



Μηχανική Λογισμικού I
Τομέας Ηλεκτρονικής και Υπολογιστών
Τμήμα HMMY
Α.Π.Θ

8^ο Εξάμηνο
Άνοιξη 2023



Embrace change with GYMFUELX
The ultimate shape changing experience

Προδιαγραφές Λογισμικού και Υλοποίηση συστήματος

Version 2.0

Χρυσόστομος Κουμίδης chrysostk@ece.auth.gr

Δήμος Κομπιτσελίδης kompitse@ece.auth.gr

Ανδρέας-Ευστάθιος Ελευθεριάδης andreassee@ece.auth.gr

Αντρέας Θεοδώρου theoantr@ece.auth.gr

Ημερομηνία: 12/6/2023



Μέλη της Ομάδας Ανάπτυξης

Όνομα	OA	Email
Χρυσόστομος Κουμίδης	27	chrysostk@ece.auth.gr
Αντρέας Θεοδώρου	27	theoantr@ece.auth.gr
Δήμος Κομπιτσελίδης	27	kompitse@ece.auth.gr
Ανδρέας-Ευστάθιος Ελευθεριάδης	27	andreassee@ece.auth.gr



Πίνακας Περιεχομένων

Πίνακας Περιεχομένων	3
Λίστα Σχημάτων.....	5
1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν.....	8
1.1 Πρότυπο Strategy	8
1.2 Πρότυπο Observer	9
2. Αρχιτεκτονική Συστήματος.....	10
2.1. Αναγνώριση Πόρων (Resources) Συστήματος	10
2.2. Τεκμηρίωση REST Διεπαφής.....	11
2.2.1. Πόρος User	11
2.2.2. Πόρος Recipe	15
2.2.3. Πόρος Report	22
2.2.4. Πόρος GymProgram.....	31
2.2.5. Πόρος Exercise	32
2.2.6. Πόρος DietProgram	34
3. Υλοποίηση Συστήματος με Node-RED	36
3.1. Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed	36
3.1.1. Ροές πόρου User	36
3.1.2. Ροές πόρου Recipe	37
3.1.3. Ροές πόρου Report.....	39
3.1.4. Ροές πόρου GymProgram.....	42
3.1.5. Ροές πόρου Exercise	42
3.1.6. Ροές πόρου DietProgram	43
3.2. Υλοποίηση Ιστοριών χρήστη.....	44
3.2.1. Ιστορία User “View profile details”	44
3.2.2. Ιστορία User “Input profile details”	44
3.2.3. Ιστορία User “Update profile details”	45
3.2.4. Ιστορία User “View Recipes”.....	45
3.2.5. Ιστορία User “View Recipe Information”.....	46
3.2.6. Ιστορία User “View Gym program”.....	46
3.2.7. Ιστορία User “View Exercise”	47
3.2.8. Ιστορία User “View Diet program”	47
3.2.9. Ιστορία User “Report Gym Program”	48
3.2.10. Ιστορία User “Report Recipe”	48
3.2.11. Ιστορία Nutritionist “View All Recipe Reports”	49
3.2.12. Ιστορία Nutritionist “View Recipe Report”	49
3.2.13. Ιστορία Nutritionist “Edit Recipe Report”	50
3.2.14. Ιστορία Nutritionist “Delete Recipe Report”	50
3.2.15. Ιστορία Nutritionist “View Recipe”	51
3.2.16. Ιστορία Nutritionist “Add Recipe”	51



3.2.17.	Ιστορία Nutritionist “Update Recipe”	52
3.2.18.	Ιστορία Nutritionist “Delete Recipe”	52
3.2.19.	Ιστορία Trainer “View All Gym Program Reports”	53
3.2.20.	Ιστορία Trainer “View Gym Program Report”	53
3.2.21.	Ιστορία Trainer “Edit Gym Program Report”	54
3.2.22.	Ιστορία Trainer “Delete Gym Program Report”	54
	Παράρτημα I – Ανοιχτά Θέματα.....	55



Λίστα Σχημάτων

Εικόνα 1: Εφαρμογή προτύπου Strategy	8
Εικόνα 2: Εφαρμογή προτύπου Observer	9
Εικόνα 3: Μοντέλο δεδομένων User	11
Εικόνα 4: Εξήγηση και παράθεση των παραμέτρων του πόρου και δοκιμή του endpoint	12
Εικόνα 5: Δοκιμή του endpoint Εικόνα 6: Αποκρίσεις του συστήματος	12
Εικόνα 7: Εξήγηση και παράθεση των παραμέτρων του πόρου. Εικόνα 8: Παράθεση των παραμέτρων του πόρου	12
Εικόνα 9: Δοκιμή του endpoint Εικόνα 10: Δοκιμή του endpoint	13
Εικόνα 11: Αποκρίσεις του συστήματος	13
Εικόνα 12: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 13: Δοκιμή του endpoint	14
Εικόνα 14: Δοκιμή του endpoint Εικόνα 15: Αποκρίσεις του συστήματος	14
Εικόνα 16: Μοντέλο δεδομένων Recipe	15
Εικόνα 17: Εξήγηση και παράθεση των παραμέτρων του πόρου και Δοκιμή του endpoint Εικόνα 18: Δοκιμή του endpoint	15
Εικόνα 19: Αποκρίσεις του συστήματος	16
Εικόνα 20: Εξήγηση και παράθεση των παραμέτρων και δοκιμή του endpoint Εικόνα 21: Δοκιμή του endpoint	16
Εικόνα 22: Αποκρίσεις του συστήματος	17
Εικόνα 23: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 24: Παράθεση των παραμέτρων του πόρου	17
Εικόνα 25: Παράθεση των παραμέτρων του πόρου και δοκιμή του endpoint	18
Εικόνα 26: Δοκιμή του endpoint	18
Εικόνα 27: Αποκρίσεις του συστήματος	18
Εικόνα 28: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 29: Δοκιμή του endpoint	19
Εικόνα 30: Αποκρίσεις του συστήματος Εικόνα 31: Αποκρίσεις του συστήματος	19
Εικόνα 32: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 33 : Παράθεση των παραμέτρων του πόρου	20
Εικόνα 34: Δοκιμή του endpoint του πόρου Εικόνα 35: Αποκρίσεις του συστήματος	20
Εικόνα 36: Δοκιμή του endpoint Εικόνα 37: Αποκρίσεις του συστήματος	21
Εικόνα 38: Μοντέλο δεδομένων Report	22
Εικόνα 39: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 40: Δοκιμή του endpoint του πόρου και αποκρίσεις του συστήματος	22
Εικόνα 41: Εξήγηση και παράθεση των παραμέτρων του πόρου	23
Εικόνα 42: Δοκιμή του endpoint του πόρου και αποκρίσεις του συστήματος	23
Εικόνα 43: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 44: Δοκιμή του endpoint	23
Εικόνα 45: Αποκρίσεις του συστήματος	24
Εικόνα 46: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 47: Δοκιμή του endpoint	24
Εικόνα 48: Αποκρίσεις του συστήματος	25
Εικόνα 49: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 50: Δοκιμή του endpoint	26
Εικόνα 51: Αποκρίσεις του συστήματος	26
Εικόνα 52: Εξήγηση και παράθεση των παραμέτρων του πόρου	27
Εικόνα 53: Δοκιμή του endpoint και αποκρίσεις του συστήματος	27
Εικόνα 54: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 55: Δοκιμή του endpoint	27
Εικόνα 56: Αποκρίσεις του συστήματος	28
Εικόνα 57: Εξήγηση και παράθεση των παραμέτρων του πόρου Εικόνα 58: Δοκιμή του endpoint	28



Εικόνα 59: Αποκρίσεις του συστήματος	29	
Εικόνα 60: Εξήγηση και παράθεση των παραμέτρων του πόρου	Εικόνα 61: Δοκιμή του endpoint	29
Εικόνα 62: Αποκρίσεις του συστήματος	29	
Εικόνα 63: Εξήγηση και παράθεση των παραμέτρων του πόρου	Εικόνα 64: Δοκιμή του endpoint	30
Εικόνα 65: Αποκρίσεις του συστήματος	30	
Εικόνα 66: Μοντέλο δεδομένων GymProgram.....	31	
Εικόνα 67: Εξήγηση και παράθεση των παραμέτρων του πόρου	Εικόνα 68: Δοκιμή του endpoint.....	31
Εικόνα 69: Αποκρίσεις του συστήματος	32	
Εικόνα 70: Μοντέλο δεδομένων Exercise.....	32	
Εικόνα 71: Εξήγηση και παράθεση των παραμέτρων του πόρου	Εικόνα 72: Δοκιμή του endpoint	33
Εικόνα 73: Αποκρίσεις του συστήματος	33	
Εικόνα 74: Μοντέλο δεδομένων DietProgram.....	34	
Εικόνα 75: Εξήγηση και παράθεση των παραμέτρων του πόρου	Εικόνα 76: Δοκιμή του endpoint	34
Εικόνα 77: Αποκρίσεις του συστήματος	35	
Εικόνα 78: Ροή NodeRed που αντιστοιχεί στο API call Get /user/:userID.....	36	
Εικόνα 79: Ροή NodeRed που αντιστοιχεί στο API call POST /user/:userID.....	36	
Εικόνα 80: Ροή NodeRed που αντιστοιχεί στο API call PUT /user/:userID.....	36	
Εικόνα 81: Ροή NodeRed που αντιστοιχεί στο API call Get /user/:userID/recipe	37	
Εικόνα 82: Ροή NodeRed που αντιστοιχεί στο API call Get /user/:userID/recipe/:recipeID	37	
Εικόνα 83: Ροή NodeRed που αντιστοιχεί στο API call POST /nutritionist/:nutritionistID/recipe.....	37	
Εικόνα 84: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/recipe/:recipelD.38	38	
Εικόνα 85: Ροή NodeRed που αντιστοιχεί στο API call PUT /nutritionist/:nutritionistID/recipe/:recipelD.38		
Εικόνα 86: Ροή NodeRed που αντιστοιχεί στο API call DELETE/nutritionist/:nutritionistID/recipe/:recipelD	38	
Εικόνα 87: Ροή NodeRed που αντιστοιχεί στο API call POST /user/:userID/gymprogram.....	39	
Εικόνα 88: Ροή NodeRed που αντιστοιχεί στο API call POST /user/:userID/recipe/:recipeID	39	
Εικόνα 89: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/report	39	
Εικόνα 90: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/report/:reportID 40		
Εικόνα 91: Ροή NodeRed που αντιστοιχεί στο API call PUT /nutritionist/:nutritionistID/report/:reportID 40		
Εικόνα 92: Ροή NodeRed που αντιστοιχεί στο API call DELETE/nutritionist/:nutritionistID/report/:reportID	40	
Εικόνα 93: Ροή NodeRed που αντιστοιχεί στο API call GET /trainer/:trainerID/report.....	41	
Εικόνα 94: Ροή NodeRed που αντιστοιχεί στο API call GET /trainer/:trainerID/report/:reportID	41	
Εικόνα 95: Ροή NodeRed που αντιστοιχεί στο API call PUT /trainer/:trainerID/report/:reportID	41	
Εικόνα 96: Ροή NodeRed που αντιστοιχεί στο API call DELETE /trainer/:trainerID/report/:reportID.....	42	
Εικόνα 97: Ροή NodeRed που αντιστοιχεί στο API call GET /user/:userID/gymprogram	42	
Εικόνα 98: Ροή NodeRed που αντιστοιχεί στο API call GET /user/:userID/gymprogram/:excerciseID	42	
Εικόνα 99: Ροή NodeRed που αντιστοιχεί στο API call GET /user/:userID/dietprogram.....	43	
Εικόνα 100:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Profile Details”.....	44	
Εικόνα 101:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Input Profile Details”.....	44	
Εικόνα 102:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Update Profile Details”.....	45	
Εικόνα 103:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipes”.....	45	
Εικόνα 104:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipe Information”.....	46	
Εικόνα 105:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Gym Program”.....	46	
Εικόνα 106:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Exercise”.....	47	
Εικόνα 107:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Diet Program”.....	47	
Εικόνα 108:Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Report Gym Program”.....	48	



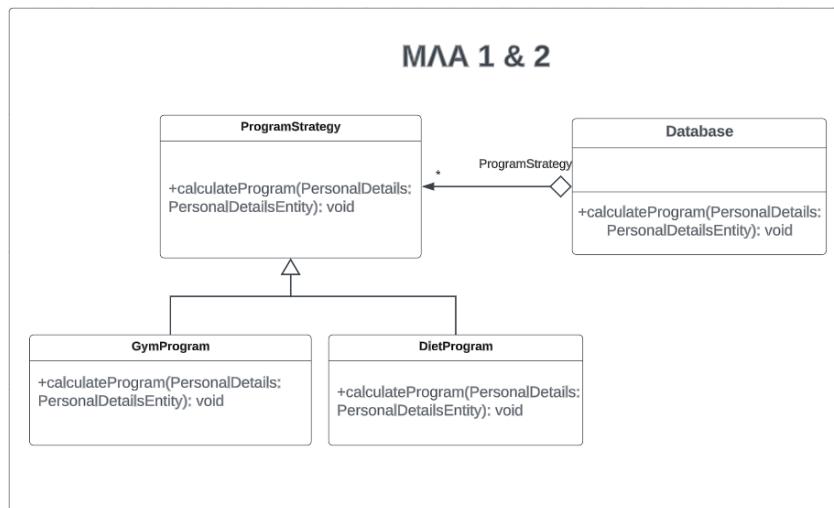
Εικόνα 109: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Report Recipe”	48
Εικόνα 110: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View All Recipe Reports”	49
Εικόνα 111: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipe Report”	49
Εικόνα 112 Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Edit Recipe Reports”	50
Εικόνα 113: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Recipe Reports”	50
Εικόνα 114: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Nutritionist View Recipe”	51
Εικόνα 115: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Add Recipe”	51
Εικόνα 116: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Update Recipe”	52
Εικόνα 117: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Recipe”	52
Εικόνα 118: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View All Gym Program Reports”	53
Εικόνα 119: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Gym Program Report”	53
Εικόνα 120: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Edit Gym Program Report”	54
Εικόνα 121: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Gym Program Report”	54



1. Πρότυπα Σχεδιασμού που υιοθετήθηκαν

1.1 Πρότυπο Strategy

Το πρότυπο **strategy** είναι ένα πρότυπο συμπεριφοράς. Στο σύστημα **GYMFUELX**, εφαρμόζεται με στόχο την κάλυψη των **ΜΛΑ-1** και **ΜΛΑ-2**, που αφορούν την δημιουργία προγράμματος γυμναστικής και διατροφής του χρήστη, καθώς και τον υπολογισμό εκτίμησης χρονικής διάρκειας του προγράμματος. Πιο συγκεκριμένα, εφαρμόζεται για την εναλλαγή μεταξύ των αλγορίθμων δημιουργίας προγράμματος γυμναστικής και δημιουργίας προγράμματος διατροφής. Ο αλγόριθμος δημιουργίας προγράμματος γυμναστικής δημιουργεί το πρόγραμμα γυμναστικής του εκάστοτε χρήστη, το αποθηκεύει στην βάση δεδομένων και υπολογίζει την εκτίμηση διάρκειας του προγράμματος, με βάση τις πληροφορίες του χρήστη. Παρομοίως, αλγόριθμος δημιουργίας προγράμματος διατροφής δημιουργεί και αποθηκεύει στην βάση δεδομένων το πρόγραμμα διατροφής του εκάστοτε χρήστη με βάση τις πληροφορίες του.

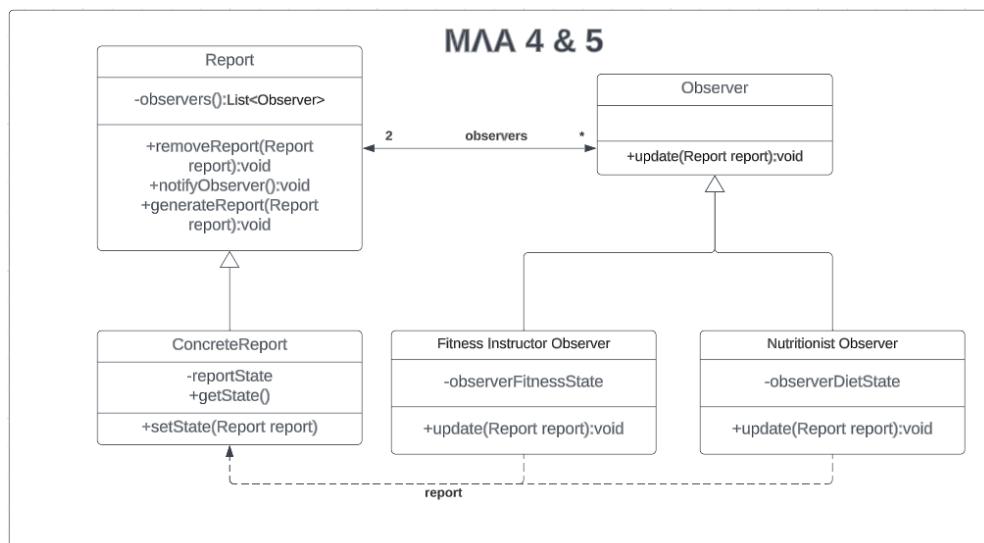


Εικόνα 1: Εφαρμογή προτύπου Strategy.



1.2 Πρότυπο Observer

Το πρότυπο **observer** είναι ένα **πρότυπο συμπεριφοράς**. Στο σύστημα **GYMFUELX**, εφαρμόζεται με στόχο την κάλυψη των **ΜΛΑ-4** και **ΜΛΑ-5**, που αφορούν την ανάθεση αναφορών συνταγών σε διατροφολόγο και αναφορών προγραμμάτων γυμναστικής σε γυμναστές. Πιο συγκεκριμένα, τα report αναπαριστούν δυο λίστες που περιέχουν τις αναφορές για τον διατροφολόγο και τον γυμναστή αντίστοιχα. Ως observer συνδέονται με το report με τις συναρτήσεις generateReport και removeReport που δημιουργούν ή διαγράφουν μια αναφορά. Το reportState περιέχει την κατάσταση κάθε λίστας. Η κατάσταση της εκάστοτε λίστας ανανεώνεται από τον αντίστοιχο observer ανάλογα με τις αλλαγές στις αναφορές. Η μια λίστα ανανεώνεται σύμφωνα με τις αλλαγές στις αναφορές των συνταγών και η άλλη με τις αλλαγές στις αναφορές των προγραμμάτων γυμναστικής. Τέλος το notify ενημερώνει τον observer όταν πραγματοποιηθεί κάποια αλλαγή.



Εικόνα 2: Εφαρμογή προτύπου Observer.



2. Αρχιτεκτονική Συστήματος

Χρησιμοποιήθηκε το εργαλείο Swaggerhub.

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

2.1. Αναγνώριση Πόρων (Resources) Συστήματος

Κλάση BEC	Πόρος REST	Endpoints (HTTP Verbs)
User, (User)	/user/{userID}	GET, POST, PUT
Recipe, (User)	/user/{userID}/recipe	GET
Recipe, (User)	/user/{userID}/recipe/{recipeID}	GET
Recipe, (Nutritionist)	/nutritionist/{NutritionistID}/recipe	POST
Recipe, (Nutritionist)	/nutritionist/{NutritionistID}/recipe/{recipeID}	GET, PUT, DELETE
Report, (User)	/user/{userID}/gymprogram	POST
Report, (User)	/user/{userID}/recipe/{recipeID}	POST
Report, (Nutritionist)	/nutritionist/{NutritionistID}/report	GET
Report, (Nutritionist)	/nutritionist/{NutritionistID}/report/{reportID}	GET, PUT, DELETE
Report, (Trainer)	/trainer/{trainerID}/report	GET
Report, (Trainer)	/trainer/{trainerID}/report/{reportID}	GET, PUT, DELETE
GymProgram, (User)	/user/{userID}/gymprogram	GET
DietProgram, (User)	/user/{userID}/dietprogram	GET
Exercise, (User)	/user/{userID}/gymprogram/{excerciseID}	GET



2.2. Τεκμηρίωση REST διεπαφής

2.2.1. Πόρος User

2.2.1.1. Μοντέλο δεδομένων User

```
User {  
    userID      integer  
    username    string  
    birthday   string($date-time)  
    gender     string  
    height     number  
    weight     number  
    meal       string  
    allergies  string  
    restrictions  string  
    level      string  
    goal       string  
}  
}
```

Εικόνα 3: Μοντέλο δεδομένων User.

2.2.1.2. Endpoint GET πόρου User, για απλό χρήστη

The screenshot shows a Swagger UI interface for a GET request to the endpoint `/user/{userID}`. The endpoint is described as "Get a user's details". A note below states: "<ΛΑ-11> Ο χρήστης πρέπει να μπορεί να επεξεργάζεται τις προσωπικές πληροφορίες στο προφίλ του." (User must be able to edit their personal information in their profile). The "Parameters" section shows a required parameter `userID` of type integer (path), with the value "456" entered into the input field. Below the parameters are "Responses" and "Snippets" sections. The "Snippets" section contains three tabs: "curl (bash)" (selected), "curl (PowerShell)", and "curl (CMD)". The "curl (bash)" tab displays the command:

```
curl -X 'GET' '\n  "https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/456' \n  -H 'accept: application/json'
```

. The "Request URL" field at the bottom also shows the URL: `https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/456`.



Εικόνα 4: Εξίγηση και παράθεση των παραμέτρων του πόρου και δοκιμή του endpoint.

Server response

Code	Details
200	Response body: <pre>{ "userID": 0, "username": "string", "birthday": "2015-07-20T15:49:04-07:00", "gender": "string", "height": 1, "weight": 1.5, "meal": "string", "allergies": "string", "restrictions": "string", "level": "string", "goal": "string" }</pre> Download
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: 151 content-type: application/json; charset=utf-8 date: Tue, 23 May 2023 15:26:36 GMT etag: W/"97-e02IWVm/ZgkohYp57A/tHOU" expires: -1 status: 200 OK x-firefox-spdy: h2</pre>	
Request duration: 1550 ms	

Εικόνα 5: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Successfully received user's details Media type application/json Controls <code>Accept</code> header. Example Value Schema <pre>{ "userID": 0, "username": "string", "birthday": "2015-07-20T15:49:04-07:00", "gender": "string", "height": 1, "weight": 1.5, "meal": "string", "allergies": "string", "restrictions": "string", "level": "string", "goal": "string" }</pre>	No links
400	Bad input parameter	No links
404	User not found	No links

Εικόνα 6: Αποκρίσεις των συστήματος.

2.2.1.3. Endpoint POST πόρου User, για απλό χρήστη

POST /user/{userID} Add details to user's profile

<ΛΑ-10> Ο χρήστης πρέπει να μπορεί να προσθέτει προσωπικές πληροφορίες στο προφίλ του

Parameters

Name	Description
userID * required integer (path)	The user's ID 456
username string (query)	The user's username andreasel
birthday * required string(\$date-time) (query)	The user's birthday 2000-04-04T18:30:00.00
gender * required string (query)	The user's gender male

Εικόνα 7: Εξίγηση και παράθεση των παραμέτρων του πόρου.

height * required number (query)	The user's height 165
weight * required number (query)	The user's weight 54
meal string (query)	The user's meal preferences meal
allergies string (query)	The user's allergies nut
restrictions string (query)	The user's dietary restrictions vegetarian
level * required string (query)	The user's workout level high

Εικόνα 8: Παράθεση των παραμέτρων του πόρου.



goal * required
string
(query)
The user's fitness goal
endurance

Execute Clear

Responses

Snippets▼

CURL (bash) CURL (PowerShell) CURL (CMD)

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456?username=andreasle&birthdays=2005-04-04T18%3A30%3A00.000Z&gender=male&height=165&weight=54&allergies=nut&restrictions=vegetarian&level=high&goal=endurance' \
  -H 'accept: application/json' \
  -d ''
```

Request URL
<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456?username=andreasle&birthdays=2005-04-04T18%3A30%3A00.000Z&gender=male&height=165&weight=54&allergies=nut&restrictions=vegetarian&level=high&goal=endurance>

Εικόνα 9: Δοκιμή του endpoint

Server response

Code Details

200 Response body

```
{
  "userID": 0,
  "username": "string",
  "birthday": "2015-07-20T15:49:04-07:00",
  "gender": "string",
  "height": 1.65,
  "weight": 1.65,
  "meal": "string",
  "allergies": "string",
  "restrictions": "string",
  "level": "string",
  "goal": "string"
}
```

Download

Response headers

```
access-control-allow-credentials: true
access-control-allow-headers: X-Requested-With,Content-Type,Accept,Origin
access-control-allow-methods: *
cache-control: no-cache
content-encoding: gzip
content-length: 151
content-type: application/json; charset=utf-8
date: Tue, 23 May 2023 15:58:55 GMT
etag: "0x8D021WVhem/ZgHn0aYp57A/tHOU"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

360 ms

Εικόνα 10: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Successfully added profile details	No links
400	Bad input parameter	No links
404	User not found	No links

Media type
application/json

Controls Accept header.

Example Value | Schema

```
{
  "userID": 0,
  "username": "string",
  "birthday": "2023-05-23T15:51:41.157Z",
  "gender": "string",
  "height": 0,
  "weight": 0,
  "meal": "string",
  "allergies": "string",
  "restrictions": "string",
  "level": "string",
  "goal": "string"
}
```

Εικόνα 11: Αποκρίσεις του συστήματος.



2.2.1.4.

Endpoint PUT πόρου User, για απλό χρήστη

PUT /user/{userID} Edit details to user's profile

<ΛΑ-11> Ο χρήστης πρέπει να μπορεί να επεξεργάζεται τις προσωπικές πληροφορίες στο προφίλ του.

Parameters

Name Description

userID * required	The user's ID
integer (path)	456
height * required	The user's height
number (query)	167
weight * required	The user's weight
number (query)	60
meal	The user's meal preferences
string (query)	meal

Cancel

Εικόνα 12: Εξήγηση και παράθεση των παραμέτρων του πόρου

allergies
string (query)
The user's allergies
allergies

restrictions
string (query)
The user's dietary restrictions
restrictions

level * required
string (query)
The user's workout level
medium

goal * required
string (query)
The user's fitness goal
endurance

Execute Clear

Responses

Snippets

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

```
curl -X 'PUT' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456?height=167&weight=60&meal=meal'
-H 'Accept: application/json'
```

Εικόνα 13: Δοκιμή του endpoint

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456?height=167&weight=60&level=medium&goal=endurance
```

Server response

Code	Details
200	Response body

```
{
  "userID": 0,
  "username": "string",
  "birthday": "2015-07-20T15:49:04-07:00",
  "gender": "string",
  "height": 1.5,
  "weight": 1.5,
  "meal": "string",
  "allergies": "string",
  "restrictions": "string",
  "level": "string",
  "goal": "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: 151
content-type: application/json; charset=utf-8
date: Tue, 23 May 2023 15:55:14 GMT
etag: W/"97-a02IwTVhem/Zgh0KaYp57A/tHOU"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

```
937 ms
```

Εικόνα 14: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Succesfully edited profile details	No links
400	Bad input parameter	No links
404	User not found	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
  "userID": 0,
  "username": "string",
  "birthday": "2023-05-23T15:55:57.204Z",
  "gender": "string",
  "height": 0,
  "weight": 0,
  "meal": "string",
  "allergies": "string",
  "restrictions": "string",
  "level": "string",
  "goal": "string"
}
```

Εικόνα 15: Αποκρίσεις του συστήματος



2.2.2. Πόρος Recipe

2.2.2.1. Μοντέλο δεδομένων Recipe

```
Recipe < {  
    recipeID      integer  
    recipeType    string  
    imgRecipe     string  
    Instructions   < [string]  
    IngredientsName < [string]  
    NutritionalTable < [string]  
    IngredientsQuantity < [string]  
    time          integer  
    difficulty    string  
    servings      string  
  
}
```

Εικόνα 16: Μοντέλο δεδομένων Recipe.

2.2.2.2. Endpoint GET πόρου Recipe, για απλό χρήστη

GET /user/{userID}/recipe Get the list of all the Recipes

</> Ο χρήστης πρέπει να μπορεί να αναζητά συνταγές και προγράμματα γεύμαστικής

Parameters

Name	Description
userID * required	the user's ID (path)

1231

Execute Clear

Responses

Snippets

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

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/1231/reci
  -H 'accept: application/json'
```

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/1231/reci
pe
```

Server response

Code	Details
200	Response body

```
[  
  {  
    "recipeID": 0,  
    "recipeType": "string",  
    "imgRecipe": "string",  
    "Instructions": [  
      "string"  
    ],  
    "IngredientsName": [  
      "string"  
    ],  
    "NutritionalTable": [  
      "string"  
    ],  
    "IngredientsQuantity": [  
      "string"  
    ],  
    "time": 0,  
    "difficulty": "string",  
    "servings": "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: 158  
content-type: application/json;charset=utf-8  
date: Tue, 23 May 2023 15:59:47 GMT  
etag: W/"9e-vWZGJ2oGcJ0J3D+kovLkQUMos"  
expires: -1  
status: 200 OK  
x-firefox-spdy: h2
```

Εικόνα 17: Εξήγηση και παράθεση των παραμέτρων του πόρου και Δοκιμή του endpoint Εικόνα 18: Δοκιμή του endpoint



Request duration
396 ms

Responses

Code	Description	Links
200	Successful operation	No links
400	Bad input parameter	No links
404	Recipe not found	No links

Media type
application/json

Controls `Accept` header.

Example Value | Schema

```
[{"recipeID": 0, "recipeType": "string", "imgRecipe": "string", "Instructions": [ "string" ], "IngredientsName": [ "string" ], "NutritionalTable": [ "string" ], "IngredientsQuantity": [ "string" ], "time": 0, "difficulty": "string", "servings": "string" }]
```

Εικόνα 19: Αποκρίσεις του συστήματος

2.2.2.3. Endpoint GET πόρου Recipe, για απλό χρήστη και συγκεκριμένο recipeID

GET /user/{userID}/recipe/{recipeID} Get a specific recipe's details

<ΛΔ> Ο χρήστης πρέπει να μπορεί να δει τις θρησκικές λεπτομέρειες για κάθε συνταγή

Parameters

Name	Description
userID * required	the user's ID
integer (path)	37
recipeID * required	The ID of the recipe
integer (path)	85

Responses

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/37/recipe/85' \
-H 'accept: application/json'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/user/37/recipe/85>

Server response

Code	Details
200	Response body

```
{ "recipeID": 0, "recipeType": "string", "imgRecipe": "string", "Instructions": [ "string" ], "IngredientsName": [ "string" ], "NutritionalTable": [ "string" ], "IngredientsQuantity": [ "string" ], "time": 0, "difficulty": "string", "servings": "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-no-cache: 0
cache-control: no-cache
content-encoding: gzip
content-type: application/json; charset=utf-8
date: Mon, 29 May 2023 13:05:28 GMT
etag: W/"9b-6y/FYKVKYLEmTvGeF0m5sT08"
expires: -
status: 200 OK
x-firebase-spdy: h2
```

Request duration

300 ms

Εικόνα 20: Εξήγηση και παράθεση των παραμέτρων και δοκιμή του endpoint

Εικόνα 21: Δοκιμή του endpoint



Responses		
Code	Description	Links
200	Successfully received recipe's details	No links
	Media type application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "recipeID": 0, "recipeType": "string", "ImgRecipe": "string", "Instructions": ["string"], "IngredientsName": ["string"], "NutritionalTable": ["string"], "IngredientsQuantity": ["string"], "time": 0, "difficulty": "string", "servings": "string" }</pre>	
400	Bad input parameter	No links
404	Recipe not found	No links

Εικόνα 22: Αποκρίσεις των συστήματος

2.2.2.4. Endpoint POST πόρου Recipe, για χρήστη διατροφολόγο

POST /nutritionist/{NutritionistID}/recipe Add a new recipe

<ΛΔ-8> Ο διατροφολόγος πρέπει να μπορεί να διαχειρίζεται τις συνταγές.

Parameters

Name	Description
NutritionistID * required	The nutritionist's ID
integer	(path)
153	
recipeType * required	The recipe type
string	(query)
breakfast	
ImgRecipe * required	The image of the recipe
string	(query)
https://pancake.com/pancake.jpg	
Instructions * required	The instructions of the recipe
array[string]	(query)
mix ingredients	
bake	
Add string item	

IngredientsName * required

array[string]

(query)

The ingredients name of the recipe.

flour

milk

[Add string item](#)

NutritionalTable * required

array[string]

(query)

The NutritionalTable of the recipe.

350 calories

45g carbs

[Add string item](#)

IngredientsQuantity * required

array[string]

(query)

The IngredientsQuantity of the recipe.

2 cups

1 cup

[Add string item](#)

time * required

integer

(query)

The time that will be needed for the recipe to be completed.

30

Εικόνα 23: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 24: Παράθεση των παραμέτρων του πόρου



difficulty * required
string
(query)
The difficulty of the recipe.
easy

servings * required
string
(query)
The servings that the recipe can produce.
2

Execute **Clear**

Responses

Snippets▼

CURL (bash) CURL (PowerShell) CURL (CMD)

```
curl -X 'POST' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/nutritionist/153/recipe?recipeType=breakfast&imgRecipe=https%3A%2F%2Fpancake.com%2Fpancake.jpg&Instructions=mix%20ingredients&Instructions=bake&IngredientsName=flour&IngredientsName=milk&NutritionalTable=350%20calories&NutritionalTable=45g%20carbs&IngredientsQuantity=2%20cups&IngredientsQuantity=1%20cup&time=30&difficulty=easy&servings=2'
```

Request URL
<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/nutritionist/153/recipe?recipeType=breakfast&imgRecipe=https%3A%2F%2Fpancake.com%2Fpancake.jpg&Instructions=mix%20ingredients&Instructions=bake&IngredientsName=flour&IngredientsName=milk&NutritionalTable=350%20calories&NutritionalTable=45g%20carbs&IngredientsQuantity=2%20cups&IngredientsQuantity=1%20cup&time=30&difficulty=easy&servings=2>

Εικόνα 25: Παράθεση των παραμέτρων του πόρου και δοκιμή του endpoint

Server response

Code	Details
200	Response body

```
{
  "recipeID": 0,
  "recipeType": "string",
  "imgRecipe": "string",
  "Instructions": [
    "string"
  ],
  "IngredientsName": [
    "string"
  ],
  "NutritionalTable": [
    "string"
  ],
  "IngredientsQuantity": [
    "string"
  ],
  "time": 0,
  "difficulty": "string",
  "servings": "string"
}
```

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: 111
content-type: application/json;charset=utf-8
date: Tue, 23 May 2023 16:14:40 GMT
etag: W/"9b-6y/fYK/VKVYLEmFtVGeF0mSsT08"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Download

Εικόνα 26: Δοκιμή του endpoint

Request duration
982 ms

Responses

Code	Description	Links
200	Recipe successfully added	No links
400	Bad input parameter	No links
404	Recipes not found	No links

Media type
application/json

Controls Accept header.

Example Value Schema

```
{
  "recipeID": 0,
  "recipeType": "string",
  "imgRecipe": "string",
  "Instructions": [
    "string"
  ],
  "IngredientsName": [
    "string"
  ],
  "NutritionalTable": [
    "string"
  ],
  "IngredientsQuantity": [
    "string"
  ],
  "time": 0,
  "difficulty": "string",
  "servings": "string"
}
```

Εικόνα 27: Αποκρίσεις του συστήματος

2.2.2.5. Endpoint GET πόρου Recipe, για χρήστη διατροφολόγο και συγκεκριμένο



recipeID

GET /nutritionist/{NutritionistID}/recipe/{recipeID} Get recipe

<ΛΑ-8> Ο διατροφολόγος πρέπει να μπορεί να διαχειρίζεται τις συνταγές.

Parameters

Name	Description
NutritionistID * required	The nutritionist ID that see the recipe.
integer (path)	153
recipeID * required	The ID of the recipe .
integer (path)	212

Execute Clear

Responses

Εικόνα 28: Εξήγηση και παράθεση των παραμέτρων του πόρου

Snippets▼

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

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIOS/Team27/2.0.0/nutritionist/153/recipe/212' \
  -H 'accept: application/json'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIOS/Team27/2.0.0/nutritionist/153/recipe/212>

Server response

Code Details

200 Response body

```
{
  "recipeID": 0,
  "recipeType": "string",
  "imgRecipe": "string",
  "Instructions": [
    "string"
  ],
  "IngredientsName": [
    "string"
  ],
  "NutritionalTable": [
    "string"
  ],
  "IngredientsQuantity": [
    "string"
  ],
  "time": 0,
  "difficulty": "string",
  "servings": "string"
}
```

Download

Εικόνα 29: Δοκιμή του endpoint.

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: 155
content-type: application/json;charset=utf-8
date: Tue,23 May 2023 16:20:59 GMT
etag: W/"9b-6y/YK/VKYLEmFTvGeF0mSsT08"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

830 ms

Responses

Code	Description	Links
200	Successful operation	No links

Media type

application/json

Controls Accept header.

Εικόνα 30: Αποκρίσεις του συστήματος

Example Value | Schema

```
{
  "recipeID": 0,
  "recipeType": "string",
  "imgRecipe": "string",
  "Instructions": [
    "string"
  ],
  "IngredientsName": [
    "string"
  ],
  "NutritionalTable": [
    "string"
  ],
  "IngredientsQuantity": [
    "string"
  ],
  "time": 0,
  "difficulty": "string",
  "servings": "string"
}
```

400 Bad input parameter No links

404 Recipe not found No links

Εικόνα 31: Αποκρίσεις του συστήματος

2.2.2.6. Endpoint PUT πόρου Recipe, για χρήστη διατροφολόγο και συγκεκριμένο



recipeID

PUT /nutritionist/{NutritionistID}/recipe/{recipeID} Update/Correct recipe

<ΛΑ-8> Ο διατροφολόγος πρέπει να μπορεί να διαχειρίζεται τις συνταγές.

Parameters

Name **Description**

NutritionistID * required	The nutritionist ID that update the recipe report
integer (path)	153
recipeID * required	The ID of the recipe that is being updated.
integer (path)	682
recipeType	The recipe type
string (query)	dinner
imgRecipe	The image of the recipe
string (query)	imgRecipe

Cancel

Εικόνα 32: Εξήγηση και παράθεση των παραμέτρων του πόρου

Instructions array[string] (query)	The instructions of the recipe
IngredientsName array[string] (query)	The ingredients Name of the recipe.
NutritionalTable array[string] (query)	The NutritionalTable of the recipe.
IngredientsQuantity array[string] (query)	The IngredientsQuantity of the recipe.
time integer (query)	The time that will be needed for the recipe to be completed.
difficulty string (query)	The difficulty of the recipe.
servings string (query)	The servings that the recipe can produce.

Εικόνα 33 : Παράθεση των παραμέτρων του πόρου

Execute **Clear**

Responses

Snippets▼

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

```
curl -X "PUT" \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/nutritionist/153/recipe/682?re' \
-H 'accept: */*' \
-e
```

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/nutritionist/153/recipe/682?recipeType=dinner&difficulty=medium&servings=4
```

Server response

Code	Details
200	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: 20 content-type: application/json;charset=utf-8 date: Tue,23 May 2023 16:27:12 GMT etag: W/"14-yIMzqGPkico39npXK6008dB3s" expires: -1 status: 200 OK x-firefox-spdy: h2</pre>

Εικόνα 34: Δοκιμή του endpoint του πόρου

Request duration	509 ms	
Responses		
Code	Description	Links
200	Successful Recipe is updated	No links
400	Bad input parameter	No links
404	Recipe not found	No links

Εικόνα 35: Αποκρίσεις του συστήματος

2.2.2.7. Endpoint **DELETE πόρου Recipe, για χρήστη διατροφολόγο και**



συγκεκριμένο recipeID

DELETE /nutritionist/{NutritionistID}/recipe/{recipeID} Delete recipe ↻

<ΛΑ-9> Ο διατροφολόγος πρέπει να μπορεί να βλέπει τις αναφορές που γίνονται από τους χρήστες στις συνταγές.

Parameters Cancel

Name	Description
NutritionistID * required integer (path)	The nutritionist ID that delete the recipe 153
recipeID * required integer (path)	The ID of the recipe that is being deleted. 287

Execute Clear

Εξήγηση και παράθεση των παραμέτρων του πόρου

Responses

Snippets▼

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

```
curl -X 'DELETE'  
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Tem27/2.0.0/nutritionist/153/recipe/287' -H  
'Accept: application/json'
```

Request URL
<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Tem27/2.0.0/nutritionist/153/recipe/287>

Server response

Code	Details
200	Response body "Successful Recipe is deleted" Download
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: 50 content-type: application/json;charset=utf-8 date: Sat,27 May 2023 09:30:01 GMT etag: W/"32-w5EASYgF3znWCndkMem+K40Bq50" expires: status: 200 OK x-firebase-spy: h2</pre>	
Request duration 350 ms	

Εικόνα 36: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Successful Recipe is deleted Media type application/json Controls Accept header. Example Value Schema "Successful Recipe is deleted"	No links
400	Bad input parameter	No links
404	Recipe not found	No links

Εικόνα 37: Αποκρίσεις του συστήματος



2.2.3. Πόρος Report

2.2.3.1. Μοντέλο δεδομένων Report

```
Report < {  
    isGym-Diet      boolean  
    ID              integer  
    ByUser         string  
}  
}
```

Εικόνα 38: Μοντέλο δεδομένων Report.

2.2.3.2. Endpoint POST πόρου Report, για απλό χρήστη για πρόγραμμα γυμναστικής

POST /user/{userID}/gymprogram Create a report to this gymprogram

<ΛΑ-6> Ο χρήστης θα πρέπει να μπορεί να κάνει αναφορά σε προγράμματα γυμναστικής για τυχόν λάθη ή δυσαρέσκεια.

Parameters

Name	Description
userID * required	the User's ID (path)

456

Execute Clear

Responses

Snippets

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

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIOS/Team27/2.0.0/user/456/gymprogram' \
  -H 'accept: */*' \
  -d ''
```

Request URL
<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIOS/Team27/2.0.0/user/456/gymprogram>

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: Tue,23 May 2023 17:20:12 GMT
etag: W/"14-yIMMzzg0Pkico39NpXK600BdB3s"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration
820 ms

Responses

Code	Description	Links
200	Operation successful	No links
400	Bad input parameter	No links
404	Gym Program not found	No links

Εικόνα 39: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 40: Δοκιμή του endpoint του πόρου και αποκρίσεις του συστήματος



2.2.3.3.

Endpoint POST πόρου Report, για απλό χρήστη και συγκεκριμένο recipeID

POST /user/{userID}/recipe/{recipeID} Create a report to this recipe

<ΛΑ-5> Ο χρήστης θα πρέπει να μπορεί να κάνει αναφορά στα συνταγές για τυχόν λάθη

Parameters

Name	Description
userID * required	the User's ID (path)
recipeID * required	the recipe's ID (path)

123
4124

Execute Clear

Responses

Snippets

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

```
curl -X 'POST' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/user/123/recipes/4124' \
  -H 'accept: */*' \
  -d ''
```

Request URL

https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/user/123/recipes/4124

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,29 May 2023 13:09:57 GMT
etag: W/"14-yDmzq8pk1ca39Np0X66OBd83s"
expires: -1
status: 200 OK
```

Request duration

699 ms

Responses

Code Description Links

200 Operation successful No links

400 Bad input parameter No links

404 Recipe not found No links

Εικόνα 41: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 42: Δοκιμή του endpoint του πόρου και αποκρίσεις του συστήματος

2.2.3.4.

Endpoint GET πόρου Report, για χρήστη διατροφολόγο

GET /nutritionist/{NutritionistID}/report Get recipe reports

<ΛΑ-9> Ο διατροφολόγος πρέπει να μπορεί να βλέπει τις αναφορές που γίνονται από τους χρήστες στις συνταγές.

Parameters

Name	Description
NutritionistID * required	The nutritionist ID that see the recipe reports (path)

153

Execute Clear

Responses

Snippets

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

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report' \
  -H 'accept: application/json'
```

Request URL

https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report

Server response

Code Details

200 Response body

```
[{"isGym-Diet": true, "ID": 0, "ByUser": "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: 74
content-type: application/json;charset=utf-8
date: Tue,23 May 2023 18:59:57 GMT
etag: W/"4a-NQG7e53cs/6+lzF0B55mRPZYDxw"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

646 ms

Εικόνα 43: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 44: Δοκιμή του endpoint



Responses

Code	Description	Links
200	Successful operation	No links
	Media type	
	application/json	
	Controls <code>Accept</code> header.	
	Example Value Schema	
	<pre>[{ "isGym-Diet": true, "ID": 0, "ByUser": "string" }]</pre>	
400	Bad input parameter	No links
404	Reports not found	No links

Εικόνα 45: Αποκρίσεις του συστήματος

2.2.3.5. Endpoint GET πόρου Report, για χρήστη διατροφολόγο και συγκεκριμένο reportID

GET /nutritionist/{NutritionistID}/report/{reportID} Get recipe report

<ΛΑ-9> Ο διατροφολόγος πρέπει να μπορεί να βλέπει τις αναφορές που γίνονται από τους χρήστες στις συνταγές.

Parameters

Name	Description
NutritionistID * required	The nutritionist ID that see the recipe report
integer (path)	153
reportID * required	The ID of the recipe report.
integer (path)	19

Execute Clear

Responses

Snippets▼

CURL (bash) CURL (PowerShell) CURL (CMD)

```
curl -X 'GET' \  
'https://virtserver.swaggerhub.com/DINOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report/19' \  
-H 'accept: application/json'
```

Request URL

https://virtserver.swaggerhub.com/DINOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report/19

Server response

Code	Details
200	Response body
	<pre>{ "isGym-Diet": true, "ID": 0, "ByUser": "string" }</pre>
	Download
	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-max-age: 3600 cache-control: no-cache content-encoding: gzip content-length: 76 content-type: application/json;charset=utf-8 date: Tue, 23 May 2023 19:03:41 GMT etag: W/\"e6-zdsNfKq+KzUmK3w/g/KSDC0LEk- expires: - status: 200 OK x-firefox-spdy: h2</pre>
	Request duration
	1419 ms

Εικόνα 46: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 47: Δοκιμή του endpoint



Responses

Code	Description	Links
200	Successful operation	No links
	Media type	
	application/json	
	Controls <code>Accept</code> header.	
	Example Value Schema	
	<pre>{ "isGym-Diet": true, "ID": 0, "ByUser": "string" }</pre>	
400	Bad input parameter	No links
404	Report not found	No links

Εικόνα 48: Αποκρίσεις του συστήματος

2.2.3.6. Endpoint PUT πόρου Report, για χρήστη διατροφολόγο και συγκεκριμένο reportID

PUT /nutritionist/{NutritionistID}/report/{reportID} Update/Correct recipe report

<ΛΑ-9> Ο διατροφολόγος πρέπει να μπορεί να βλέπει τις αναφορές που γίνονται από τους χρήστες στις συνταγές.

Parameters

Name Description

NutritionistID * required integer (path) The nutritionist ID that update the recipe report 153

reportID * required integer (path) The ID of the recipe report that is being updated. 19

dayDietProgramIDs array[array] (query) The Diet Program Details 52
Add integer item -
37
Add integer item -

Responses

Snippets

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

```
curl -X 'PUT' \ 'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/nutritionist/153/report/19?dayDietProgramIDs=52&dayDietProgramIDs=37&dayDietProgramIDs=36&dayDietProgramIDs=68'
```

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/nutritionist/153/report/19?dayDietProgramIDs=52&dayDietProgramIDs=37&dayDietProgramIDs=36&dayDietProgramIDs=68
```



Εικόνα 49: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 50: Δοκιμή του endpoint

Server response

Code	Details
200	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: 20 content-type: application/json;charset=utf-8 date: Tue,23 May 2023 19:07:52 GMT etag: W/"14-yIMMzzqGPkico39NpxK600BdB3s" expires: -1 status: 200 OK x-firefox-spdy: h2</pre> Request duration <pre>396 ms</pre>

Responses

Code	Description	Links
200	Successful Recipe Report is updated	No links
400	Bad input parameter	No links
404	Recipe Report not found	No links

Εικόνα 51: Αποκρίσεις του συστήματος

2.2.3.7. Endpoint **DELETE** πόρου **Report**, για χρήστη διατροφολόγο και

Responses

Snippets >

cURL (bash)	cURL (PowerShell)	cURL (CMD)
<pre>curl -X 'DELETE' \ 'https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report/19' \ -H "Accept: */*"</pre>		

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELDIS/Team27/2.0.0/nutritionist/153/report/19
```

Server response

Code	Details	
200	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: 20 content-type: application/json;charset=utf-8 date: Sat,27 May 2023 09:39:43 GMT etag: W/"14-yIMMzzqGPkico39NpxK600BdB3s" expires: -1 status: 200 OK x-firefox-spdy: h2</pre> Request duration <pre>837 ms</pre>	
200	Successful Recipe Report is deleted	No links
400	Bad input parameter	No links
404	Recipe Report not found	No links

/nutritionist/{NutritionistID}/report/{reportID} delete recipe report

Parameters

Name Description

NutritionistID * required integer (path) 153

reportID * required integer (path) 19

Execute Clear



συγκεκριμένο reportID

Εικόνα 52: Εξήγηση και παράθεση των παραμέτρων του πόρου

2.2.3.8. Endpoint GET πόρου Report, για χρήστη γυμναστή

GET /trainer/{trainerID}/report Get gym program reports

<ΠΛA-7> Ο γυμναστής θα πρέπει να έχει την δυνατότητα να βλέπει και να διορθώνει τα προγράμματα γυμναστικής από τυχό λάθη

Parameters

Name	Description
trainerID * required	The trainer Id

String (path)

32

Cancel

Execute Clear

Responses

Snippets

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

```
curl -X 'GET' \ 
-H 'Accept: application/json' \
-H 'Content-Type: application/json' \
-H 'Host: virtserver.swaggerhub.com' \
-H 'User-Agent: curl/7.64.1' \
-H 'Connection: keep-alive' \
-H 'Cache-Control: no-cache' \
-H 'Postman-Token: 33333333-3333-3333-3333-333333333333' \
-H 'Accept-Encoding: gzip, deflate' \
-H 'Content-Length: 74' \
-H 'Content-Type: application/json; charset=utf-8' \
-H 'Date: Tue, 23 May 2023 19:12:38 GMT' \
-H 'Etag: W/"4a-NQG7e3cs/6+lzF0BS5mRPZYDxw"' \
-H 'Expires: -1' \
-H 'Status: 200 OK' \
-H 'X-Firefox-Spdy: h2'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/trainer/32/report>

Εικόνα 54: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 53: Δοκιμή του endpoint και αποκρίσεις του συστήματος

2.2.3.8. Endpoint GET πόρου Report, για χρήστη γυμναστή

Server response

Code Details

200 Response body

```
[{"isGym-Diet": true, "ID": 0, "ByUser": "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: 74
content-type: application/json; charset=utf-8
date: Tue, 23 May 2023 19:12:38 GMT
etag: W/"4a-NQG7e3cs/6+lzF0BS5mRPZYDxw"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

300 ms

Εικόνα 55: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Successful operation	No links
400	Bad input parameter	No links
404	Gym Program Reports not found	No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
[{"isGym-Diet": true, "ID": 0, "ByUser": "string"}]
```

400 Bad input parameter

404 Gym Program Reports not found



Εικόνα 56: Αποκρίσεις του συστήματος

2.2.3.9. Endpoint GET πόρου Report, για γυμναστή και συγκεκριμένο reportID

GET /trainer/{trainerID}/report/{reportID} Get gym program report

<ΛΔ> Ο γυμναστής θα πρέπει να έχει την δυνατότητα να βλέπει και να διορθώνει τα προγράμματα γυμναστικής από τυχόν λάθη

Parameters

Name	Description
trainerID * required	The trainer ID
integer (path)	32
reportID * required	The report ID
integer (path)	41

Cancel

Execute Clear

Responses

Snippets

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/trainer/32/report/41' \
-H 'accept: application/json'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/trainer/32/report/41>

Server response

Code Details

200 Response body

```
{ "isGym-Diet": true, "ID": 0, "ByUser": "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: 76
content-type: application/json;charset=utf-8
date: Tue,23 May 2023 19:16:03 GMT
etag: W-/4zdsWfKq+Kc2uHK3w/g/K5dC0Lk"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

609 ms

Εικόνα 57: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 58: Δοκιμή του endpoint

Responses

Code	Description	Links
200	Successful operation	No links
400	Bad input parameter	No links
404	Gym Program Report not found	No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
{ "isGym-Diet": true, "ID": 0, "ByUser": "string" }
```

400 Bad input parameter

404 Gym Program Report not found



Εικόνα 59: Αποκρίσεις του συστήματος

2.2.3.10. Endpoint PUT πόρου Report, για γυμναστή και συγκεκριμένο reportID

PUT /trainer/{trainerID}/report/{reportID} Update/Correct fitness program report

<ΛΔ-7> Ο γυμναστής θα πρέπει να έχει την δυνατότητα να βλέπει και να διορθώνει τα προγράμματα γυμναστικής από τυχόν λάθη

Parameters

Name	Description
trainerID * required	The trainer
integer (path)	32
reportID * required	The report ID
integer (path)	41
GymProgramDetails * required	The Gym Program Details
array(string) (query)	<ul style="list-style-type: none">dummy text 1dummy text 2dummy text 3Add string item

Execute Clear

Responses

Snippets

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

```
curl -X 'PUT' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/trainer/32/report/41?GymProgramDetails=dummy%20text%20&GymProgramDetails=dummy%20text%20'
-H 'accept: */*' 
```

Request URL

```
https://virtserver.swaggerhub.com/DIMOSKOMPITSELIDIS/Team27/2.0.0/trainer/32/report/41?GymProgramDetails=dummy%20text%20&GymProgramDetails=dummy%20text%20
```

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: Tue, 23 May 2023 19:20:02 GMT
etag: W/"14-yIMMzzqGPkico39NpXK600Bd3s"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Εικόνα 60: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 61: Δοκιμή του endpoint

Request duration
1049 ms

Responses

Code	Description	Links
200	Successful Gym Program Report is updated	No links
400	Bad input parameter	No links
404	Gym Program Report not found	No links

Εικόνα 62: Αποκρίσεις του συστήματος



2.2.3.11. Endpoint DELETE πόρου Report, για γυμναστή και συγκεκριμένο reportID

DELETE /trainer/{trainerID}/report/{reportID} Get gym program reports

<ΛΑ-7> Ο γυμναστής θα πρέπει να έχει την δυνατότητα να βλέπει και να διορθώνει τα προγράμματα γυμναστικής από τυχόν λάθη

Parameters

Cancel

Name	Description
trainerID * required	The trainer ID that deletes the gym program
integer (path)	32
reportID * required	The ID of the report that is being deleted.
integer (path)	41

Execute Clear

Responses

Snippets▼

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

```
curl -X 'DELETE' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/trainer/32/report/41' \
  -H 'accept: */*' [Copy]
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/trainer/32/report/41>

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: Sat,27 May 2023 09:42:57 GMT
etag: W/"14-yIMzqzGPKic039NpXX600Bd3s"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

667 ms

Εικόνα 63: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 64: Δοκιμή του endpoint

Responses		
Code	Description	Links
200	Successful Gym Program Report is deleted	No links
400	Bad input parameter	No links
404	Gym Program Report not found	No links

Εικόνα 65: Αποκρίσεις του συστήματος



2.2.4. Πόρος GymProgram

2.2.4.1. Μοντέλο δεδομένων GymProgram



Εικόνα 66: Μοντέλο δεδομένων GymProgram.

2.2.4.2. Endpoint GET πόρου GymProgram, για απλό χρήστη

The screenshot shows the Swagger UI interface for the GET /user/{userID}/gymprogram endpoint. It includes fields for Parameters (userID), a Request URL (https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/user/456/gymprogram), and a curl command snippet.

Parameters:

Name	Description
userID * required	the user's ID (path)

Responses:

Snippets:

```
cURL (bash) cURL (PowerShell) cURL (CMD)
curl -X 'GET' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIIDIS/Team27/2.0.0/user/456/gymprogram' \
-H 'accept: application/json'
```

The screenshot shows the Swagger UI interface displaying the response for the GET /user/{userID}/gymprogram endpoint. It includes a 'Code' tab showing the JSON response body and a 'Details' tab showing the response headers.

Server response:

Code:

```
{
  "GymProgramDetails": [
    {
      "exerciseID": 0,
      "exerciseTitle": "string",
      "exerciseDescription": "string",
      "explanationVideo": "string"
    }
  ],
  "gymProgramID": 0
}
```

Details:

Response body:

```
{
  "GymProgramDetails": [
    {
      "exerciseID": 0,
      "exerciseTitle": "string",
      "exerciseDescription": "string",
      "explanationVideo": "string"
    }
  ],
  "gymProgramID": 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: 128
content-type: application/json;charset=utf-8
date: Tue, 23 May 2023 16:56:59 GMT
etag: W/"80-icZ/SeShdCAUy0IRUD4UH3Ay488"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration: 915 ms

Εικόνα 67: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 68: Δοκιμή του endpoint



Code	Description	Links
200	Successful operation	No links
	Media type application/json ▾ Controls Accept header.	
	Example Value Schema	
	<pre>{ "GymProgramDetails": [{ "exerciseID": 0, "exerciseTitle": "string", "exerciseDescription": "string", "explanationVideo": "string" }], "gymProgramID": 0 }</pre>	
400	Bad input parameter	No links
404	Gym Program not found	No links

Εικόνα 69: Αποκρίσεις των συστήματος

2.2.5. Πόρος Exercise

2.2.5.1. Μοντέλο δεδομένων Exercise

```
Exercise ▶ {  
  exerciseID      integer  
  exerciseTitle   string  
  exerciseDescription string  
  explanationVideo string  
  
}
```

Εικόνα 70: Μοντέλο δεδομένων Exercise.



2.2.5.2. Endpoint GET πόρου Exercise, για απλό χρήστη και συγκεκριμένο excerciseID

GET /user/{userID}/gymprogram/{excerciseID} Get a specific excercise's details

<ΛΑ-4> Ο χρήστης θα πρέπει να μπορεί να δει πληροφορίες για κάθε άσκηση γυμναστικής

Parameters

Name	Description
userID * required	The user's ID integer (path)
excerciseID * required	The ID of the excercise integer (path)

456

63

Execute Clear

Responses

Snippets▼

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

```
curl -X 'GET' \
'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456/gymprogram/63' \
-H 'accept: application/json'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/456/gymprogram/63>

Server response

Code	Details
200	Response body

```
{
  "exerciseID": 0,
  "excerciseTitle": "string",
  "excerciseDescription": "string",
  "explanationVideo": "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: 89
content-type: application/json;charset=utf-8
date: Tue,23 May 2023 17:03:48 GMT
etag: W/"59-OfekuRpM4vCQqUs9ExJ+wYn9b4"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

758 ms

Εικόνα 71: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 72: Δοκιμή του endpoint

Responses

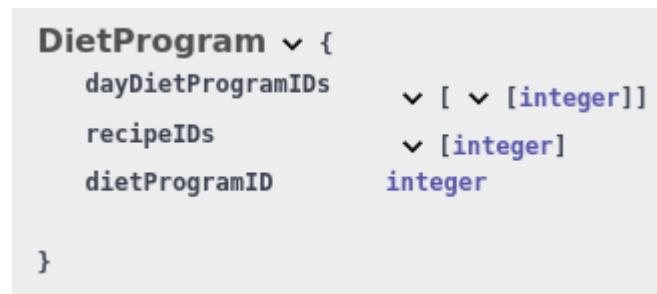
Code	Description	Links
200	Successfully received excercise's details	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	{ "exerciseID": 0, "excerciseTitle": "string", "excerciseDescription": "string", "explanationVideo": "string" }	
400	Bad input parameter	No links
404	Gym Program not found	No links

Εικόνα 73: Αποκρίσεις του συστήματος



2.2.6. Πόρος DietProgram

2.2.6.1. Μοντέλο δεδομένων DietProgram



Εικόνα 74: Μοντέλο δεδομένων DietProgram.

2.2.6.2. Endpoint GET πόρου DietProgram, για απλό χρήστη

GET /user/{userID}/dietprogram Get user's diet program

<ΛΑ-2> Ο χρήστης πρέπει να μπορεί να λαμβάνει προγράμματα γυμναστικής και πρόγραμμα διατροφής βασισμένα σε προσωπικές πληροφορίες.

Parameters

Name	Description
userID * required	The user's ID
integer	(path)
465	

Cancel

Execute Clear

Responses

Snippets

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

```
curl -X 'GET' \
  'https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/465/dietprogram' \
  -H 'accept: application/json'
```

Request URL

<https://virtserver.swaggerhub.com/DIMOSKOMPITSELIidis/Team27/2.0.0/user/465/dietprogram>

Server response

Code Details

200 Response body

```
{
  "dayDietProgramIDs": [
    [
      0
    ],
    "recipeIDs": [
      0
    ],
    "dietProgramID": 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: 75
content-type: application/json;charset=utf-8
date: Tue, 23 May 2023 16:59:55 GMT
etag: W/"4b-kX13Cte6IPLTs0fqfVdE6zBhBbs"
expires: -1
status: 200 OK
x-firefox-spdy: h2
```

Request duration

869 ms

Εικόνα 75: Εξήγηση και παράθεση των παραμέτρων του πόρου

Εικόνα 76: Δοκιμή του endpoint



Responses		
Code	Description	Links
200	Successfully received Diet Program	No links
	Media type	
	<input checked="" type="button"/> application/json	<input type="button"/>
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "dayDietProgramIDs": [[0]], "recipeIDs": [0], "dietProgramID": 0 }</pre>	
400	Bad input parameter	No links
404	Diet Program not found	No links

Εικόνα 77: Αποκρίσεις του συστήματος



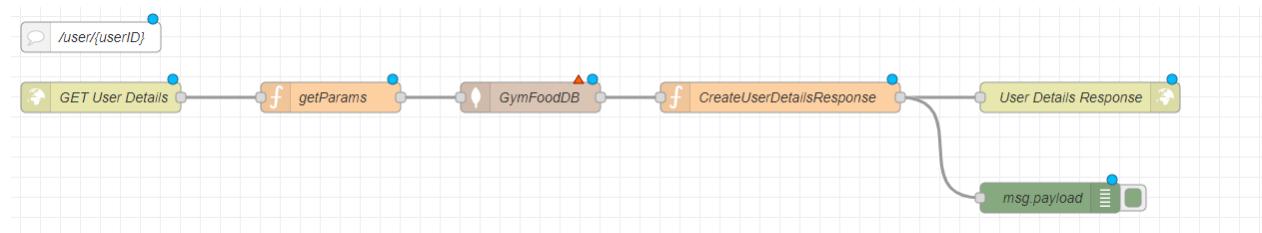
3. Υλοποίηση Συστήματος με Node-RED

3.1. Αντιστοίχιση των REST Υπηρεσιών σε Ροές NodeRed

- Σύνδεσμος για το αρχείο τύπου zip που περιέχει τα flows.

3.1.1. Ρόες πόρου User

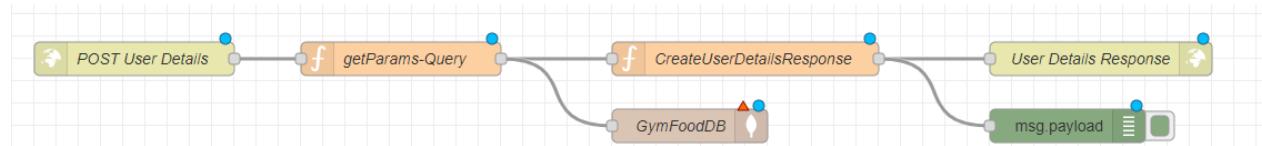
Ρόη endpoint GET /user/{userID}



Εικόνα 78: Ρόη NodeRed που αντιστοιχεί στο API call Get /user/:userID.

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή των προσωπικών στοιχείων του χρήστη. Μέσω της συνάρτησης Get Params, επιστρέφεται το ζήτημα με συγκεκριμένο userID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateUserDetailsResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

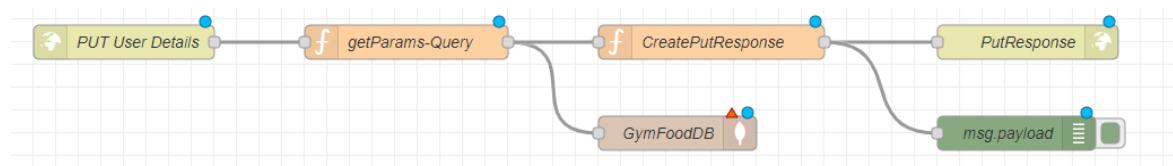
Ρόη endpoint POST /user/{userID}



Εικόνα 79: Ρόη NodeRed που αντιστοιχεί στο API call POST /user/:userID.

Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να καταχωρήσει τα προσωπικά του στοιχεία στην βάση GymFoodDB όπου ανακτά το userID μέσω της συνάρτησης Get Params-Query για να αποθηκευτεί αναλόγως. Ακόμη η CreateUserDetailsResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

Ρόη endpoint PUT /user/{userID}



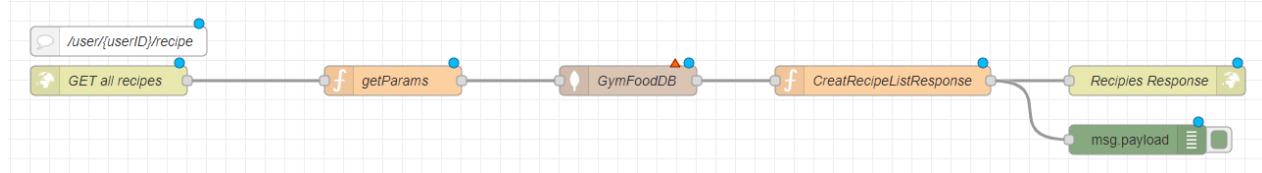
Εικόνα 80: Ρόη NodeRed που αντιστοιχεί στο API call PUT /user/:userID.



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να ενημερώσει τα προσωπικά του στοιχεία στην βάση GymFoodDB όπου μέσω της συνάρτησης Get Params-Query ανακτά το userID και τις αντίστοιχες αλλαγές που έκανε ο χρήστης στα προσωπικά του στοιχεία. Ακόμη η CreatePUTResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή των νέων στοιχείων.

3.1.2. Ροές πόρου Recipe

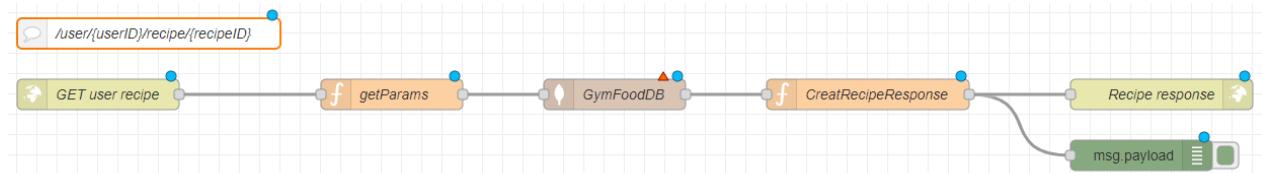
Ροή endpoint GET /user/{userID}/recipe



Εικόνα 81: Ροή NodeRed που αντιστοιχεί στο API call Get /user/:userID/recipe

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή όλων των συνταγών. Μέσω της συνάρτησης Get Params, επιστρέφει τη συνταγή με συγκεκριμένο userID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateRecipeListResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

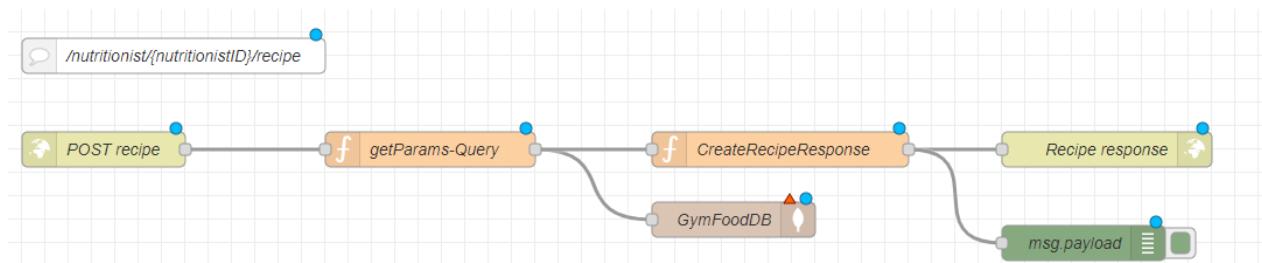
Ροή endpoint GET /user/{userID}/recipe/{recipeID}



Εικόνα 82: Ροή NodeRed που αντιστοιχεί στο API call Get /user/:userID/recipe/:recipeID.

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή μίας συγκεκριμένης συνταγής. Μέσω της συνάρτησης Get Params, επιστρέφει τη συνταγή με συγκεκριμένο userID και recipeID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateRecipeResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

Ροή endpoint POST /nutritionist/{nutritionistID}/recipe

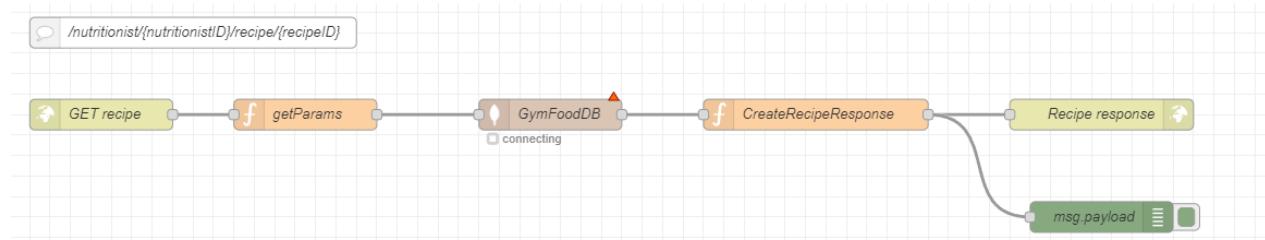


Εικόνα 83: Ροή NodeRed που αντιστοιχεί στο API call POST /nutritionist/:nutritionistID/recipe.



Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας διατροφολόγος να καταχωρίσει στοιχεία της συνταγή στην βάση GymFoodDB όπου ανακτά το nutritionistID μέσω της συνάρτησης Get Params-Query για να αποθηκευτεί αναλόγως. Ακόμη η CreateRecipeResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή της αντίστοιχης συνταγής.

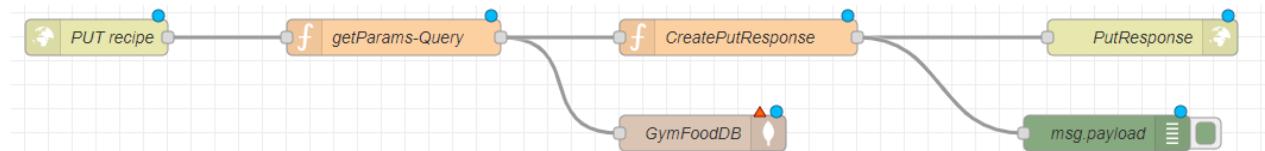
Ροή endpoint GET /nutritionist/{nutritionistID}/recipe/{recipeID}



Εικόνα 84: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/recipe/:recipeID.

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή συγκεκριμένης συνταγής. Μέσω της συνάρτησης Get Params, επιστρέφει τη συνταγή με συγκεκριμένο nutritionistID και recipeID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateRecipeResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

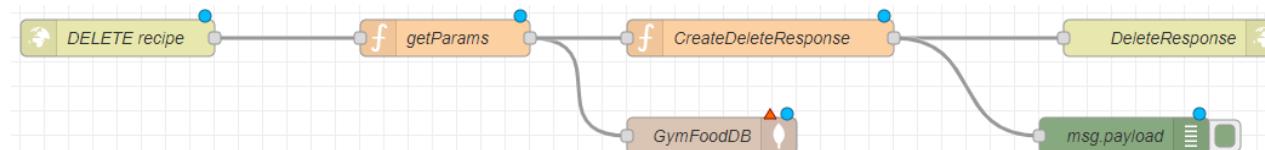
Ροή endpoint PUT /nutritionist/{nutritionistID}/recipe/{recipeID}



Εικόνα 85: Ροή NodeRed που αντιστοιχεί στο API call PUT /nutritionist/:nutritionistID/recipe/:recipeID.

Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας διατροφολόγος να επεξεργαστεί τα στοιχεία της συνταγή στην βάση GymFoodDB όπου μέσω της συνάρτησης Get Params-Query ανακτούμε τα nutritionistID, recipeID και τις αντίστοιχες αλλαγές που έκανε ο διατροφολόγος στα στοιχεία της συνταγή. Ακόμη η CreatePUTResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

Ροή endpoint DELETE /nutritionist/{nutritionistID}/recipe/{recipeID}



Εικόνα 86: Ροή NodeRed που αντιστοιχεί στο API call DELETE /nutritionist/:nutritionistID/recipe/:recipeID.

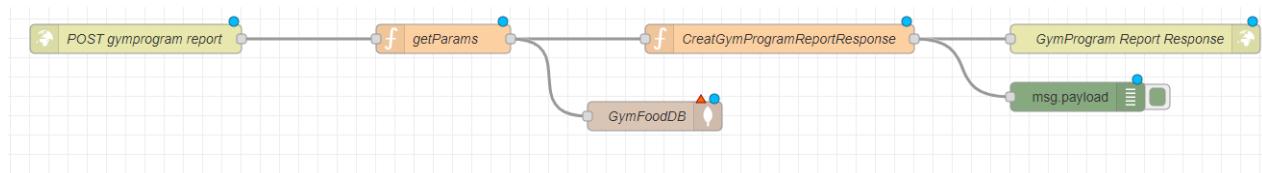
Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας διατροφολόγος να διαγράψει μία



συνταγή από την βάση GymFoodDB όπου μέσω της συνάρτησης Get Params ανακτούμε τα nutritionistID και recipeID. Ακόμη η CreateDELETEResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

3.1.3. Ροές πόρου Report

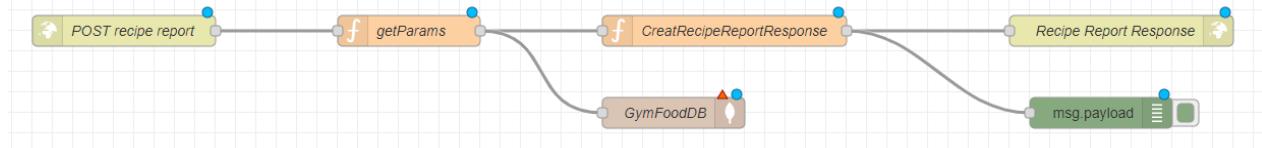
Ροή endpoint POST /user/{userID}/gymprogram



Εικόνα 87: Ροή NodeRed που αντιστοιχεί στο API call POST /user/:userID/gymprogram.

Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να δημιουργήσει αναφορά σε ένα πρόγραμμα στην βάση GymFoodDB όπου μέσω της συνάρτησης Get Params ανακτούμε το userID. Ακόμη η CreateGymProgramReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

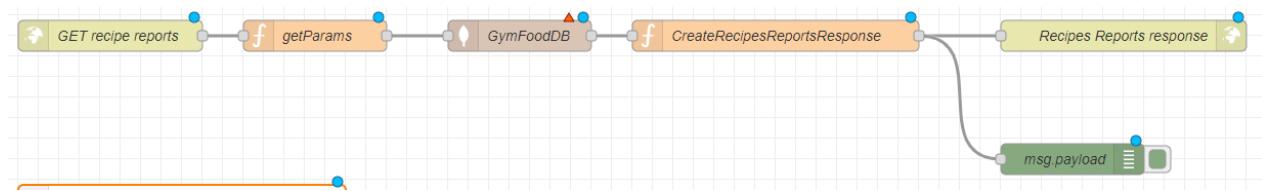
Ροή endpoint POST /user/{userID}/recipe/{recipeID}



Εικόνα 88: Ροή NodeRed που αντιστοιχεί στο API call POST /user/:userID/recipe/:recipeID

Η ροή που υλοποιεί την υπηρεσία, μέσω της οποίας μπορεί ένας χρήστης να δημιουργήσει αναφορά σε συγκεκριμένη συνταγή στην βάση GymFoodDB όπου μέσω της συνάρτησης Get Params ανακτούμε το userID και recipeID. Ακόμη η CreateRecipeReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

Ροή endpoint GET /nutritionist/{nutritionistID}/report

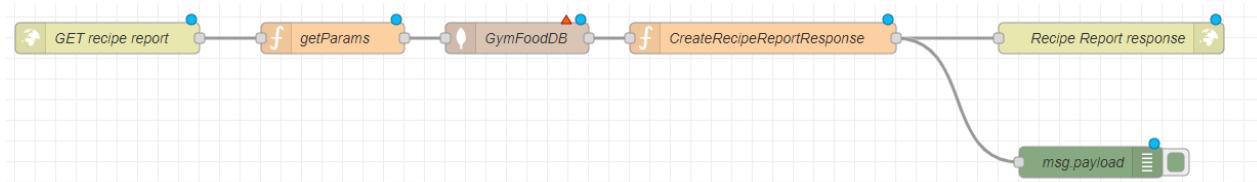


Εικόνα 89: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/report

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την προβολή όλων των αναφορών χρήστη στον διατροφολόγο. Μέσω της συνάρτησης Get Params, επιστρέφεται οι αναφορές συνταγών με συγκεκριμένο nutritionistID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateRecipesReportsResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.



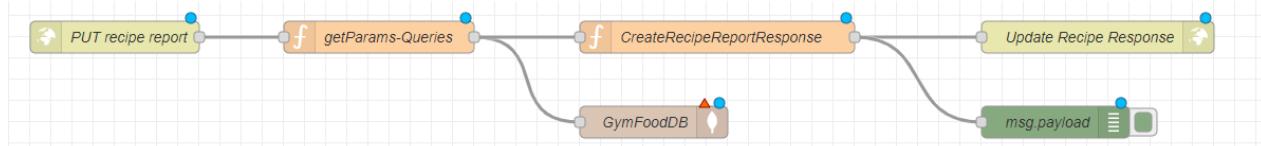
Ροή endpoint GET /nutritionist/{nutritionistID}/report/{reportID}



Εικόνα 90: Ροή NodeRed που αντιστοιχεί στο API call GET /nutritionist/:nutritionistID/report/:reportID

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την προβολή μιας αναφοράς χρήστη στον διατροφολόγο. Μέσω της συνάρτησης Get Params, επιστρέφεται η αναφορά συνταγής με συγκεκριμένο nutritionistID και reportID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateRecipeReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

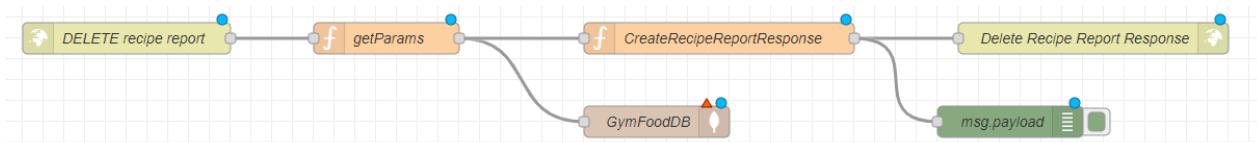
Ροή endpoint PUT /nutritionist/{nutritionistID}/report/{reportID}



Εικόνα 91: Ροή NodeRed που αντιστοιχεί στο API call PUT /nutritionist/:nutritionistID/report/:reportID

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επεξεργασία μιας αναφοράς χρήστη από τον διατροφολόγο. Μέσω της συνάρτησης Get Params-Query ανακτούμε τα nutritionistID, reportID και τις αντίστοιχες αλλαγές που έκανε ο διατροφολόγος στα στοιχεία της αναφερόμενης συνταγής. Ακόμη η CreateRecipeReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

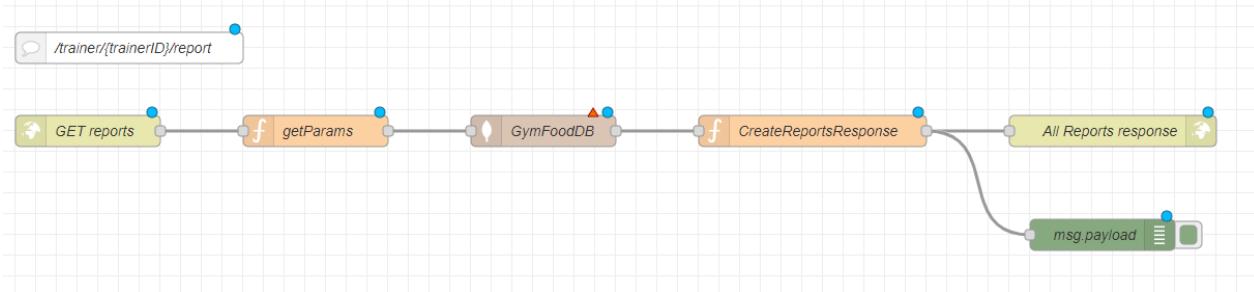
Ροή endpoint DELETE /nutritionist/{nutritionistID}/report/{reportID}



Εικόνα 92: Ροή NodeRed που αντιστοιχεί στο API call DELETE /nutritionist/:nutritionistID/report/:reportID

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την διαγραφή μιας αναφοράς χρήστη από τον διατροφολόγο. Μέσω της συνάρτησης Get Params ανακτούμε τα nutritionistID, reportID. Ακόμη η CreateRecipeReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

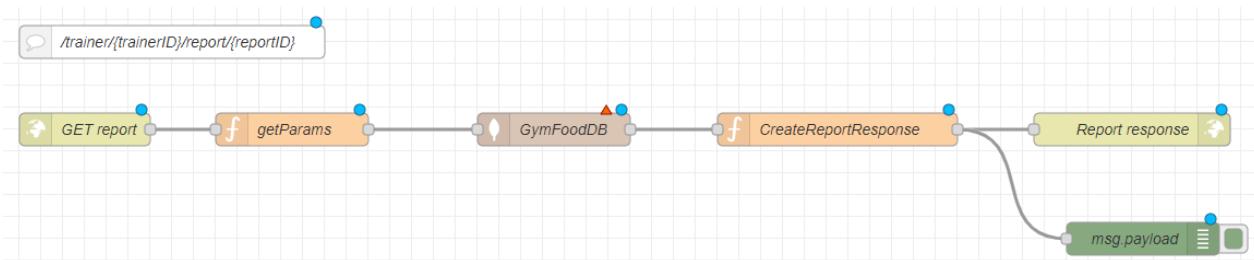
Ροή endpoint GET /trainer/{trainerID}/report



Εικόνα 93: Ροή NodeRed που αντιστοιχεί στο API call GET /trainer/:trainerID/report

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την προβολή όλων των αναφορών χρήστη στον γυμναστή. Μέσω της συνάρτησης Get Params, επιστρέφονται οι αναφορές προγράμματος γυμναστικής με συγκεκριμένο trainerID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateReportsResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

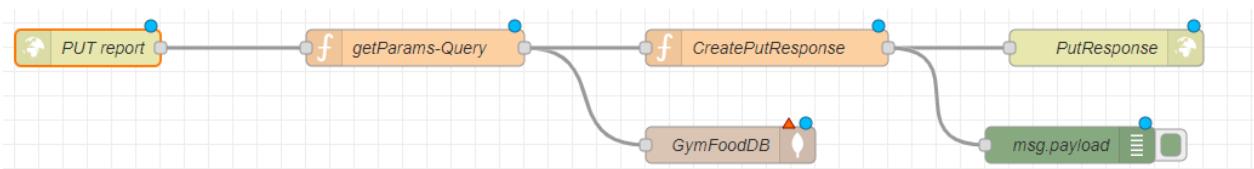
Ροή endpoint GET /trainer/{trainerID}/report/{reportID}



Εικόνα 94: Ροή NodeRed που αντιστοιχεί στο API call GET /trainer/:trainerID/report/:reportID

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την προβολή μιας αναφοράς χρήστη στον γυμναστή. Μέσω της συνάρτησης Get Params, επιστρέφεται η αναφορά προγράμματος γυμναστικής με συγκεκριμένο trainerID και reportID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateReportResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

Ροή endpoint PUT /trainer/{trainerID}/report/{reportID}

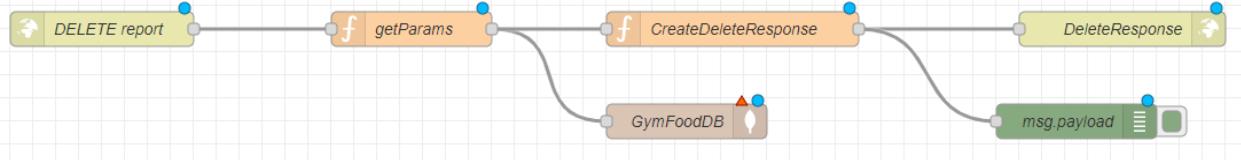


Εικόνα 95: Ροή NodeRed που αντιστοιχεί στο API call PUT /trainer/:trainerID/report/:reportID

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επεξεργασία μιας αναφοράς χρήστη από τον γυμναστή. Μέσω της συνάρτησης Get Params-Query ανακτούμε τα trainerID, reportID και τις αντίστοιχες αλλαγές που έκανε ο γυμναστής στα στοιχεία της αναφερόμενου προγράμματος γυμναστικής. Ακόμη η CreatePUTResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.



Ροή endpoint DELETE /trainer/{trainerID}/report/{reportID}

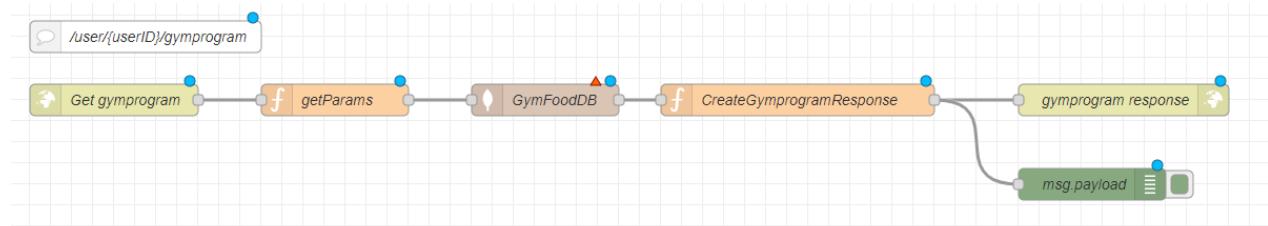


Εικόνα 96: Ροή NodeRed που αντιστοιχεί στο API call `DELETE /trainer/:trainerID/report/:reportID`

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την διαγραφή μιας αναφοράς χρήστη από τον γυμναστή. Μέσω της συνάρτησης Get Params ανακτούμε τα trainerID, reportID. Ακόμη η CreateDeleteResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

3.1.4. Ροές πόρου GymProgram

Ροή endpoint GET /user/{userID}/gymprogram

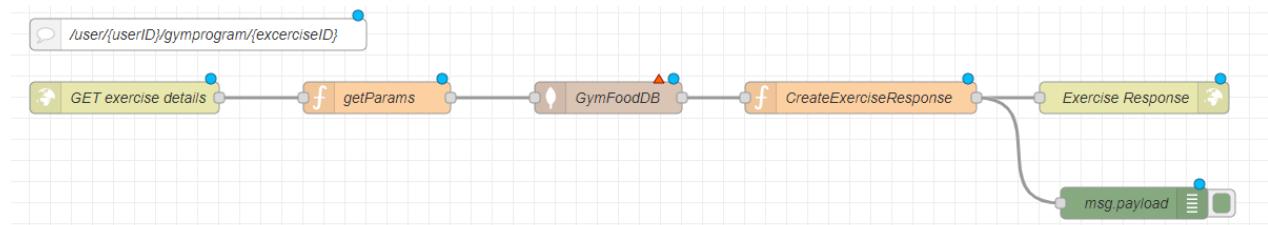


Εικόνα 97: Ροή NodeRed που αντιστοιχεί στο API call `GET /user/:userID/gymprogram`

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την δημιουργία και επιστροφή εξατομικευμένου προγράμματος διατροφής στον χρήστη με βάση τις απαιτήσεις του. Μέσω της συνάρτησης Get Params, επιστρέφεται το πρόγραμμα γυμναστικής με συγκεκριμένο userID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateGymprogramResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

3.1.5. Ροές πόρου Exercise

Ροή endpoint GET /user/{userID}/gymprogram/{excerciseID}



Εικόνα 98: Ροή NodeRed που αντιστοιχεί στο API call `GET /user/:userID/gymprogram/:excerciseID`

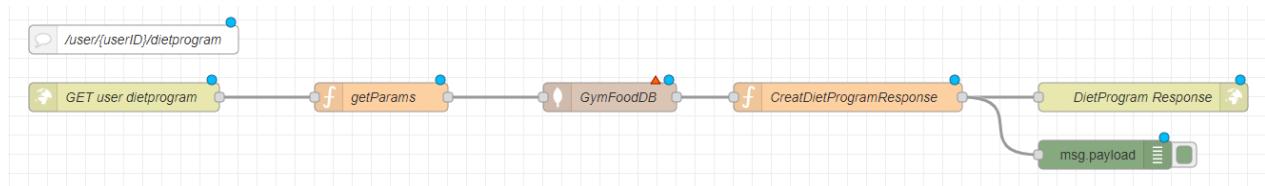
Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την επιστροφή των λεπτομερειών μιας άσκησης. Μέσω της συνάρτησης Get Params, επιστρέφεται η άσκηση με συγκεκριμένο userID και



exerciseID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateExerciseResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.

3.1.6. Ροές πόρου DietProgram

Ροή endpoint GET /user/{userID}/dietprogram



Εικόνα 99: Ροή NodeRed που αντιστοιχεί στο API call GET /user/:userID/dietprogram

Η ροή που υλοποιεί την υπηρεσία, που είναι υπεύθυνη για την δημιουργία και επιστροφή εξατομικευμένου προγράμματος διατροφής στον χρήστη με βάση τις απαιτήσεις του. Μέσω της συνάρτησης Get Params, επιστρέφεται το πρόγραμμα διατροφής με συγκεκριμένο userID, όπως ορίζεται στο URL parameter από την βάση GymFoodDB, ενώ η CreateDietProgramResponse, φέρνει το HTTP response που επιστρέφεται στην επιθυμητή μορφή.



3.2. Υλοποίηση Ιστοριών χρήστη

3.2.1. Ιστορία User “View profile details”



Εικόνα 100: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Profile Details”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει τις πληροφορίες του προφίλ του.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Get Users Details	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID} η οποία επιστρέφει τις πληροφορίες προφίλ του εκάστοτε χρήστη.
Msg.payload	debug	Τυπώνει στην κονσόλα τις πληροφορίες προφίλ του χρήστη.

3.2.2. Ιστορία User “Input profile details”



Εικόνα 101: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Input Profile Details”.

Ροή μέσω της οποίας ο χρήστης μπορεί να εισάγει τις πληροφορίες του στο προφίλ του.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Post User Details	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID} η οποία επιτρέπει την προσθήκη προσωπικών πληροφοριών στο προφίλ του εκάστοτε χρήστη.
Msg.payload	debug	Τυπώνει στην κονσόλα τις πληροφορίες προφίλ του χρήστη που θα προστεθούν.



3.2.3. Ιστορία User “Update profile details”



Εικόνα 102: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Update Profile Details”.

Ροή μέσω της οποίας ο χρήστης μπορεί να ενημερώσει τις πληροφορίες του προφίλ του.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Put User Details	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID} η οποία επιτρέπει την επεξεργασία προσωπικών πληροφοριών στο προφίλ του.
Msg.payload	debug	Τυπώνει στην κονσόλα τις ενημερωμένες πληροφορίες προφίλ του χρήστη.

3.2.4. Ιστορία User “View Recipes”



Εικόνα 103: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipes”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει τις διαθέσιμες συνταγές.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Get All Recipes	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID}/recipe οποία επιστρέφει λίστα με συνταγές.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα με τις όλες τις συνταγές.



3.2.5. Ιστορία User “View Recipe Information”



Εικόνα 104: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipe Information”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει τις λεπτομέρειες μίας συνταγής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Get Recipe Information	http-request	Κάνει κλήση προς την υπηρεσία /user/{userID}/recipe/{recipeID} η οποία επιστρέφει τις λεπτομέρειες μίας συνταγής.
Msg.payload	debug	Τυπώνει στην κονσόλα τις λεπτομέρειες της συνταγής.

3.2.6. Ιστορία User “View Gym program”



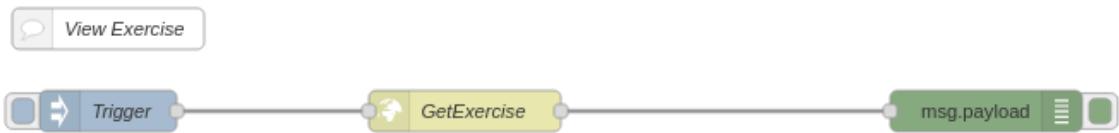
Εικόνα 105: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Gym Program”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει το πρόγραμμα γυμναστικής του.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GymProgram Request	http-request	Κάνει κλήση προς την υπηρεσία user/{userID}/gymprogram η οποία επιστρέφει ένα πρόγραμμα γυμναστικής.
Msg.payload	debug	Τυπώνει στην κονσόλα το πρόγραμμα γυμναστικής.



3.2.7. Ιστορία User “View Exercise”



Εικόνα 106: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Exercise”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει τις λεπτομέρειες μιας άσκησης.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetExercise	http-request	Κάνει κλήση προς την υπηρεσία user/{userID}/gymprogram/{exerciseID} η οποία επιστρέφει περιγραφή και επεξήγηση για μία άσκηση.
Msg.payload	debug	Τυπώνει στην κονσόλα την επεξήγηση και περιγραφή της άσκησης.

3.2.8. Ιστορία User “View Diet program”



Εικόνα 107: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Diet Program”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δει το πρόγραμμα διατροφής του.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetDietProgram	http-request	Κάνει κλήση προς την υπηρεσία user/{userID}/dietprogram η οποία επιστρέφει ένα πρόγραμμα διατροφής.
Msg.payload	debug	Τυπώνει στην κονσόλα το πρόγραμμα διατροφής με τις ανάλογες συνταγές.



3.2.9. Ιστορία User “Report Gym Program”



Εικόνα 108: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Report Gym Program”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δημιουργήσει αναφορά για το πρόγραμμα γυμναστικής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
Post GymProgram report	http-request	Κάνει κλήση προς την υπηρεσία user/{userID}/gymprogram η οποία επιτρέπει στον χρήστη να κάνει αναφορά σε ένα πρόγραμμα γυμναστικής.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Operation Successful.

3.2.10. Ιστορία User “Report Recipe”



Εικόνα 109: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Report Recipe”.

Ροή μέσω της οποίας ο χρήστης μπορεί να δημιουργήσει αναφορά για μια συνταγή.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
ReportRecipeRequest	http-request	Κάνει κλήση προς την υπηρεσία user/{userID}/recipe/{recipeID} η οποία επιτρέπει στο χρήστη να κάνει αναφορά σε μία συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Operation Successful.



3.2.11. Ιστορία Nutritionist “View All Recipe Reports”



Εικόνα 110: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View All Recipe Reports”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να δει όλες τις αναφορές συνταγών που του έχουν ανατεθεί.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetRecipesReportsRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/report οποία επιτρέπει στον διατροφολόγο να βλέπει τις αναφορές που έγιναν στις συνταγές.
Msg.payload	debug	Τυπώνει στην κονσόλα λίστα με όλες τις αναφορές συνταγών που του έχουν ανατεθεί.

3.2.12. Ιστορία Nutritionist “View Recipe Report”



Εικόνα 111: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Recipe Report”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να δει την αναφορά μιας συνταγής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetRecipeReportRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/report/{reportID} η οποία επιτρέπει στον διατροφολόγο να βλέπει αναφορά που έγινε από έναν χρήστη σε μία συγκεκριμένη συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα την αναφορά της συνταγής.



3.2.13. Ιστορία Nutritionist “Edit Recipe Report”



Εικόνα 112: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Edit Recipe Reports”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να επεξεργαστεί την αναφορά μιας συνταγής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
UpdateRecipeReportRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/report/{reportID} η οποία επιτρέπει στον διατροφολόγο να επεξεργάζεται την αναφορά που έγινε για μια συγκεκριμένη συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Recipe Report is updated

3.2.14. Ιστορία Nutritionist “Delete Recipe Report”



Εικόνα 113: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Recipe Reports”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να διαγράψει την αναφορά μιας συνταγής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
DeleteRecipeReportRequest	http-request	Κάνει κλήση προς την υπηρεσία nutrtionist/{nutrtionistID}/report/{reportID} η οποία επιτρέπει στον διατροφολόγο να διαγράψει μία αναφορά συνταγής.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Recipe Report is deleted.



3.2.15. Ιστορία Nutritionist “View Recipe”



Εικόνα 114: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Nutritionist View Recipe”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να δει μία συνταγή.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetRecipeRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/recipe/{recipeID} η οποία επιτρέπει στον διατροφολόγο να βλέπει μία συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα την συνταγή.

3.2.16. Ιστορία Nutritionist “Add Recipe”



Εικόνα 115: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Add Recipe”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να προσθέσει μία συνταγή.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
AddRecipeRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/recipe η οποία επιτρέπει στον διατροφολόγο να προσθέτει μία συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα την συνταγή η οποία προστέθηκε.



3.2.17. Ιστορία Nutritionist “Update Recipe”

Nutritionist Update/Correct a Recipe



Εικόνα 116: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Update Recipe”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να ενημερώσει τις λεπτομέρειες μιας συνταγής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
UpdateRecipeRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/recipe/{recipeID} η οποία επιτρέπει στον διατροφολόγο να ενημερώνει τις λεπτομέρειες μιας συνταγής.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Recipe is updated.

3.2.18. Ιστορία Nutritionist “Delete Recipe”

Nutritionist Delete a Recipe



Εικόνα 117: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Recipe”.

Ροή μέσω της οποίας ο διατροφολόγος μπορεί να διαγράψει μια συνταγή.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
DeleteRecipeRequest	http-request	Κάνει κλήση προς την υπηρεσία nutritionist/{nutritionistID}/recipe/{recipeID} η οποία επιτρέπει στον διατροφολόγο να διαγράψει μία συνταγή.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Recipe is deleted.



3.2.19. Ιστορία Trainer “View All Gym Program Reports”



Εικόνα 118: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View All Gym Program Reports”.

Ροή μέσω της οποίας ο γυμναστής μπορεί να δει όλες τις αναφορές προγραμμάτων γυμναστικής που του έχουν ανατεθεί.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetGymProgramReportsList	http-request	Κάνει κλήση προς την υπηρεσία trainer/{trainerID}/report η οποία επιτρέπει στον γυμναστή να βλέπει όλες τις αναφορές προγραμμάτων γυμναστικής που του έχουν ανατεθεί.
Msg.payload	debug	Τυπώνει στην κονσόλα την λίστα με όλες τις αναφορές.

3.2.20. Ιστορία Trainer “View Gym Program Report”



Εικόνα 119: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “View Gym Program Report”.

Ροή μέσω της οποίας ο γυμναστής μπορεί να δει την αναφορά ενός προγράμματος γυμναστικής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
GetGymProgramReport	http-request	Κάνει κλήση προς την υπηρεσία /trainer/{trainerID}/report/{reportID} η οποία επιτρέπει στον γυμναστή να βλέπει την αναφορά που έγινε από έναν χρήστη στο πρόγραμμα γυμναστικής του.
Msg.payload	debug	Τυπώνει στην κονσόλα την αναφορά του προγράμματος γυμναστικής.



3.2.21. Ιστορία Trainer “Edit Gym Program Report”



Εικόνα 120: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Edit Gym Program Report”.

Ροή μέσω της οποίας ο γυμναστής μπορεί να επεξεργαστεί την αναφορά ενός προγράμματος γυμναστικής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
PutGymProgramReport	http-request	Κάνει κλήση προς την υπηρεσία /trainer/{trainerID}/report/{reportID} η οποία επιτρέπει στον γυμναστή να επεξεργάζεται μια αναφορά που έγινε σε ένα πρόγραμμα γυμναστικής.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Gym Program Report is updated.

3.2.22. Ιστορία Trainer “Delete Gym Program Report”



Εικόνα 121: Ροή NodeRed που αντιστοιχεί στην ιστορία χρήστη “Delete Gym Program Report”.

Ροή μέσω της οποίας ο γυμναστής μπορεί να διαγράψει την αναφορά ενός προγράμματος γυμναστικής.

Όνομα κόμβου	Τύπος κόμβου	Περιγραφή
Trigger	Inject	Χρησιμοποιείται για την ενεργοποίηση της εκτέλεσης της ροής.
DeleteGymProgramReport	http-request	Κάνει κλήση προς την υπηρεσία /trainer/{trainerID}/report/{reportID} η οποία επιτρέπει στον γυμναστή να διαγράψει μια αναφορά προγράμματος γυμναστικής.
Msg.payload	debug	Τυπώνει στην κονσόλα το μήνυμα Successful Gym Program Report is deleted.



Παράτημα I – 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.
- Οι πόροι που σχετίζονται με την διαχείριση χρηστών δεν τεκμηριώνονται, αφού δεν αποτελούν μέρος της εργασίας, σε ένα ρεαλιστικό σύστημα όμως είναι απαραίτητοι.
- Απαιτείται επίσης σχεδιασμός των ροών που θα χρησιμοποιούνται για το front-end μέρος του συστήματος.