



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022



PartyUp

Find the best party around

Προδιαγραφές Λογισμικού και Υλοποίηση συστήματος

Version 0.6

Βασίλειος Ευαγγελόπουλος vasileie@ece.auth.gr

Σταύρος Βασίλειος Μπουλιόπουλος smpoulio@ece.auth.gr

Νικόλαος Γιαννόπουλος ngiannop@ece.auth.gr

Εμμανουήλ Κωτούλας emmakoto@ece.auth.gr

Ημερομηνία 04/06/2022



Ιστορικό Αλλαγών

Όνομα	Ημερομηνία	Αλλαγή	Έκδοση
Α. Συμεωνίδης	17/05/2007	Δημιουργία εγγράφου. Προσαρμογή των προτύπων του K. E. Wiegers ¹² και του M. Smialek's.	0.1
Α. Συμεωνίδης	29/3/2014	Μικρή αναθεώρηση – τροποποίηση ενότητων	0.1.3
Χ. Ζολώτας	10/4/2020	Μεγάλη αναθεώρηση – αφαίρεση ενότητων	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
Βασίλειος Ευαγγελόπουλος	25	vasileie@ece.auth.gr
Σταύρος Βασίλειος Μπουλιόπουλος	25	smpoulis@ece.auth.gr
Νικόλαος Γιαννόπουλος	25	ngiannop@ece.auth.gr
Εμμανουήλ Κωτούλας	25	emmakoto@ece.auth.gr

¹² Copyright © 2002 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document. Original template is available at: <http://www.processimpact.com/>



Πίνακας Περιεχομένων

Πίνακας Περιεχομένων	3
1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν	6
1.1 Πρότυπο σχεδίασης Bridge	6
1.2 Πρότυπο σχεδίασης Proxy	6
1.3 Πρότυπο σχεδίασης Facade	7
2. Αρχιτεκτονική Συστήματος	9
2.1 Αναγνώριση Πόρων (Resources) Συστήματος	9
2.2 Τεκμηρίωση REST διεπαφής	10
2.2.1 Πόρος User	10
2.2.1.1 Μοντέλο δεδομένων User	10
2.2.1.2 Endpoint POST πόρου User	11
2.2.1.3 Endpoint PUT πόρου User	12
2.2.1.4 Endpoint GET πόρου User Φίλοι συγκεκριμένου User	13
2.2.1.5 Endpoint PUT πόρου User Φίλοι συγκεκριμένου User	14
2.2.2 Πόρος Party	15
2.2.2.1 Μοντέλο δεδομένων Party	15
2.2.2.2 Endpoint POST party/host	15
2.2.2.3 Endpoint GET party/activeParty	18
2.2.2.4 Endpoint GET party/activeParty/{activePartyID}	20
2.2.2.5 Endpoint PUT party/activeParty/{activePartyID}	22
2.2.2.6 Endpoint GET party/partyHistory	24
2.2.2.7 Endpoint PUT party/partyHistory/{partyHistoryID}/review	25
2.2.2.8 Endpoint GET party/activePartyId/{activePartyID}/participants	26
2.2.3 Πόρος Payment	28
2.2.3.1 Endpoint Post πόρου Payment	28
2.2.4 Πόρος SearchResults	29
2.2.4.1 Endpoint Get πόρου SearchResults	30
3. Υλοποίηση Συστήματος με Node-RED	31
3.1 Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed	31
3.1.1 Ροές πόρου User	31
3.2 Υλοποίηση Ιστοριών χρήστη	34
3.2.1 Ιστορία Χρήστη host a party	34



3.2.2	Ιστορία χρήστη review a party	34
3.2.3	Ιστορία χρήστη join και leaves a party	35
3.2.4	Ιστορία χρήστη view και filter active parties	35
3.2.5	Ιστορία χρήστη add a friend	36
3.2.6	Ιστορία χρήστη views participants	36
3.2.7	Ιστορία χρήστη search users	36
3.2.8	Ιστορία χρήστη donates money	37
3.2.9	Ιστορία χρήστη views recently joined parties	37
Παράρτημα Ι – Ανοιχτά Θέματα		38

Λίστα Σχημάτων

Εικόνα 1	Πρότυπο σχεδίασης Bridge προσαρμοσμένο στο αντικείμενο τύπου PaymentProxy.	6
Εικόνα 2	Πρότυπο σχεδίασης Proxy προσαρμοσμένο στα αντικείμενα τύπου PartyWidget.	7
Εικόνα 3	Πρότυπο σχεδίασης Facade προσαρμοσμένο στο υποσύστημα Principal Component Package.	8
Εικόνα 4	Πόρος User	10
Εικόνα 5	Endpoint POST πόρου User	11
Εικόνα 6	Endpoint PUT πόρου User	12
Εικόνα 7	Endpoint GET πόρου User Φίλοι συγκεκριμένου User	13
Εικόνα 8	Endpoint PUT πόρου User Φίλοι συγκεκριμένου User	14
Εικόνα 9	Πόρος Party	15
Εικόνα 10	Endpoint POST party/host	17
Εικόνα 11	Endpoint GET party/activeParty	19
Εικόνα 12	Endpoint GET party/activeParty/{activePartyID}	21
Εικόνα 13	Endpoint PUT party/activeParty/{activePartyID}	23
Εικόνα 14	Endpoint GET party/partyHistory	24
Εικόνα 15	Endpoint PUT party/partyHistory/{partyHistoryID}/review	25
Εικόνα 16	Endpoint GET party/activePartyId/{activePartyID}/participants	27
Εικόνα 17	Πόρος Payment	28
Εικόνα 18	Endpoint Post πόρου Payment	29
Εικόνα 19	Πόρος SearchResults	29
Εικόνα 20	Endpoint Get πόρου SearchResults	30
Εικόνα 21	Ροή NodeRed που αντιστοιχεί στο API call Post /user.	31



Εικόνα 22 Ροή NodeRed που αντιστοιχεί στο API call Put /user/login.	31
Εικόνα 23 Ροή NodeRed που αντιστοιχεί στο API call Post /user/:userID/party/host.	31
Εικόνα 24 Ροή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/activeParty.	32
Εικόνα 25 Ροή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/activeParty/{activePartyID}.	32
Εικόνα 26 Ροή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/activeParty/{activePartyID}.	32
Εικόνα 27 Ροή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/partyHistory/.	32
Εικόνα 28 Ροή NodeRed που αντιστοιχεί στο API call PUT /user/{userID}/party/partyHistory/{partyHistoryID}/review/.	32
Εικόνα 29 Ροή NodeRed που αντιστοιχεί στο API call POST /user/{userID}/party/activeParty/{activePartyID}/payment/.	33
Εικόνα 30 Ροή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/party/activePartyID/{activePartyID}/participants/.	33
Εικόνα 31 Ροή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/friends/.	33
Εικόνα 32 Ροή NodeRed που αντιστοιχεί στο API call PUT /user/:userID/friends/.	33
Εικόνα 33 Ροή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/friends/searchResults/.	33
Εικόνα 34 Ροή NodeRed που αντιστοιχεί στο API call GET /user.	34
Εικόνα 35 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “host a party”.	34
Εικόνα 36 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “review a party”.	34
Εικόνα 37 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “join a party” και “leave a party”.	35
Εικόνα 38 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “view active parties” και “filter active parties”.	35
Εικόνα 39 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “add a friend”.	36
Εικόνα 40 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “view participants”.	36
Εικόνα 41 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “search users”.	36
Εικόνα 42 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “donate money”.	37
Εικόνα 43 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “view recently joined parties”.	37

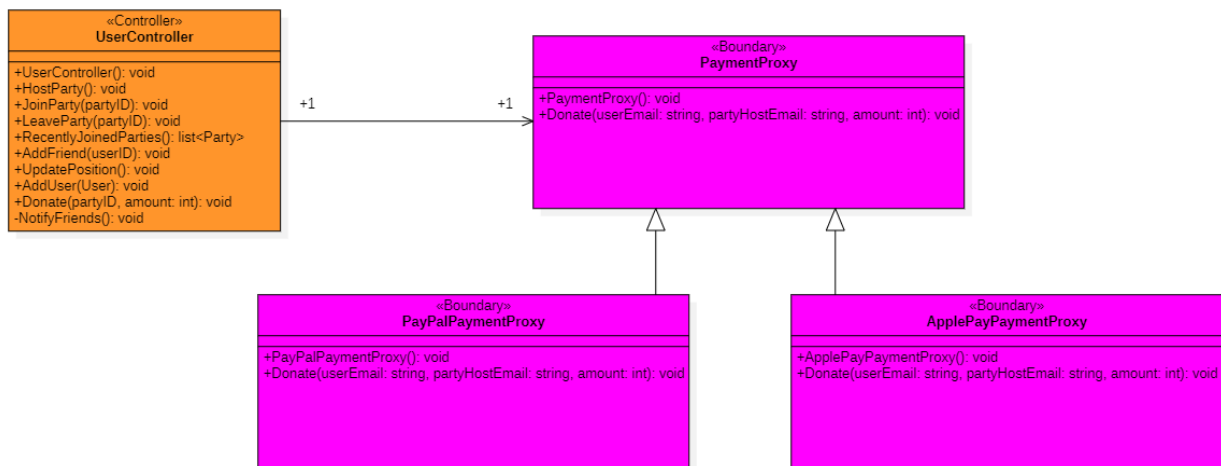


1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν

1.1 Πρότυπο σχεδίασης Bridge

- Πρότυπο Bridge
- Δομικό πρότυπο

Η υλοποίηση του συγκεκριμένου προτύπου εξυπηρετεί στην εύκολη επεκτασιμότητα του συστήματος από την στιγμή που η προσθήκη νέας λειτουργικότητας στο σύστημα δεν απαιτεί την τροποποίηση ήδη υπάρχοντος κώδικα. Χρησιμοποιείται για την μελλοντική επέκταση της λειτουργικότητας που σχετίζεται με την λειτουργική απαίτηση ΛΑ-7. Με τον τρόπο αυτό, σχετικά εύκολα, θα μπορούν να προστεθούν στην εφαρμογή και άλλοι τρόποι πληρωμής έτσι ώστε οι χρήστες να έχουν στην διάθεσή τους μία γκάμα επιλογών από την οποία ο κάθε χρήστης θα μπορεί να επιλέξει τον τρόπο που του ταιριάζει καλύτερα.



Εικόνα 1 Πρότυπο σχεδίασης Bridge προσαρμοσμένο στο αντικείμενο τύπου PaymentProxy.

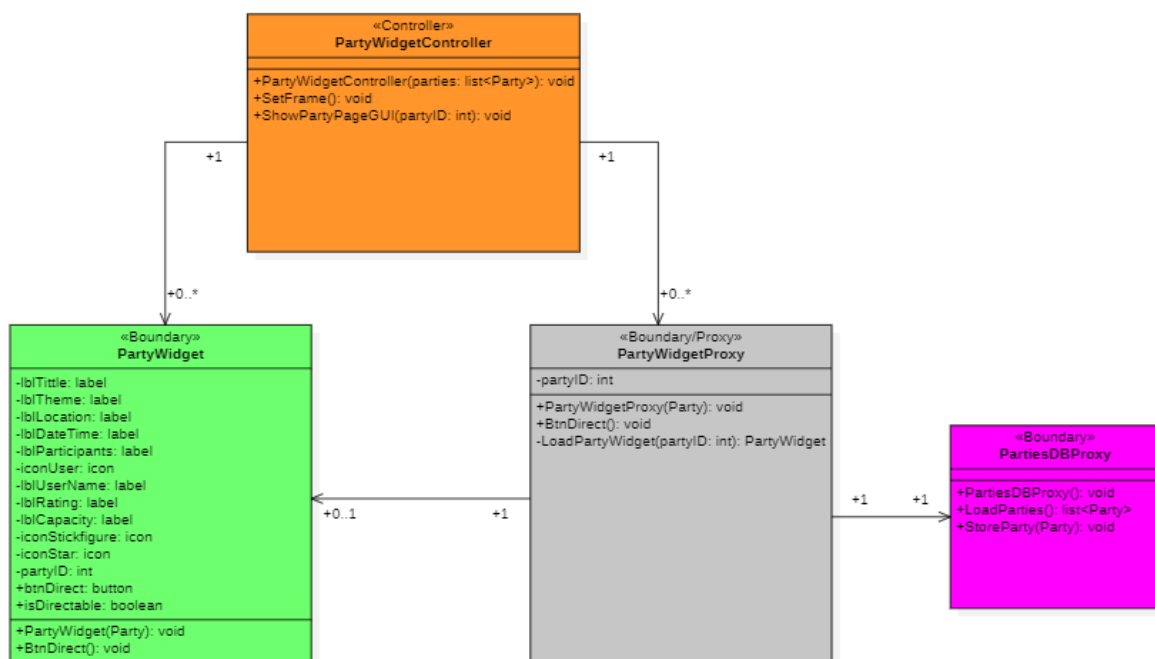
1.2 Πρότυπο σχεδίασης Proxy

- Πρότυπο Proxy
- Δομικό πρότυπο

Η υλοποίηση του συγκεκριμένου σχεδιαστικού προτύπου τύπου διαμεσολαβητή ικανοποιεί την ανάγκη της γρήγορης φόρτωσης και παρουσίασης της διεπαφής χρήστη. Αυτό συμβαίνει επειδή φορτώνουμε τα



“πραγματικά” αντικείμενα (τα οποία είναι ακριβά να δημιουργηθούν) μόνο όταν τα ζητήσει/χρειαστεί ο χρήστης. Το συγκεκριμένο πρότυπο σχεδίασης ικανοποιεί την μη λειτουργική απαίτηση γρήγορης απόκρισης της εφαρμογής (**ΜΛΑ-1**). Πιο συγκεκριμένα, εφαρμόζεται στα αντικείμενα *PartyWidget* έτσι ώστε να φορτώνονται κάθε φορά από την βάση δεδομένων μόνο τα πάρτι τα οποία μπορεί να δει εκείνη την στιγμή ο χρήστης (ο περιορισμός αυτός προκύπτει από τα όρια της οθόνης) και όχι όλα τα διαθέσιμα πάρτι που πληρούν τα κριτήρια. Μόνο όταν ο χρήστης χρειαστεί τα υπόλοιπα πάρτι αυτά θα φορτωθούν από την βάση δεδομένων (π.χ κάνοντας scroll).



Εικόνα 2 Πρότυπο σχεδίασης Proxy προσαρμοσμένο στα αντικείμενα τύπου *PartyWidget*.

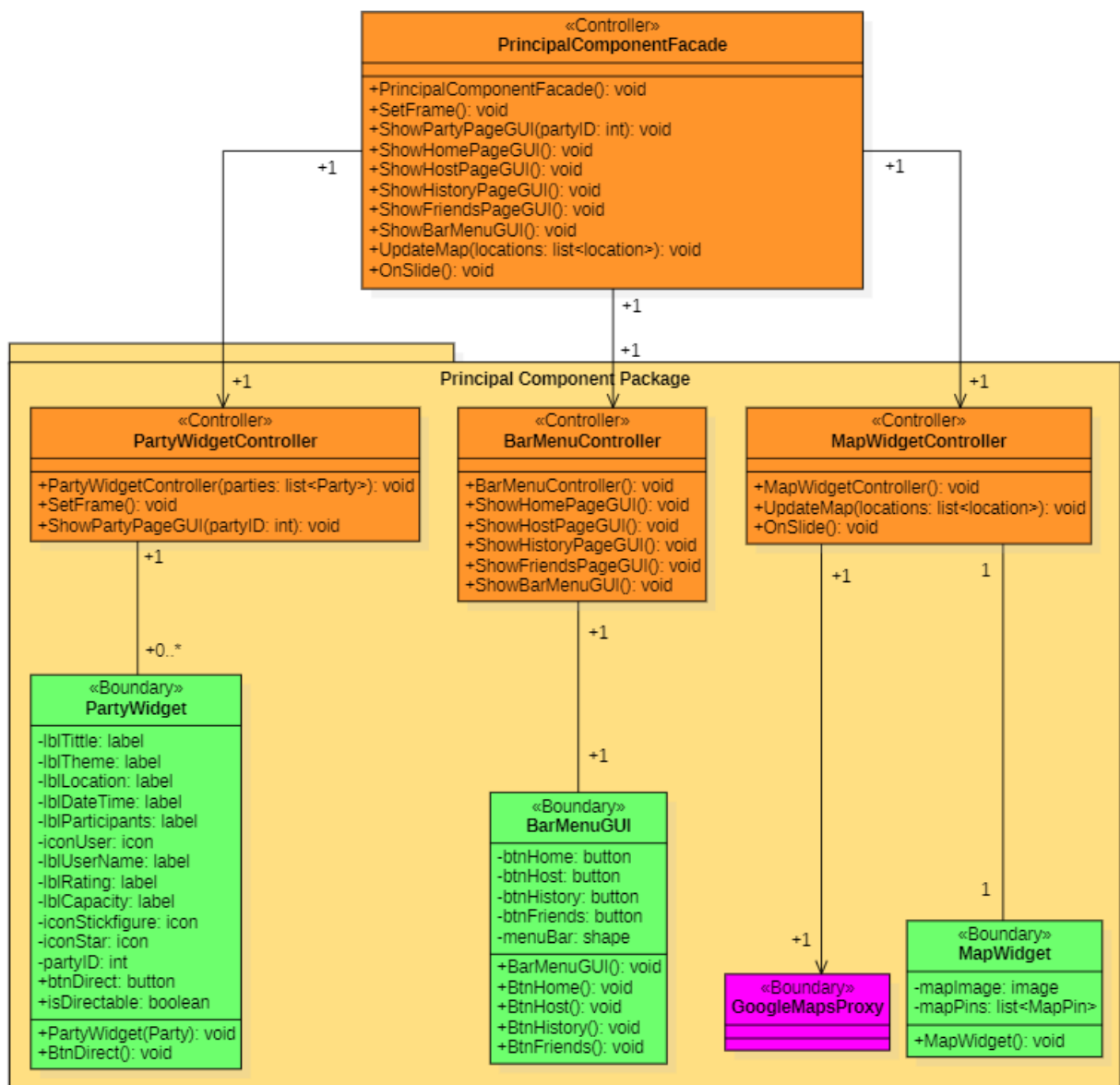
1.3 Πρότυπο σχεδίασης Facade

- Πρότυπο Facade
- Δομικό πρότυπο

Η υλοποίηση του συγκεκριμένου προτύπου εξυπηρετεί στην “ασθενή” σύνδεση μεταξύ των υποσυστημάτων του συστήματος. Με αυτόν τον τρόπο μειώνεται η πολυπλοκότητα του συστήματος που απορρέει λόγω των πολλαπλών συνδέσεων μεταξύ των υποσυστημάτων και επιπλέον αυξάνεται και η διατηρησιμότητα αφού με αυτόν τον τρόπο μειώνονται τα σημεία στα οποία πρέπει να πραγματοποιηθούν αλλαγές (οι αλλαγές περιορίζονται στο αντικείμενο Facade). Συγκεκριμένα, το πρότυπο αυτό θα εφαρμοστεί στο υποσύστημα *Principal Component Package* το οποίο συνδέεται σχεδόν με όλα τα άλλα υποσυστήματα.



Η εφαρμογή του προτύπου Facade στην συγκεκριμένη περίπτωση θα μείωνε την πολυπλοκότητα των συνδέσεων μεταξύ των υποσυστημάτων, άρα και την πολυπλοκότητα του συστήματος, με αποτέλεσμα να έχουμε μία πιο καθαρή εικόνα για το ίδιο το σύστημα και επομένως και για την εφαρμογή που θα υλοποιήσουμε.



Εικόνα 3 Πρότυπο σχεδίασης Facade προσαρμοσμένο στο υποσύστημα Principal Component Package.



2. Αρχιτεκτονική Συστήματος

Χρησιμοποιήθηκε το εργαλείο Swaggerhub.

- [partyUp API.yaml](#)
- [partyUP API.json](#)
- [Άμεση προβολή μέσω Swaggerhub](#)

2.1 Αναγνώριση Πόρων (Resources) Συστήματος

Κλάση BEC	Πόρος REST	Endpoints (HTTP Verbs)
User	/user	POST
	/user/login	PUT
	/user/{userID}/friends	GET,PUT
Party	/user/{userID}/party/host/	POST
	/user/{userID}/party/activeParty	GET,PUT
	/user/{userID}/party/activeParty/{activePartyID}	GET, PUT
	/user/{userID}/party/partyHistory/	GET
	/user/{userID}/party/partyHistory/{partyHistoryID}/review/	PUT
	/user/{userID}/party/activePartyID/{activePartyID}/participants/	GET
Payment	/user/{userID}/party/activeParty/{activePartyID}/payment/	POST
SearchResults	/user/{userID}/friends/searchResults/	GET



2.2 Τεκμηρίωση REST διεπαφής

2.2.1 Πόρος User

2.2.1.1 Μοντέλο δεδομένων User

```
User {  
  username      string  
  password      string  
  age           integer($int32)  
  gender        string  
  email         string  
  reviewScore   number($double)  
  userID        integer($int32)  
  joinedParties  
    {  
      list of partyID  
      integer($int32)  
    }  
  friends  
    {  
      list of userID  
      integer($int32)  
    }  
}
```

Εικόνα 4 Πόρος User



2.2.1.2 Endpoint POST πόρου User

The screenshot displays the Swagger UI for the POST /user endpoint. The interface is divided into several sections:

- POST /user postUser**: The endpoint name and method.
- Create a new user user**: A description of the endpoint.
- Parameters**: A section for parameters, currently empty.
- Content-Type**: A dropdown menu set to `application/json`.
- body**: A required field of type `object`. The `Edit Value` tab shows a JSON object with the following structure:

```
{  "username": "makis",  "password": "1234567890qwerty",  "age": 69,  "gender": "nonbinary",  "email": "makis@auth.gr",  "reviewScore": 3.6,  "userID": 15231,  "joinedParties": [    4512123,    213254,    421312,    214123  ]}
```
- Responses**: A section for responses, currently empty.
- Snippets**: A section for code snippets, showing a `curl` command for the POST request.

The `curl` command is as follows:

```
curl -X 'POST' \  'https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user' \  -H 'accept: application/json' \  -H 'Content-Type: application/json' \  -d '{  "username": "makis",  "password": "1234567890qwerty",  "age": 69,  "gender": "nonbinary",  "email": "makis@auth.gr",  "reviewScore": 3.6,  "userID": 15231,  "joinedParties": [    4512123,    213254,    421312,    214123  ]}'
```

The response is shown as a 200 status code with the following headers:

```
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: Tue,31 May 2022 18:44:45 GMT  etag: W/"14-40h36nhnrn3soolwXlJ6FqT6xz/E"  expires: -1  status: 200 OK  x-firefox-spdy: h2
```

The request duration is 236 ms.

The response body is a JSON object with the following structure:

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

Εικόνα 5 Endpoint POST πόρου User



2.2.1.3 Endpoint PUT πόρου User

PUT /user/login loginUser

Logs user into the system

Parameters

Name	Description
email required	The user email for login
string	
(query)	
<input type="text" value="makis@ece.auth.gr"/>	
password required	The password for login in clear text
string	
(query)	
<input type="text" value="1234567890qwerty"/>	

Execute

Clear

Responses

Response content type **application/json**

Snippets

cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'PUT' \
  https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/login?email=makis@ece.auth.gr&password=1234567890qwerty \
  -H 'accept: application/json'
```

Request URL

https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/login?email=makis@ece.auth.gr&password=1234567890qwerty

Server response

Code	Details
200	<div>Response body</div> <div><pre>{ "token": "string" }</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: 44 content-type: application/json; charset=utf-8 date: Tue, 31 May 2022 18:56:33 GMT etag: W/"2c-kMQDwXrHYvwwL6yfzci0FED++lc" expires: -1 status: 200 OK x-firefox-spdy: h2</pre></div>

Request duration

553 ms

| Responses | | Code | Description | |------|----------------| | 200 | login response | Example Value | Model ``` { "token": "string" } ``` |
| default | unexpected error |

Example Value | Model

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

Εικόνα 6 Endpoint PUT πόρου User



2.2.1.4 Endpoint GET πόρου User Φίλοι συγκεκριμένου User

GET /user/{userID}/friends/ viewAllFriends

View all friends of user

Parameters

userID * required
integer (\$int32)
(path)
124125

Execute Clear

Responses

Response content type: application/json

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/124125/friends/' \
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/124125/friends/
```

Server response

Code Details

200 Response body

```
{
  "username": "string",
  "password": "string",
  "age": 0,
  "gender": "string",
  "email": "string",
  "reviewScore": 1.100000023841858,
  "userID": 0,
  "joinedParties": [
    0
  ],
  "friends": [
    0
  ]
}
```

Response headers

```
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: 147
content-type: application/json;charset=utf-8
date: Thu, 02 Jun 2022 08:08:36 GMT
etag: W/"93-0qHzRDwVAY/4gZKhyXRzZnAzo"
expires: -1
status: 200 OK
```

Request duration

234 ms

Responses

Code Description

200 view friends

Example Value Model

default unexpected error

Example Value Model

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

Εικόνα 7 Endpoint GET πόρου User Φίλοι συγκεκριμένου User

2.2.1.5 Endpoint PUT πόρου User Φίλοι συγκεκριμένου User

PUT

/user/{userID}/friends/ addFriend

FR - 8 The user must be able to add a friend.

Parameters

Name

Description

userID * required

integer (Sint32)

(path)

The id of the user who will add a friend

21341

username * required

string

(query)

The username given from user to add him

konstantis

Execute

Responses

Response content typeapplication/json

Code

Description

200

successful editing of friend relationship

Example Value | Model

{
 "username": "string",
 "password": "string",
 "age": 0,
 "gender": "string",
 "email": "string",
 "reviewScore": 0,
 "userID": 0,
 "joinedParties": [
 0
],
 "friends": [
 0
]
}

default

Default response

Example Value | Model

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

Εικόνα 8 Endpoint PUT πόρου User Φίλοι συγκεκριμένου User



2.2.2 Πόρος Party

2.2.2.1 Μοντέλο δεδομένων Party

Party	
name	string
address	string
theme	string
capacity	integer(\$int32)
partyID	integer(\$int32)
date	string
rating	number(\$double)
isActive	boolean
ammountParticipants	integer(\$int32)
participants	▼ [User > {...}]
host	User > {...}
}	

Εικόνα 9 Πόρος Party

2.2.2.2 Endpoint POST party/host

POST /user/{userID}/party/host addParty

FR1 - The user must be able to host a party.

Parameters

Name	Description
userID * required	The ID of the user of the user that hosts a new party
string (path)	<input type="text" value="123"/>
Content-Type	<input type="text" value="application/json"/>
string (header)	
body * required	Party model
object (body)	<div>Edit Value Model</div> <pre>{ "name": "DRINK IT ALL", "address": "Martiou 25", "theme": "string", "capacity": 50, "partyID": 3333, "date": "03/4/2022", "rating": 0, "isActive": true, "ammountParticipants": 0, "participants": [{ </pre>



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

body * required

Party model

object

Edit Value | Model

(body)

```
{
  "name": "DRINK IT ALL",
  "address": "Martiou 25",
  "theme": "string",
  "capacity": 50,
  "partyID": 3333,
  "date": "03/4/2022",
  "rating": 0,
  "isActive": true,
  "amountParticipants": 0,
  "participants": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
        0
      ]
    }
  ],
  "host": {
    "username": "string",
    "password": "string",
    "age": 0,
    "gender": "string",
    "email": "string",
    "reviewScore": 0,
    "userID": 0,
    "joinedParties": [
      0
    ],
    "friends": [
      0
    ]
  }
}
```

Cancel

```
    },
    "friends": [
      0
    ]
  }
}
```

Cancel

Parameter content type

application/json

Execute

Clear

Responses

Response content type application/json

Snippets

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

```
curl -X 'POST' \
'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/123/party/host' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "name": "string",
  "address": "string",
  "theme": "string",
  "capacity": 0,
  "partyID": 0,
  "date": "string",
  "rating": 0,
  "isActive": true,
  "amountParticipants": 0,
  "participants": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
        0
      ]
    }
  ],
  "host": {
    "username": "string",
    "password": "string",
    "age": 0,
    "gender": "string",
    "email": "string",
    "reviewScore": 0,
    "userID": 0,
    "joinedParties": [
      0
    ],
    "friends": [
      0
    ]
  }
}
```




Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

```
    "theme": "string",
    "capacity": 0,
    "partyID": 0,
    "date": "string",
    "rating": 0,
    "isActive": true,
    "amountParticipants": 0,
    "participants": [
      {
        "username": "string",
        "password": "string",
        "age": 0,
        "gender": "string",
        "email": "string",
        "reviewScore": 0,
        "userID": 0,
        "joinedParties": [
          0
        ],
        "friends": [
          0
        ]
      }
    ],
    "host": {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
        0
      ]
    }
  }
}
```

Request URL

https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/123/party/host

Server response

Code	Details
200	<p>Response body</p> <pre>{ "name": "string", "address": "string", "theme": "string", "capacity": 0, "partyID": 0, "date": "string", "rating": 1.100000023841858, "isActive": true, "amountParticipants": 0, "participants": [{ "username": "string", "password": "string", "age": 0, "gender": "string", "email": "string", "reviewScore": 1.100000023841858, "userID": 0, "joinedParties": [0], "friends": [</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 content-length: 233 content-type: application/json;charset=utf-8 date: Tue, 31 May 2022 18:36:07 GMT etag: W/"e9-Qs6LYaZTee5pAP/aA0Zy/d3S/ao" expires: -1 status: 200 OK</pre> <p>Response duration</p>

Εικόνα 10 Endpoint POST party/host



2.2.2.3 Endpoint GET party/activeParty

GET `/user/{userID}/party/activeParty` viewParties

FR2 The user must be able to view the active parties.

Parameters

Name	Description
userID * required integer(\$int32) (path)	The id of the user viewing the active parties
capacity integer(\$int32) (query)	filter by capacity of party
theme string (query)	theme of the party
distance integer(\$int32) (query)	distance from the party
rating integer(\$int32) (query)	rating of host to filter by

Execute

Clear

Responses

Response content type **application/json**

Snippets

cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/1/party/activeParty?capacity=20&theme=mixed&distance=2&rating=4' \
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/1/party/activeParty?capacity=20&theme=mixed&distance=2&rating=4
```

Server response



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

Request URL

https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/1/party/activeParty?capacity=20&theme=mixed&distance=2&rating=4

Server response

Code

Details

200

Response body

```
[
  {
    "name": "string",
    "address": "string",
    "theme": "string",
    "capacity": 0,
    "partyID": 0,
    "date": "string",
    "rating": 1.100000023841858,
    "isActive": true,
    "amountParticipants": 0,
    "participants": [
      {
        "username": "string",
        "password": "string",
        "age": 0,
        "gender": "string",
        "email": "string",
        "reviewScore": 1.100000023841858,
        "userID": 0,
        "joinedParties": [
          0
        ]
      }
    ]
  }
]
```

Download

Response headers

```

],
"friends": [
  0
]
},
"host": {
  "username": "string",
  "password": "string",
  "age": 0,
  "gender": "string",
  "email": "string",
  "reviewScore": 1.100000023841858,
  "userID": 0,
  "joinedParties": [
    0
  ],
  "friends": [
    0
  ]
}
}
]
```

Download

Response headers

```
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: 234
content-type: application/json; charset=utf-8
date: Tue, 31 May 2022 18:49:37 GMT
etag: W/"ea-IViY96b7kuHMEGc85Iwyg4Rr1E"
expires: -1
status: 200 OK
```

Request duration

607 ms

Responses

Request duration

607 ms

Responses

Code

Description

200

party response

Example Value | Model

Εικόνα 11 Endpoint GET party/activeParty



2.2.2.4 Endpoint GET party/activeParty/{activePartyID}

GET

/user/{userID}/party/activeParty/{activePartyID} get party by id

^ 🔒 ↶

FR2 The user must be able to view the active parties.

Parameters

Cancel

Name	Description
userID * required	The id of the user viewing the active parties
integer(\$int32)	
(path)	9999
activePartyID * required	The id of the active party the user is viewing
integer(\$int32)	
(path)	12323

ExecuteClear

Responses

Response content type application/json

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/9999/party/activeParty/12323' \
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/9999/party/activeParty/12323
```

Server response

Code	Details
200	<div>Response body</div> <pre>{ "name": "string", "address": "string", "theme": "string", "capacity": 0, "partyID": 0, "date": "string", "rating": 1.100000023841858, "isActive": true, "amountParticipants": 0, "participants": [{ "username": "string", "password": "string", "age": 0, "gender": "string", "email": "string", "reviewScore": 1.100000023841858, "userID": 0, </pre>



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

256 ms

Responses

Code	Description
200	party response

Example Value | Model

```
{
  "name": "string",
  "address": "string",
  "theme": "string",
  "capacity": 0,
  "partyID": 0,
  "date": "string",
  "rating": 0,
  "isActive": true,
  "amountParticipants": 0,
  "participants": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [

```

default unexpected error

Εικόνα 12 Endpoint GET party/activeParty/{activePartyID}



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

2.2.2.5 Endpoint PUT party/activeParty/{activePartyID}

PUT /user/{userID}/party/activeParty/{activePartyID} joinParty

FR-3 - The user must be able to join a party
FR-12 - The user must be able to leave a party

Parameters Cancel

Name	Description
userID * required integer(\$int32) (path)	The id of the user viewing the active parties
activePartyID * required integer(\$int32) (path)	The id of the active party the user is viewing
Content-Type string (header)	<div>application/json</div>
body * required object (body)	Party model <div>Edit Value Model</div>
body * required object (body)	Party model <div>Edit Value Model</div> <div><pre>{ "name": "string", "address": "string", "theme": "string", "capacity": 0, "partyID": 0, "date": "string", "rating": 0, "isActive": true, "amountParticipants": 0, "participants": [{ "username": "string", "password": "string", "age": 0, "gender": "string", "email": "string", "reviewScore": 0, "userID": 0, "joinedParty": 0 }]}</pre></div>

Cancel

Parameter content type

application/json

Execute

Clear

Responses Response content type application/json



Responses Response content type: application/json

Snippets

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

```
curl -X 'PUT' \
  'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/993/party/activeParty/6969' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "string",
    "address": "string",
    "theme": "string",
    "capacity": 0,
    "partyID": 0,
    "date": "string",
    "rating": 0,
    "isActive": true,
    "ammountParticipants": 0,
    "participants": [
      {
        "username": "string",
        "password": "string",
        "age": 0,
        "gender": "string",
        "email": "string",
        "reviewScore": 0,
        "userID": 0,
        "joinedParties": [
          0
        ],
        "friends": [
          0
        ]
      }
    ]
  }'
```

Request URL

https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/993/party/activeParty/6969

Server response

Code	Details
200	<p>Response body</p> <pre>{ "name": "string", "address": "string", "theme": "string", "capacity": 0, "partyID": 0, "date": "string", "rating": 1.100000023841858, "isActive": true, "ammountParticipants": 0, "participants": [{ "username": "string", "password": "string", "age": 0, "gender": "string", "email": "string", "reviewScore": 0, "userID": 0, "joinedParties": [0], "friends": [0] }] }</pre>

Εικόνα 13 Endpoint PUT party/activeParty/{activePartyID}



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

2.2.2.6 Endpoint GET party/partyHistory

GET /user/{userID}/party/partyHistory partyHistory

FR - 14 The user must be able to see the parties in which he/she has participated

Parameters

Name	Description
userID * required	The id of the user viewing their history
integer (\$int32)	
(path)	1111

Execute Clear

Responses

Response content type: application/json

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/1111/party/partyHistory' \
-H 'accept: application/json'
```

Request URL

https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/1111/party/partyHistory

Server response

Code Details

200

Response body

```
[
  {
    "name": "string",
    "address": "string",
    "theme": "string",
    "capacity": 0,
    "partyID": 0,
    "date": "string",
    "rating": 1.100000023841858,
    "isActive": true,
    "amountParticipants": 0,
    "participants": [
      {
        "username": "string",
        "password": "string",
        "age": 0,
        "gender": "string",
        "email": "string",
        "reviewScore": 1.100000023841858,
        "userID": 0,
        "joinedParties": [
          0
        ]
      }
    ]
  }
],
```

Response headers

```
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: 234
content-type: application/json; charset=utf-8
date: Tue, 31 May 2022 19:04:11 GMT
etag: W/"ea-1VIY96b7kuHWEgc851wyg4Kr1E"
expires: -1
status: 200 OK
```

Εικόνα 14 Endpoint GET party/partyHistory



2.2.2.7 Endpoint PUT party/partyHistory/{partyHistoryID}/review

PUT

/user/{userID}/party/partyHistory/{partyHistoryID}/review reviewParty

⌵ 🔒 ↶

FR - 4 The user must be able to review past parties

Parameters

Try it out

Name	Description
userID * required integer(\$int32) (path)	The id of the user reviewing a party
partyHistoryID * required integer(\$int32) (path)	The id of the party the user is reviewing
stars * required integer(\$int32) (query)	The review a user gives to a party

Responses

Response content type application/json

Code	Description
200	successfull update review score of party's host <div>Example Value Model</div> <pre>{ "username": "string", "password": "string", "age": 0, "gender": "string", "email": "string", "reviewScore": 0, "userID": 0, "joinedParties": [0], "friends": [0]}</pre>
default	unexpected error <div>Example Value Model</div> <pre>{ "code": 0, "message": "string"}</pre>

Εικόνα 15 Endpoint PUT party/partyHistory/{partyHistoryID}/review



2.2.2.8 Endpoint GET party/activePartyId/{activePartyID}/participants

GET /user/{userID}/party/activePartyID/{activePartyID}/participants/ viewParticipants

FR-11 The user must be able to see the users participating in the party.

Parameters Cancel

Name	Description
userID * required integer(\$int32) (path)	The id of the user
activePartyID * required integer(\$int32) (path)	The id of the party the user is showing participants

Execute Clear

Responses Response content type application/json

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/909/party/activePartyID/222/participants/' \
-H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/studen57/party-up_api/4.7.8/user/909/party/activePartyID/222/participants/
```

Server response

Code **Details**

200

Response body

```
{
  "name": "string",
  "address": "string",
  "theme": "string",
  "capacity": 0,
  "partyID": 0,
  "date": "string",
  "rating": 1.100000023841858,
  "isActive": true,
  "ammountParticipants": 0,
  "participants": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 1.100000023841858,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
```

Download



Τεχνολογία Λογισμικού

Τομέας Ηλεκτρονικής και Υπολογιστών

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Α.Π.Θ.

8^ο Εξάμηνο

Άνοιξη 2022

Code	Description
200	successful retrieval participant list from party

Example Value | Model

```
{
  "name": "string",
  "address": "string",
  "theme": "string",
  "capacity": 0,
  "partyID": 0,
  "date": "string",
  "rating": 0,
  "isActive": true,
  "amountParticipants": 0,
  "participants": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [

```

Response headers

```
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: 233
content-type: application/json;charset=utf-8
date: Tue,31 May 2022 19:08:12 GMT
etag: W/"e9-Qs6LYaZTee5pAP/aA0Zy/d3S/ao"
expires: -1
status: 200 OK
```

Request duration

580 ms

Εικόνα 16 Endpoint GET party/activePartyId/{activePartyID}/participants



2.2.3 Πόρος Payment

Payment {	
userID	integer(\$int32) payer
activePartyID	integer(\$int32) target party's owner
service	string Paypal method
amount	integer(\$int32) price amount to be delivered
}	

Εικόνα 17 Πόρος Payment

2.2.3.1 Endpoint Post πόρου Payment

POST

/user/{userID}/party/activeParty/{activePartyID}/payment/

amount

FR-7 The user must be able to contribute financially to the party.

Parameters

Name	Description
userID * required integer(\$int32) (path)	The id of the user
activePartyID * required integer(\$int32) (path)	The id of the party the user viewing
amount * required number(\$double) (query)	The amount of user gives to hoster of party

Execute

Clear

Responses

Response content type application/json

Snippets

cURL (bash)

cURL (PowerShell)

cURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/21412/party/activeParty/214551' \
  -H 'accept: application/json' \
  -d ''
```

Request URL

```
https://virtserver.swaggerhub.com/Emkot87/party-up_api/4.7.8/user/21412/party/activeParty/214551/payment/?amount=2143
```

Server response

Code	Details
200	<div>Response body</div> <div><pre>{ "userID": 0, "activePartyID": 0, "service": "string", "amount": 0 }</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: 78 content-type: application/json;charset=utf-8 date: Wed,01 Jun 2022 17:37:20 GMT etag: W/"4e-3bNSQMaOXvn8l13Ba3GOKXNNPw" expires: -1 status: 200 OK</pre></div>



Request duration	
270 ms	
Responses	
Code	Description
200	payment response
Example Value Model	
<pre>{ "userID": 0, "activePartyID": 0, "service": "string", "amount": 0 }</pre>	
default	unexpected error
Example Value Model	
<pre>{ "code": 0, "message": "string" }</pre>	

Εικόνα 18 Endpoint Post πόρου Payment

2.2.4 Πόρος SearchResults

```
SearchResults {
  inputName string
  results {
    User {
      username string
      password string
      age integer($int32)
      gender string
      email string
      reviewScore number($double)
      userID integer($int32)
      joinedParties > [...]
      friends > [...]
    }
  }
}
```

Εικόνα 19 Πόρος SearchResults



2.2.4.1 Endpoint Get πόρου SearchResults

GET /user/{userID}/friends/searchResults/ viewResultsSearch

FR - 8 The user must be able to search for other users.

Parameters

Name	Description
userID * required	The id of the user
integer(int32)	
(path)	
	15231
inputName * required	The input username given from user
string	
(query)	
	Makis

Execute Clear

Responses

Response content type: application/json

Snippets

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

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/Enkot87/party-up_api/4.7.8/user/15231/friends/searchResults/?inputName=Makis'
```

Server response

Code Details

200

Response body

```
{
  "inputName": "string",
  "results": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 1.100000023841858,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
        0
      ]
    }
  ]
}
```

Response headers

```
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
```

expires: -1
status: 200 OK
x-firefox-spdly: h2

Request duration

592 ms

Responses

Code	Description
200	view friends

Example Value | Model

```
{
  "inputName": "string",
  "results": [
    {
      "username": "string",
      "password": "string",
      "age": 0,
      "gender": "string",
      "email": "string",
      "reviewScore": 0,
      "userID": 0,
      "joinedParties": [
        0
      ],
      "friends": [
        0
      ]
    }
  ]
}
```

default

unexpected error

Example Value | Model

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

Εικόνα 20 Endpoint Get πόρου SearchResults



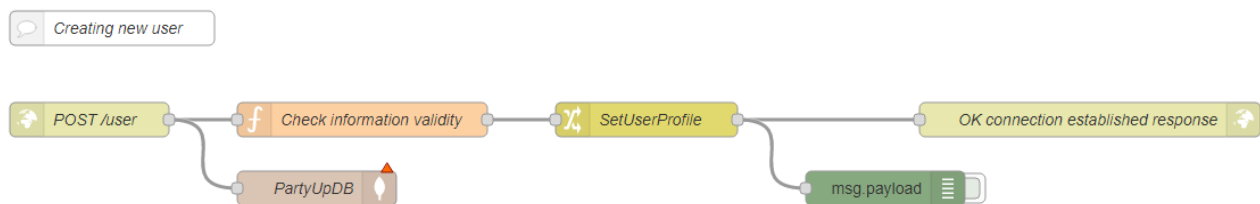
3. Υλοποίηση Συστήματος με Node-RED

3.1 Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed

- [partyUp_Flow.json](#)

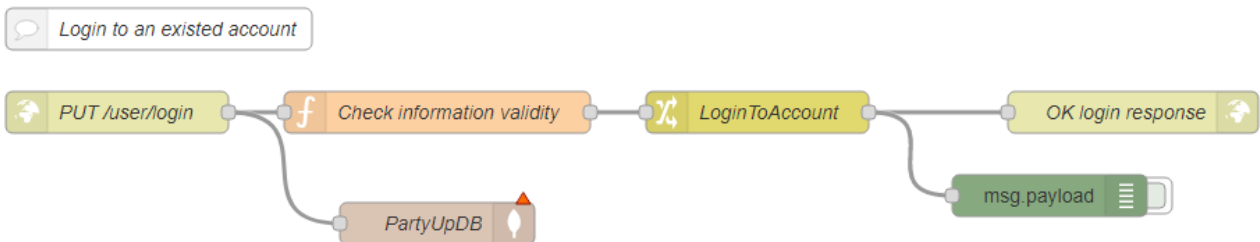
3.1.1 Ροές πόρου User

Ροή endpoint POST /user



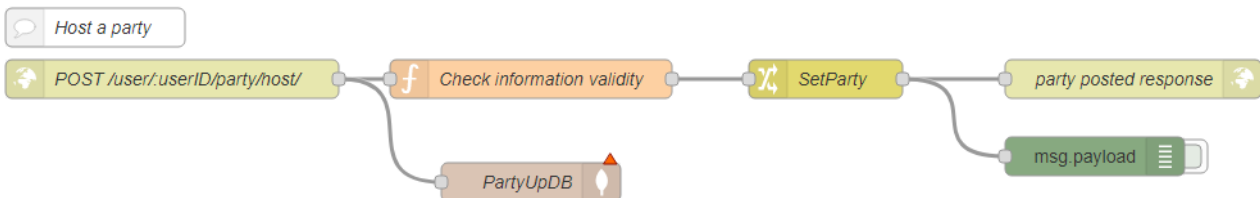
Εικόνα 21 Ροή NodeRed που αντιστοιχεί στο API call Post /user.

Ροή endpoint PUT /user/login



Εικόνα 22 Ροή NodeRed που αντιστοιχεί στο API call Put /user/login.

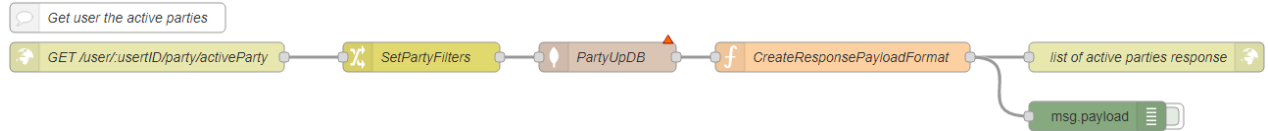
Ροή endpoint POST /user/{userID}/party/host



Εικόνα 23 Ροή NodeRed που αντιστοιχεί στο API call Post /user/:userID/party/host.

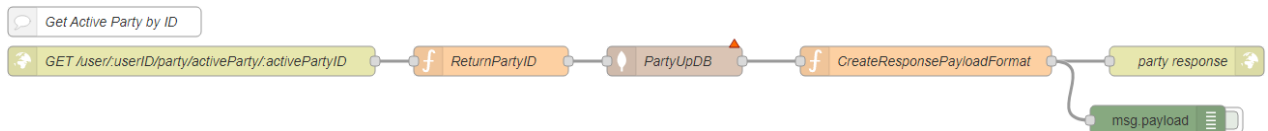


Ποή endpoint GET /user/{userID}/party/activeParty



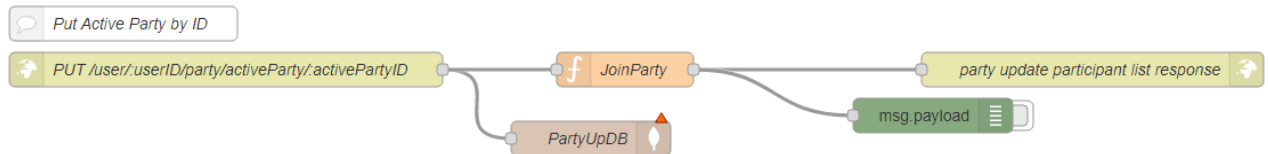
Εικόνα 24 Ποή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/activeParty.

Ποή endpoint GET /user/{userID}/party/activeParty/{activePartyID}



Εικόνα 25 Ποή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/activeParty/{activePartyID}.

Ποή endpoint PUT /user/{userID}/party/activeParty/{activePartyID}



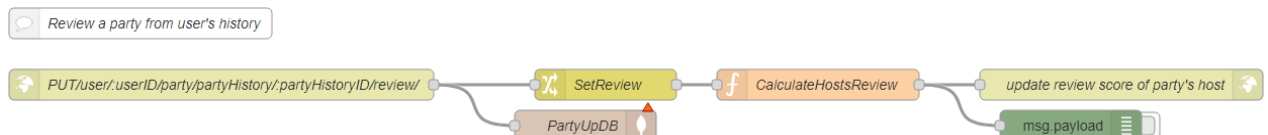
Εικόνα 26 Ποή NodeRed που αντιστοιχεί στο API call Put /user/{userID}/party/activeParty/{activePartyID}.

Ποή endpoint GET /user/{userID}/party/partyHistory/



Εικόνα 27 Ποή NodeRed που αντιστοιχεί στο API call Get /user/{userID}/party/partyHistory/.

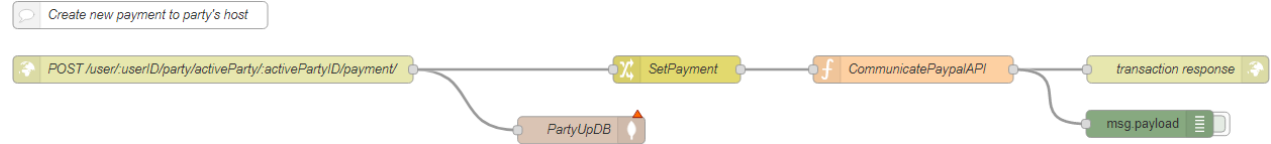
Ποή endpoint PUT /user/{userID}/party/partyHistory/{partyHistoryID}/review/



Εικόνα 28 Ποή NodeRed που αντιστοιχεί στο API call PUT /user/{userID}/party/partyHistory/{partyHistoryID}/review/.



Ποή endpoint POST /user/{userID}/party/activeParty/{activePartyID}/payment/



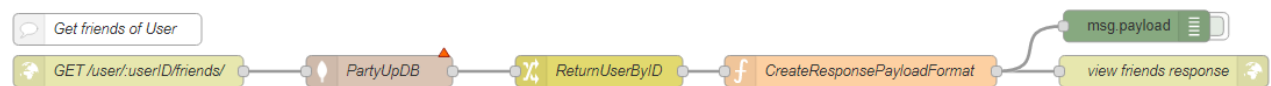
Εικόνα 29 Ποή NodeRed που αντιστοιχεί στο API call POST /user/{userID}/party/activeParty/{activePartyID}/payment/.

Ποή endpoint GET /user/{userID}/party/activePartyID/{activePartyID}/participants/



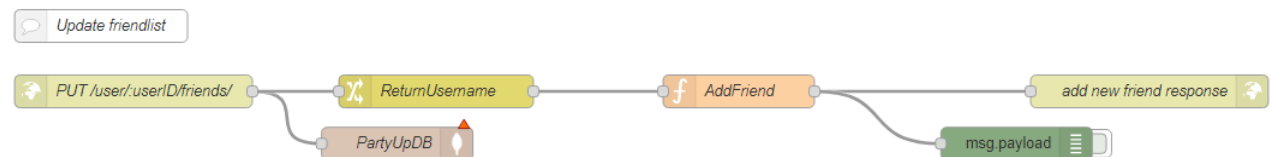
Εικόνα 30 Ποή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/party/activePartyID/{activePartyID}/participants/.

Ποή endpoint GET /user/{userID}/friends/



Εικόνα 31 Ποή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/friends/.

Ποή endpoint



Εικόνα 32 Ποή NodeRed που αντιστοιχεί στο API call PUT /user/{userID}/friends/.

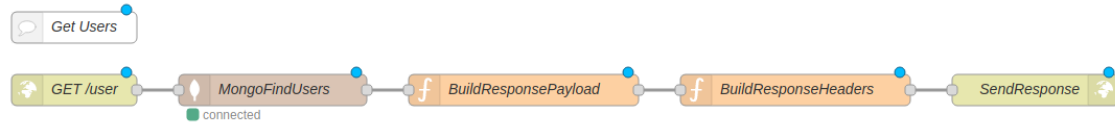
Ποή endpoint GET /user/{userID}/friends/searchResults/



Εικόνα 33 Ποή NodeRed που αντιστοιχεί στο API call GET /user/{userID}/friends/searchResults/.



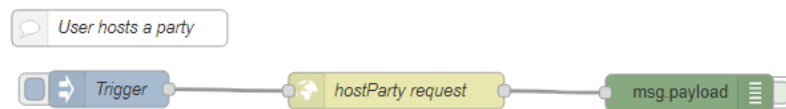
Ροή endpoint GET /user



Εικόνα 34 Ροή NodeRed που αντιστοιχεί στο API call GET /user.

3.2 Υλοποίηση Ιστοριών χρήστη

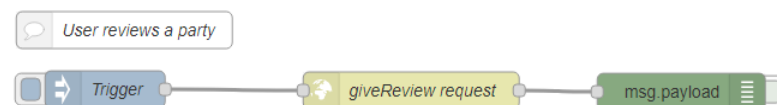
3.2.1 Ιστορία Χρήστη host a party



Εικόνα 35 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη "host a party".

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
hostParty request	http-request	Κάνει κλήση προς την υπηρεσία POST /user/{userID}/party/host η οποία δημιουργεί ένα πάρτι.
Msg.payload	debug	Τυπώνει στην κονσόλα το πάρτι που μόλις δημιουργήθηκε.

3.2.2 Ιστορία χρήστη review a party

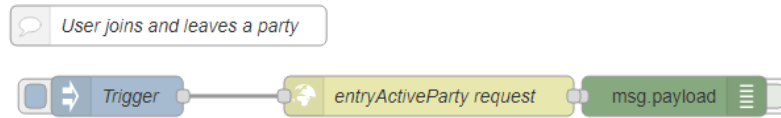


Εικόνα 36 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη "review a party".

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
giveReview request	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID}/party/partyHistory/{partyHistoryID}/review/ η οποία προσθέτει μία αξιολόγηση σε ένα πάρτι.
Msg.payload	debug	Τυπώνει στην κονσόλα την ανανεωμένη αξιολόγηση του πάρτι.



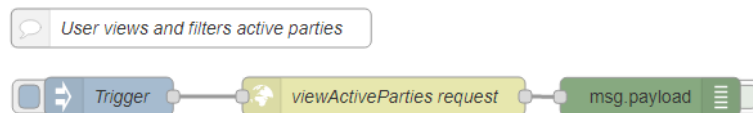
3.2.3 Ιστορία χρήστη join και leaves a party



Εικόνα 37 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “join a party” και “leave a party”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
entryActivePartyrequest	http-request	Κάνει κλήση προς την υπηρεσία PUT /user/{userID}/party/activeParty/{activePartyID} η οποία προσθέτει ή αφαιρεί τον χρήστη στο συγκεκριμένο πάρτι.
Msg.payload	debug	Τυπώνει στην κονσόλα την ανανεωμένη λίστα συμμετεχόντων του πάρτι.

3.2.4 Ιστορία χρήστη view και filter active parties

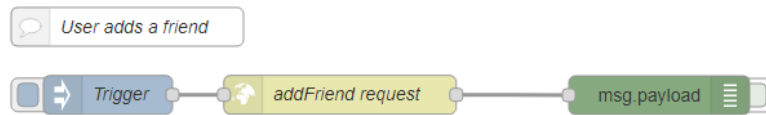


Εικόνα 38 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “view active parties” και “filter active parties”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
viewActiveParties request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{userID}/party/activeParty η οποία επιστρέφει τα ενεργά πάρτι βάσει συγκεκριμένων φίλτρων.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα των ενεργών πάρτι.



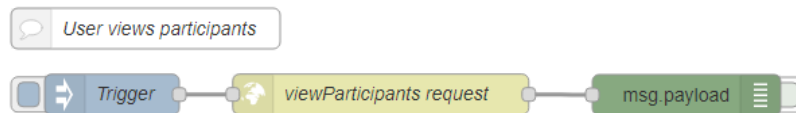
3.2.5 Ιστορία χρήστη add a friend



Εικόνα 39 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “add a friend”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
addFriend request	http-request	Κάνει κλήση προς την υπηρεσία PUT /user/{userID}/friends/ η οποία προσθέτει τον συγκεκριμένο χρήστη στην λίστα φίλων.
Msg.payload	debug	Τυπώνει στην κονσόλα την ανανεωμένη λίστα φίλων του χρήστη.

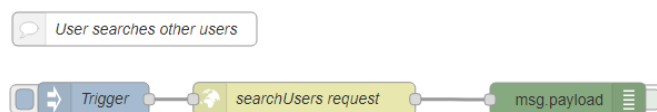
3.2.6 Ιστορία χρήστη views participants



Εικόνα 40 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήση “view participants”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
viewParticipants request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{userID}/party/activePartyID/{activePartyID}/participants/η οποία επιστρέφει την λίστα συμμετεχόντων του συγκεκριμένου πάρτι.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα συμμετεχόντων του πάρτι.

3.2.7 Ιστορία χρήστη search users

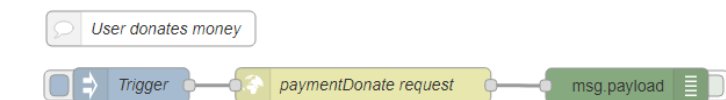


Εικόνα 41 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήση “search users”.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
searchUsers request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{userID}/friends/searchResults/ η οποία αναζητά χρήστες βάσει του συγκεκριμένου username και επιστρέφει μία λίστα χρηστών με τα καλύτερα ταιριάσματα.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα χρηστών που επιστρέφεται.

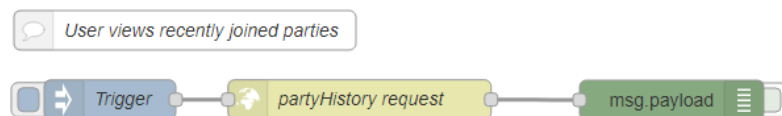
3.2.8 Ιστορία χρήστη donates money



Εικόνα 42 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “donate money”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
paymentDonate request	http-request	Κάνει κλήση προς την υπηρεσία POST /user/{userID}/party/activeParty/{activePartyID}/payment/ η οποία επικοινωνεί με το API της PayPal και ολοκληρώνει την χρηματική συναλλαγή.
Msg.payload	debug	Τυπώνει στην κονσόλα τις πληροφορίες της χρηματικής συναλλαγής.

3.2.9 Ιστορία χρήστη views recently joined parties



Εικόνα 43 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “view recently joined parties”.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
viewRecentlyJoinedParties request	http-request	Κάνει κλήση προς την υπηρεσία GET /user/{userID}/party/partyHistory/ η οποία επιστρέφει τα πάρτι στα οποία συμμετείχε ο χρήστης προσφάτως.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα με τα πάρτι που συμμετείχε ο χρήστης προσφάτως.



Παράρτημα Ι – Ανοιχτά Θέματα

- Τα πρότυπα σχεδίασης που υλοποιήσαμε δεν καλύπτουν όλες τις λειτουργικές και μη λειτουργικές απαιτήσεις. Συγκεκριμένα, απαιτείται εμπεριστατωμένη ανάλυση των κλάσεων του προηγούμενου παραδοτέου προκειμένου να έχουμε ένα συνολικό πλάνο σχεδιασμού για το οποίο θα αποδοθούν συγκεκριμένα πρότυπα σχεδίασης.
- Στον ορισμό του REST API μας δεν ασχοληθήκαμε με την διασύνδεση του συστήματος με το Google Maps API καθώς κάναμε την παραδοχή ότι το application επικοινωνεί άμεσα με το API αυτό.