



Τεχνολογία Λογισμικού
Τομέας Ηλεκτρονικής και
Υπολογιστών
Τμήμα ΗΜΜΥ
Α.Π.Θ.

8ο Εξάμηνο
Άνοιξη 2020



Προδιαγραφές Λογισμικού και Υλοποίηση συστήματος

Del.1.2

Version 0.6

Κούτρα Χαριτίνη charitink@ece.auth.gr
Κωστοπούλου Ναταλία nkostopou@ece.auth.gr
Νάστος Βίκτωρ viktorna@ece.auth.gr
Πορτοκαλίδης Σταύρος stavport@ece.auth.gr

08/06/2020



Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Α. Συμεωνίδης		asymeon@issel.ee.auth.gr
Κούτρα Χαριτίνη	ETHER (Ομάδα 18)	charitink@ece.auth.gr
Κωστοπούλου Ναταλία	ETHER (Ομάδα 18)	nkostopou@ece.auth.gr
Νάστος Βίκτωρ	ETHER (Ομάδα 18)	viktorna@ece.auth.gr
Πορτοκαλίδης Σταύρος	ETHER (Ομάδα 18)	stavport@ece.auth.gr



ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Πίνακας Περιεχομένων.....	3
Λίστα Σχημάτων	4
1. Πρότυπα Σχεδιασμού	5
2. Αρχιτεκτονική Συστήματος	11
2.1 Αναγνώριση Πόρων (Resources) Συστήματος.....	13
2.2 Τεκμηρίωση REST διεπαφής	13
2.2.1 Πόρος Flight.....	13
2.2.1.1 Μοντέλο δεδομένων Flight.....	13
2.2.1.2 Endpoint πόρου Flight.....	13
2.2.2 Πόρος Search.....	16
2.2.2.1 Μοντέλο δεδομένων Search.....	16
2.2.2.2 Endpoint πόρου Search.....	16
2.2.2.3 Endpoint πόρου Search.....	19
2.2.2.4 Endpoint πόρου Search.....	21
2.2.2.5 Endpoint πόρου Search.....	23
2.2.3 Πόρος Supplier	25
2.2.3.1 Μοντέλο δεδομένων Supplier.....	25
2.2.3.2 Endpoint πόρου Supplier.....	25
2.2.3.3 Endpoint πόρου Supplier.....	28
2.2.3.3 Endpoint πόρου Supplier.....	30
2.2.4 Πόρος Travel.....	33
2.2.4.1 Μοντέλο δεδομένων Travel.....	33
2.2.4.2 Endpoint πόρου Travel.....	33
2.2.5 Πόρος User	35
2.2.5.1 Μοντέλο δεδομένων User	35
2.2.5.2 Endpoint πόρου User	35
2.2.5.3 Endpoint πόρου User	37
3. Υλοποίηση Συστήματος με Node-RED	40
3.1 Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed	40
3.1.1 Ροές Πόρου XX.....	40
3.2 Υλοποίηση της Εφαρμογής Πελάτη	45
3.2.1 Σενάριο Χρήσης XX.....	45
Παράρτημα Ι – Ανοιχτά Θέματα.....	50



ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ

- Σχήμα 1. Διάγραμμα κλάσεων Bridge Pattern
- Σχήμα 2. Διάγραμμα κλάσεων Bridge Pattern
- Σχήμα 3. Διάγραμμα κλάσεων Bridge Pattern
- Σχήμα 4. Διάγραμμα κλάσεων Bridge Pattern
- Σχήμα 5. Διάγραμμα κλάσεων Bridge Pattern
- Σχήμα 6. Διάγραμμα κλάσεων συνδυασμός Bridge-Strategy Pattern
- Σχήμα 7. Διάγραμμα κλάσεων συνδυασμός Observer Pattern
- Σχήμα 8. Διάγραμμα κλάσεων συνδυασμός Observer Pattern
- Σχήμα 9. Ροή Πόρου Flight
- Σχήμα 10. Ροή Πόρου Search
- Σχήμα 11. Ροή Πόρου Search
- Σχήμα 12. Ροή Πόρου Search
- Σχήμα 13. Ροή Πόρου Search
- Σχήμα 14. Ροή Πόρου Supplier
- Σχήμα 15. Ροή Πόρου Supplier
- Σχήμα 16. Ροή Πόρου Supplier
- Σχήμα 17. Ροή Πόρου Supplier
- Σχήμα 18. Ροή για Ιστορία χρήστη 1
- Σχήμα 19. Ροή για Ιστορία χρήστη 3
- Σχήμα 20. Ροή για Ιστορία χρήστη 3
- Σχήμα 21. Ροή για Ιστορία χρήστη 4
- Σχήμα 22. Ροή για Ιστορία χρήστη 5
- Σχήμα 23. Ροή για Ιστορία χρήστη 6
- Σχήμα 24. Ροή για Ιστορία χρήστη 7
- Σχήμα 25. Ροή για Ιστορία χρήστη 8



1. ΠΡΟΤΥΠΑ ΣΧΕΔΙΑΣΜΟΥ

Τα πρότυπα σχεδιασμού που υιοθετήθηκαν βασίζονται στις παρακάτω Λειτουργικές και Μη Λειτουργικές απαιτήσεις:

ΜΛΑ 1

Το σύστημα πρέπει να μπορεί να συνδέεται σε noSQL βάσεις δεδομένων , MongoDB και ElasticSearch

ΜΛΑ 2

Το σύστημα πρέπει να μπορεί να ενημερώνεται αυτόματα σε περίπτωση αλλαγής κάποιου από τα δεδομένα του.

ΛΑ 1

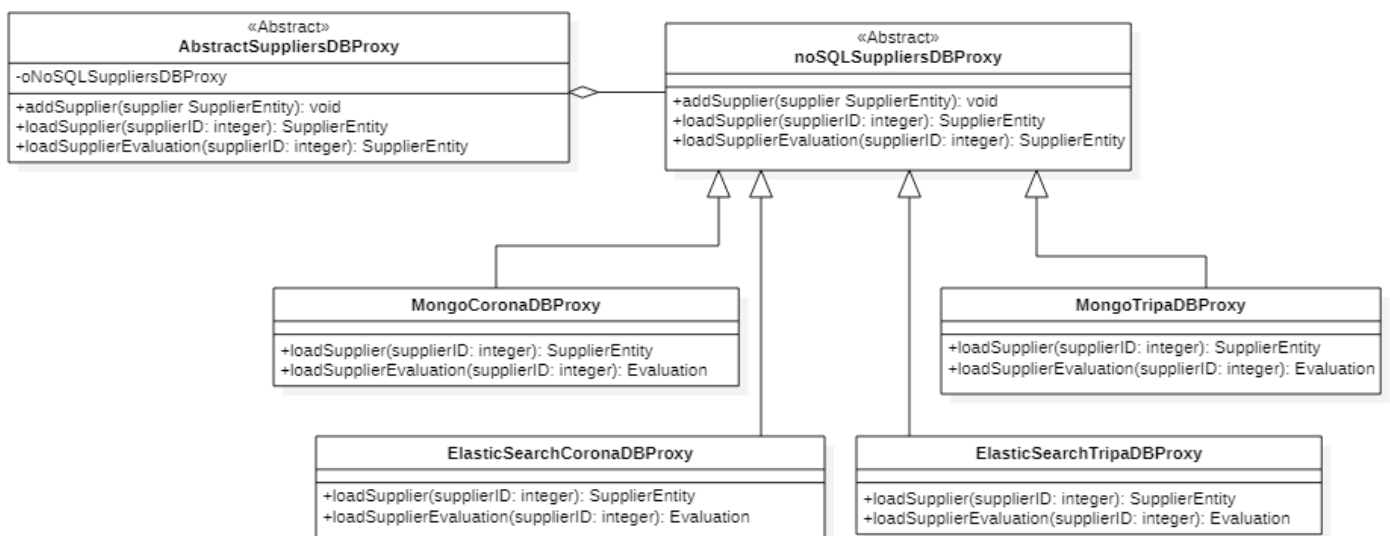
Ο χρήστης πρέπει να μπορεί να επιλέξει την ταξινόμηση των αποτελεσμάτων

1.1 Bridge Pattern

Ταξινομία : Δομικό Πρότυπο

1.1.1 Bridge Pattern για SupplierDBProxy

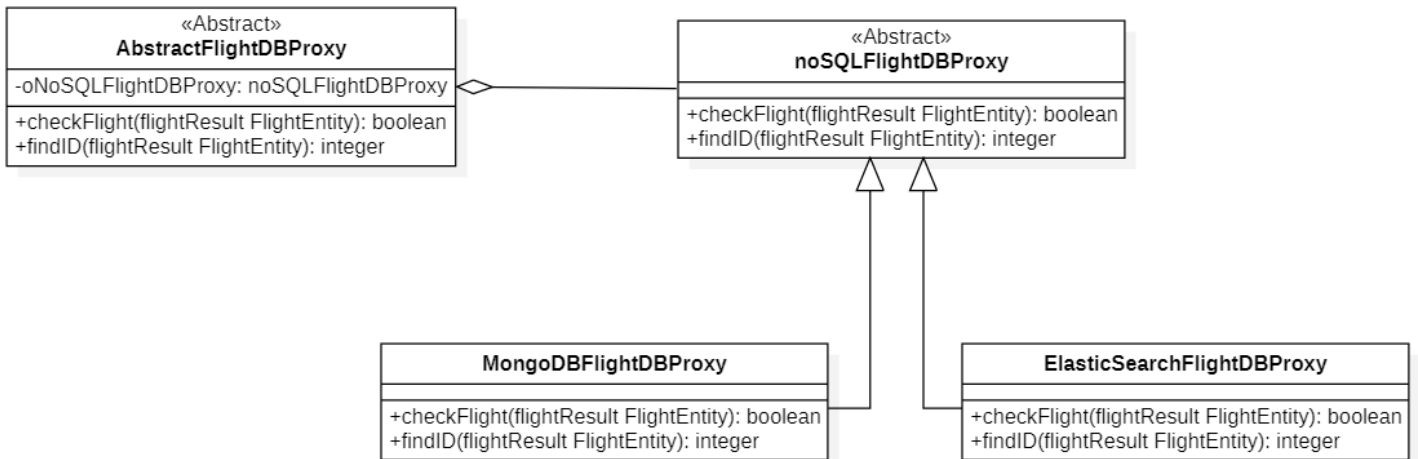
Εφαρμόζουμε το συγκεκριμένο πρότυπο σχεδίασης στην κλάση SupplierDBProxy καθώς θέλουμε ανεξαρτησία υλοποίησης και το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων των παρόχων. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου. Σημειώνεται ότι δίνονται ενδεικτικά κάποια παραδείγματα από τους παρόχους που συνεργάζονται με το σύστημα.



Σχήμα 1. Διάγραμμα κλάσεων Bridge Pattern

1.1.2 Bridge Pattern για FlightDBProxy

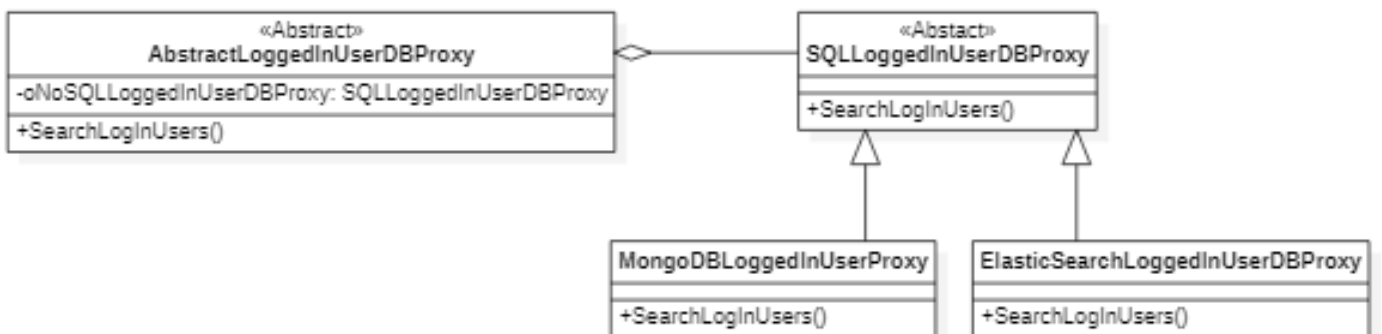
Εφαρμόζουμε το συγκεκριμένο πρότυπο σχεδίασης στην κλάση FlightDBProxy καθώς θέλουμε ανεξαρτησία υλοποίησης και το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων του. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου.



Σχήμα 2. Διάγραμμα κλάσεων Bridge Pattern

1.1.3 Bridge Pattern για LoggedInUserDBProxy

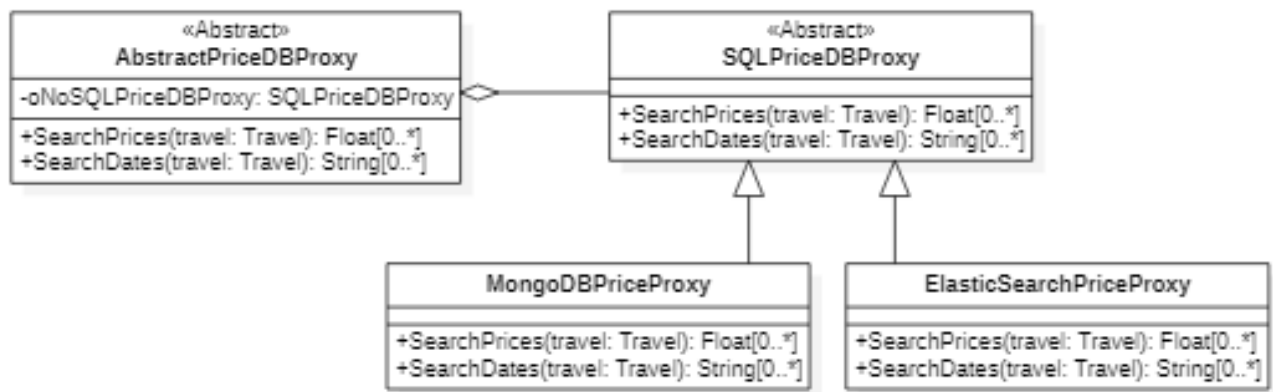
Εφαρμόζουμε το συγκεκριμένο πρότυπο σχεδίασης στην κλάση LoggedInUserDBProxy καθώς θέλουμε το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων του. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου.



Σχήμα 3. Διάγραμμα κλάσεων Bridge Pattern

1.1.4 Bridge Pattern για PriceDBProxy

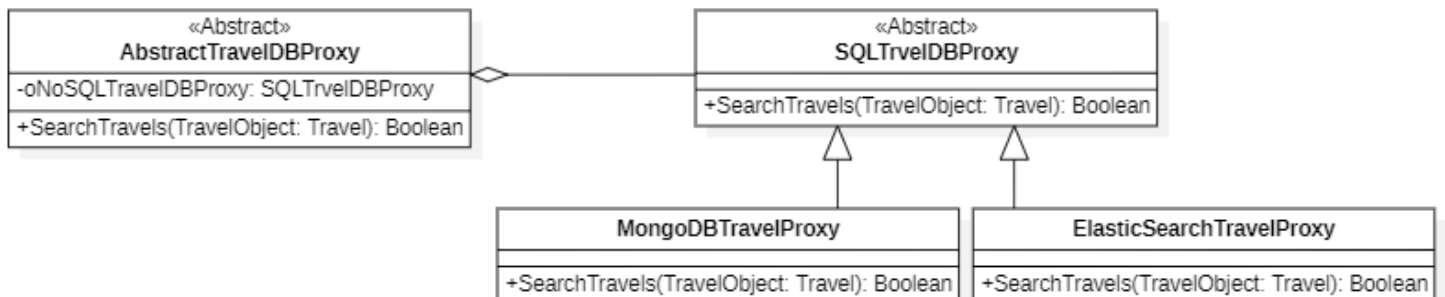
Εφαρμόζουμε το συγκεκριμένο πρότυπο σχεδίασης στην κλάση PriceDBProxy καθώς θέλουμε το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων των παρόχων. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου.



Σχήμα 4. Διάγραμμα κλάσεων Bridge Pattern

1.1.5 Bridge Pattern για TravelDBProxy

Εφαρμόζουμε το συγκεκριμένο πρότυπο σχεδίασης στην κλάση TravelDBProxy καθώς θέλουμε το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων του συστήματος. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου.



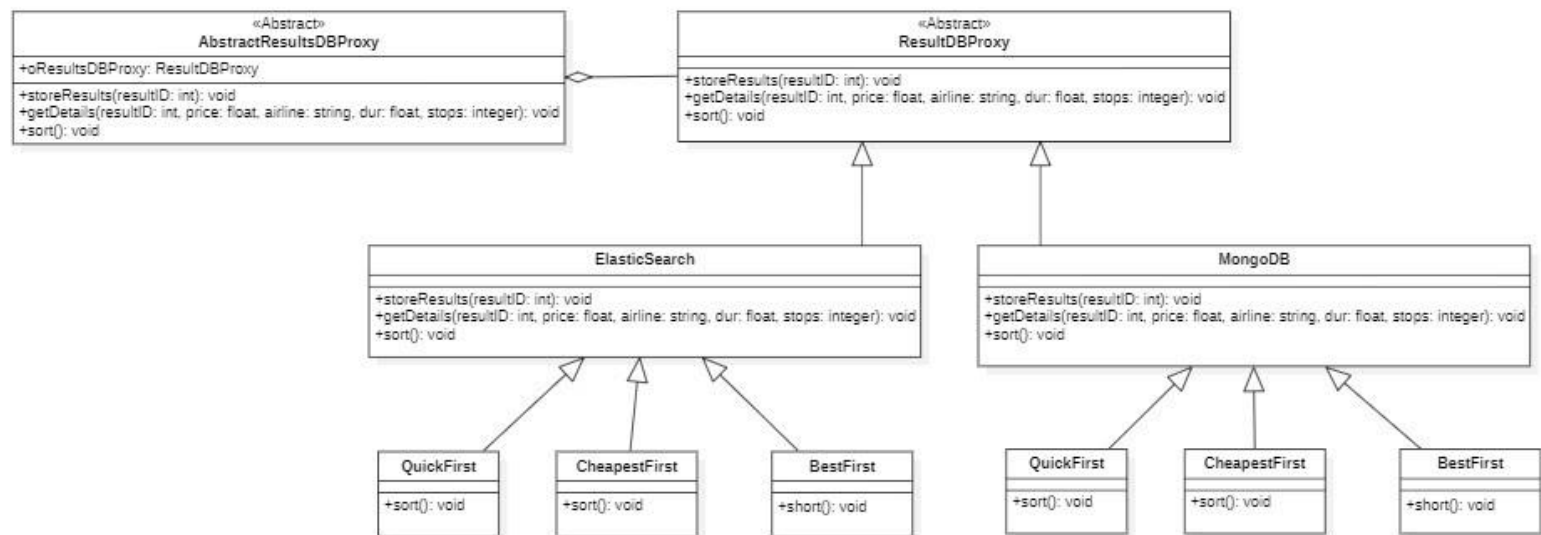
Σχήμα 5. Διάγραμμα κλάσεων Bridge Pattern

1.2 Strategy Pattern

Ταξινόμια : Πρότυπο συμπεριφοράς

1.2.1 Συνδυασμός Bridge και Strategy pattern για ResultDBProxy

Εφαρμόζουμε το Bridge pattern στην κλάση ResultDBProxy καθώς θέλουμε ανεξαρτησία υλοποίησης και το σύστημα μας να επικοινωνεί με τις noSQL βάσεις δεδομένων . Εφαρμόζουμε το Strategy pattern, σε συνδυασμό με το bridge pattern, στην κλάση ResultDBProxy, με σκοπό την ταξινόμηση μια λίστας αποτελεσμάτων στις βάσεις δεδομένων. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή των συγκεκριμένων προτύπων.



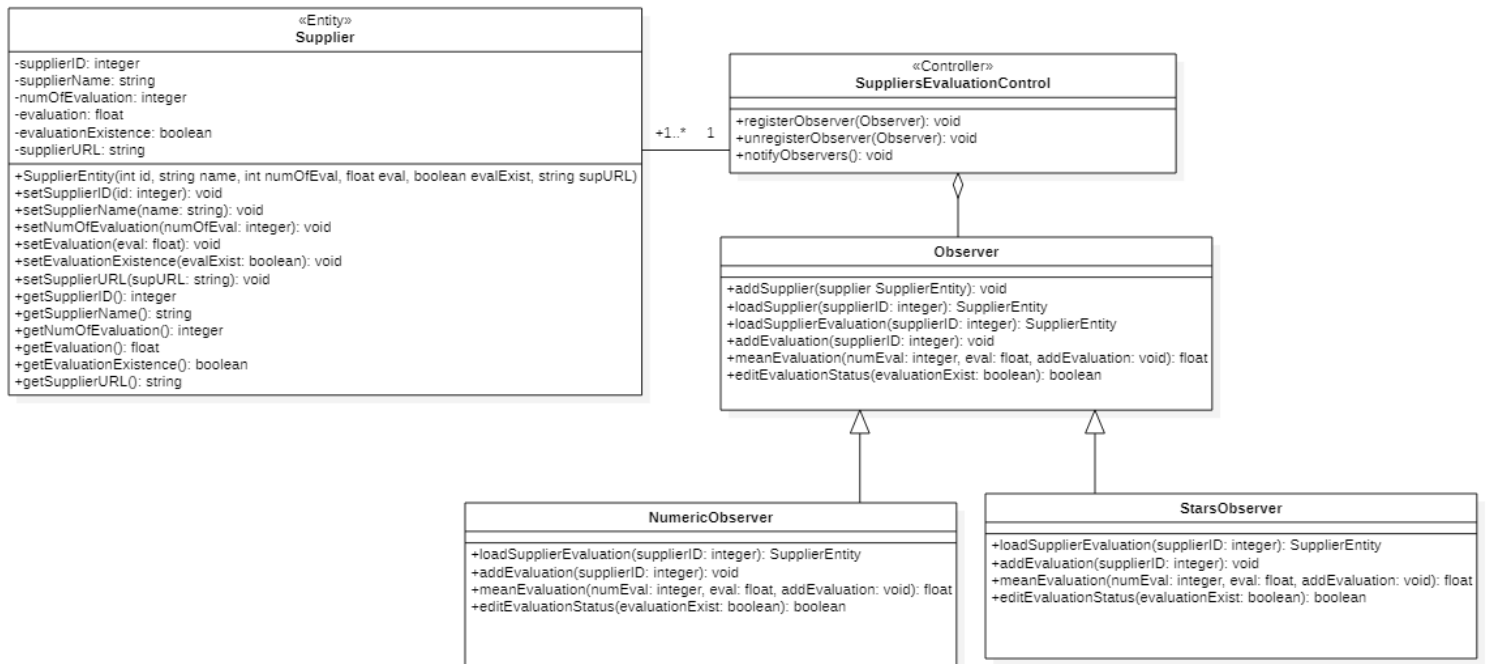
Σχήμα 6. Διάγραμμα κλάσεων συνδυασμός Bridge-Strategy Pattern

1.3 Observer Pattern

Ταξινόμια: Πρότυπο συμπεριφοράς

1.3.1 Observer Pattern για SupplierEntity

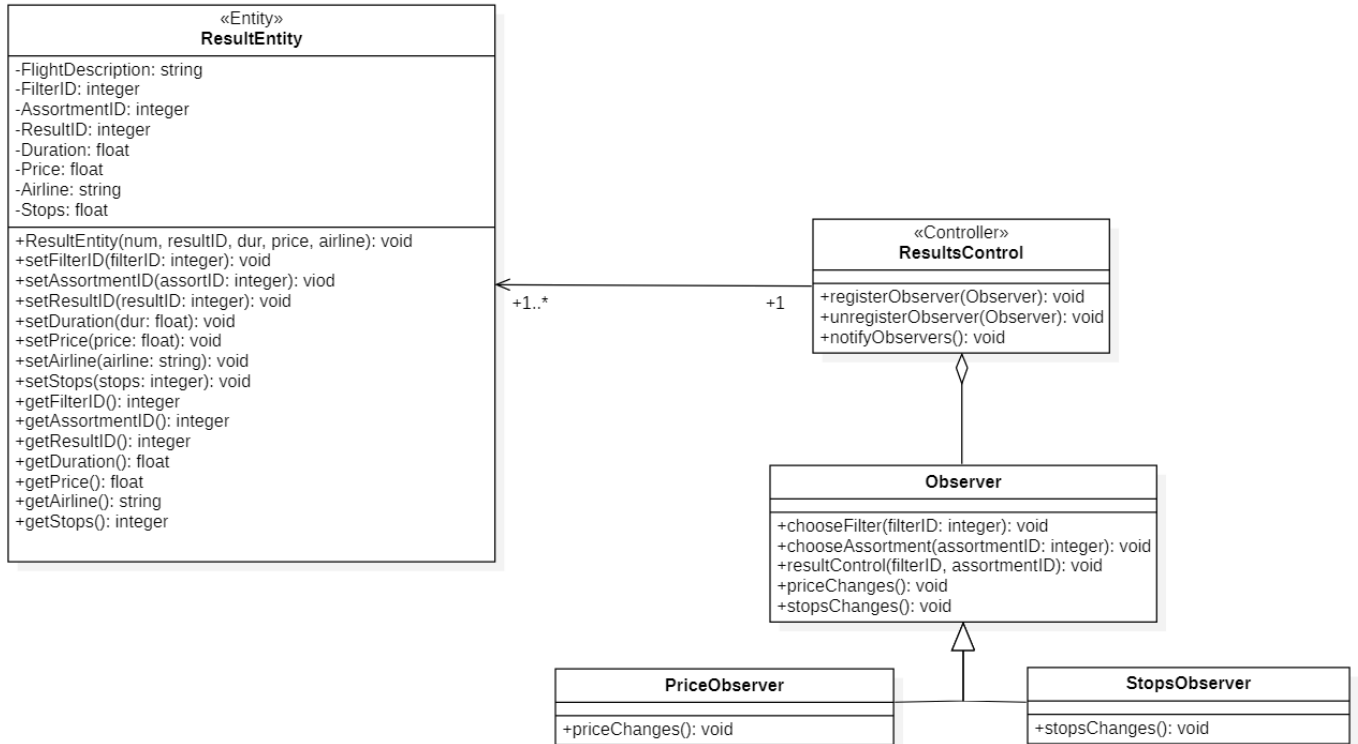
Σκοπός της εφαρμογής του συγκεκριμένου pattern στην κλάση SupplierEntity είναι η αυτόματη ενημέρωση όλων των οθονών που δείχνουν την αξιολόγηση των παρόχων τόσο στην αριθμητική τιμή όσο και στα αστεράκια αξιολόγησης. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου. Επειδή το διάγραμμα κλάσεων μας ακολουθεί το πρότυπο BEC ουσιαστικά έχουμε έναν συνδυασμό των δύο προτύπων.



Σχήμα 7. Διάγραμμα κλάσεων συνδυασμός Observer Pattern

1.3.2 Observer Pattern για ResultsEntity

Σκοπός της εφαρμογής του συγκεκριμένου pattern στην κλάση ResultEntity είναι η αυτόματη ενημέρωση της οθόνης που δείχνει τα αποτελέσματα της αναζήτησης. Παρακάτω παρουσιάζεται το συνοπτικό διάγραμμα κλάσεων που προκύπτει μετά την εφαρμογή του συγκεκριμένου προτύπου. Επειδή το διάγραμμα κλάσεων μας ακολουθεί το πρότυπο BEC ουσιαστικά έχουμε έναν συνδυασμό των δύο προτύπων.



Σχήμα 8. Διάγραμμα κλάσεων συνδυασμός Observer Pattern



2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ

Link: <https://www.dropbox.com/sh/8yg4gawo9lf0om9/AACqqyZ5hmjPEKrcqUIPh67Wa?dl=0>

Swaggerhub: https://app.swaggerhub.com/apis/haritini_koutra/ether/1.0.0

2.1 Αναναγνώριση Πόρων (Resources) Συστήματος

Resouces	Endpoint	HTTP Verb	Πόρος Rest	Parameters
user	user/login	GET	Πόρος one-off	username:string password:string
	user/{username}	GET	Πόρος one-off	username:string
	user/{username}/flight /search	GET	Αλγοριθμικός Πόρος	username:string
	user/{username}/checkPriceChange	GET	Αλγοριθμικός Πόρος	username:string
flight	flight/search	GET	Αλγοριθμικός Πόρος	date:string(path) startLocation:string(path) endLocation:sting(path) numberOfTravellers:integer cabinClass:string
travel	travel/search	GET	Αλγοριθμικός Πόρος	date:string maxBudget:integer minBudget:integer
search	/search	GET	Αλγοριθμικός Πόρος	
	/search/sort	GET	Αλγοριθμικός Πόρος	
	/search/filter	GET	Αλγοριθμικός Πόρος	
	/search/result/{resultID}	GET	Αλγοριθμικός Πόρος	resultID:integer



Resousces	Endpoint	HTTP Verb	Πόρος Rest	Parameters
supplier	/search/result/{resultID}/supplier/{supplierID}	GET	Αλγοριθμικός Πόρος	resultID:integer supplierID:integer
	/search/result/{resultID}/supplier/{supplierID}	GET	Αλγοριθμικός Πόρος	resultID:integer supplierID:integer
	/user/{username}/supplier/{supplierID}/evaluate	POST	Αλγοριθμικός Πόρος	username:string supplierID:integer



2.2 Τεκμηρίωση REST διεπαφής

2.2.1 Πόρος Flight

2.2.1.1 Μοντέλο δεδομένων flight

```
Flight {  
  flight      integer($int64)  
  date        string  
  startLocation string  
  endLocation string  
  numberOfTravellers integer($int32)  
  cabinClass  string  
}
```

2.2.1.2 Endpoint πόρου flight (/flight/search/{date}/{startLocation}/{endLocation})

flight Operations about flights

GET /flight/search/{date}/{startLocation}/{endLocation} Search flights

A user must be able to search flights

Parameters Cancel

Name	Description
date * required string (path)	date <input type="text" value="6-6-2020"/>
startLocation * required string (path)	start location <input type="text" value="ATH"/>
endLocation * required string (path)	end location <input type="text" value="JFK"/>



numberOfTravellers * required
integer(\$int32)
(query)

Number of travellers

cabinClass * required
string
(query)

Cabin class

Execute

Clear

Responses

Response content type

application/json

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/flight/search/6-6-2020/%CE%91%CE%A4%CE%97/JFK?numberOfTravellers=4&cabinClass=business" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/flight/search/6-6-2020/%CE%91%CE%A4%CE%97/JFK?numberOfTravellers=4&cabinClass=business
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>[{ "resultID": 0, "Flight": { "flight": 0, "date": "string", "startLocation": "string", "endLocation": "string", "numberOfTravellers": 0, "cabinClass": "string" }, "price": 0, "duration": 0, "stops": 0, "airline": "string" }]</pre></div><div>Download</div></div>



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: 164
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 14:36:22 GMT
etag: W/"a4-ENCpQ855TmsI0sNeoJtGJvCf8mA"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 94c165dd7b9a9d68d15ef70552eaba76.cloudfront.net (CloudFront)
x-amz-cf-id: PGiWVI07QDlWMB3X-y_XUu0sgg0YySqZa-xEwjeNzPjr0jZ9zc0Y_Q==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

613 ms

Responses

Code	Description
------	-------------

200	successful operation
-----	----------------------

Example Value | Model

```
[
  {
    "resultID": 0,
    "Flight": {
      "flight": 0,
      "date": "string",
      "startLocation": "string",
      "endLocation": "string",
      "numberOfTravellers": 0,
      "cabinClass": "string"
    },
    "price": 0,
    "duration": 0,
    "stops": 0,
    "airline": "string"
  }
]
```

404	Flight not found
-----	------------------



2.2.2 Πόρος Search

2.2.2.1 Μοντέλο δεδομένων search

```
Search ▾ [Search ▾ {  
  resultID      integer($int64)  
  Flight        Flight > {...}  
  price          integer($int32)  
  duration       integer($int32)  
  stops         integer($int32)  
  airline       string  
}]
```

2.2.2.2 Endpoint πόρου search (/search/filter)

search Operations about search results

GET **/search/filter** Filter results

A user must be able to apply filters to the results

Parameters Cancel

Name	Description
filterForDuration * required	Filters to apply
boolean (query)	<div>false</div>
filterForPrice * required	Filters to apply
boolean (query)	<div>false</div>
filterForAirline * required	Filters to apply
boolean (query)	<div>true</div>



filterForStops * required
boolean
(query)

Filters to apply

false

Execute

Clear

Responses

Response content type

application/json

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/filter?filterForDuration=false&filterForPrice=false&filterForAirline=true&filterForStops=false" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/filter?filterForDuration=false&filterForPrice=false&filterForAirline=true&filterForStops=false
```

Server response

Code	Details
200	<div>Response body</div> <div><pre>[{ "resultID": 0, "Flight": { "flight": 0, "date": "string", "startLocation": "string", "endLocation": "string", "numberOfTravellers": 0, "cabinClass": "string" }, "price": 0, "duration": 0, "stops": 0, "airline": "string" }]</pre></div> <div>Download</div>



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: 164
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 15:03:02 GMT
etag: W/"a4-ENCpQ855TmsI0sNeoJtGJvCf8mA"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 f8fb5342430dbbd465e18ccbef143b23.cloudfront.net (CloudFront)
x-amz-cf-id: v-c08RYo8NyuqizrjUq0inRNvDme7GbIHms3E3imfQ18hwdeROG6QQ==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

634 ms

Responses

Code	Description
------	-------------

200	successful operation
-----	----------------------

Example Value | Model

```
[
  {
    "resultID": 0,
    "Flight": {
      "flight": 0,
      "date": "string",
      "startLocation": "string",
      "endLocation": "string",
      "numberOfTravellers": 0,
      "cabinClass": "string"
    },
    "price": 0,
    "duration": 0,
    "stops": 0,
    "airline": "string"
  }
]
```



2.2.2.3 Endpoint πύρου search (/search/sort)

GET **/search/sort** Sort results

A user must be able to sort the results

Parameters Cancel

Name	Description
bestFirst * required boolean (query)	Assortment by best flight first <div>false</div>
cheapestFirst * required boolean (query)	Assortment by cheapest flight first <div>true</div>
quickestFirst * required boolean (query)	Assortment by quickest flight first <div>false</div>

Execute

Clear

Responses Response content type application/json

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/sort?bestFirst=false&cheapestFirst=true&quickestFirst=false" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/sort?bestFirst=false&cheapestFirst=true&quickestFirst=false
```



Server response

Code

Details

200

Response body

```
[
  {
    "resultID": 0,
    "Flight": {
      "flight": 0,
      "date": "string",
      "startLocation": "string",
      "endLocation": "string",
      "numberOfTravellers": 0,
      "cabinClass": "string"
    },
    "price": 0,
    "duration": 0,
    "stops": 0,
    "airline": "string"
  }
]
```

[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: 164
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 15:17:20 GMT
etag: W/"a4-ENCpQ855TmsI0sNeoJtGJvCfBmA"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 f4a2a568790100f41e907486ba960ec7.cloudfront.net (CloudFront)
x-amz-cf-id: rvnZzydFw0h3hvtSRee7gdb0d8auMTJ-50shKx2pfgoVMI9E-u_hfw==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

554 ms



Responses

Code	Description
200	successful operation

Example Value | Model

```
[
  {
    "resultID": 0,
    "Flight": {
      "flight": 0,
      "date": "string",
      "startLocation": "string",
      "endLocation": "string",
      "numberOfTravellers": 0,
      "cabinClass": "string"
    },
    "price": 0,
    "duration": 0,
    "stops": 0,
    "airline": "string"
  }
]
```

2.2.2.4 Endpoint πόρου search (/search/result/{resultID})

GET /search/result/{resultID} Flight result

A user must be able to choose a flight

Parameters Cancel

Name	Description
resultID * required	Specific number of the result
integer(\$int32)	
(path)	

Execute

Clear

Responses Response content type application/json



Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/3141" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/3141
```

Server response

Code

Details

200

Response body

```
{
  "resultID": 0,
  "flight": {
    "flight": 0,
    "date": "string",
    "startLocation": "string",
    "endLocation": "string",
    "numberOfTravellers": 0,
    "cabinClass": "string"
  },
  "price": 0,
  "duration": 0,
  "stops": 0,
  "airline": "string"
}
```

[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: 160
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 15:23:57 GMT
etag: W/"a0-uHExRAVUMohs250VgxhzaYVJLVQ"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 14f9681cda1f73138148aa327582a6ee.cloudfront.net (CloudFront)
x-amz-cf-id: Ax9Ab3moiBzoEvY1eLHKuoY_Fd6m1oFs9_cLq0_0Nny5B8k5rEDhJg==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

392 ms



Responses	
Code	Description
200	successful operation
Example Value Model	
<pre>{ "resultID": 0, "Flight": { "flight": 0, "date": "string", "startLocation": "string", "endLocation": "string", "numberOfTravellers": 0, "cabinClass": "string" }, "price": 0, "duration": 0, "stops": 0, "airline": "string" }</pre>	
404	Result not found

2.2.2.5 Endpoint πύρου search (/search)

GET **/search** Search Flights result

A user must be able to see the flight search results

Parameters Cancel

No parameters

Execute

Clear

Responses Response content type application/json

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search" -H
"accept: application/json"
```



Server response

Code	Details
------	---------

200

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: 20
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 15:23:35 GMT
etag: W/"14-40hJ6nhrn3soo1wXlJ6FqT6xz/E"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 14f9681cda1f73138148aa327582a6ee.cloudfront.net (CloudFront)
x-amz-cf-id: sftdpstPb1PE_g6V_CeMsAkbnMeHPbz3Jvbe2r0Sc3HFcK58VZmUuQ==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

554 ms

Responses

Code	Description
------	-------------

200

successful operation



2.2.3 Πόρος Supplier

2.2.3.1 Μοντέλο δεδομένων supplier

```
Supplier {  
  supplierID integer($int64)  
  name string  
  evaluation number  
  url string  
  evaluationExistence boolean  
}
```

2.2.3.2 Endpoint πόρου supplier (/user/{username}/supplier/{supplierID}/evaluate)

supplier Operations about supplier

POST /user/{username}/supplier/{supplierID}/evaluate Evaluate supplier

A logged in user must be able to evaluate suppliers

Parameters Cancel

Name	Description
username * required string (path)	User to return
supplierID * required integer(\$int64) (path)	Supplier to evaluate



evaluation * required

object
(body)

Evaluation

Edit Value | Model

```
{  "supplierID": 0,  "name": "string",  "evaluation": 0,  "url": "string",  "evaluationExistence": true}
```

Cancel

Parameter content type

application/json

Execute

Clear

Responses

Response content type

application/json

Curl

```
curl -X POST  "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/kwstas/supplier/3242/evaluate"  -H "accept: application/json" -H "Content-Type: application/json" -d "{ \"supplierID\": 0,  \"name\": \"string\", \"evaluation\": 0, \"url\": \"string\", \"evaluationExistence\": true}"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/kwstas/supplier/3242/evaluate
```



Server response	
Code	Details
200	<div>Response body<div><pre>{ "supplierID": 0, "name": "string", "evaluation": 1.5, "url": "string", "evaluationExistence": true }</pre></div><div>Download</div></div> <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: 96 content-type: application/json; charset=utf-8 date: Mon, 08 Jun 2020 15:50:19 GMT etag: W/"60-nZTr25XiXkVpmoyljvJODswqcME" expires: -1 server: nginx/1.14.0 + Phusion Passenger 5.3.4 status: 200, 200 OK via: 1.1 7b83ac68a47a443291524106f7b5d3dd.cloudfront.net (CloudFront) x-amz-cf-id: lZm20Kol-mRsJxMrm-MPGLLMw2S5ZWyRnlce5m5jmlrwMAZVshlrwg== x-amz-cf-pop: ATH50-C1 x-cache: Miss from cloudfront x-powered-by: Express, Phusion Passenger 5.3.4</pre></div></div> <div>Request duration<div>961 ms</div></div>
Responses	
Code	Description
200	<div>successful operation</div> <div>Example Value Model<div><pre>{ "supplierID": 0, "name": "string", "evaluation": 0, "url": "string", "evaluationExistence": true }</pre></div></div>
400	<div>Not a logged in user</div>



2.2.3.3 Endpoint πόρου supplier (/search/result/{resultID}/supplier/{supplierID}/connect)

GET `/search/result/{resultID}/supplier/{supplierID}/connect` Connect with a supplier ↩

A user must be able to connect with a supplier

Parameters Cancel

Name	Description
resultID * required integer(\$int32) (path)	specific number of a result
<input type="text" value="8127"/>	
supplierID * required integer(\$int32) (path)	specific number of a supplier
<input type="text" value="5368"/>	

Execute Clear

Responses Response content type **application/json** ▼

Curl

```
curl -X GET
"https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/8127/supplier/5368/connect" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/8127/supplier/5368/connect
```



Server response

Code

Details

200

Response body

```
[
  {
    "resultID": 0,
    "Flight": {
      "flight": 0,
      "date": "string",
      "startLocation": "string",
      "endLocation": "string",
      "numberOfTravellers": 0,
      "cabinClass": "string"
    },
    "price": 0,
    "duration": 0,
    "stops": 0,
    "airline": "string"
  }
]
```

[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: 164
content-type: application/json; charset=utf-8
date: Mon, 08 Jun 2020 15:58:39 GMT
etag: W/"a4-ENCpQ855TmsI0sNeoJtGJvCfBmA"
expires: -1
server: nginx/1.14.0 + Phusion Passenger 5.3.4
status: 200, 200 OK
via: 1.1 b968b41049fe2eec709f0d8dbb0fa593.cloudfront.net (CloudFront)
x-amz-cf-id: FRhtZm6nXrNlhtcLS5-dB5Vp3u-WIMaewfYnWJs70743EFnShFtzkg==
x-amz-cf-pop: ATH50-C1
x-cache: Miss from cloudfront
x-powered-by: Express, Phusion Passenger 5.3.4
```

Request duration

743 ms



Responses	
Code	Description
200	successful operation
<div>Example Value Model</div> <pre>[{ "resultID": 0, "Flight": { "flight": 0, "date": "string", "startLocation": "string", "endLocation": "string", "numberOfTravellers": 0, "cabinClass": "string" }, "price": 0, "duration": 0, "stops": 0, "airline": "string" }]</pre>	
503	Error the server is currently unavailable

2.2.3.4 Endpoint πόρου supplier (/search/result/{resultID}/supplier/{supplierID})

GET /search/result/{resultID}/supplier/{supplierID} Evaluate a supplier

A user must be able to see the evaluation of a supplier

Parameters Cancel

Name	Description
resultID * required	specific number of a result
integer (path)	<input type="text" value="6463"/>
supplierID * required	specific number of supplier
integer(\$int32) (path)	<input type="text" value="0982"/>

Execute Clear



Responses Response content type application/json

Curl

```
curl -X GET
"https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/6463/supplier/0982" -
H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/search/result/6463/supplier/0982
```

Server response

Code	Details
200	<div>Response body<pre>{ "supplierID": 0, "name": "string", "evaluation": 1.5, "url": "string", "evaluationExistence": true }</pre>Download</div> <div>Response headers<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: 96 content-type: application/json; charset=utf-8 date: Mon, 08 Jun 2020 16:08:18 GMT etag: W/"60-nZTr25XiXkvpmoyljvJODswqcME" expires: -1 server: nginx/1.14.0 + Phusion Passenger 5.3.4 status: 200, 200 OK via: 1.1 a642bb64a88fc6131a3257b53be23f26.cloudfront.net (CloudFront) x-amz-cf-id: cwrHH5IQImJAVHqK_kwLhYyFqyc_cFjY-4ht4quUT0j01-6Xkbtrdw== x-amz-cf-pop: ATH50-C1 x-cache: Miss from cloudfront x-powered-by: Express, Phusion Passenger 5.3.4</pre></div> <div>Request duration<pre>704 ms</pre></div>



Responses	
Code	Description
200	successful operation
Example Value Model	
<pre>{ "supplierID": 0, "name": "string", "evaluation": 0, "url": "string", "evaluationExistence": true }</pre>	
404	Evaluation not found

2.2.4 Πόρος Travel

2.2.4.1 Μοντέλο δεδομένων travel

```
Travel {
  date          string
  minBudget     integer
  maxBudget     integer
}
```

2.2.4.2 Endpoint πόρου supplier (/travel/search/{date})

travel Operations about travel results

GET `/travel/search/{date}` Search trip

A user must be able to search trip within a budget

Parameters Cancel

Name	Description
date * required	date
string (path)	<input type="text" value="3-8-2020"/>
minBudget * required	Min budget
integer(\$int32) (query)	<input type="text" value="50"/>



maxBudget * required
integer(\$int32)
(query)

Max budget

Execute

Clear

Responses

Response content type

application/json

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/travel/search/3-8-2020?minBudget=50&maxBudget=350" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/travel/search/3-8-2020?minBudget=50&maxBudget=350
```

Server response

Code	Details
404	<div>Error:</div> <div>Response body</div> <div>Download file</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-length: 0 content-type: application/octet-stream date: Mon, 08 Jun 2020 16:19:12 GMT etag: W/"0-2jmj7l5rSw0yVb/vlWAYkK/YBwk" expires: -1 server: nginx/1.14.0 + Phusion Passenger 5.3.4 status: 404, 404 Not Found via: 1.1 ede741ecd9db6e9b69f0c722dcde4f3.cloudfront.net (CloudFront) x-amz-cf-id: -bh2HahPcQXEVI5dFDpVELy0IxtZu4TwMEyhMjXudBWhkQZOF8ebqQ== x-amz-cf-pop: ATH50-C1 x-cache: Error from cloudfront x-powered-by: Express, Phusion Passenger 5.3.4</pre></div>



Responses	
Code	Description
200	successful operation
Example Value Model	
<pre>[{ "resultID": 0, "Flight": { "flight": 0, "date": "string", "startLocation": "string", "endLocation": "string", "numberOfTravellers": 0, "cabinClass": "string" }, "price": 0, "duration": 0, "stops": 0, "airline": "string" }]</pre>	
404	Travel not found

2.2.5 Πόρος User

2.2.5.1 Μοντέλο δεδομένων user

```
User {
  username      string
  firstName     string
  lastName      string
  email         string
  password      string
  phone         string
}
```

2.2.5.2 Endpoint πόρου user (/user/login)

user Simple user accounts

PUT **/user/login** Logs user into the system

A user must be able to login to his/her account by providing his/her username and password

Parameters

Cancel



Name	Description
username * required	The username for login
string (query)	<input type="text" value="kwstas"/>
password * required	The password for login
object (body)	Edit Value Model
	<pre>{ "username": "string", "firstName": "string", "lastName": "string", "email": "string", "password": "string", "phone": "string" }</pre>
<div>Cancel</div>	
Parameter content type	
<div>application/json</div>	
<div>Execute</div> <div>Clear</div>	
Responses	
Response content type <div>application/json</div>	
Curl	
<pre>curl -X PUT "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/login?username=kwstas" -H "accept: application/json" -H "Content-Type: application/json" -d "{ \"username\": \"string\", \"firstName\": \"string\", \"lastName\": \"string\", \"email\": \"string\", \"password\": \"string\", \"phone\": \"string\"}"</pre>	
Request URL	
<pre>https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/login?username=kwstas</pre>	



Request duration

1176 ms

Responses

Code	Description	
200	successful operation	
Example Value Model		
string		
Headers:		
Name	Description	Type
X-Rate-Limit	calls per hour allowed by the user	integer
X-Expires-After	date in UTC when token expires	string
400	Invalid username/password supplied	

2.2.5.3 Endpoint πύρου user (/user/{username})

GET /user/{username} Get user's information

A user must be able to see his information by providing his username

Parameters Cancel

Name	Description
username * required	User to return
string (path)	<input type="text" value="kwstas"/>

Execute Clear



Responses Response content type **application/json** ▼

Curl

```
curl -X GET "https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/kwstas" -H "accept: application/json"
```

Request URL

```
https://virtserver.swaggerhub.com/haritini_koutra/ether/1.0.0/user/kwstas
```

Server response

Code	Details
200	<div>Response body<pre>{ "username": "string", "firstName": "string", "lastName": "string", "email": "string", "password": "string", "phone": "string"}</pre>Download</div> <div>Response headers<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: 84 content-type: application/json; charset=utf-8 date: Mon, 08 Jun 2020 16:27:02 GMT etag: W/"54-p1Id7B8D2kI+sLWMoFJbeMo8fLE" expires: -1 server: nginx/1.14.0 + Phusion Passenger 5.3.4 status: 200, 200 OK via: 1.1 bb8a874d65e0b595aaa3d9aa3f930102.cloudfront.net (CloudFront) x-amz-cf-id: BLERg3yCEvog8VEHW3HGMMo_hXafhM820jcMYH3jnwZeGcqjJFS5Qw== x-amz-cf-pop: ATH50-C1 x-cache: Miss from cloudfront x-powered-by: Express, Phusion Passenger 5.3.4</pre></div> <div>Request duration<pre>516 ms</pre></div>



Responses

Code	Description
------	-------------

200	successful operation
-----	----------------------

Example Value | Model

```
{
  "username": "string",
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "phone": "string"
}
```

400	Invalid username supplied
-----	---------------------------

404	User not found
-----	----------------



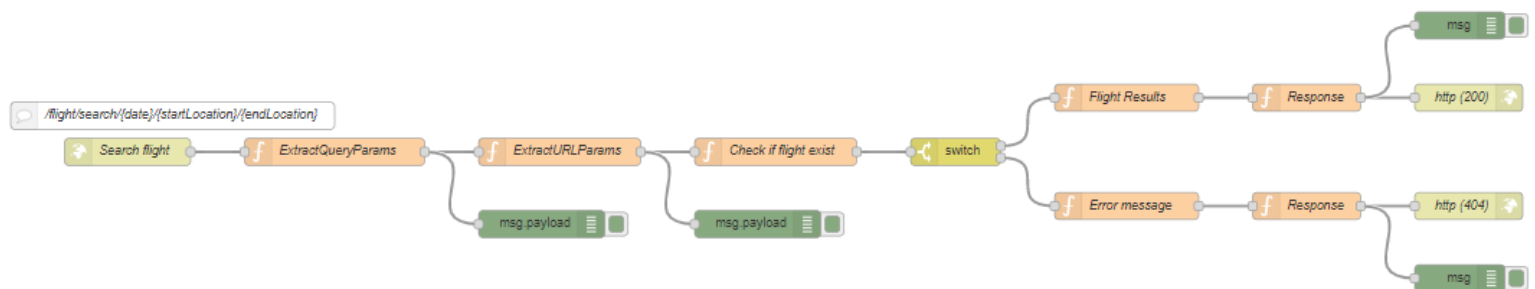
3. ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ ΜΕ NODE-RED

3.1 Αντιστοίχιση των REST υπηρεσιών σε ροές Node-RED

Link: <https://www.dropbox.com/s/la75kpbxl4l1b41/flows%20%281%29.json?dl=0>

3.1.1 Ροές Πόρου Flight

Ροή endpoint get /flight/search/{date}/{startLocation}/{endLocation}

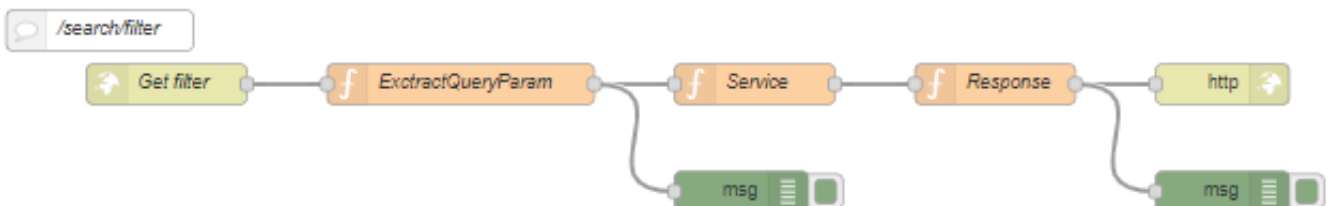


Σχήμα 9. Ροή Πόρου Flight

Η παραπάνω ροή υλοποιεί την υπηρεσία Get-evaluation η οποία δέχεται ως path παραμέτρους (μέσω της συνάρτησης ExtractURLParam) τον μοναδικό χαρακτηριστικό αριθμό ενός αποτελέσματος (resultID) και τον μοναδικό χαρακτηριστικό αριθμό του παρόχου (supplierID). Με τη βοήθεια της συνάρτησης *Check evaluation existence* γίνεται ο έλεγχος για την ύπαρξη των αξιολογήσεων. Συνεπώς η ροή αυτή υλοποιεί και happy path (ύπαρξη αξιολόγησης) με response 200 και unhappy path (μη διαθέσιμη αξιολόγηση) με response 404.

3.1.2 Ροές Πόρου Search

Ροή endpoint get /search/filter

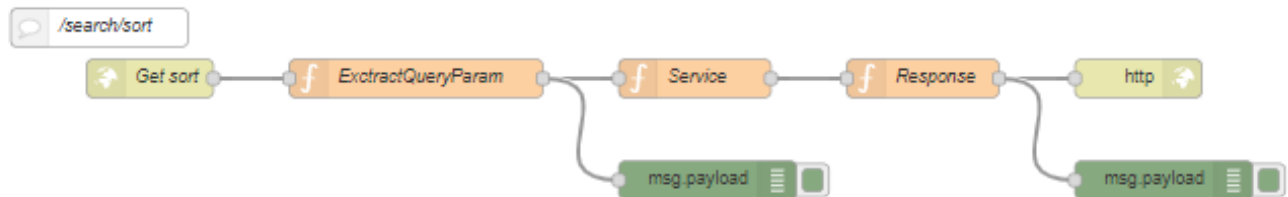


Σχήμα 10. Ροή Πόρου Search



Η παραπάνω ροή υλοποιεί την υπηρεσία Get-filter η οποία δέχεται ως query παράμετρο (μέσω της συνάρτησης ExtractQueryParam) το επιθυμητό φίλτρο και το επιστρέφει. Η ροή αυτή υλοποιεί μόνο happy path με response 200.

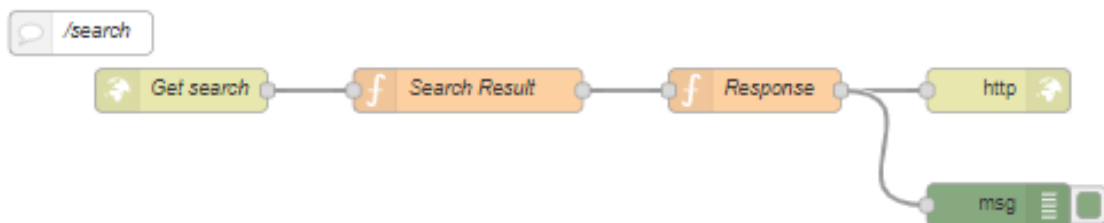
Ροή endpoint get /search/sort



Σχήμα 11. Ροή Πόρου Search

Η παραπάνω ροή υλοποιεί την την υπηρεσία Get-sort η οποία δέχεται ως query (μέσω της συνάρτησης ExtractqueryParam) παράμετρο την επιθυμητή ταξινόμηση και την επιστρέφει. Η ροή αυτή υλοποιεί μόνο happy path με response 200.

Ροή endpoint get /search

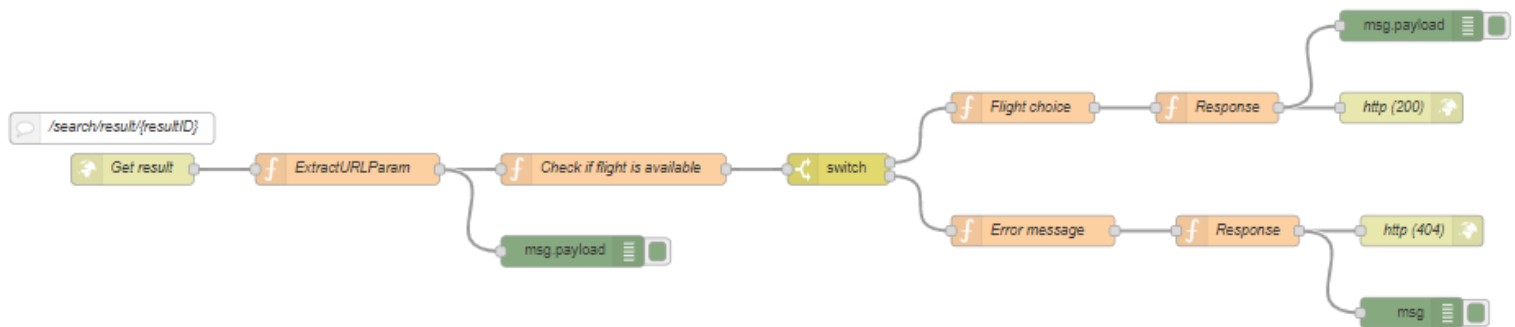


Σχήμα 12. Ροή Πόρου Search

Η παραπάνω ροή υλοποιεί την υπηρεσία Get-search η οποία επιστρέφει μία αναζήτηση .Η ροή αυτή υλοποιεί μόνο happy path με response 200.



Ροή endpoint get /search/result/resultID

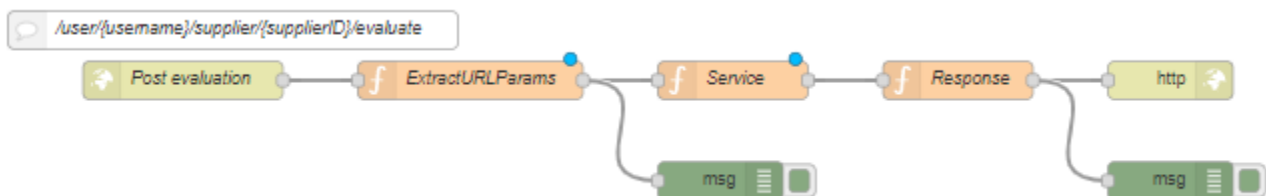


Σχήμα 13. Ροή Πόρου Search

Η παραπάνω ροή υλοποιεί την υπηρεσία Get-Result, η οποία δέχεται ως path παραμέτρους (μέσω της συνάρτησης ExtractURLParam) την συγκεκριμένη πτήση που επέλεξε ο χρήστης. Με την χρήση της συνάρτησης *Check if flight is available*, γίνεται ο έλεγχος για την διαθεσιμότητα της πτήσης. Η ροή υλοποιεί το happy path σε περίπτωση που υπάρχει η πτήση, ενώ υλοποιεί και το unhappy path σε περίπτωση που η πτήση δεν υπάρχει.

3.1.3 Ροές Πόρου Supplier

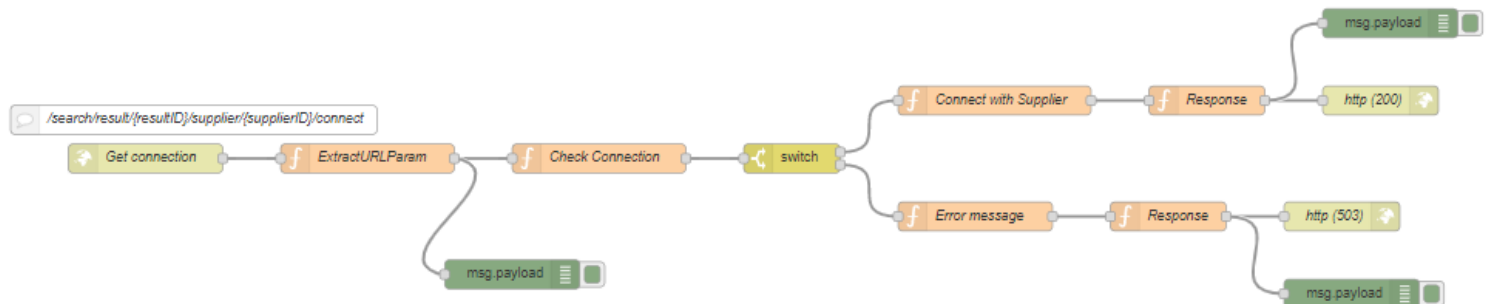
Ροή endpoint post user/{username}/supplier/{supplierID}/evaluate



Σχήμα 14. Ροή Πόρου Supplier

Η παραπάνω ροή υλοποιεί την υπηρεσία Post evaluation στην οποία όταν δέχεται μία νέα αξιολόγηση και επιπλέον ως path παραμέτρους το username του εγγεγραμμένου χρήστη και το supplierID (μέσω της συνάρτησης ExtractURLParams) και επιστρέφει την νέα αξιολόγηση καθώς επίσης τον μοναδικό αριθμό που περιγράφει τον κάθε πάροχο. Τέλος, αυτή η ροή έχει μόνο happy path με response 200.

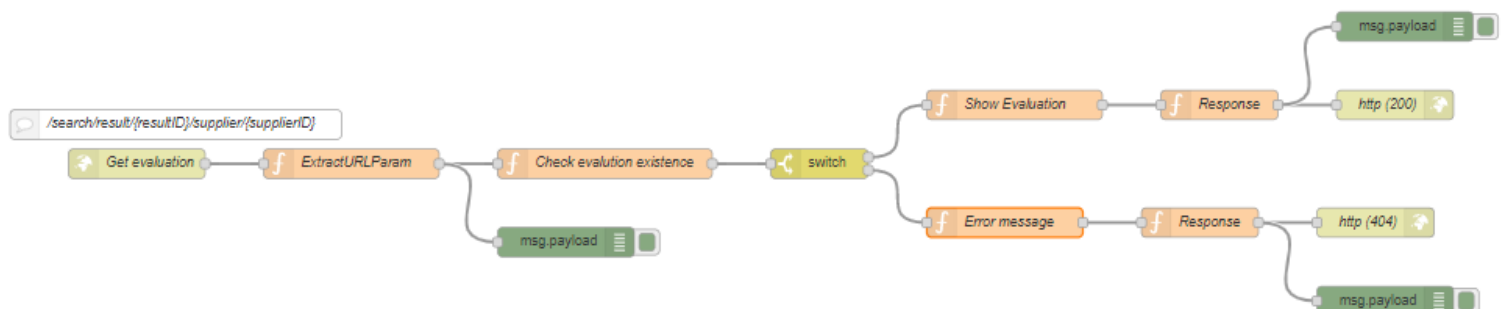
Ροή endpoint get search/result/{resultID}/supplier/{supplierID}/connect



Σχήμα 15. Ροή Πόρου Supplier

Η παραπάνω ροή υλοποιεί την υπηρεσία Get-connection, η οποία δέχεται ως path παραμέτρους (μέσω της συνάρτησης ExtractURLParam) το resultID που είναι ο χαρακτηριστικός αριθμός της πτήσης και το supplierID που είναι ο χαρακτηριστικός αριθμός του παρόχου. Με την χρήση της συνάρτησης Check Connection, γίνεται ο έλεγχος για το αν η σύνδεση πραγματοποιείται με επιτυχία. Η ροή υλοποιεί το happy path σε περίπτωση που πραγματοποιείται η σύνδεση (response 200), ενώ υλοποιεί και το unhappy path σε περίπτωση που η σύνδεση δεν μπορεί να πραγματοποιηθεί (response 503).

Ροή endpoint get search/result/{resultID}/supplier/{supplierID}



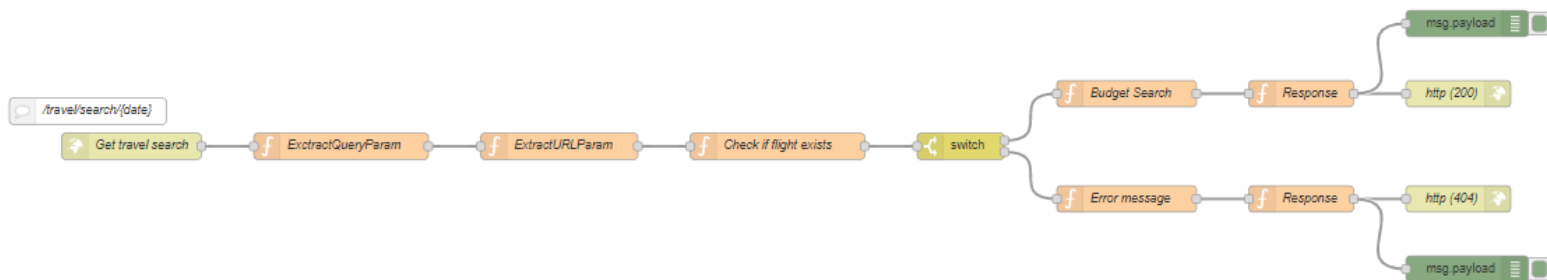
Σχήμα 16. Ροή Πόρου Supplier

Η παραπάνω ροή υλοποιεί την υπηρεσία Get-evaluation η οποία δέχεται ως path παραμέτρους (μέσω της συνάρτησης ExtractURLParam) τον μοναδικό χαρακτηριστικό αριθμό ενός αποτελέσματος (resultID) και τον μοναδικό χαρακτηριστικό αριθμό του παρόχου (supplierID). Με τη βοήθεια της συνάρτησης Check evaluation existence γίνεται ο έλεγχος για την ύπαρξη των αξιολογήσεων. Συνεπώς η ροή αυτή υλοποιεί και happy path (ύπαρξη αξιολόγησης) με response 200 και unhappy path (μη διαθέσιμη αξιολόγηση) με response 404.



3.1.4 Ροές Πόρου Travel

Ροή endpoint post user/{username}/supplier/{supplierID}/evaluate



Σχήμα 17. Ροή Πόρου Supplier

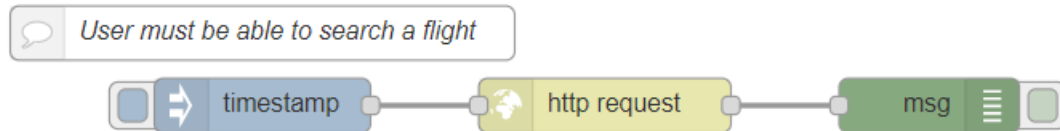
Η παραπάνω ροή υπολοιεί την υπηρεσία Get travel search η οποία δέχεται ως URL παραμέτρους (μέσω της συνάρτησης ExtractURLParams) την ημερομηνία (date) και ως query παραμέτρους (μέσω της συνάρτησης ExtractQueryParams) το minbudget και maxbudget. Μέσω της συνάρτησης Check if flight exist γίνεται ο έλεγχος για την ύπαρξη πτήσεων με τα συγκεκριμένα χαρακτηριστικά. Για το happy path(200) επιστρέφει ένα πίνακα με πτήσεις, ενώ υλοποιεί unhappy path(404) σε περίπτωση που δεν υπάρχουν πτήσεις με τα συγκεκριμένα χαρακτηριστικά.



3.2 Υλοποίηση Ιστοριών χρήστη

3.2.1 Ιστορία Χρήστη 1

<Ο χρήστης πρέπει να μπορεί να αναζητεί μία πτήση>



Σχήμα 18. Ροή για Ιστορία χρήστη 1

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να αναζητήσει μια πτήση. Αυτό επιτυγχάνεται με κλήση προς την υπηρεσία με `[uri:/flight/search/{date}/{startLocation}/{endLocation}]` και στη συνέχεια εκτυπώνει τα αποτελέσματα που φέρουν τα χαρακτηριστικά που δόθηκαν από τον χρήστη.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει κλήση προς την υπηρεσία <code>/flight/search/{date}/{startLocation}/{endLocation}</code> η οποία επιστρέφει μια λίστα από πτήσεις.
msg	debug	Τυπώνει στην κονσόλα την λίστα με τις πτήσεις.

3.2.2 Ιστορία Χρήστη 2

<Ο χρήστης πρέπει να μπορεί να επιλέγει την ταξινόμηση των αποτελεσμάτων>



Σχήμα 19. Ροή για Ιστορία χρήστη 3

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να επιλέξει την επιθυμητή ταξινόμηση των αποτελεσμάτων. Συγκεκριμένα υλοποιείται το τμήμα της εφαρμογής, στο οποίο γίνεται κλήση προς την υπηρεσία με `[uri: / search/sort]` και στην συνέχεια εφαρμόζεται ταξινόμηση στα δεδομένα με βάση την ταξινόμηση που επιλέχθηκε. Το αποτέλεσμα της εκτέλεσης του παραπάνω τμήματος θα πρέπει να είναι η εκτύπωση της ταξινομημένης λίστας των αποτελεσμάτων.



Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει κλήση προς την υπηρεσία /search/sort η οποία επιστρέφει την επιλογή της ταξινόμησης
Assortment Selection and Sort	function	Εφαρμόζεται ταξινόμηση στα δεδομένα (λίστα πτήσεων) και επιστρέφει την λίστα με τις ταξινομημένες πτήσεις
msg	debug	Τυπώνει στην κονσόλα την λίστα από τις ταξινομημένες πτήσεις

3.2.3 Ιστορία Χρήστη 3

< Ο χρήστης πρέπει να μπορεί να επιλέξει φίλτρα για τα αποτελέσματα>



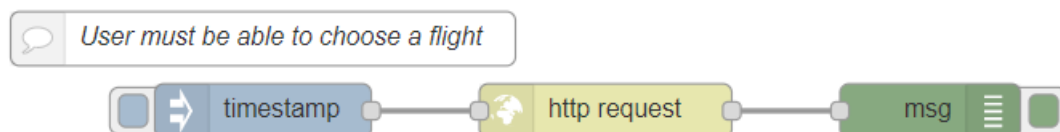
Σχήμα 20. Ροή για Ιστορία χρήστη 3

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να επιλέξει τα επιθυμητά φίλτρα των αποτελεσμάτων. Αυτό επιτυγχάνεται με κλήση προς την υπηρεσία με [uri:/search/filter]. Το αποτέλεσμα της παραπάνω κλήσης θα πρέπει να είναι η εκτύπωση των αποτελεσμάτων (πτήσεις) με τα φίλτρα που εφαρμόστηκαν.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει κλήση προς την υπηρεσία /search/filter η οποία επιστρέφει την επιλογή φίλτρου του χρήστη στο σύστημα.
Filter selection and Filter	function	Εφαρμόζει φιλτράρισμα στα δεδομένα (λίστα πτήσεων) και επιστρέφει τη λίστα πτήσεων με τα φίλτρα που εφαρμόστηκαν.
msg	debug	Τυπώνει στην κονσόλα τη λίστα πτήσεων με τα επιλεγμένα φίλτρα.

3.2.4 Ιστορία Χρήστη 4

< Ο χρήστης πρέπει να μπορεί να επιλέξει μια πτήση>



Σχήμα 21. Ροή για Ιστορία χρήστη 4

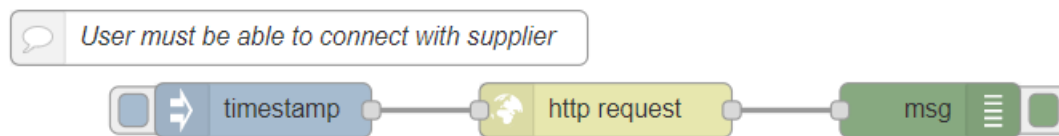


Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να επιλέξει την πτήση που επιθυμεί, αυτό επιτυγχάνεται με κλήση προς την υπηρεσία [uri: /search/result/{resultID}]. Το αποτέλεσμα της εκτέλεσης του παραπάνω τμήματος θα πρέπει να είναι η εκτύπωση των χαρακτηριστικών της πτήσης που επιλέχθηκε από τον χρήστη.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει την κλήση προς την υπηρεσία: /search/result/{resultID} η οποία επιστρέφει την πτήση που επιλέχθηκε.
msg	debug	Τυπώνει στην κονσόλα την πτήση που επιλέχθηκε.

3.2.5 Ιστορία Χρήστη 5

< Ο χρήστης πρέπει να μπορεί να συνδεθεί με τον πάροχο των εισιτηρίων που επέλεξε>



Σχήμα 22. Ροή για Ιστορία χρήστη 5

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να συνδεθεί με τον πάροχο εισιτηρίων που επέλεξε, αυτό επιτυγχάνεται με κλήση προς την υπηρεσία [uri: /search/result/{resultID}/supplier/{supplierID}/connect]. Το αποτέλεσμα της εκτέλεσης του παραπάνω τμήματος θα πρέπει να είναι η εκτύπωση του url του παρόχου των εισιτηρίων.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει την κλήση προς την υπηρεσία: /search/result/{resultID}/supplier/{supplierID}/connect Το οποίο επιστρέφει το URL για την σύνδεση με το πάροχο του εισιτηρίου.
msg	debug	Τυπώνει στην κονσόλα το URL του παρόχου.

3.2.7 Ιστορία Χρήστη 6

< Ο εγγεγραμμένος χρήστης πρέπει να μπορεί να αξιολογήσει τον πάροχο των εισιτηρίων>



Σχήμα 23. Ροή για Ιστορία χρήστη 6

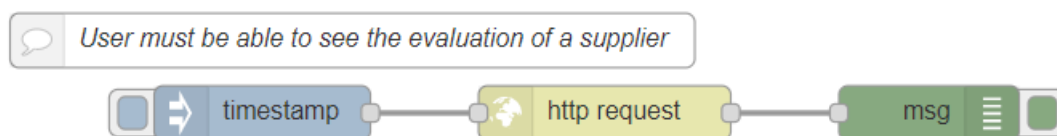


Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Post Evaluation	function	Προσθέτει την νέα αξιολόγηση του παρόχου
http request	http-request	Κάνει κλήση προς την υπηρεσία /user/{username}/supplier/{supplierID}/evaluate η οποία επιστρέφει την αξιολόγηση που δίνει ο χρήστης
Update evaluation	function	Εφαρμόζεται ενημέρωση στην ήδη υπάρχουσα αξιολόγηση του παρόχου
msg	debug	Τυπώνει στην κονσόλα τα στοιχεία του παρόχου με την νέα αξιολόγηση του

Σε αυτή τη ροή υλοποιείται η δυνατότητα του εγγεγραμμένου χρήστη να αξιολογεί έναν πάροχο. Υλοποιείται το τμήμα της εφαρμογής, στο οποίο μετά την προσθήκη της νέας αξιολόγησης γίνεται κλήση προς την υπηρεσία με `[uri: /user/{username}/supplier/{supplierID}/evaluate]` και στην συνέχεια ενημερώνεται η ήδη υπάρχουσα αξιολόγηση του παρόχου. Το αποτέλεσμα είναι η εκτύπωση των στοιχείων του παρόχου με την νέα αξιολόγηση.

3.2.7 Ιστορία Χρήστη 7

< Ο χρήστης πρέπει να μπορεί να δει την αξιολόγηση των παρόχων.>



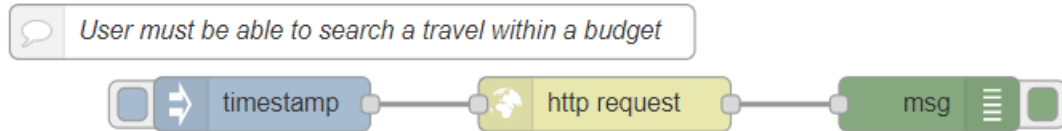
Σχήμα 24. Ροή για Ιστορία χρήστη 7

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να δει τις αξιολογήσεις του παρόχου. Αυτό επιτυγχάνεται με κλήση προς την υπηρεσία με `[uri:/search/result/{resultID}/supplier/{supplierID}]`. Το αποτέλεσμα της παραπάνω κλήσης θα πρέπει να είναι η εκτύπωση των αξιολογήσεων του παρόχου.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει κλήση προς την υπηρεσία /search/result/{resultID}/supplier/{supplierID} η οποία επιστρέφει τη λίστα με τις αξιολογήσεις του παρόχου.
msg	debug	Τυπώνει στην κονσόλα τη λίστα με τις αξιολογήσεις του παρόχου.

3.2.8 Ιστορία Χρήστη 8

< Ο χρήστης θα έπρεπε να μπορεί να αναζητήσει ταξίδια αναλόγως το budget.>



Σχήμα 25. Ροή για Ιστορία χρήστη 8

Σε αυτή τη ροή υλοποιείται η δυνατότητα του χρήστη να αναζητήσει μια πτήση μέσα σε ένα συγκεκριμένο budget. Αυτό επιτυγχάνεται με κλήση προς την υπηρεσία με [uri:/travel/search/{date}] και στη συνέχεια εκτυπώνει τα αποτελέσματα που βρησκονται μέσα στο συγκεκριμένο budget που δόθηκε από τον χρήστη.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Timestamp	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
http request	http-request	Κάνει κλήση προς την υπηρεσία /travel/search/{date} η οποία επιστρέφει μια λίστα από πτήσεις.
msg	debug	Τυπώνει στην κονσόλα την λίστα με τις πτήσεις.



● ΠΑΡΑΡΤΗΜΑ Ι – ΑΝΟΙΧΤΑ ΘΕΜΑΤΑ

Στο δεύτερο μέρος της εργασίας υλοποιείται ο πόρος User ο οποίος δεν ανήκει στις απαιτήσεις του συστήματος μας, αλλά πραγματοποιήθηκε για λόγους πληρότητας.