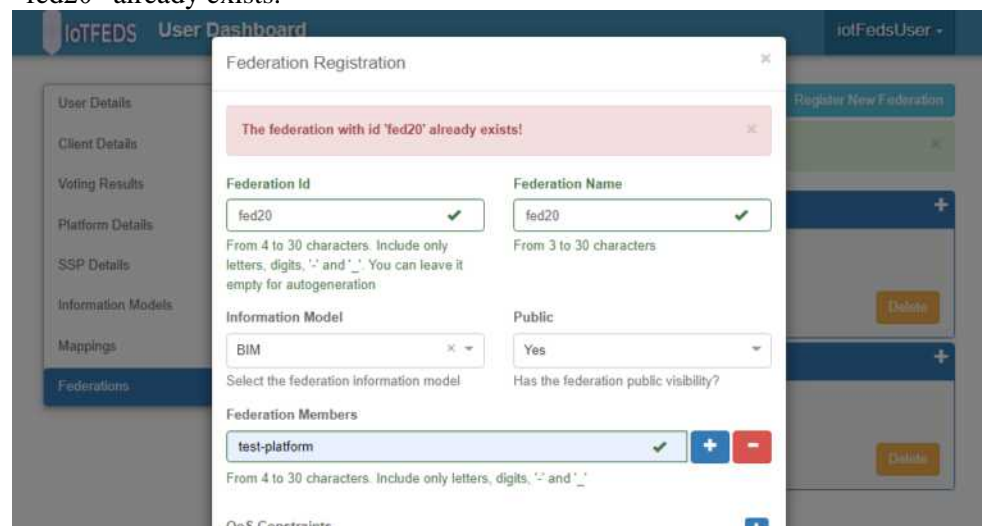
Ο χρήστης συμπληρώνει όλα τα υποχρεωτικά πεδία της φόρμας και πατά το κουμπί “Submit” το οποίο μέχρι να συμπληρωθούν όλα τα υποχρεωτικά πεδία της φόρμας παραμένει απενεργοποιημένο. Η εγγραφή της ομοσπονδίας επιτυγχάνεται όταν οι τιμές που έχουν εισαχθεί στη φόρμα είναι σωστές.

Το αποτέλεσμα της επιτυχούς εγγραφής της ομοσπονδίας στο IoTFeds σύστημα συνοδεύεται με την εμφάνιση του μηνύματος: “Registration of federation fed20 was successful”.

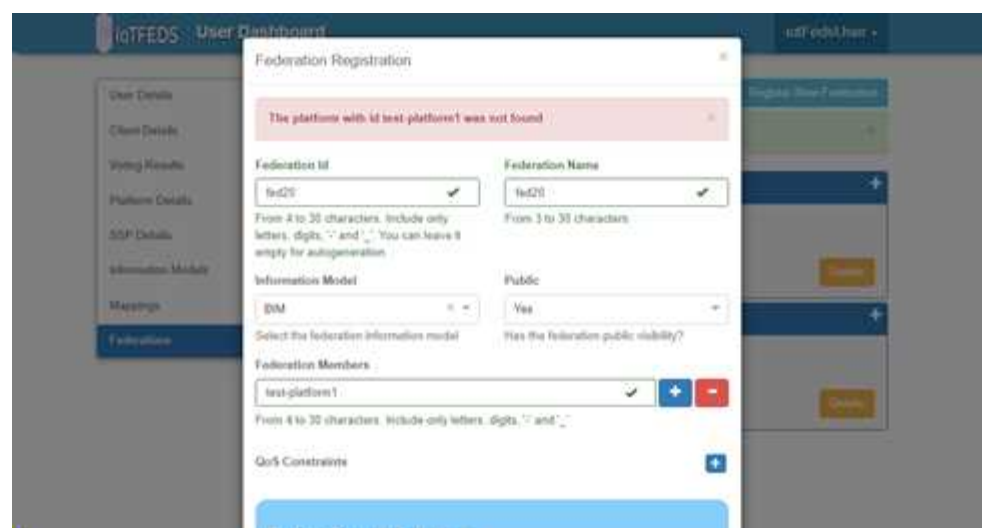


Σε περίπτωση σφάλματος εμφανίζεται αντίστοιχο μήνυμα.

Στην περίπτωση όπου ο χρήστης επιλέξει ταυτότητα ομοσπονδίας (federation id) που έχει καταχωρηθεί στο σύστημα εμφανίζεται μήνυμα λάθους: The federation with id “fed20” already exists.



Στην περίπτωση όπου ο χρήστης επιλέξει μέλος ομοσπονδίας (federation member) που δεν υπάρχει (πχ test-platform1) τότε το σύστημα εμφανίζει μήνυμα λάθους: The platform with id “test-platform1” was not found.



PASS

**AdminGUI-6:**

Η δοκιμή αυτή εξασφαλίζει ότι ένα ομόσπονδο μέλος μπορεί να ζητήσει την συμμετοχή ενός νέου μέλους στην ομοσπονδία.

Ο χρήστης προσκαλεί ένα νέο μέλος στην επιθυμητή ομοσπονδία από τη λίστα “Federations List” του μενού “Federations”.

**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-5 (Εγγραφή ομοσπονδίας στο IoTFeds σύστημα).

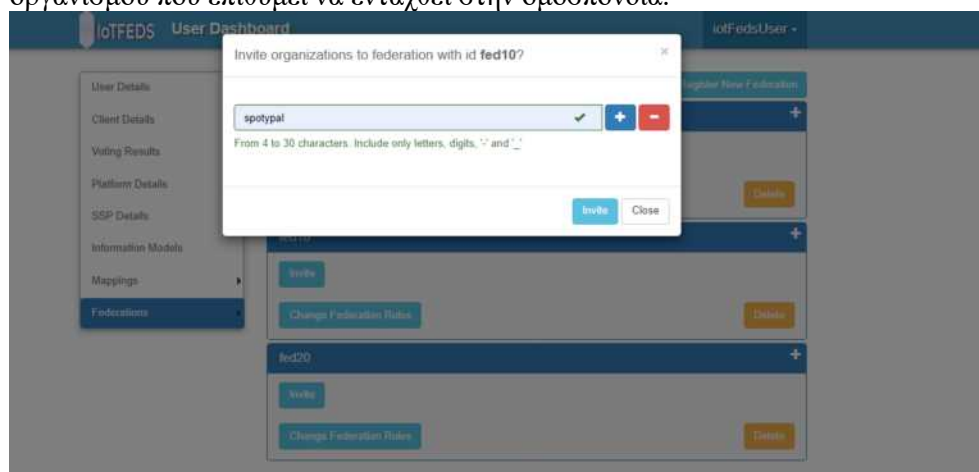
**Προαπαιτούμενο:** Ο χρήστης γνωρίζει το όνομα του μέλους που πρόκειται να προσκληθεί ως νέο μέλος στην ομοσπονδία. Το μέλος έχει εγγραφεί στο IoTFeds σύστημα με τη διαδικασία που περιγράφεται στην περίπτωση δοκιμής AdminGUI-4.

**Αναμενόμενα αποτελέσματα:**

Η φόρμα για τη συμπλήρωση των στοιχείων του οργανισμού που πρόκειται να προστεθεί στην ομοσπονδία εμφανίζεται στην οθόνη όταν ο χρήστης πατάει το κουμπί “Invite”.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στο χρήστη. Ο χρήστης συμπληρώνει το όνομα του οργανισμού που επιθυμεί να ενταχθεί στην ομοσπονδία.



Με το αίτημα ενεργοποιείται η διαδικασία της ψηφοφορίας για αποδοχή ή όχι του οργανισμού στην ομοσπονδία από τα υπόλοιπα μέλη της ομοσπονδίας.

Με το πάτημα του κουμπιού “Invite” προωθείται στο σύστημα IoTFeds το αίτημα ένταξης του οργανισμού στην επιθυμητή ομοσπονδία (όπως fed10) και ενεργοποιείται η διαδικασία της ψηφοφορίας για αποδοχή ή όχι του οργανισμού στην ομοσπονδία από τα υπόλοιπα μέλη της ομοσπονδίας.

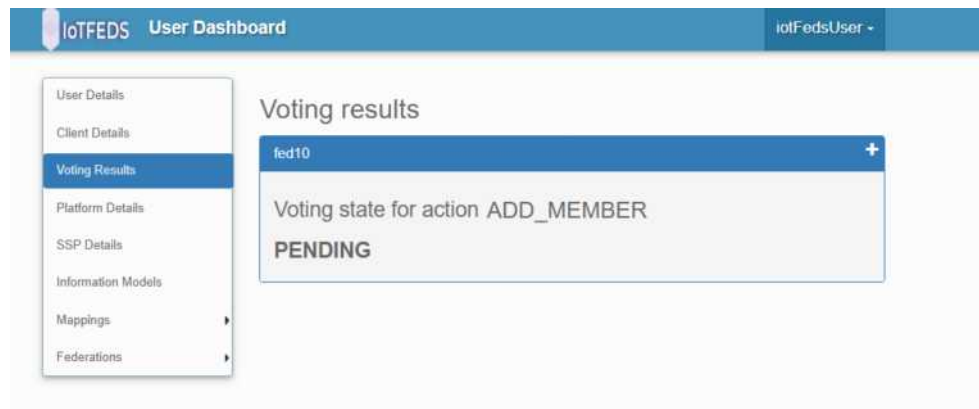
Η ψηφοφορία ολοκληρώνεται και ο χρήστης ενημερώνεται.

Η επιτυχημένη αίτηση για ψηφοφορία συνοδεύεται με το παρακάτω μήνυμα στην οθόνη του χρήστη:

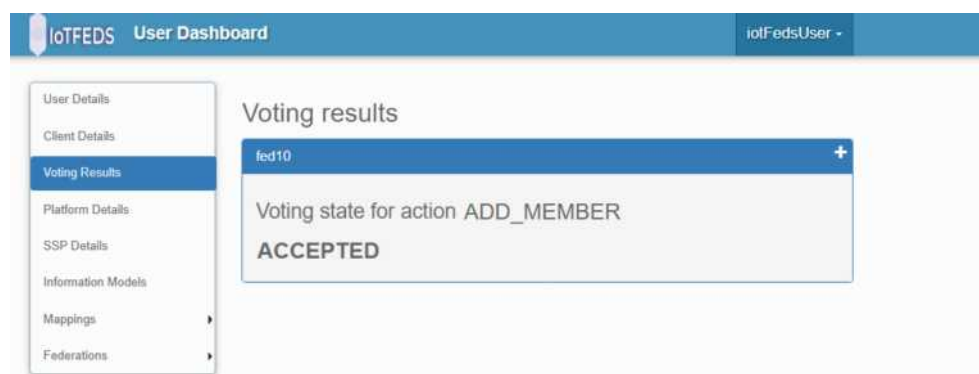
Invitation to federation "fed10" was successful!

Ο χρήστης που έστειλε την πρόσκληση ένταξης μπορεί να μάθει το αποτέλεσμα της ψηφοφορίας επιλέγοντας από το κεντρικό μενού την επιλογή “Voting Results”.

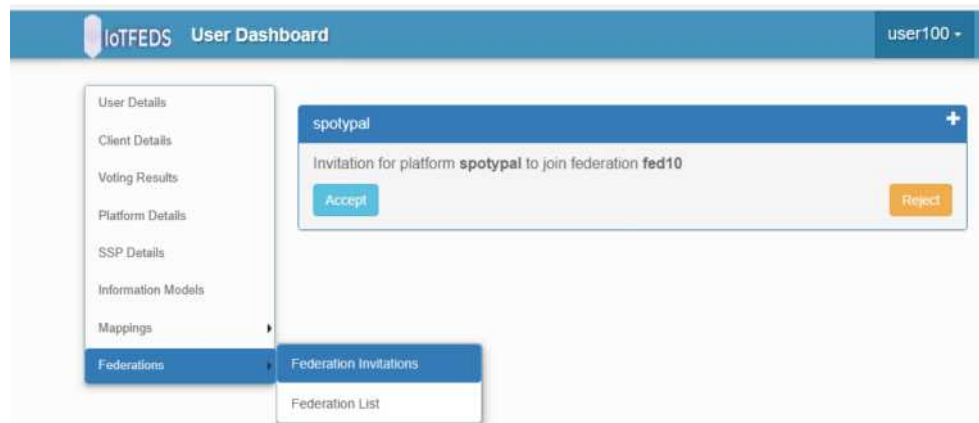
Μέχρι την ολοκλήρωση της ψηφοφορίας θα εμφανίζεται η παρακάτω ενημέρωση:



Στην περίπτωση που το αποτέλεσμα της ψηφοφορίας για ένταξη στην ομοσπονδία είναι θετικό τότε εμφανίζεται στην οθόνη του χρήστη η παρακάτω ενημέρωση:



Στη συνέχεια το νέο μέλος της ομοσπονδίας δέχεται την πρόσκληση στο λογαριασμό του στο IoTFeds μέσα από το μενού Federations→Federation Invitations.



Ο χρήστης αποδεχόμενος την πρόσκληση πατάει το κουμπί “Accept” και εμφανίζεται τότε στην οθόνη του το παρακάτω μήνυμα:

The invitation to platform with id "spotypal" to join the federation with id "fed10" was accepted successfully

Παράλληλα ο χρήστης που έστειλε την πρόσκληση ενημερώνεται για την αποδοχή της πρόσκλησης με το ίδιο μήνυμα στο μενού Federations→Federation Invitations.



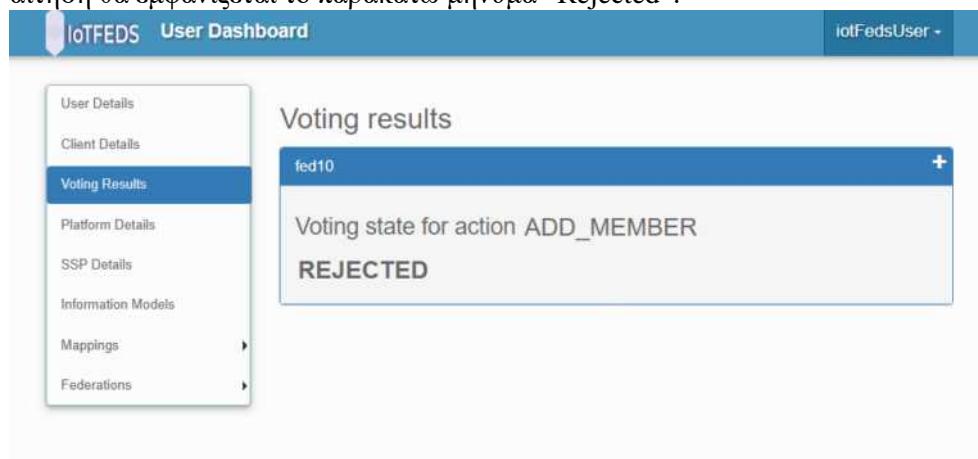
The invitation to platform with id "spotypal" to join the federation with id "fed10" was accepted successfully

Στην περίπτωση που ο χρήστης που δέχεται την πρόσκληση δεν επιθυμεί την ένταξη του οργανισμού στην ομοσπονδία, τότε πατά το κουμπί “Reject” και τότε εμφανίζεται το παρακάτω μήνυμα στην οθόνη του:

The invitation to platform with id "spotypal" to join the federation with id "fed10" was rejected successfully

Ταυτόχρονα το ίδιο μήνυμα εμφανίζεται στην οθόνη του χρήστη που έκανε την πρόσκληση.

Στην περίπτωση όπου η αίτηση για ένταξη στην ομοσπονδία απορρίφθηκε στην ψηφοφορία τότε στη σελίδα Voting Results στο λογαριασμό του χρήστη που έκανε την αίτηση θα εμφανίζεται το παρακάτω μήνυμα “Rejected”.



#### AdminGUI-7:

#### PASS

Η δοκιμή αυτή εξασφαλίζει ότι ένα μέλος της IoTFeds πλατφόρμας μπορεί να ζητήσει την συμμετοχή του σε μία υπάρχουσα ομοσπονδία.

Ο χρήστης έχει πρόσβαση στις διαθέσιμες ομοσπονδίες.

#### Προαπαιτούμενη περίπτωση δοκιμής:

AdminGUI-4 (Ο χρήστης έχει κάνει εγγραφή οργανισμού στο IoTFeds σύστημα).

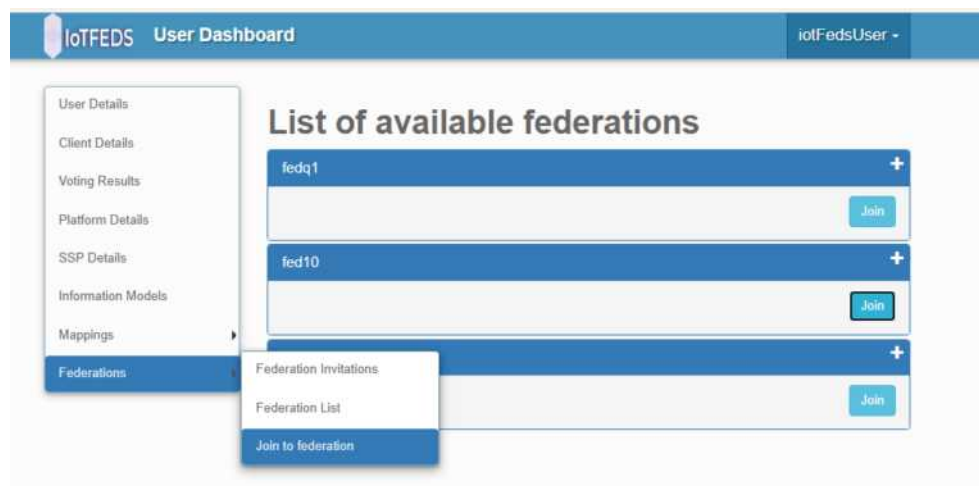
AdminGUI-5 (Άλλοι χρήστες της IoTFeds πλατφόρμας έχουν εγγράψει ομοσπονδίες και είναι έτοιμες να υποδεχτούν νέα μέλη).

#### Αναμενόμενα αποτελέσματα:

Η λίστα με τις ομοσπονδίες εμφανίζεται στην οθόνη του χρήστη με την επιλογή “Join to federation” στο μενού “Federations”.

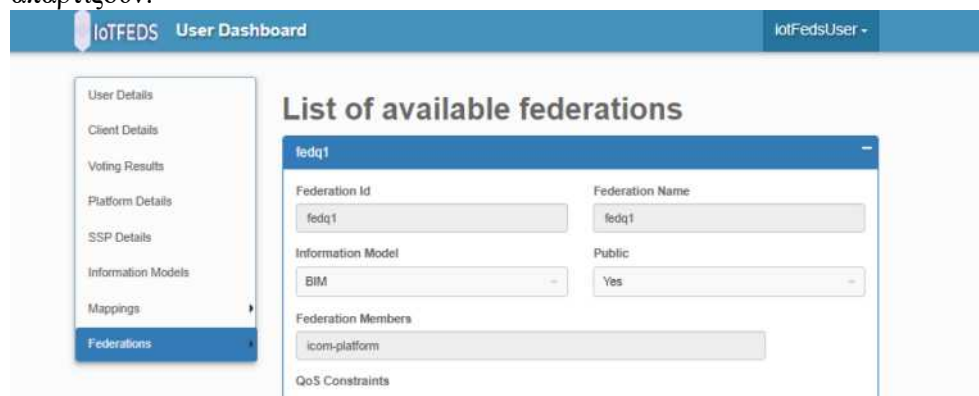
#### Αποτελέσματα που παρατηρήθηκαν:

Στην οθόνη του χρήστη εμφανίζεται μια λίστα από ομοσπονδίες στις οποίες μπορεί να ενταχθεί.



Ο χρήστης έχει πρόσβαση στις πληροφορίες της ομοσπονδίας.

Ο χρήστης πατώντας το “+” κουμπί βλέπει στην οθόνη του τους κανόνες του Smart Contract που ισχύει για κάθε ομοσπονδία καθώς και ποια μέλη (οργανισμοί) την απαρτίζουν.



Ο χρήστης στέλνει το αίτημα συμμετοχής στην επιθυμητή ομοσπονδία (fedq1 στη δοκιμή) που εκκινεί διαδικασία ψηφοφορίας.

Ο χρήστης πατώντας το κουμπί “Join” στέλνει το αίτημα συμμετοχής στην επιθυμητή ομοσπονδία (fedq1 στη δοκιμή) και ξεκινά η διαδικασία ψηφοφορίας από μέλη που συμμετέχουν στην ομοσπονδία fedq1.

#### **Αναμενόμενα αποτελέσματα:**

Ο χρήστης μπορεί να παρακολουθήσει το αίτημα του στη λίστα των “Pending requests”.

#### **Αποτελέσματα που παρατηρήθηκαν:**

Στην περίπτωση που η ψηφοφορία είναι σε εξέλιξη ο χρήστης στην οθόνη του έχει την παρακάτω ενημέρωση:

The screenshot shows the 'IoT FEDS User Dashboard'. On the left is a sidebar menu with options: User Details, Client Details, Voting Results (highlighted), Platform Details, SSP Details, Information Models, Mappings, and Federations. The main content area is titled 'Voting results' and contains a table with one row for 'fedq1'. The table shows the 'Voting state for action ADD\_MEMBER' as 'PENDING'.

Client	Voting state for action ADD_MEMBER
fedq1	PENDING

Κάθε μέλος της ομοσπονδίας ενημερώνεται για το αίτημα και την ψηφοφορία.

**Αναμενόμενα αποτελέσματα:**

Κάθε ψηφοφόρος αποκτά πρόσβαση στην ψηφοφορία για να υποβάλει την ψήφο του.

**Αποτελέσματα που παρατηρήθηκαν:**

Ο κάθε εμπλεκόμενος χρήστης ενημερώνεται με email με σύνδεσμο για την ψηφοφορία. Ο σύνδεσμος επιτρέπει στο χρήστη να υποβάλλει επιτυχώς όσο οι προϋποθέσεις για την εγκυρότητα του πληρούνται (π.χ χρονικό διάστημα ψηφοφορίας). Λεπτομέρειες σχετικά με τις ψηφοφορίες και περισσότερα σενάρια υπάρχουν στο Π2.1. Στην περίπτωση που το αποτέλεσμα της ψηφοφορίας είναι θετικό ο χρήστης στην οθόνη του έχει την παρακάτω ενημέρωση:

This screenshot is identical to the previous one, but the voting state for 'fedq1' is now 'ACCEPTED'.

Client	Voting state for action ADD_MEMBER
fedq1	ACCEPTED

Στην περίπτωση που το αποτέλεσμα της ψηφοφορίας είναι αρνητικό, ο χρήστης στην οθόνη του έχει την παρακάτω ενημέρωση:

This screenshot is identical to the previous ones, but the voting state for 'fedq1' is now 'REJECTED'.

Client	Voting state for action ADD_MEMBER
fedq1	REJECTED

**AdminGUI-8:****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένα ομόσπονδο μέλος (οργανισμός) μπορεί να αποχωρήσει από την ομοσπονδία στην οποία ανήκει.

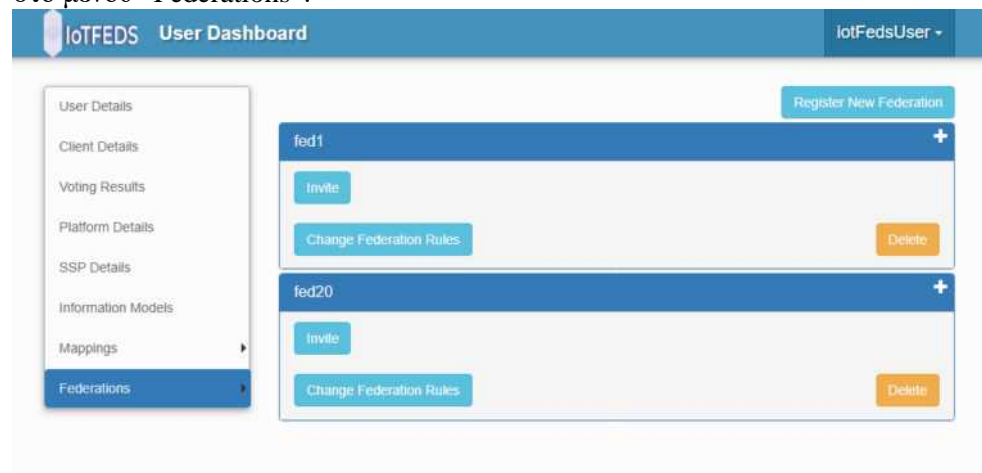
Ο χρήστης έχει πρόσβαση στις διαθέσιμες ομοσπονδίες του.

**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-6,  
AdminGUI-7

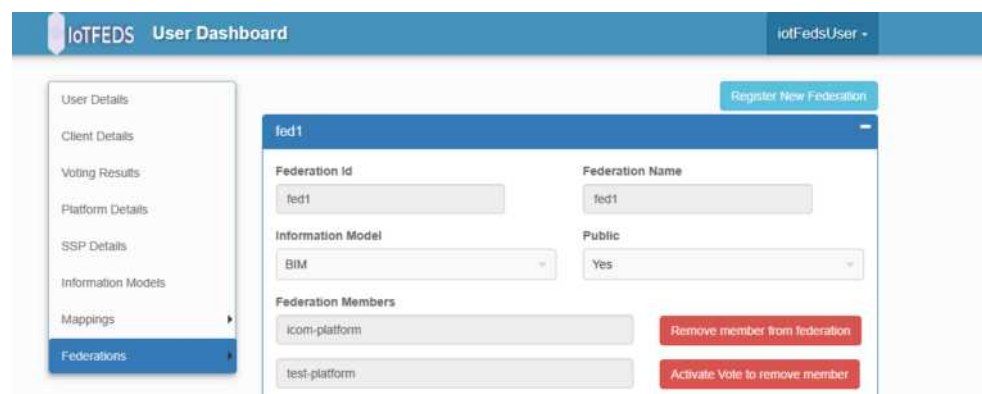
**Αναμενόμενα αποτελέσματα:**

Οι ομοσπονδίες στις οποίες ανήκει ο χρήστης εμφανίζονται στη λίστα “Federations List” στο μενού “Federations”.



Ο χρήστης επιλέγει το μέλος προς αποχώρηση από την ομοσπονδία.

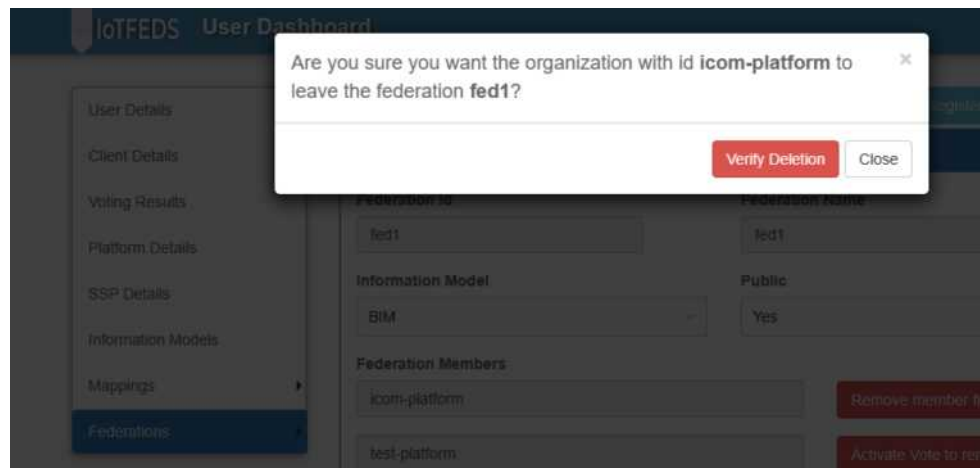
Ο χρήστης πατάει το “+” κουμπί για να δει τη λίστα των μελών μιας ομοσπονδίας (fed1 στη δοκιμή). Από τη λίστα επιλέγει το όνομα του οργανισμού του (icom-platform στη δοκιμή) που επιθυμεί να απομακρύνει από την ομοσπονδία fed1 .



Στη συνέχεια επιλέγει το “Remove member from federation” κουμπί για να ενεργοποιήσει τη διαδικασία αποχώρησης του μέλους icom-member από την ομοσπονδία fed1.

**Αποτελέσματα που παρατηρήθηκαν:**

Με το πάτημα του κουμπιού “Remove member from federation” εμφανίζεται στην οθόνη του χρήστη μήνυμα επιβεβαίωσης;



Ο χρήστης στέλνει το αίτημα για αποχώρηση από την ομοσπονδία.

#### Αναμενόμενα αποτελέσματα:

Ο χρήστης ενημερώνεται για το αποτέλεσμα του αιτήματος και ανανεώνεται η λίστα ομοσπονδιών που του εμφανίζεται σε περίπτωση επιτυχίας.

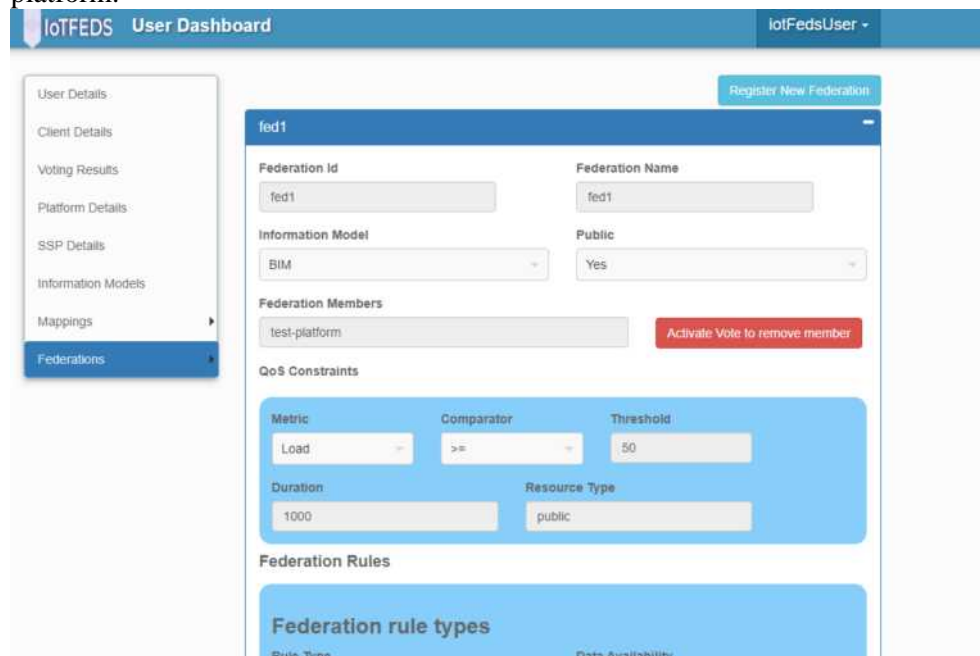
#### Αποτελέσματα που παρατηρήθηκαν:

Στην οθόνη του χρήστη εμφανίζεται το παρακάτω μήνυμα:

The platform with id "icom-platform" left the federation "fed1" successfully!

Με το ίδιο μήνυμα ενημερώνονται και τα υπόλοιπα μέλη της ομοσπονδίας για την απομάκρυνση του οργανισμού icom-platform.

Η λίστα των ομοσπονδιών ανανεώνεται και εμφανίζεται χωρίς πλέον το μέλος icom-platform:



**AdminGUI-10:****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένα ομόσπονδο μέλος της IoTFeds πλατφόρμας ζητά την απομάκρυνση ενός άλλου ομόσπονδου μέλους της ίδιας ομοσπονδίας.

Ο χρήστης έχει πρόσβαση στις ομοσπονδίες που συμμετέχει και στις πληροφορίες τους.

**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-6

AdminGUI-7

**Αναμενόμενα αποτελέσματα:**

Τα μέλη της ομοσπονδίας εμφανίζονται στη φόρμα στοιχείων της ομοσπονδίας και υπάρχει επιλογή για την απομάκρυνση ενός άλλου μέλους με τη διαδικασία της ψηφοφορίας.

**Αποτελέσματα που παρατηρήθηκαν:**

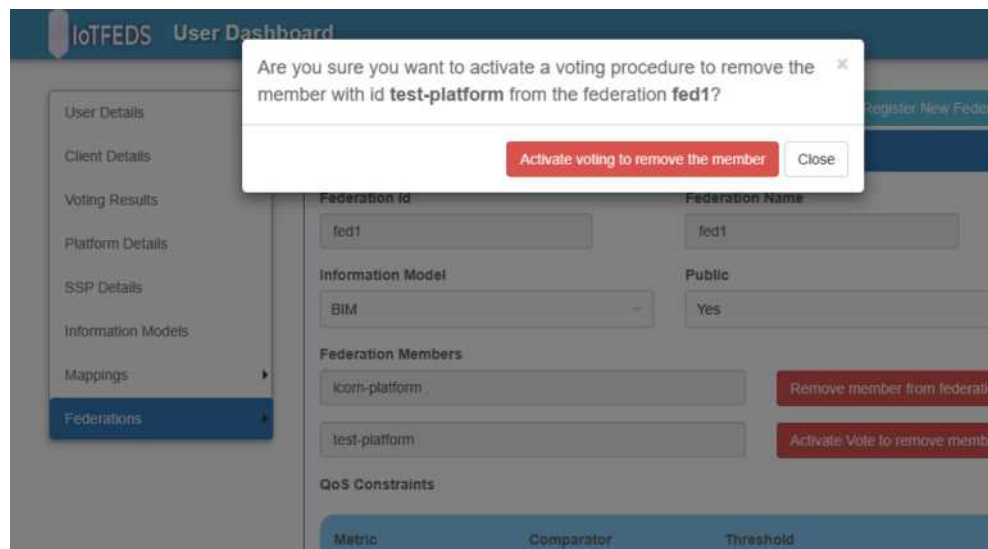
Στην οθόνη του χρήστη εμφανίζεται η σελίδα που περιέχει τη λίστα των μελών της ομοσπονδίας (Federation Members). Από τη λίστα, τα μέλη (οργανισμοί) που η απομάκρυνση τους απαιτεί την ενεργοποίηση ψηφοφορίας συνοδεύονται με το κουμπί “Activate Vote to remove member”.

Ο χρήστης επιλέγει το μέλος για αποχώρηση από την ομοσπονδία και εκκινεί τη διαδικασία ψηφοφορίας.

Ο χρήστης επιλέγει το κουμπί “Activate Vote to remove member” στο μέλος που επιθυμεί να αποχωρήσει.

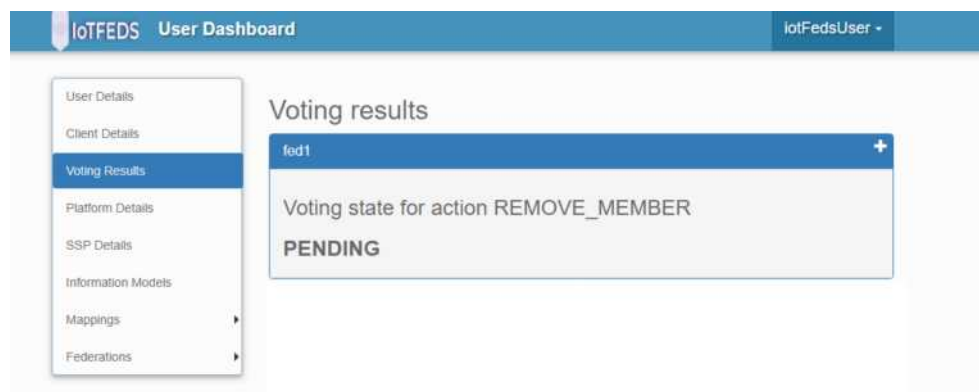
**Αναμενόμενα αποτελέσματα:**

Το μήνυμα επιβεβαίωσης για απομάκρυνση του μέλους από την ομοσπονδία εμφανίζεται στην οθόνη του χρήστη.



#### Αποτελέσματα που παρατηρήθηκαν:

Πατώντας το κουμπί “Activate voting to remove the member”, το αίτημα εμφανίζεται στη λίστα με τα αιτήματα ψηφοφορίας του χρήστη με την ένδειξη ότι η ψηφοφορία είναι σε εξέλιξη.



Κάθε μέλος της ομοσπονδίας ενημερώνεται για το αίτημα και αποκτάει πρόσβαση στην ψηφοφορία.

#### Αναμενόμενα αποτελέσματα:

Ο κάθε εμπλεκόμενος χρήστης ενημερώνεται με συνδεσμο για την ψηφοφορία.

#### Αποτελέσματα που παρατηρήθηκαν:

Ο κάθε εμπλεκόμενος χρήστης ενημερώνεται με συνδεσμο με email για την ψηφοφορία και υποβάλλει την ψήφο του.

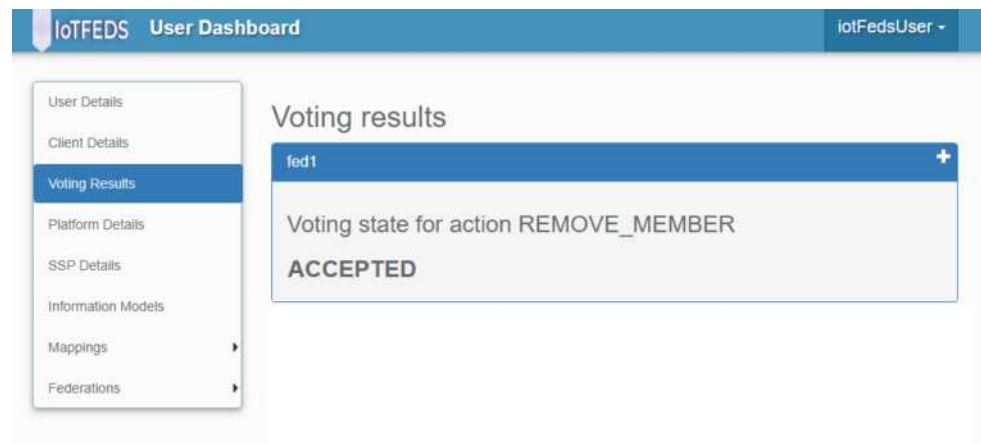
Μετά την ολοκλήρωση της ψηφοφορίας, κάθε μέλος της ομοσπονδίας ενημερώνεται για το αίτημα.

#### Αναμενόμενα αποτελέσματα:

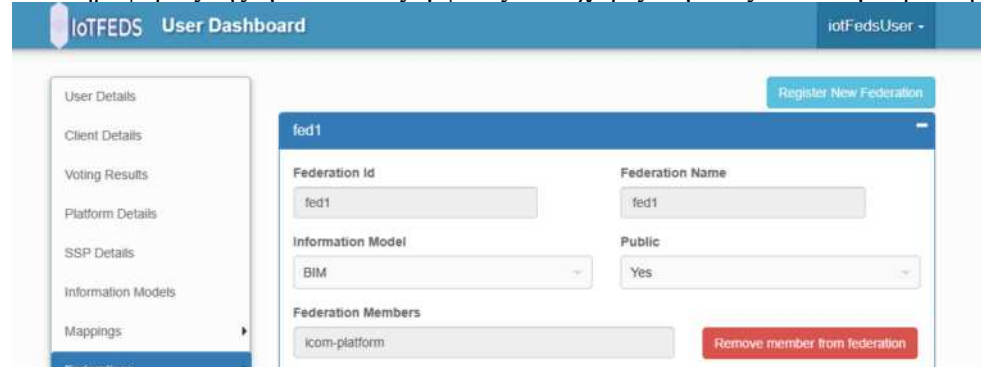
Στην περίπτωση θετικής ψήφου (που πληρούν τους κανόνες της ομοσπονδίας και όπως περιγράφεται στο Π2.1), ο χρήστης ενημερώνεται για την απομάκρυνσή του και η λίστα μελών της ομοσπονδίας ενημερώνεται.

#### Αποτελέσματα που παρατηρήθηκαν:

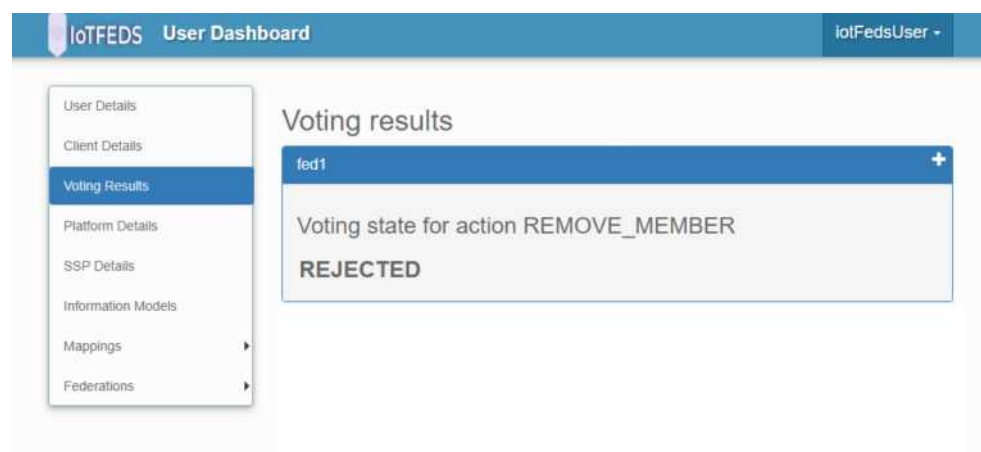
Στην περίπτωση που το αποτέλεσμα της ψηφοφορίας είναι θετικό για την απομάκρυνση του μέλους, τότε εμφανίζεται στην οθόνη της σελίδας “Voting Results” το παρακάτω μήνυμα αποτελέσματος της ψηφοφορίας:



Οι πληροφορίες της ομοσπονδίας εμφανίζονται χωρίς το μέλος που απομακρύνθηκε:



Στην περίπτωση που το αποτέλεσμα της ψηφοφορίας είναι αρνητικό για την απομάκρυνση του μέλους τότε εμφανίζεται στην οθόνη της σελίδας “Voting Results” το παρακάτω μήνυμα αποτελέσματος της ψηφοφορίας:



#### AdminGUI-11:

#### PASS

Η δοκιμή αυτή εξασφαλίζει ότι ένα ομόσπονδο μέλος της IoTFeds πλατφόρμας μπορεί να ζητήσει την τροποποίηση κανόνων σε ομοσπονδίες που συμμετέχει.

Ο χρήστης έχει πρόσβαση στις ομοσπονδίες που συμμετέχει και στις πληροφορίες τους.

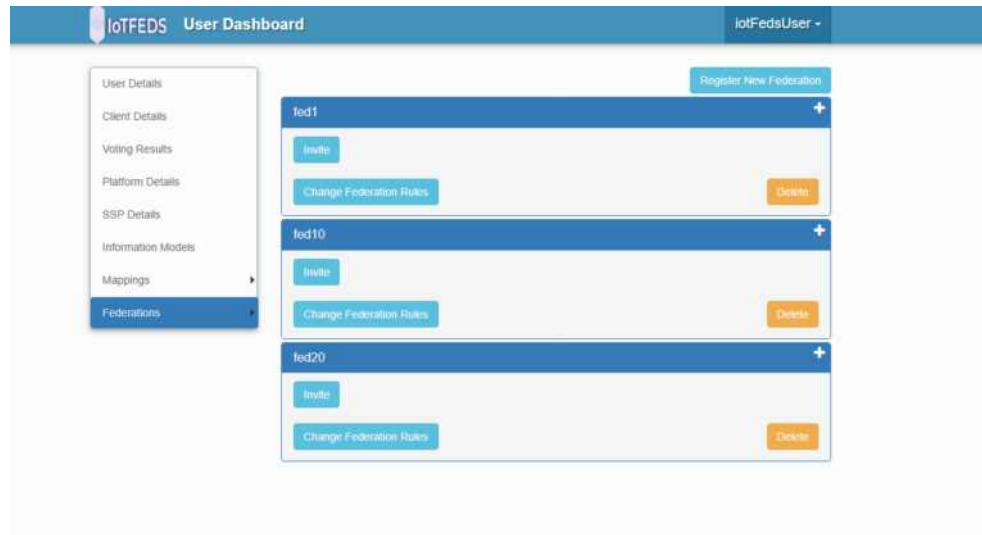


**Προαπαιτούμενη περίπτωση δοκιμής:**

AdminGUI-5 (Εγγραφή ομοσπονδίας στο IotFeds σύστημα).

**Αναμενόμενα αποτελέσματα:**

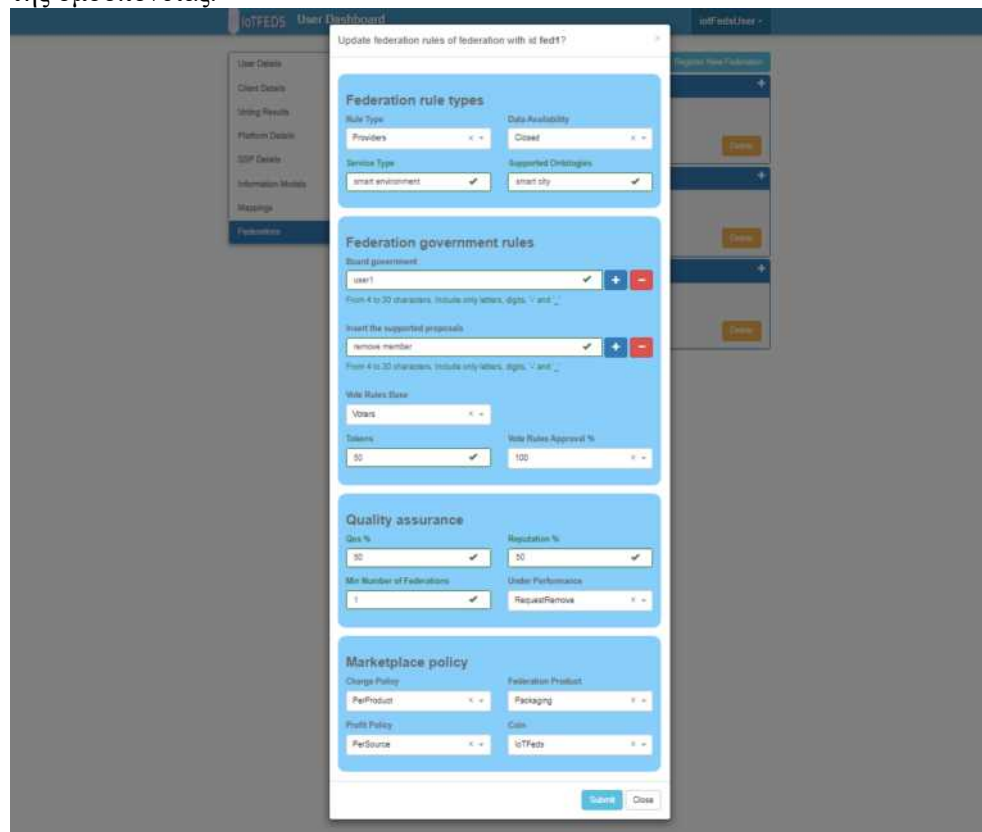
Ο χρήστης εισέρχεται στη σελίδα με τη λίστα των ομοσπονδιών:



Στη συνέχεια επιλέγει την ομοσπονδία στην οποία επιθυμεί να γίνει τροποποίηση κανόνων.

**Αποτελέσματα που παρατηρήθηκαν:**

Πατώντας το κουμπί “Change Federation Rules” εμφανίζεται η φόρμα με τους κανόνες της ομοσπονδίας.



Ο χρήστης επιλέγει την τροποποίηση των κανόνων της ομοσπονδίας.

**Αναμενόμενα αποτελέσματα:**

Ο χρήστης αφού ολοκληρώσει την τροποποίηση των κανόνων πατάει το κουμπί “Submit” για να προωθηθεί η τροποποίηση των κανόνων σε ψηφοφορία από τα μέλη της ομοσπονδίας.

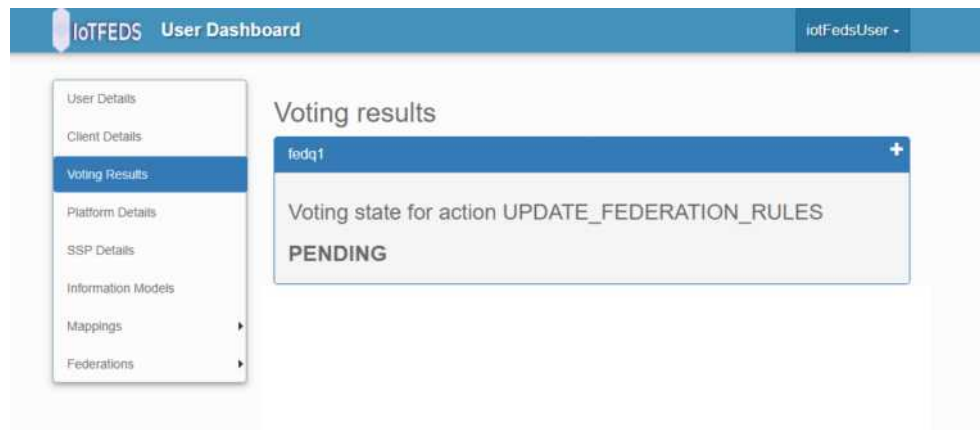
**Αποτελέσματα που παρατηρήθηκαν:**

Το παρακάτω μήνυμα ενημερώνει το χρήστη για την επιτυχημένη προώθηση της αίτησης για τροποποίηση κανόνων της ομοσπονδίας.

Your change federation rules request is forwarded to voting procedure



Στη σελίδα Voting Results αρχικά εμφανίζεται το μήνυμα ότι η ψηφοφορία για τροποποίηση κανόνων είναι σε εξέλιξη.



Κάθε μέλος της ομοσπονδίας ενημερώνεται για το αίτημα και αποκτάει πρόσβαση στην ψηφοφορία.

**Αναμενόμενα αποτελέσματα:**

Ο κάθε εμπλεκόμενος χρήστης ενημερώνεται με συνδεσμο για την ψηφοφορία.

**Αποτελέσματα που παρατηρήθηκαν:**

Ο κάθε εμπλεκόμενος χρήστης ενημερώνεται με συνδεσμο για την ψηφοφορία και υποβάλλει την ψήφο του.

Μετά την ολοκλήρωση της ψηφοφορίας σε περίπτωση θετικής ψήφου, οι κανόνες της ομοσπονδίας τροποποιούνται και ο χρήστης ενημερώνεται με τα αποτελέσματα της ψηφοφορίας.

**Αναμενόμενα αποτελέσματα:**

Ο χρήστης ενημερώνεται με το αποτέλεσμα της ψηφοφορίας και οι κανόνες της ομοσπονδίας στη φόρμα στοιχείων της εμφανίζονται τροποποιημένοι μόνο μετά από θετική ψήφο.

**Αποτελέσματα που παρατηρήθηκαν:**

Στην περίπτωση που η τροποποίηση των κανόνων έχει εγκριθεί από την ψηφοφορία, τότε στη σελίδα Voting Results εμφανίζεται το παρακάτω ενημερωτικό μήνυμα:



Στην περίπτωση που η τροποποίηση των κανόνων δεν έχει εγκριθεί από την ψηφοφορία, τότε στη σελίδα Voting Results εμφανίζεται το παρακάτω ενημερωτικό μήνυμα:

**AdminGUI-12****PASS**

Η δοκιμή αυτή εξασφαλίζει ότι ένας χρήστης μπορεί να διαγράψει μία πλατφόρμα του από το IoTFeds σύστημα.

Ο χρήστης έχει πρόσβαση στις πλατφόρμες του και τις πληροφορίες τους.

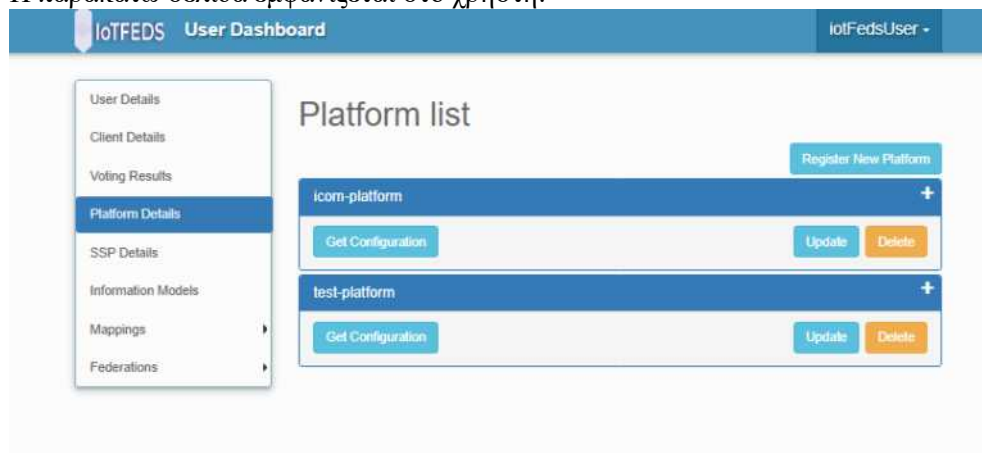
**Προαπαιτούμενη περίπτωση δοκιμής:**  
AdminGUI-4

**Αναμενόμενα αποτελέσματα:**

Οι πλατφόρμες του χρήστη εμφανίζονται σε λίστα στο μενού “Platform Details”.

**Αποτελέσματα που παρατηρήθηκαν:**

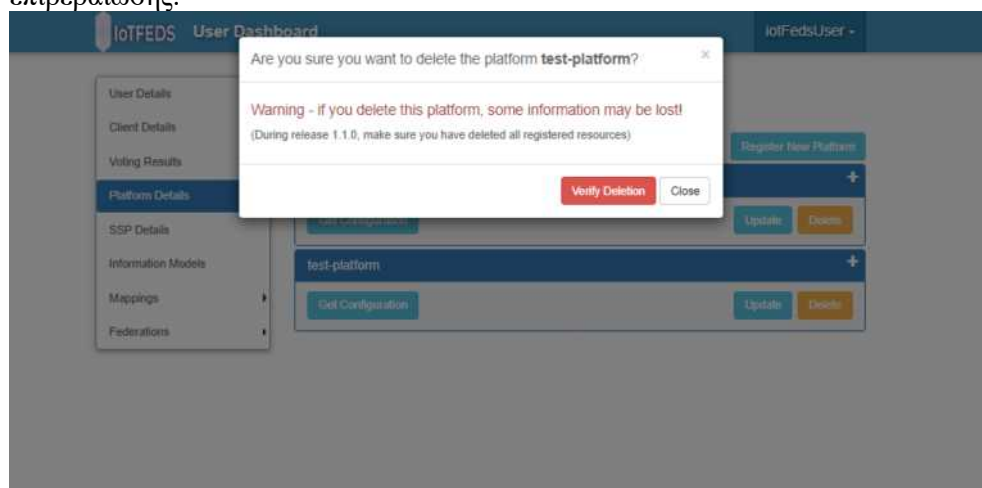
Η παρακάτω σελίδα εμφανίζεται στο χρήστη:



Ο χρήστης επιλέγει τη διαγραφή της πλατφόρμας του.

**Αναμενόμενα αποτελέσματα:**

Πατώντας το κουμπί “Delete” εμφανίζεται στην οθόνη του χρήστη μήνυμα επιβεβαίωσης.



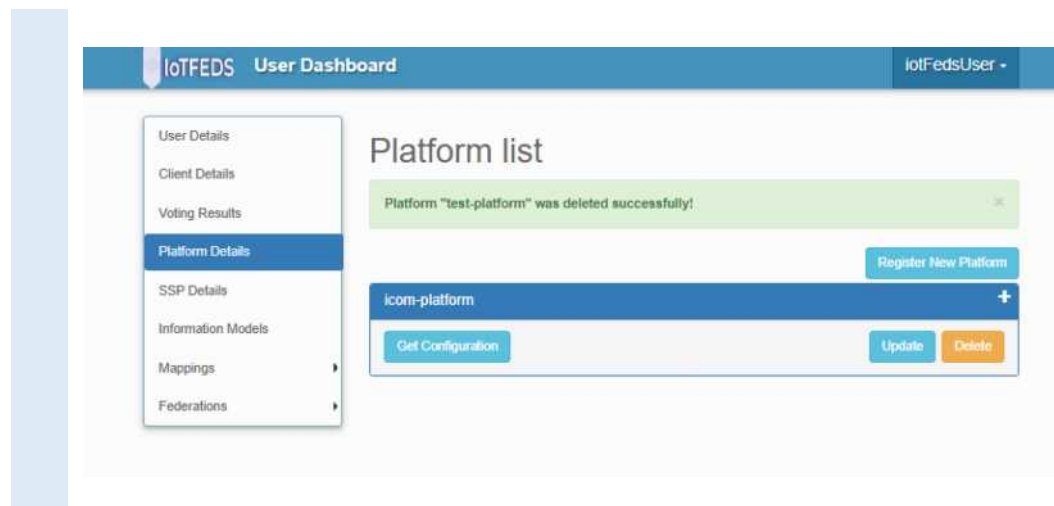
Στη συνέχεια ο χρήστης επιβεβαιώνει την πρόθεσή του για διαγραφή της πλατφόρμας. Ο χρήστης πατάει το κουμπί “Verify Deletion”.

**Αναμενόμενα αποτελέσματα:**

Το IoTfeds σύστημα επιβεβαιώνει την επιτυχή διαγραφή της πλατφόρμας εμφανίζοντας μήνυμα στην οθόνη του χρήστη. Το μενού “Platform Details” ενημερώνεται εμφανίζοντας μια ανανεωμένη λίστα από πλατφόρμες χωρίς την πλατφόρμα που διαγράφηκε.

**Αποτελέσματα που παρατηρήθηκαν:**

Η παρακάτω σελίδα εμφανίζεται στο χρήστη.



## 5 Συμπεράσματα

Στο παραδοτέο αυτό περιγράφεται η τελική έκδοση του ολοκληρωμένου συστήματος της πλατφόρμας IoTFeds. Επιπλέον, τεκμηριώνεται η ενσωμάτωση των υπηρεσιών των στοιχείων λογισμικού που αναπτύχθηκαν ή και επεκτάθηκαν στα πλαίσια του έργου. Η τελική αυτή έκδοση αποτελεί συνέχεια της ενδιάμεσης έκδοσης που παρουσιάστηκε στο παραδοτέο Π1.4 «Ενδιάμεση Έκδοση Ολοκληρωμένου Συστήματος» ενσωματώνοντας πλήρως και επιτυχώς όλους τους μηχανισμούς και λειτουργικότητες που αναπτύχθηκαν στις επιμέρους τεχνικές δράσεις του έργου (EE2, EE3).

Το παραδοτέο αρχικά συνοψίζει τη μεθοδολογία που ακολουθήθηκε για την ενσωμάτωση και τη δοκιμή της τελικής έκδοσης του ολοκληρωμένου συστήματος και των βασικών υποδομών που αξιοποιήθηκαν. Επιπλέον, περιγράφει την αρχιτεκτονική του τελικού πρωτοτύπου με την ενσωμάτωση των επιμέρους κατηγοριών υπηρεσιών: για τη διαχείριση ομοσπονδιών IoT, τις υπηρεσίες ελέγχου πρόσβασης, κατανεμημένες υπηρεσίες αναζήτησης και ανακάλυψης, υπηρεσίες φήμης και εμπιστοσύνης και τέλος υπηρεσίες αγοράς όπως ο έλεγχος και διεκπεραίωση συναλλαγών.

Στη συνέχεια παρουσιάζονται τα στοιχεία λογισμικού των υποσυστημάτων που συμμετέχουν στην υλοποίηση των μηχανισμών αυτών και οι αντίστοιχες διεπαφές που συμπεριλήφθηκαν στην τελική έκδοση του ολοκληρωμένου συστήματος.

Τέλος, ακολουθεί η λειτουργική δοκιμή της IoTFeds πλατφόρμας και η επίδειξη της χρήσης της όπως υλοποιήθηκε στην τελική έκδοση του ολοκληρωμένου συστήματος.

## Αναφορές

**There are no sources in the current document.**