



BidCoin

BID WITH CONFIDENCE

Προδιαγραφές Λογισμικού και Υλοποίηση συστήματος

Del.1.2

Version 0.6



Μηχανική Λογισμικού Ι
Τομέας Ηλεκτρονικής και Υπολογιστών
Τμήμα ΗΜΜΥ
Α.Π.Θ

8^ο Εξάμηνο
Άνοιξη 2023

Φίλιππος Γερμανόπουλος germanof@ece.auth.gr
Τρύφων Καλταπανίδης tryfkalt@ece.auth.gr
Φίλιππος Τόλιας fttolias@ece.auth.gr
Γεώργιος Παπακωστίδης papgeokon@ece.auth.gr

11/6/2023



Ιστορικό Αλλαγών

Όνομα	Ημερομηνία	Αλλαγή	Έκδοση
Α. Συμεωνίδης	17/05/2007	Δημιουργία εγγράφου. Προσαρμογή των προτύπων του K. E. Wiegers και του M. Smialek's.	0.1
Α. Συμεωνίδης	29/3/2014	Μικρή αναθεώρηση – τροποποίηση ενοτήτων	0.1.3
Χ. Ζολώτας	10/4/2020	Μεγάλη αναθεώρηση – αφαίρεση ενοτήτων	0.4
Χ. Ζολώτας	15/4/2020	Μεγάλη αναθεώρηση – προσθήκη ενότητας REST προδιαγραφών	0.5.3
Κ. Παναγιώτου	25/4/2020	Μεγάλη αναθεώρηση – προσθήκη ενότητας Nodered περιγραφής	0.5.7
Α. Συμεωνίδης	30/4/2020	Αναθεώρηση και τελική δομή προτύπου	0.6

Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Γερμανόπουλος Φίλιππος	17	germanof@ece.auth.gr
Καλαττανίδης Τρύφων	17	tryfkalt@ece.auth.gr
Τόλιας Φίλιππος	17	fttolias@ece.auth.gr
Παπακωστίδης Γεώργιος	17	papgeokon@ece.auth.gr



Πίνακας Περιεχομένων

Πίνακας Περιεχομένων	4
Λίστα Σχημάτων	6
1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν	8
2. Αρχιτεκτονική Συστήματος	11
2.1. Αναγνώριση Πόρων (Resources) Συστήματος	11
2.2. Τεκμηρίωση REST διεπαφής	12
2.2.1. Πόρος Auction	12
2.2.1.1. Μοντέλο δεδομένων Auction	12
2.2.1.2. Endpoint POST πόρου Auction, για “ενδιαφερόμενο” και συγκεκριμένο auction ID	13
2.2.1.3. Endpoint PUT πόρου Auction, για δημοπράτη και συγκεκριμένο auction ID	14
2.2.1.4. Endpoint GET πόρου Auction, για συγκεκριμένο auction ID	17
2.2.1.5. Endpoint POST πόρου Auction, για πωλητή και συγκεκριμένο user ID	19
2.2.2. ΠΟΡΟΣ USER	21
2.2.2.1. Μοντέλο δεδομένων User	21
2.2.2.5. Endpoint GET πόρου User, για συγκεκριμένο sender_ID , Chat_ID και message	28
2.2.2.6. Endpoint POST πόρου User, για συγκεκριμένο sender_ID , Chat_ID και δημιουργία chat	30
2.2.2.7. Endpoint GET πόρου User, για συγκεκριμένο sender_ID , Chat_ID και παραλαβή chat	32
2.2.3. ΠΟΡΟΣ ITEM	34
2.2.3.1. Μοντέλο δεδομένων item	34
2.2.3.2. Endpoint POST πόρου Item, για συγκεκριμένο user ID	35
2.2.3.4. Endpoint DELETE πόρου Item, για συγκεκριμένο user ID και item ID	39
2.2.3.5. Endpoint PUT πόρου Item, για συγκεκριμένο item ID	41
2.2.3.6. Endpoint POST πόρου Item, για συγκεκριμένο item ID	43
3. Υλοποίηση Συστήματος με Node-RED	45
3.1. Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed	45
3.1.1. Ροές πόρου User	45
3.1.2. Ροές πόρου Item	45
3.1.3. Ροές πόρου Auction	47
3.1.4. Ροές πόρου Chat	49
3.2. Υλοποίηση Ιστοριών χρήστη	50
3.2.1. Ιστορία Χρήστη User Search Item	50
3.2.2. Ιστορία Χρήστη User List Item	50
3.2.3. Ιστορία Χρήστη User Edit Item Profile	51
3.2.4. Ιστορία Χρήστη User Delete Item	51



3.2.5. Ιστορία Χρήστη User Declare Bid	52
3.2.6. Ιστορία Χρήστη User Create Event (Create Auction)	52
3.2.7. Ιστορία Χρήστη User Control Auction	53
3.2.8. Ιστορία Χρήστη User Join Auction	53
3.2.9. Ιστορία Χρήστη User Pay Item	54
3.2.10. Ιστορία Χρήστη User Puts Item in Favourite	54
3.2.11. Ιστορία Χρήστη User Edit Profile	55
3.2.12. Ιστορία Χρήστη User Messaging	55
Παράρτημα Ι – Πίνακας Ιχνηλασιμότητας	58
Παράρτημα ΙΙ – Ανοιχτά Θέματα	59



Λίστα Σχημάτων

- Σχήμα 1: Εφαρμογή προτύπου Abstract Factory
Σχήμα 2: Εφαρμογή προτύπου Proxy
Σχήμα 3: Εφαρμογή συνδυασμού προτύπων Bridge-Template
Σχήμα 4: Μοντέλο δεδομένων Auction και μοντέλο δεδομένων Bid
Σχήμα 5: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 6: Αποκρίσεις του συστήματος
Σχήμα 7: Δοκιμή του endpoint
Σχήμα 8: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 9: Αποκρίσεις του συστήματος
Σχήμα 10: Δοκιμή του endpoint
Σχήμα 11: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 12: Αποκρίσεις του συστήματος
Σχήμα 13: Δοκιμή του endpoint
Σχήμα 14: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 15: Αποκρίσεις του συστήματος
Σχήμα 16: Δοκιμή του endpoint
Σχήμα 17: Μοντέλο δεδομένων User, μοντέλο δεδομένων Chat και μοντέλο δεδομένων Message
Σχήμα 18: Εξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 19: Αποκρίσεις του συστήματος
Σχήμα 20: Δοκιμή του endpoint
Σχήμα 21: Εξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 22: Αποκρίσεις του συστήματος
Σχήμα 23: Δοκιμή του endpoint
Σχήμα 24: Εξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 25: Αποκρίσεις του συστήματος
Σχήμα 26: Δοκιμή του endpoint
Σχήμα 27: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 28: Αποκρίσεις του συστήματος
Σχήμα 29: Δοκιμή του endpoint
Σχήμα 30: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 31: Αποκρίσεις του συστήματος
Σχήμα 32: Δοκιμή του endpoint
Σχήμα 33: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 34: Αποκρίσεις του συστήματος
Σχήμα 35: Δοκιμή του endpoint
Σχήμα 36: Μοντέλο δεδομένων Item και μοντέλο δεδομένων Pay
Σχήμα 37: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 38: Αποκρίσεις του συστήματος
Σχήμα 39: Δοκιμή του endpoint
Σχήμα 40: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 41: Αποκρίσεις του συστήματος
Σχήμα 42: Δοκιμή του endpoint
Σχήμα 43: Επεξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 44: Αποκρίσεις του συστήματος
Σχήμα 45: Δοκιμή του endpoint
Σχήμα 46: Εξήγηση και παράθεση των παραμέτρων του πόρου
Σχήμα 47: Αποκρίσεις του συστήματος
Σχήμα 48: Δοκιμή του endpoint



Μηχανική Λογισμικού Ι
Τομέας Ηλεκτρονικής και Υπολογιστών
Τμήμα ΗΜΜΥ
Α.Π.Θ

8^ο Εξάμηνο
Άνοιξη 2023

Σχήμα 49:Εξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 50:Αποκρίσεις του συστήματος

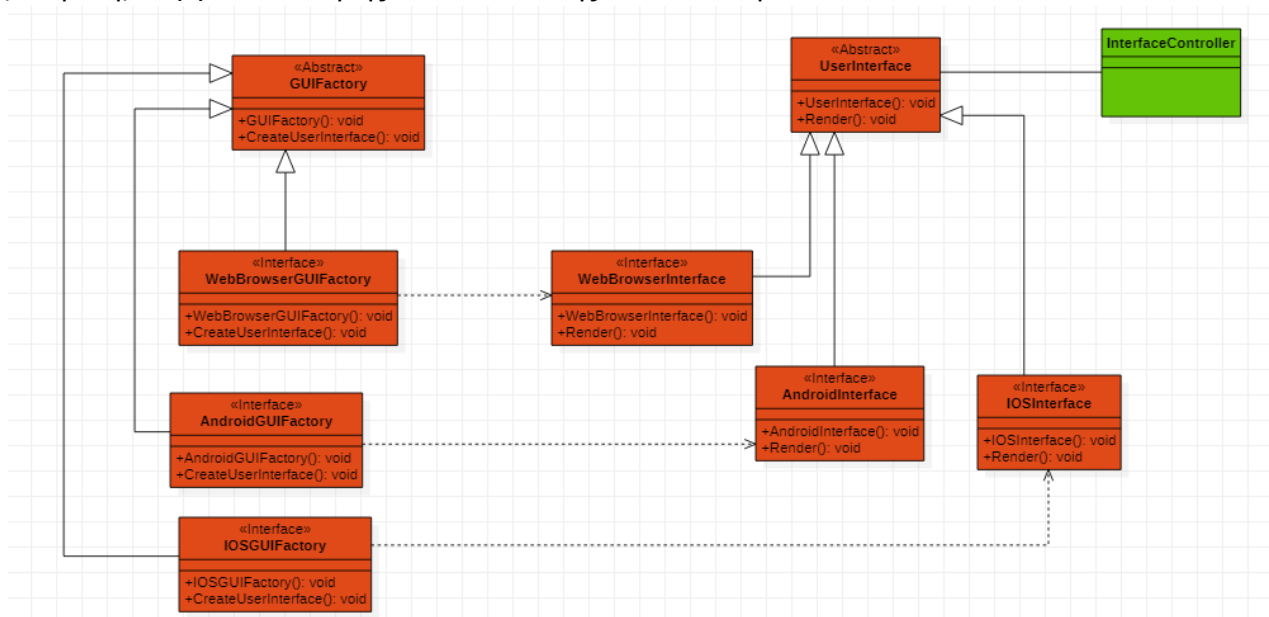
Σχήμα 51:Δοκιμή του endpoint



1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν

1.1 Πρότυπο Abstract Factory

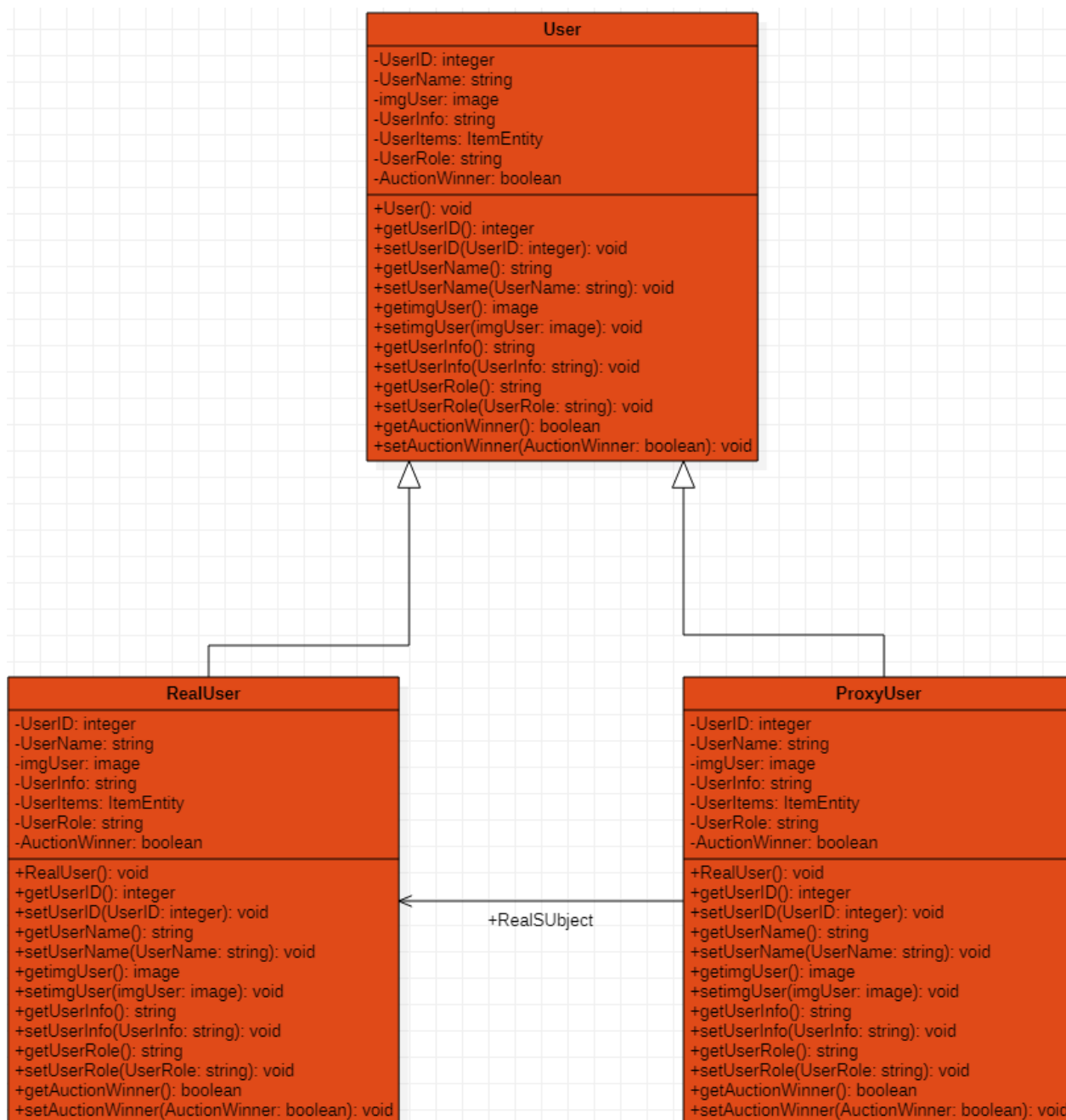
Πρόκειται για ένα Δημιουργικό Πρότυπο. Στο σύστημα μας με την χρήση του προτύπου Abstract Factory, το σύστημα μας μπορεί να υποστηρίξει διαφορετικού τύπου συσκευές (Android, iOS καθώς και web browsers) και να επικοινωνεί μαζί τους, κάτι το οποίο είναι απαραίτητο για την κάλυψη της ΜΛΑ-1 που ορίστηκε στο έγγραφο απαιτήσεων χρήστη. Συγκεκριμένα, μέσω των υλοποιήσεων του Abstract Product `UserInterface`, είναι εφικτή η πρόσβαση στα αρχεία πολυμέσων σε συσκευές διαφόρων τύπων. Επίσης με χρήση αυτού του προτύπου δίνεται η δυνατότητα, εάν στο μέλλον υπάρξουν και νέες δημοφιλείς συσκευές, να μπορεί το σύστημα μας να αλληλεπιδρά και με αυτές με την δημιουργία κατάλληλης concrete κλάσης στο κατάλληλο Abstract Product.



Σχήμα 1: Εφαρμογή προτύπου Abstract Factory.

1.2 Πρότυπο Proxy

Το πρότυπο proxy είναι ένα δομικό πρότυπο. Στο σύστημα BidCoin, εφαρμόζεται με στόχο την κάλυψη της ΜΛΑ-2 που πραγματεύεται την αξιοπιστία και ασφάλεια του συστήματος έναντι κακόβουλων επιθέσεων και την προστασία των προσωπικών δεδομένων των χρηστών κατά GDPR, μέσω του ελέγχου πρόσβασης σε ένα αντικείμενο της κλάσης `User`. Πιο συγκεκριμένα, ένα ζήτημα μπορεί να προβληθεί μόνο από ταυτοποιημένους χρήστες, οι οποίοι μάλιστα έχουν διαφορετικά δικαιώματα πρόσβασης ανάλογα με την ιδιότητα τους (ενδιαφερόμενος, πωλητής ή δημοπράτης). Επομένως, στην κλάση `User` εφαρμόσαμε ένα protection proxy. Επίσης το Proxy πρότυπο, με κατάλληλη υλοποίηση μπορεί να λειτουργήσει και ως virtual proxy, προκειμένου να φορτώνει ένα προσωρινό αντικείμενο (εικόνα), έως ότου δοθεί πρόσβαση στο πραγματικό περιεχόμενο ενός αντικειμένου. Η περίπτωση αυτή συμβάλει στην ομαλότερη εμπειρία χρήστη.

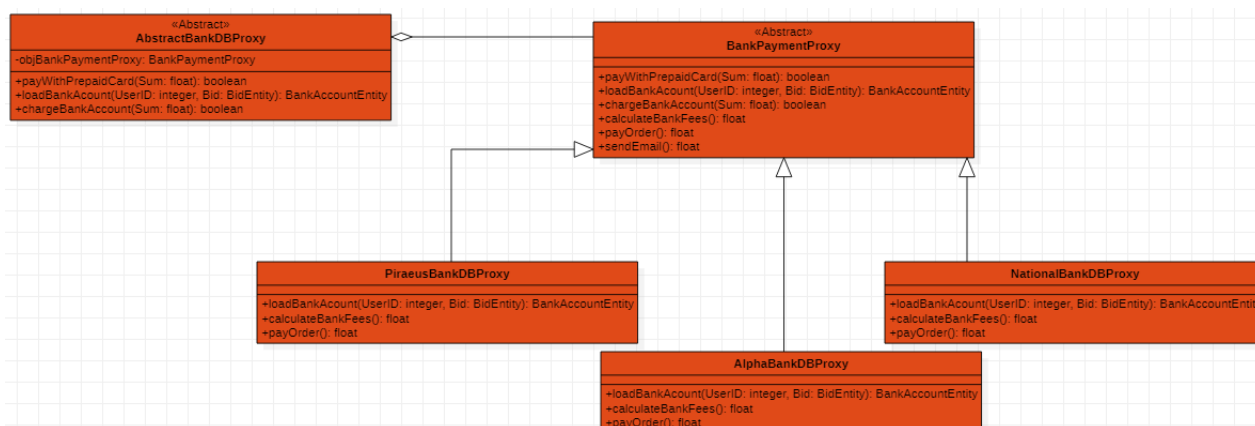


Σχήμα 2: Εφαρμογή προτύπου Proxy.



1.3 Συνδυασμός Προτύπων Bridge- Template

Πρόκειται για έναν συνδυασμό δομικού προτύπου και προτύπου συμπεριφοράς. Εφαρμόζεται με στόχο την κάλυψη της ΜΛΑ-6 που αφορά τους τρόπους πληρωμής του συστήματος για την αγορά ενός αντικειμένου. Ειδικότερα, πρόκειται για μία διαδικασία αυτόνομη η οποία μπορεί να υπόκειται σε διαφοροποιήσεις, αναλόγως τον τρόπο πληρωμής, ή την τράπεζα με την οποία συνεργάζεται ο εκάστοτε χρήστης. Επιτρέπει λοιπόν το σύστημα να εκπονήει την διαδικασία πληρωμής εκ νέου, ανεξάρτητα από τις διαφοροποιήσεις αυτές. Η εκμετάλλευση του Bridge προτύπου ορίζει υπάρχουσες και μελλοντικές διεπαφές τραπεζικών συστημάτων. Η προσθήκη του προτύπου Template δίνει την δυνατότητα στις τραπεζικές συναλλαγές να ολοκληρώνονται, σύμφωνα με τις εξατομικευμένες προβλεπόμενες διαδικασίες της κάθε τράπεζας, αφήνοντας έτσι περιθώριο για μεγαλύτερη ευελιξία και επεκτασιμότητα στο σύστημα.



Σχήμα 3: Εφαρμογή συνδυασμού προτύπων Bridge-Template



2. Αρχιτεκτονική Συστήματος

Χρησιμοποιήθηκε το εργαλείο Swaggerhub.

- [Σύνδεσμος](#) για το αρχείο τύπου JSON με τις προδιαγραφές του API.
- [Σύνδεσμος](#) για το αρχείο τύπου zip που περιέχει τον κώδικα για τον server της εφαρμογής.
- [Σύνδεσμος](#) για το API στο Swaggerhub.

2.1. Αναγνώριση Πόρων (Resources) Συστήματος

Κλάση BEC	Πόρος REST	Endpoints(HTTP Verbs)
Auction,(User)	/auction/{auction_id}/bid	POST
Auction,(User)	/auction/{auction_id}	GET, PUT
Auction,(User)	/user/{user_id}/auction	POST
Item,(User)	/user/{user_id}/item	POST
Item,(User)	/user/{user_id}/item/{item_id}	PUT, DELETE
Item,(User)	/item/search	GET
Item,(User)	/item/{item_id}	PUT
Item,(User)	/item/{item_id}/pay	POST
User,(User)	/user/{user_id}	PUT
User,(User)	/user/{sender_id}/chat/{chat_id}/message	GET, POST
User,(User)	/user/{sender_id}/chat	GET
User,(User)	/user/{sender_id}/chat/{chat_id}	POST



2.2. Τεκμηρίωση REST διεπαφής

2.2.1. Πόρος Auction

2.2.1.1. Μοντέλο δεδομένων Auction

```
Auction ∨ {  
  name*                string  
  user_id*              integer($int64)  
  Date*                 string  
  item_id*              integer($int64)  
  clockTimer*           integer($int64)  
  clockStart*           boolean  
  max_bid_increment*    integer($int64)  
  min_bid_increment*    integer($int64)  
  auction_id*           integer($int64)  
  
}
```

```
Bid ∨ {  
  user_id*              integer($int64)  
  sum*                  integer($int64)  
  item_id*              integer($int64)  
  
}
```

Σχήμα 4: Μοντέλο δεδομένων Auction και μοντέλο δεδομένων Bid.



2.2.1.2.Endpoint POST πόρου Auction, για “ενδιαφερόμενο” και συγκεκριμένο auction ID

POST /auction/{auction_id}/bid

FR1: The "Interested" user must be able to declare a bid for an item

Parameters

Name	Description
auction_id * required	ID of auction to bid in
integer(\$int64) (path)	
	35

Request body required

application/json

Sum of the bid

```
{  "user_id": 0,  "sum": 0,  "item_id": 0}
```

Σχήμα 5: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Code	Description	Links
201	new bid response	No links
Media type application/json		
Controls Accept header.		
Example Value Schema		
{ "user_id": 0, "sum": 0, "item_id": 0}		
default	unexpected error	No links
Media type application/json		
Example Value Schema		
{ "code": 0, "message": "string"}		

Σχήμα 6: Αποκρίσεις του συστήματος.



cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/35/bid' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "user_id": 0,
    "sum": 0,
    "item_id": 0
  }'
```

Request URL

https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/35/bid

Server response

Code	Details
201	<div>Response body</div> <div><pre>{ "user_id": 0, "sum": 0, "item_id": 0 }</pre></div> <div>Download</div> <div>Response headers</div> <div><pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip content-length: 52</pre></div>

Σχήμα 7: Δοκιμή του endpoint.

2.2.1.3.Endpoint PUT πόρου Auction, για δημοπράτη και συγκεκριμένο auction ID



PUT /auction/{auction_id} Control the flow of an auction. ^ ↩

FR7: The "Auctioneer" user must be able to control the flow of the auction

Parameters Cancel

Name	Description
auction_id * required integer(\$int64) (path)	ID of auction which needs to be edited

Request body * required application/json

Auction

```
{
  "name": "string",
  "user_id": 0,
  "date": "string",
  "item_id": 0,
  "clockTimer": 0,
  "clockStart": true,
  "max_bid_increment": 0,
  "min_bid_increment": 0,
  "auction_id": 0
}
```

Responses		
Code	Description	Links
200	edit profile response	No links
default	unexpected error	No links
Media type		
application/json		
Controls Accept header.		
Example Value Schema		
<pre>{ "code": 0, "message": "string" }</pre>		

Σχήμα 8: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 9: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'PUT' \
'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/12' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "name": "string",
  "user_id": 0,
  "Date": "string",
  "item_id": 0,
  "clockTimer": 0,
  "clockStart": true,
  "max_bid_increment": 0,
  "min_bid_increment": 0,
  "auction_id": 0
}'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/12
```

Server response

Code	Details
200	<p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: *</pre>

Σχήμα 10: Δοκιμή του endpoint.



2.2.1.4.Endpoint GET πόρου Auction, για συγκεκριμένο auction ID

GET /auction/{auction_id}

FR17: The user must be able to participate in auctions

Parameters

Name	Description
auction_id * required	ID of auction to participate in
integer(\$int64)	
(path)	

35

Execute Clear

Σχήμα 11: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Code	Description	Links
200	auction to participate in	No links
Media type: application/json		
Controls: Accept header.		
Example Value Schema		
<pre>{ "name": "string", "user_id": 0, "date": "string", "item_id": 0, "clockTimer": 0, "clockStart": true, "max_bid_increment": 0, "min_bid_increment": 0, "auction_id": 0}</pre>		
default	unexpected error	No links
Media type: application/json		
Example Value Schema		
<pre>{ "code": 0, "message": "GetItem"</pre>		

Σχήμα 12: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/35' \
  -H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/auction/35
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "name": "string", "user_id": 0, "Date": "string", "item_id": 0, "clockTimer": 0, "clockStart": true, "max_bid_increment": 0, "min_bid_increment": 0, "auction_id": 0 }</pre> <p>Download</p>

Response headers

Σχήμα 13: Δοκιμή του endpoint.



2.2.1.5.Endpoint POST πόρου Auction, για πωλητή και συγκεκριμένο user ID

POST /user/{user_id}/auction

FR8: The "Seller" user must be able to create an auction event

Parameters

Name	Description
user_id * required	ID of auction which needs to be edited
integer(\$int64)	(path)
	23

Request body required application/json

Auction

```
{  "name": "string",  "user_id": 0,  "Date": "string",  "item_id": 0,  "clockTime": 0,  "clockStart": true,  "max_bid_increment": 0,  "min_bid_increment": 0}
```

Σχήμα 14: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Code	Description	Links
201	edit profile response	No links
Media type: application/json		
Controls: Accept header.		
Example Value Schema		
<pre>{ "name": "string", "user_id": 0, "Date": "string", "item_id": 0, "clockTime": 0, "clockStart": true, "max_bid_increment": 0, "min_bid_increment": 0, "auction_id": 0}</pre>		
default	unexpected error	No links
Media type: application/json		
Example Value Schema		
<pre>{ "code": 0,</pre>		

Σχήμα 15: Αποκρίσεις του συστήματος.



cURL (bash)cURL (PowerShell)cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/15/auction' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "string",
    "user_id": 0,
    "Date": "string",
    "item_id": 0,
    "clockTimer": 0,
    "clockStart": true,
    "max_bid_increment": 0,
    "min_bid_increment": 0
  }'
```

Request URL

https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/15/auction

Server response

Code	Details
201	<div>Response body</div> <pre>{ "name": "string", "user_id": 0, "Date": "string", "item_id": 0, "clockTimer": 0, "clockStart": true, "max_bid_increment": 0, "min_bid_increment": 0, "auction_id": 0 }</pre> <div>Download</div>

Σχήμα 16: Δοκιμή του endpoint.



2.2.2. ΠΟΡΟΣ USER

2.2.2.1. Μοντέλο δεδομένων User

```
User ∨ {  
  user_id*           integer($int64)  
  name               string  
  userInfo*          string  
  UserRole*          string  
  AuctionWinner      boolean  
}
```

```
Chat ∨ {  
  user_id             integer($int64)  
  recipient_id*       integer($int64)  
  read*               boolean  
  chat_id*            integer($int64)  
}
```

```
Message ∨ [  
  required: List [ "content" ]  
  properties: OrderedMap { "content": OrderedMap { "type": "string" } }  
  Message ∨ { ↵  
    content*          string  
  }  
}]
```

Σχήμα 17: Μοντέλο δεδομένων User, μοντέλο δεδομένων Chat και μοντέλο δεδομένων Message.



2.2.2.2. Endpoint PUT πόρου User, για συγκεκριμένο user ID

PUT /user/{user_id}

FR5: The user must be able to edit his profile

Parameters Cancel

Name	Description
user_id * required	ID of user who wants to edit a profile
integer(\$int64)	
(path)	
	<input type="text" value="563"/>

Request body required application/json

User

```
{  "user_id": 0,  "name": "string",  "userInfo": "string",  "UserRole": "string",  "AuctionWinner": true}
```

Σχήμα 18: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Responses		
Code	Description	Links
200	edit profile response	No links
default	unexpected error	No links
Media type		
application/json		
Controls Accept header.		
Example Value Schema		
<pre>{ "code": 0, "message": "string"}</pre>		

Σχήμα 19: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'PUT' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/563' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "user_id": 0,
    "name": "string",
    "userInfo": "string",
    "UserRole": "string",
    "AuctionWinner": true
  }'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/563
```

Server response

Code	Details
200	<p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip content-length: 20</pre>

Σχήμα 20: Δοκιμή του endpoint.



2.2.2.3. Endpoint GET πόρου User για την αναζήτηση Item

GET /item/search

FR10: The user must be able to search for an item

Parameters

Name	Description
ItemName	Name of the searched item

string
(query)

ItemName

Execute Clear

Responses

Responses

Code	Description	Links
200	search item	No links

Media type
application/json

Controls Accept header.

Example Value Schema

```
{
  "name": "string",
  "user_id": 0,
  "FavouriteItem": true,
  "ItemDescription": "string",
  "StartingPrice": 0,
  "item_id": 0
}
```

default

unexpected error No links |

Media type
application/json

Example Value Schema

```
{
  "code": 0,
  "message": "string"
}
```

Σχήμα 21: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 22: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/item/search?ItemName=ItemName' \
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/item/search?ItemName=ItemName
```

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "name": "string", "user_id": 0, "FavouriteItem": true, "ItemDescription": "string", "StartingPrice": 1.1, "item_id": 0 }]</pre> <p>Response headers</p>

Σχήμα 23: Δοκιμή του endpoint.



2.2.2.4. Endpoint POST πόρου User, για συγκεκριμένο sender_ID , Chat_ID και message

POST

/user/{sender_id}/chat/{chat_id}/message

create a chat with other users.

FR15: The user must be able to send a personal message to other users

Parameters

Name	Description
chat_id * required	ID of chat
integer(\$int64)	
(path)	
	52
sender_id * required	ID of sender
integer(\$int64)	
(path)	
	12
recipient_id	ID of recipient
integer(\$int64)	
(query)	
	532

Request body required

application/json

NewMessage

{
 "content": "string"
}

Code	Description	Links
201	Message	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	[{ "content": "string" }]	
default	unexpected error	No links
	Media type	
	application/json	
	Example Value Schema	
	{ "code": 0, "message": "string" }	

Σχήμα 24: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 25: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/12/chat/52/' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "content": "string"
  }'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/12/chat/52/
message?recipient_id=532
```

Server response

Code	Details
201	<p>Response body</p> <pre>{ "content": "string" }</pre> <p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content- Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: *</pre>

Σχήμα 26: Δοκιμή του endpoint.



2.2.2.5. Endpoint GET πόρου User, για συγκεκριμένο sender_ID , Chat_ID και message

GET `/user/{sender_id}`
`/chat/{chat_id}`
`/message`

FR15: The user must be able to send a personal message to other users

Parameters Cancel

Name	Description
chat_id * required	ID of chat
integer(\$int64) (path)	
	<input type="text" value="21"/>
sender_id * required	ID of sender
integer(\$int64) (path)	
	<input type="text" value="3"/>
recipient_id	ID of recipient
integer(\$int64) (query)	
	<input type="text" value="24"/>

Σχήμα 27: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Code	Description	Links
201	message	No links
	Media type <div>application/json</div> <div>Controls Accept header.</div> <div>Example Value Schema</div> <div><pre>[{ "content": "string" }]</pre></div>	
default	unexpected error	No links
	Media type <div>application/json</div> <div>Example Value Schema</div> <div><pre>{ "code": 0, "message": "string" }</pre></div>	

Σχήμα 28: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/3/chat/21/m
  -H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/3/chat/21/m
essage?recipient_id=24
```

Server response

Code	Details
201	<div>Response body</div> <div><pre>{ "content": "string" }</pre></div> <div>Response headers</div> <div><pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content- Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip content-length: 46 content-type: application/json;charset=utf-8</pre></div>

Σχήμα 29: Δοκιμή του endpoint.



2.2.2.6. Endpoint POST πόρου User, για συγκεκριμένο sender_ID , Chat_ID και δημιουργία chat

POST	/user/{sender_id}/chat/{chat_id} Send direct messages to other users.	Code	Description	Links								
FR15: The user must be able to send a personal message to other users		201	Chat	No links								
<div>Parameters</div> <table><thead><tr><th>Name</th><th>Description</th></tr></thead><tbody><tr><td>chat_id * required integer(\$int64) (path)</td><td>ID of chat</td></tr><tr><td>sender_id * required integer(\$int64) (path)</td><td>ID of sender</td></tr><tr><td>recipient_id integer(\$int64) (query)</td><td>ID of recipient</td></tr></tbody></table>		Name	Description	chat_id * required integer(\$int64) (path)	ID of chat	sender_id * required integer(\$int64) (path)	ID of sender	recipient_id integer(\$int64) (query)	ID of recipient	<div>Media type application/json</div> <div>Controls Accept header.</div> <div>Example Value Schema</div> <pre>{ "user_id": 0, "recipient_id": 0, "read": true, "chat_id": 0}</pre>		
Name	Description											
chat_id * required integer(\$int64) (path)	ID of chat											
sender_id * required integer(\$int64) (path)	ID of sender											
recipient_id integer(\$int64) (query)	ID of recipient											
<div>Request body required</div> <div>application/json</div> <div>Newchat</div> <pre>{ "user_id": 0, "recipient_id": 0, "read": true}</pre>		default	unexpected error	No links								
			<div>Media type application/json</div> <div>Example Value Schema</div> <pre>{ "code": 0, "message": "string"}</pre>									

Σχήμα 30: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 31: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/12/chat/123' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "user_id": 0,
    "recipient_id": 0,
    "read": true
  }'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/12/chat/123?recipient_id=244
```

Server response

Code	Details
201	<p>Response body</p> <pre>{ "user_id": 0, "recipient_id": 0, "read": true, "chat_id": 0 }</pre> <p>Response headers</p>

Σχήμα 32: Δοκιμή του endpoint.



2.2.2.7. Endpoint GET πόρου User, για συγκεκριμένο sender_ID , Chat_ID και παραλαβή chat

GET

/user/{sender_id}
/chat/{chat_id}

FR15: The user must be able to send a personal message to other users

Parameters

Name	Description
chat_id * required	ID of chat
integer(\$int64) (path)	
	213
sender_id * required	ID of sender
integer(\$int64) (path)	
	3421
recipient_id	ID of recipient
integer(\$int64) (query)	
	222

ExecuteClear

Code	Description	Links
201	Chat	No links
	Media type application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "user_id": 0, "recipient_id": 0, "read": true, "chat_id": 0 }</pre>	
default	unexpected error	No links
	Media type application/json	
	Example Value Schema	
	<pre>{ "code": 0, "message": "string" }</pre>	

Σχήμα 33: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Σχήμα 34: Αποκρίσεις του συστήματος.



Responses

Snippets ▾

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/3421/chat/213?recipient_id=222' \
  -H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/3421/chat/213?recipient_id=222
```

Server response

Code	Details
201	<p>Response body</p> <pre>{ "user_id": 0, "recipient_id": 0, "read": true, "chat_id": 0 }</pre> <p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip</pre>

Σχήμα 35: Δοκιμή του endpoint.



2.2.3. ΠΟΡΟΣ ITEM

2.2.3.1. Μοντέλο δεδομένων item

```
Item ▾ {  
  name*                string  
  user_id*             integer($int64)  
  FavouriteItem*       boolean  
  ItemDescription*     string  
  StartingPrice*       number($float)  
  item_id*             integer($int64)  
  
}
```

```
Pay ▾ {  
  item_id*             integer($int32)  
  sum*                 integer($int32)  
  pay_id*              integer($int64)  
  
}
```

Σχήμα 36: Μοντέλο δεδομένων Item και μοντέλο δεδομένων Pay.



2.2.3.2.Endpoint POST πόρου Item, για συγκεκριμένο user ID

POST `/user/{user_id}/item` List a new item.

FR3: The "Seller" user must be able to list an item

Parameters

Name	Description
user_id * required	ID of user who wants to list an item
integer(\$int64)	
(path)	

122

Request body required

application/json

New Item

```
{  "name": "string",  "user_id": 0,  "FavouriteItem": true,  "ItemDescription": "string",  "StartingPrice": 0}
```

Σχήμα 37: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Responses

Code	Description	Links
201	new item response	No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
{  "name": "string",  "user_id": 0,  "FavouriteItem": true,  "ItemDescription": "string",  "StartingPrice": 0,  "item_id": 0}
```

default

unexpected error

Media type

application/json

Example Value | Schema

```
{  "code": 0,  "message": "string"}
```

Σχήμα 38: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/122/item' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "string",
    "user_id": 0,
    "FavouriteItem": true,
    "ItemDescription": "string",
    "StartingPrice": 0
  }'
```

Request URL

https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/122/item

Server response

Code	Details
201	<p>Response body</p> <pre>{ "name": "string", "user_id": 0, "FavouriteItem": true, "ItemDescription": "string", "StartingPrice": 1.1, "item_id": 0 }</pre> <p>Download</p>

Σχήμα 39: Δοκιμή του endpoint.



2.2.3.3. Endpoint PUT πόρου Item, για συγκεκριμένο user ID και item ID

PUT `/user/{user_id}/item/{item_id}` Edit item profile.

FR4: The "Seller" user must be able to edit his item's details

Parameters Cancel

Name	Description
user_id * required integer(\$int64) (path)	ID of user who wants to edit his item
<input type="text" value="123"/>	
item_id * required integer(\$int64) (path)	ID of user who wants to edit his item
<input type="text" value="1122"/>	

Request body required application/json

Item

```
{
  "name": "string",
  "user_id": 0,
  "FavouriteItem": true,
  "ItemDescription": "string",
  "StartingPrice": 0,
  "item_id": 0
}
```

Σχήμα 40: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Responses		
Code	Description	Links
200	item response	No links
default	unexpected error	No links
Media type		
<div>application/json</div>		
Controls Accept header.		
Example Value Schema		
<pre>{ "code": 0, "message": "string" }</pre>		

Σχήμα 41: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'PUT' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/123/item/1122' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "string",
    "user_id": 0,
    "FavouriteItem": true,
    "ItemDescription": "string",
    "StartingPrice": 0,
    "item_id": 0
  }'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/123/item/1122
```

Server response

Code	Details
200	<p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip</pre>

Σχήμα 42: Δοκιμή του endpoint.



2.2.3.4.Endpoint DELETE πόρου Item, για συγκεκριμένο user ID και item ID

DELETE `/user/{user_id}/item/{item_id}`

FR4: The "Seller" user must be able to edit his item's details

Parameters Cancel

Name	Description
user_id * required integer(\$int64) (path)	ID of user who wants to edit his item
item_id * required integer(\$int64) (path)	ID of user who wants to edit his item

Execute Clear

Σχήμα 43: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Code	Description	Links
200	item response	No links
default	unexpected error	No links

Media type
application/json

Controls Accept header.

Example Value | Schema

```
{  "code": 0,  "message": "string"}
```

Σχήμα 44: Αποκρίσεις του συστήματος.



cURL (bash)cURL (PowerShell)cURL (CMD)

```
curl -X 'DELETE' \
'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/21/item/23'
-H 'accept: application/json'
```

Request URL

'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/21/item/23'

Server response

Code	Details
200	<div>Response headers<div><pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip content-length: 20 content-type: application/json;charset=utf-8 date: Mon,19 Jun 2023 20:11:33 GMT etag: W/"14-yIMMzzqGPKico39NpXK600BdB3s" expires: -1 status: 200 OK</pre></div><div>Request duration<div>325 ms</div></div></div>

Σχήμα 45: Δοκιμή του endpoint.



2.2.3.5.Endpoint PUT πόρου Item, για συγκεκριμένο item ID

PUT /item/{item_id}

FR11: The "Interested" user must be able to mark an item as favourite

Parameters

Name	Description
item_id * required	ID of favourite item
integer(\$int64)	
(path)	

2

Execute Clear

Σχήμα 46: Επεξήγηση και παράθεση των παραμέτρων του πόρου

403

Error: response status is 403

Response headers

```
content-length: 0
date: Sun, 11 Jun 2023 19:09:19 GMT
status: 403 Forbidden
```

Responses

Code	Description	Links
200	favourite item	No links
default	unexpected error	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
  "code": 0,
  "message": "string"
}
```

Σχήμα 47: Αποκρίσεις του συστήματος.



Responses

Snippets▼

cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'DELETE' \
'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/21'
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/user/21/item/23
```

Server response

Code	Details
200	<p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: * cache-control: no-cache content-encoding: gzip content-length: 20 content-type: application/json;charset=utf-8 date: Mon,19 Jun 2023 10:50:07 GMT etag: W/"14-yIMzzqGpkico39NpXK600BdB3s" expires: -1 status: 200 OK</pre>

Σχήμα 48: Δοκιμή του endpoint.



2.2.3.6.Endpoint POST πόρου Item, για συγκεκριμένο item ID

POST /item
/{item_id}
/pay

FR13: The "Interested" user must be able to pay for the item

Parameters

Name	Description
item_id * required	ID of favourite item
integer(\$int64) (path)	
	243

Request body required

application/json

Pay

```
{  "item_id": 0,  "sum": 0}
```

Σχήμα 49: Επεξήγηση και παράθεση των παραμέτρων του πόρου

Responses		
Code	Description	Links
201	pay for an item	No links
Media type		
application/json		
Controls Accept header.		
Example Value Schema		
{ "item_id": 0, "sum": 0, "pay_id": 0}		
default	unexpected error	No links
Media type		
application/json		
Example Value Schema		
{ "code": 0, "message": "string"}		

Σχήμα 50: Αποκρίσεις του συστήματος.



Snippets▼

cURL (bash) cURL (PowerShell) cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/item/24
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "item_id": 0,
    "sum": 0
  }'
```

Request URL

```
https://virtserver.swaggerhub.com/GERMANOF_1/Bidcoin_API/3.0.0/item/24
3/pay
```

Server response

Code	Details
201	<p>Response body</p> <pre>{ "item_id": 0, "sum": 0, "pay_id": 0 }</pre> <p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: X-Requested-With,Content- Type,Accept,Origin access-control-allow-methods: * access-control-allow-origin: *</pre>

Σχήμα 51: Δοκιμή του endpoint.



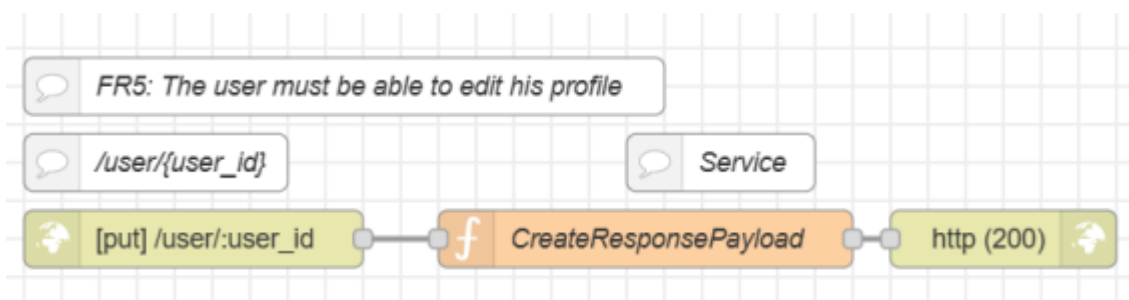
3. Υλοποίηση Συστήματος με Node-RED

3.1. Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed

- [Σύνδεσμος](#) για το αρχείο τύπου zip που περιέχει τα flows.

3.1.1. Ροές πόρου User

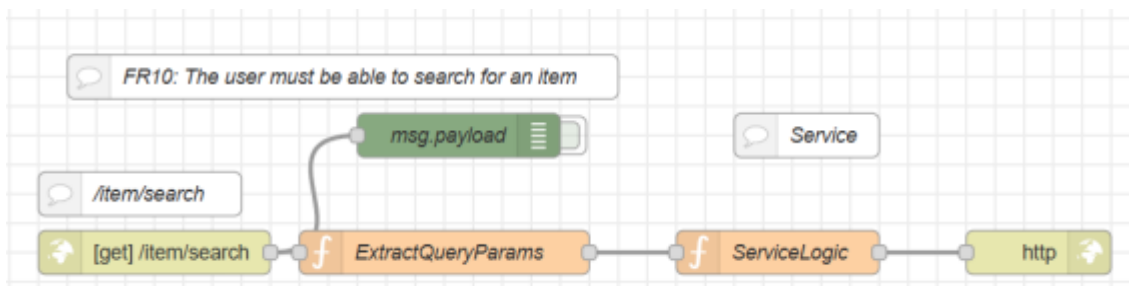
Ροή endpoint put /user/{user_id}



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να επεξεργαστεί τις πληροφορίες στο προφίλ του.

3.1.2. Ροές πόρου Item

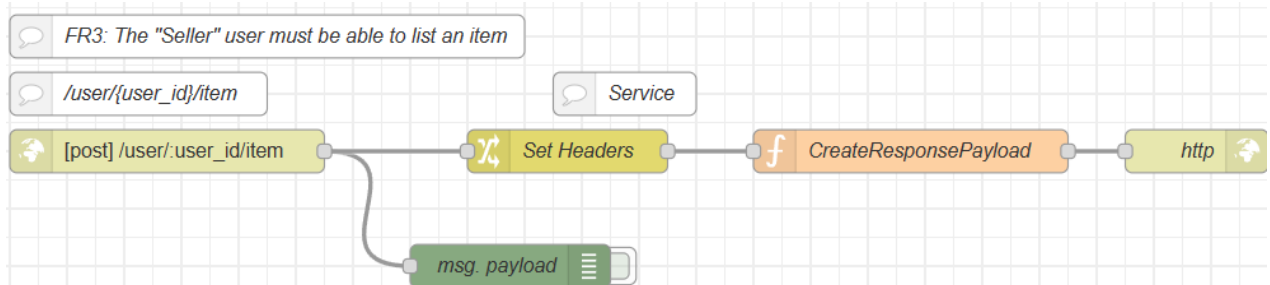
Ροή endpoint get /item/search



Ροή που υλοποιεί την υπηρεσία κατά την οποία ο χρήστης μπορεί να αναζητήσει αντικείμενα με βάση την ονομασία τους.

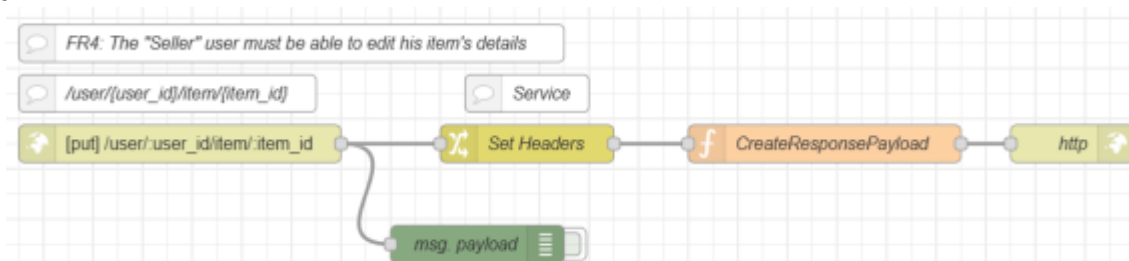


Ροή endpoint post /user/{user_id}/item



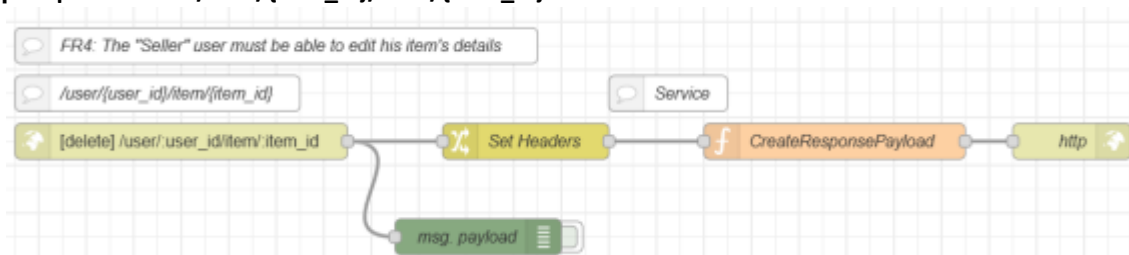
Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να καταχωρήσει ένα νέο αντικείμενο (να δημιουργήσει το προφίλ) στο σύστημα.

Ροή endpoint put /user/{user_id}/item/{item_id}



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να ενημερώσει το περιεχόμενο ενός αντικειμένου (δηλαδή το προφίλ του) που καταχώρησε νωρίτερα στο σύστημα.

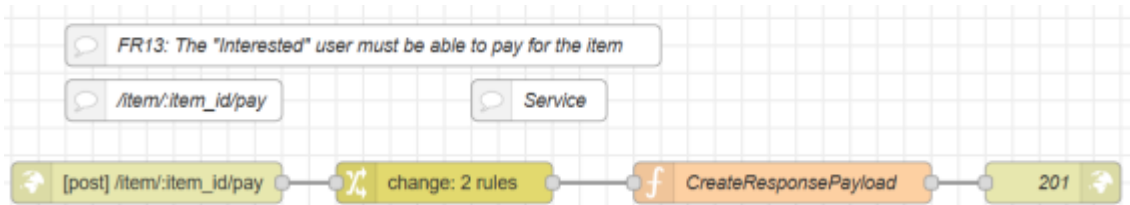
Ροή endpoint delete /user/{user_id}/item/{item_id}



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να διαγράψει ένα νέο αντικείμενο στο σύστημα.



Ροή endpoint post /item/{item_id}/pay



Ροή που φτιάχνει την υπηρεσία κατά την οποία ο χρήστης μπορεί να πληρώσει για το αντικείμενο που έχει κερδίσει.

Ροή endpoint put /item/{item_id}



Ροή που υλοποιεί την υπηρεσία κατά την οποία ο χρήστης μπορεί να θέσει ένα αντικείμενο ως “Αγαπημένο” ενημερώνοντας την ιδιότητά του.

3.1.3. Ροές πόρου Auction

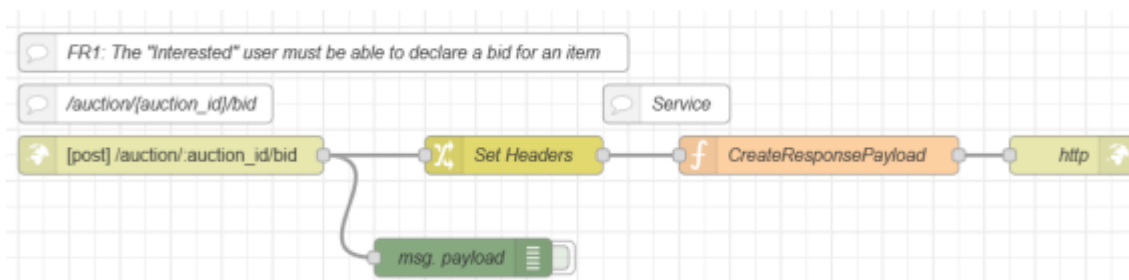
Ροή endpoint post /user/{user_id}/auction



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να καταχωρήσει μία νέα δημοπρασία στο σύστημα.



Ροή endpoint post /auction/{auction_id}/bid



Ροή που δημιουργεί την υπηρεσία κατά την οποία ο χρήστης μπορεί να ορίσει ποσό για δημοπρασία.

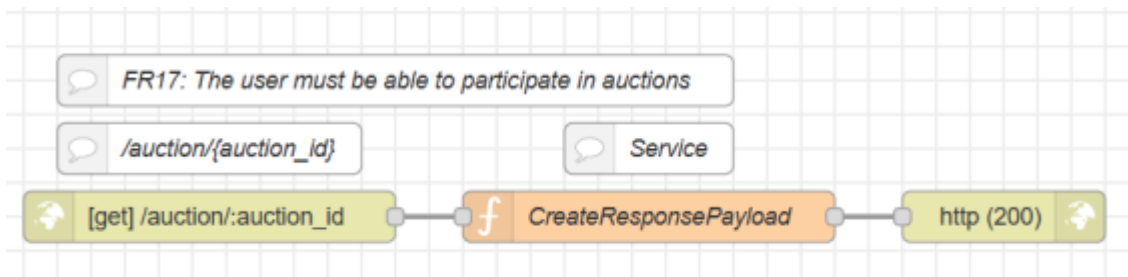
Ροή endpoint put /auction/{auction_id}



Ροή που δημιουργεί την υπηρεσία κατά την οποία ο χρήστης μπορεί να ελέγξει την πορεία της δημοπρασίας αλλάζοντας:

- 1) Το ρολόι μέτρησης
- 2) Το μέγιστο και ελάχιστο βήμα αύξησης προσφορών.

Ροή endpoint get auction/{auction_id}



Ροή που υλοποιεί την υπηρεσία κατά την οποία ο χρήστης μπορεί να συμμετέχει σε μια δημοπρασία.

Η υπηρεσία επιστρέφει στον χρήστη τα στοιχεία της δημοπρασίας. Μέσω της συνάρτησης CreateResponsePayload, επιστρέφεται μια δημοπρασία με ειδικό ID: {auction_id}, όπως ορίζεται στο URL Path parameter.



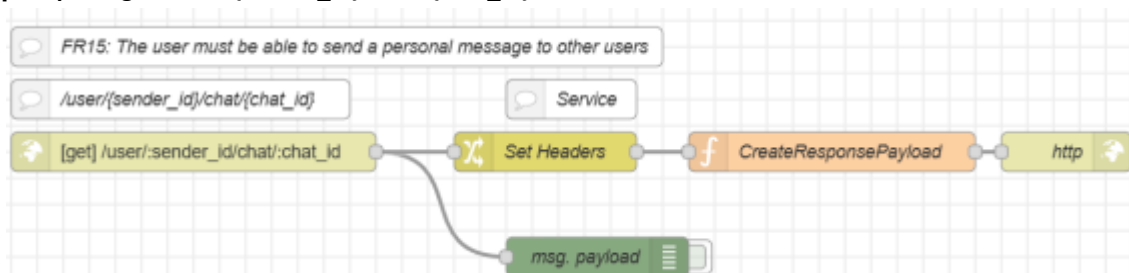
3.1.4. Ροές πόρου Chat

Ροή endpoint post /user/{sender_id}/chat



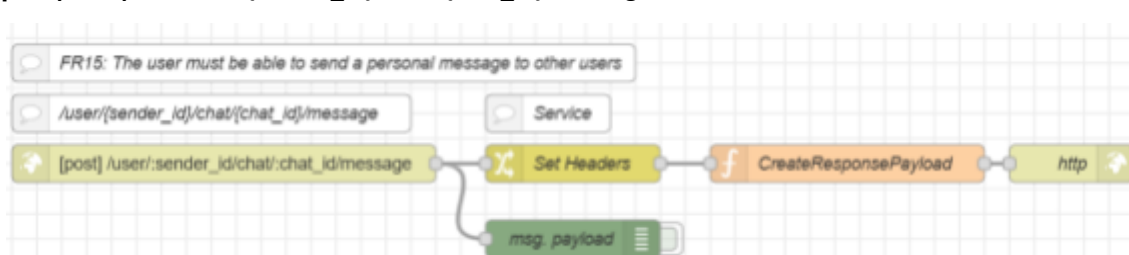
Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να καταχωρήσει μία νέα συνομιλία στο σύστημα.

Ροή endpoint get /user/{sender_id}/chat/{chat_id}



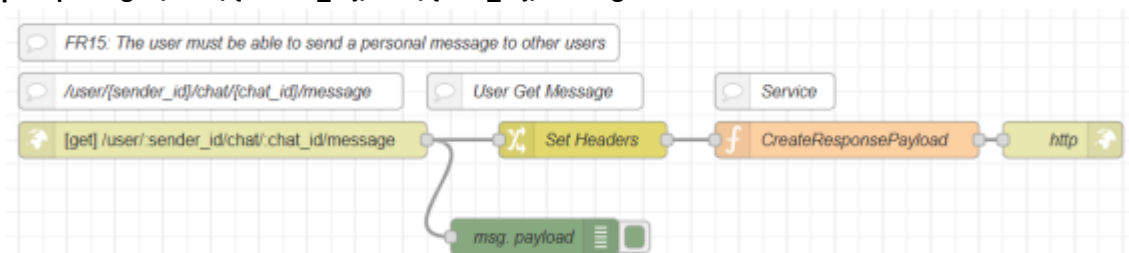
Ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή μίας συνομιλία.

Ροή endpoint post /user/{sender_id}/chat/{chat_id}/message



Ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να καταχωρήσει ένα νέο μήνυμα στο σύστημα.

Ροή endpoint get /user/{sender_id}/chat/{chat_id}/message



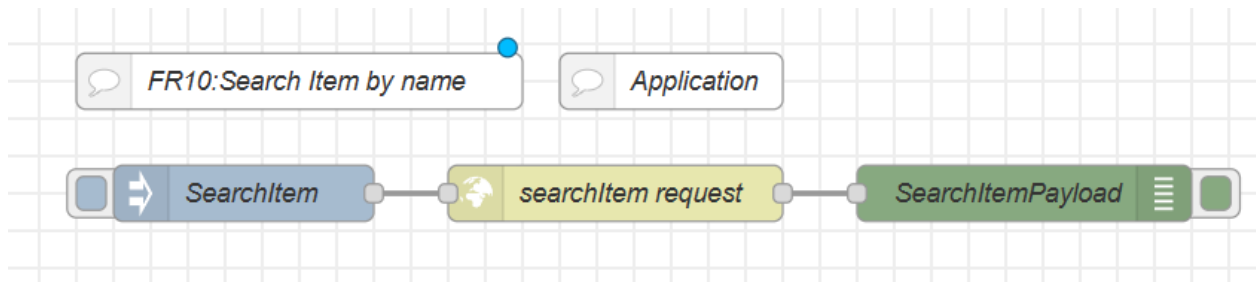
Ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή ενός νέου μηνύματος στον χρήστη.



3.2. Υλοποίηση Ιστοριών χρήστη

3.2.1. Ιστορία Χρήστη User Search Item

Ροή κατά την οποία ο χρήστης μπορεί να αναζητήσει αντικείμενα με βάση την ονομασία τους.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
SearchItem	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
searchItem request	http-request	Κάνει κλήση προς την υπηρεσία GET /item/search η οποία επιστρέφει στον χρήστη τα αποτελέσματα της αναζήτησης αντικειμένων με βάση την ονομασία τους.
SearchItemPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή το αντικείμενο που αναζητά.

3.2.2. Ιστορία Χρήστη User List Item

Ροή κατά την οποία ο χρήστης μπορεί να κάνει List κάποιο αντικείμενο δημιουργώντας έτσι το προφίλ του.

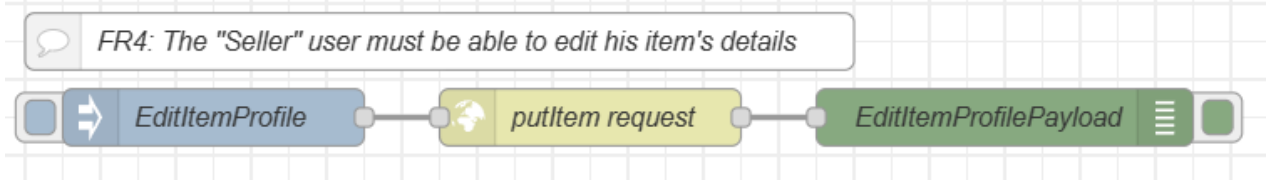


Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
AddItem	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
postItem request	http-request	Κάνει κλήση προς την υπηρεσία Post /user/{user_id}/item η οποία δίνει στο χρήστη με ID: {user_id} την δυνατότητα να προσθέσει ένα καινούριο αντικείμενο στην λίστα αντικειμένων του.
PostItemProfilePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή το αντικείμενο που δημιούργησε.



3.2.3. Ιστορία Χρήστη User Edit Item Profile

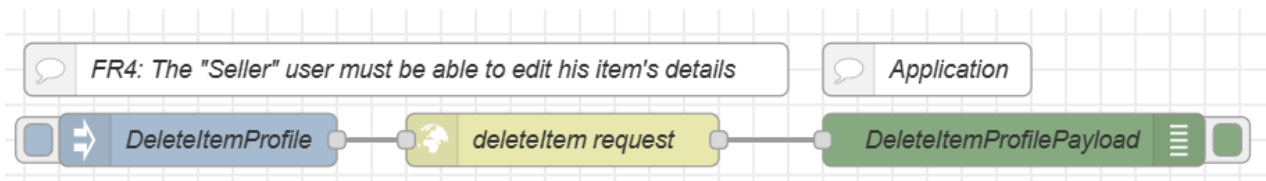
Ροή κατά την οποία ο χρήστης μπορεί να τροποποιήσει το προφίλ κάποιου αντικειμένου.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
EditItemProfile	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
putItem request	http-request	Κάνει κλήση προς την υπηρεσία Put /user/{user_id}/item/{item_id} η οποία δίνει στον χρήστη με ID: {user_id} τη δυνατότητα να τροποποιήσει το προφίλ του αντικείμενου με ID: {item_id}.
EditItemProfilePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή το αντικείμενο που τροποποίησε.

3.2.4. Ιστορία Χρήστη User Delete Item

Ροή κατά την οποία ο χρήστης μπορεί να διαγράψει κάποιο αντικείμενο.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
EditItemProfile	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
putItem request	http-request	Κάνει κλήση προς την υπηρεσία Put /user/{user_id}/item/{item_id} η οποία δίνει στον χρήστη με ID: {user_id} τη δυνατότητα να σβήσει αντικείμενο με ID: {item_id}.
EditItemProfilePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη.???????



3.2.5. Ιστορία Χρήστη User Declare Bid

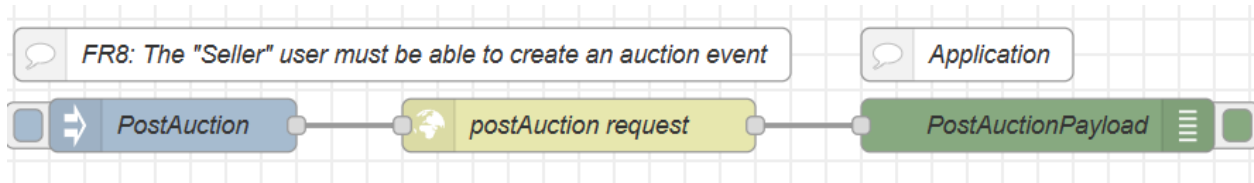
Ροή κατά την οποία ο χρήστης μπορεί να ορίσει ποσό για δημοπρασία.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
DeclareBid	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
postBid request	http-request	Κάνει κλήση προς την υπηρεσία Post /auction/{auction_id}/bid η οποία δίνει στον χρήστη τη δυνατότητα να δηλώσει ποσό κατά την διάρκεια της δημοπρασίας με ID: {auction_id}
postBidPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή την προσφορά του χρήστη.

3.2.6. Ιστορία Χρήστη User Create Event (Create Auction)

Ροή κατά την οποία ο χρήστης μπορεί να ορίσει ένα event δημοπρασίας.



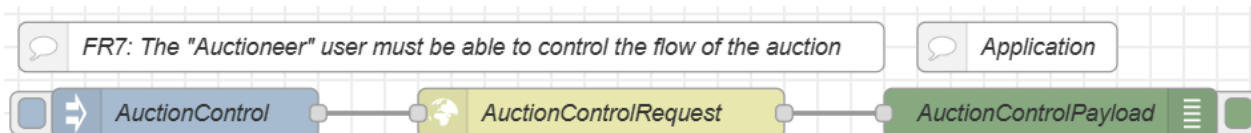
Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
PostAuction	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
postAuction request	http-request	Κάνει κλήση προς την υπηρεσία Post /user/{user_id}/auction η οποία δίνει στον χρήστη με ID: {user_id} τη δυνατότητα να ορίσει ένα event δημοπρασίας.
PostAuctionPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία της δημοπρασίας που δημιουργήσε.



3.2.7. Ιστορία Χρήστη User Control Auction

Ροή κατά την οποία ο χρήστης μπορεί να ελέγξει την πορεία της δημοπρασίας αλλάζοντας

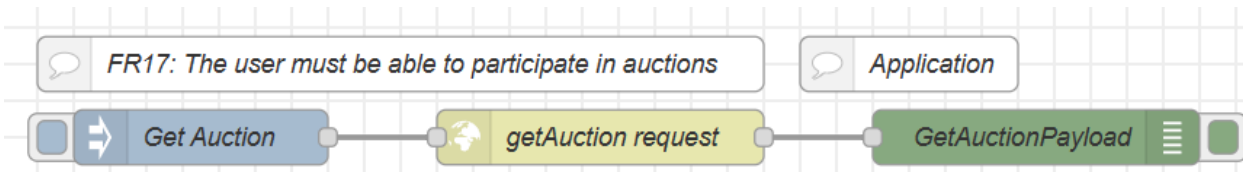
- 1) Το ρολόι μέτρησης
- 2) Το μέγιστο και ελάχιστο βήμα αύξησης προσφορών.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
AuctionControl	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
AuctionControl request	http-request	Κάνει κλήση προς την υπηρεσία Put /auction/{auction_id} η οποία δίνει στον χρήστη "δημοπράτη" τη δυνατότητα να ελέγχει την ροή της δημοπρασίας με ID: {auction_id}
AuctionControlPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα τροποποιημένα στοιχεία της δημοπρασίας που αφορούν το ρολόι και τα βήματα αύξησης.

3.2.8. Ιστορία Χρήστη User Join Auction

Ροή κατά την οποία ο χρήστης μπορεί να συμμετέχει σε μια δημοπρασία.

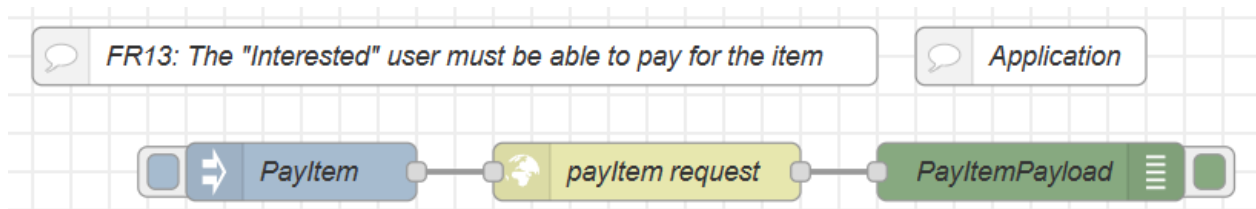


Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Get Auction	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
getAuction request	http-request	Κάνει κλήση προς την υπηρεσία Get auction/{auction_id} η οποία δίνει στον χρήστη τη δυνατότητα να συμμετέχει σε μια δημοπρασία με ID: {auction_id}.
GetAuctionPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία της δημοπρασίας που συμμετέχει.



3.2.9. Ιστορία Χρήστη User Pay Item

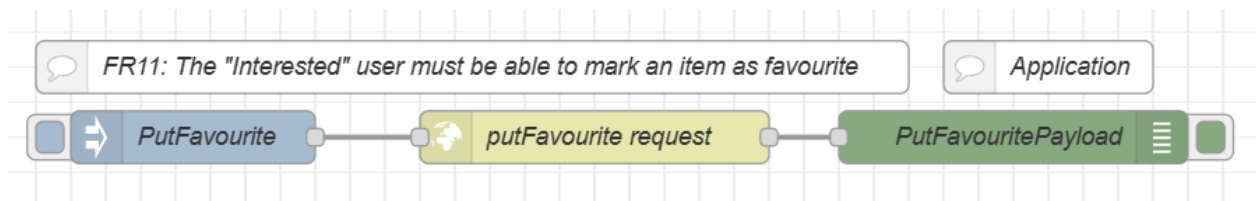
Ροή κατά την οποία ο χρήστης μπορεί να πληρώσει για το αντικείμενο που έχει κερδίσει.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
PayItem	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
payItem request	http-request	Κάνει κλήση προς την υπηρεσία Post /item/{item_id}/pay η οποία δίνει στον χρήστη τη δυνατότητα να πληρώσει για το αντικείμενο με ID: {item_id} που έχει κερδίσει.
PayItemPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τις πληροφορίες πληρωμής για το αντικείμενο που κέρδισε.

3.2.10. Ιστορία Χρήστη User Puts Item in Favourite

Ροή κατά την οποία ο χρήστης μπορεί να θέσει ένα αντικείμενο ως “Αγαπημένο”.

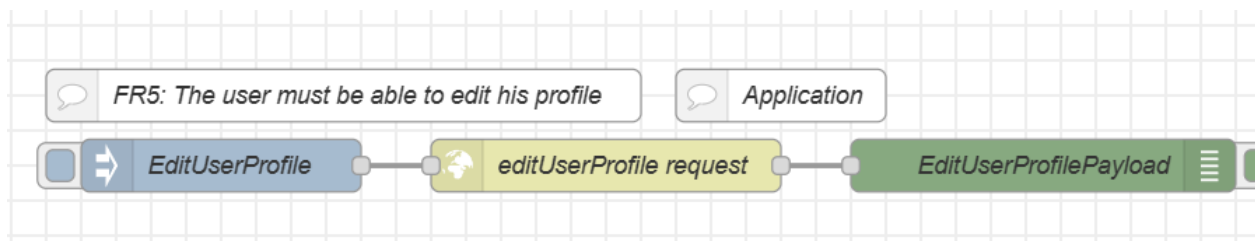


Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
PutFavourite	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
putFavourite request	http-request	Κάνει κλήση προς την υπηρεσία Put /item/{item_id} οποία δίνει στον χρήστη τη δυνατότητα να θέσει ένα αντικείμενο με ID: {item_id} ως “αγαπημένο”.
PutFavouritePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία του αντικειμένου με τροποποιημένη την ιδιότητα “Αγαπημένο”.



3.2.11. Ιστορία Χρήστη User Edit Profile

Ροή κατά την οποία ο χρήστης μπορεί να τροποποιήσει το προφίλ του.



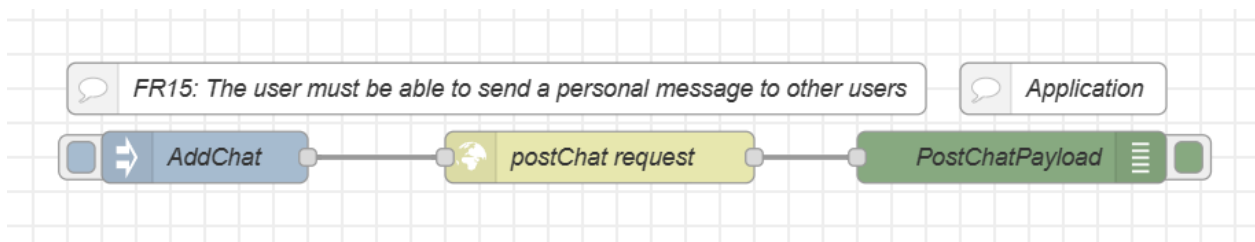
Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
EditUserProfile	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
editUserProfile request	http-request	Κάνει κλήση προς την υπηρεσία Put/user/{user_id} η οποία δίνει στον χρήστη με ID: {user_id} τη δυνατότητα να τροποποιήσει το προφίλ του.
EditUserProfilePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία του προφίλ του.

3.2.12. Ιστορία Χρήστη User Messaging

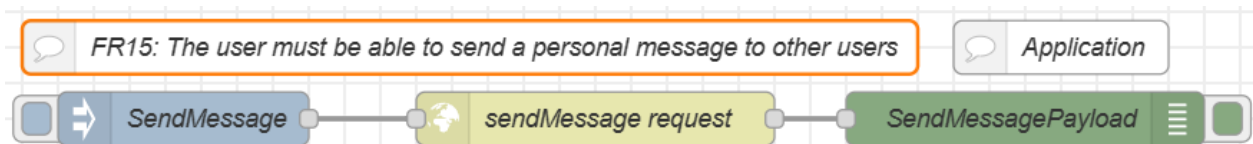
Ροή κατά την οποία ο χρήστης μπορεί να στείλει και να ανταλλάξει μηνύματα,

1) Δημιουργώντας ChatRoom

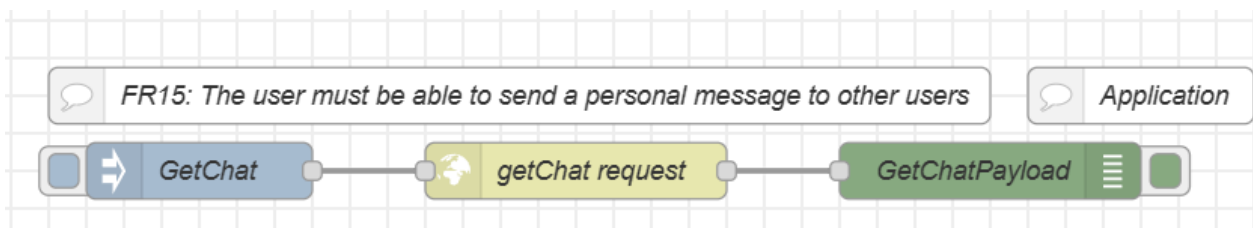
2) Δημιουργώντας ένα μήνυμα μέσα στο ChatRoom



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Add Chat	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
postChat request	http-request	Κάνει κλήση προς την υπηρεσία Post /user/{sender_id}/chat η οποία δίνει στον χρήστη με ID: {sender_id} τη δυνατότητα να δημιουργήσει ένα ChatRoom.
PostChatPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία του ChatRoom που δημιουργήσε και το περιεχόμενο των μηνυμάτων μέσα σε αυτό.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
SendMessage	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
sendMessage request	http-request	Κάνει κλήση προς την υπηρεσία Post /user/{sender_id}/chat/{chat_id}/message η οποία δίνει στον χρήστη τη δυνατότητα να δημιουργήσει ένα μήνυμα και να το στείλει στο ChatRoom με ID: {chat_id}
SendMessagePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
GetChat	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
getChat request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{sender_id}/chat/{chat_id} η οποία επιστρέφει στον χρήστη το ChatRoom με ID : {chat_id} με τα αντίστοιχα μηνύματα που υπάρχουν μέσα.
GetChatPayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή το ChatRoom με τα αντίστοιχα μηνύματα.





Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
ReceiveMessage	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
getMessage request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{sender_id}/chat/{chat_id}/message η οποία επιστρέφει στον χρήστη με ID: {sender_id}, στο ChatRoom με ID : {chat_id} τα αντίστοιχα στοιχεία του μηνύματος.
ReceiveMessagePayload	debug	Τυπώνει στην κονσόλα το response του συστήματος στην κλήση του χρήστη, δηλαδή τα στοιχεία των μηνυμάτων που βρίσκονται στο ChatRoom.



Μηχανική Λογισμικού Ι
Τομέας Ηλεκτρονικής και Υπολογιστών
Τμήμα ΗΜΜΥ
Α.Π.Θ

8^ο Εξάμηνο
Άνοιξη 2023

Παράρτημα Ι – Πίνακας Ιχνηλασιμότητας

<Εφόσον χρειάζεται, ορίστε έναν πίνακα Ιχνηλασιμότητας>.



Παράρτημα II – Ανοιχτά Θέματα

- Για την κάλυψη αναγκών του παρόντος παραδοτέου, χρειάστηκε να προστεθεί ένας πόρος Message, ο οποίος δεν υπήρχε στο πρώτο παραδοτέο. Έτσι ο χρήστης δημιουργεί μία συνομιλία (Chat) με άλλον χρήστη, η οποία φέρει ένα συγκεκριμένο περιεχόμενο, δηλαδή τα μηνύματα (Messages) τα οποία ανταλλάσσουν οι χρήστες μεταξύ τους.
- Απαιτείται μία βάση δεδομένων για τη δυναμική επεξεργασία και αποθήκευση των δεδομένων και την ορθή λειτουργία των ροών που δημιουργήθηκαν.
- Δεν υλοποιήθηκαν ορισμένες μέθοδοι στα endpoints (πχ. POST στον user, δηλαδή τη δημιουργία προφίλ που γίνεται κατά την εγγραφή) που αντιστοιχούν σε συγκεκριμένες λειτουργικότητες του συστήματος, σύμφωνα με τις εντολές που δόθηκαν στο πλαίσιο της εργασίας για την τήρηση της απλότητας. Επίσης λειτουργικότητες όπως GET user/{user_id}, όπου επιστρέφει το προφίλ του χρήστη σε κάποιον άλλον χρήστη δεν υλοποιήθηκαν καθώς ως απλούστερη λειτουργικότητα δεν αναφέρθηκαν στις ΛΑ. Συνεπώς συνίσταται έντονα η προσθήκη τους για την πραγματική και ολόπλευρη πλήρωση της εφαρμογής.
- Τα πρότυπα σχεδίασης που χρησιμοποιήθηκαν δεν καλύπτουν όλες τις μη λειτουργικές απαιτήσεις, αλλά έγινε η απαιτούμενη χρήση τουλάχιστον 3 διαφορετικών προτύπων.