

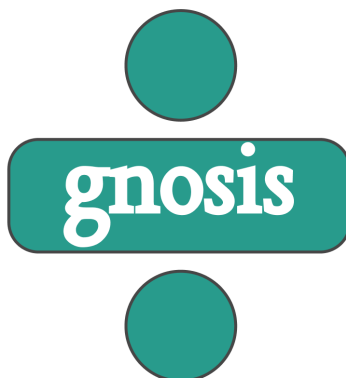


8<sup>ο</sup> Εξάμηνο

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



Άνοιξη 2024



**/gnosis**

Giving a piece of advice for your health.

# Σχεδίαση και Ανάπτυξη Διεπαφών REST του Συστήματος

Version 1.0

30/5/2024

Θεόδωρος Βλαχάκης [vlathegeo@ece.auth.gr](mailto:vlathegeo@ece.auth.gr)

Κωνσταντίνος Καλαμάρας [kkalamar@ece.auth.gr](mailto:kkalamar@ece.auth.gr)

Κωνσταντίνος Φώτιος Παπαδάκης [kpapadak@ece.auth.gr](mailto:kpapadak@ece.auth.gr)

Νικόλαος Ευαγγελίδης [nievange@ece.auth.gr](mailto:nievange@ece.auth.gr)



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



8<sup>ο</sup> Εξάμηνο

Άνοιξη 2024

## Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Α. Συμεωνίδης	*	<a href="mailto:asymeon@issel.ee.auth.gr">asymeon@issel.ee.auth.gr</a>
Θεόδωρος Βλαχάκης	50	<a href="mailto:vlathegeo@ece.auth.gr">vlathegeo@ece.auth.gr</a>
Κωνσταντίνος Καλαμάρας	50	<a href="mailto:kkalamar@ece.auth.gr">kkalamar@ece.auth.gr</a>
Κωνσταντίνος Φώτιος Παπαδάκης	50	<a href="mailto:kpapadak@ece.auth.gr">kpapadak@ece.auth.gr</a>
Νικόλαος Ευαγγελίδης	50	<a href="mailto:nievange@ece.auth.gr">nievange@ece.auth.gr</a>



## Πίνακας Περιεχομένων

<b>Πίνακας Περιεχομένων</b>	<b>3</b>
<b>1. Αναγνώριση Πόρων (Resources) Συστήματος</b>	<b>5</b>
1.1 Πόροι One-Off	5
1.2 Πόροι Δεδομένων	5
1.3 Αλγοριθμικοί Πόροι	6
<b>2. Τεκμηρίωση REST API</b>	<b>7</b>
2.1 Πόρος Patient	7
2.1.1 Μοντέλο Δεδομένων Patient	7
2.1.2 Τερματικό GET πόρου Patient για συγκεκριμένο patientID	9
2.1.2.1 Παράμετροι εισόδου	9
2.1.2.2 Μοντέλο δεδομένων εξόδου	10
2.2 Πόρος HealthRequest	11
2.2.1 Μοντέλο Δεδομένων HealthRequest	12
2.2.2 Τερματικό POST πόρου HealthRequest για συγκεκριμένο patientID	12
2.2.2.1 Παράμετροι εισόδου	12
2.2.2.2 Μοντέλο δεδομένων εξόδου	13
2.2.3 Τερματικό PUT πόρου HealthRequest για συγκεκριμένο HealthRequestID	14
2.2.2.1 Παράμετροι εισόδου	14
2.2.2.2 Μοντέλο δεδομένων εξόδου	14
2.3 Πόρος MedicalHistoryDocument	15
2.3.1 Μοντέλο Δεδομένων MedicalHistoryDocument	16
2.3.2 Τερματικό POST πόρου MedicalHistoryDocument για συγκεκριμένο patientID	16
2.3.2.1 Παράμετροι εισόδου	16
2.2.2.2 Μοντέλο δεδομένων εξόδου	17
2.4 Πόρος Appointment	17
2.4.1 Μοντέλο Δεδομένων Appointment	18
2.4.2 Τερματικό POST πόρου Appointment για συγκεκριμένο patientID	18
2.4.2.1 Παράμετροι εισόδου	18
2.2.2.2 Μοντέλο δεδομένων εξόδου	19
2.5 Πόρος DeleteAccount (One-Off)	19
2.5.1 Τερματικό DELETE πόρου DeleteAccount για συγκεκριμένο patientID	20
2.5.1.1 Παράμετροι εισόδου	20
2.5.1.2 Μοντέλο δεδομένων εξόδου	20
2.6 Πόρος RecommendDoctors (Αλγοριθμικός)	21
2.6.1 Τερματικό GET πόρου RecommendDoctors για συγκεκριμένη ειδικότητα γιατρού doctorSpecialty	22



2.6.1.1 Παράμετροι εισόδου	22
2.6.1.2 Μοντέλο δεδομένων εξόδου	22
2.7 Πόρος RequestSpecificDoctor (Αλγοριθμικός)	23
2.7.1 Τερματικό GET πόρου RequestSpecificDoctor για συγκεκριμένο όνομα γιατρού doctorName	23
2.7.1.1 Παράμετροι εισόδου	23
2.7.1.2 Μοντέλο δεδομένων εξόδου	24
2.8 Πόρος Notification	25
2.8.1 Μοντέλο δεδομένων	25
2.8.2 Τερματικό POST /patient/{patientId}/notification	25
2.8.2.1 Παράμετροι εισόδου	25
2.8.2.2 Μοντέλο δεδομένων εξόδου	26
2.8.3 Τερματικό GET /patient/{patientId}/notification/{notificationId}	26
2.8.3.1 Παράμετροι εισόδου	27
2.8.3.2 Μοντέλο δεδομένων εξόδου	27
2.8.4 Τερματικό PUT /patient/{patientId}/notification/{notificationId}	27
2.8.4.1 Παράμετροι εισόδου	27
2.8.4.2 Μοντέλο δεδομένων εξόδου	28
2.8.5 Τερματικό DELETE patient/{patientId}/notification/{notificationId}	28
2.8.5.1 Παράμετροι εισόδου	29
2.8.5.2 Μοντέλο δεδομένων εξόδου	29
2.9 Πόρος HealthRequestList	29
2.9.1 Μοντέλο δεδομένων	30
2.9.2 Τερματικό GET /patient/{patientId}/healthRequestList	30
2.9.2.1 Παράμετροι εισόδου	31
2.9.2.2 Μοντέλο δεδομένων εξόδου	31
2.9.3 Τερματικό GET /patient/{patientId}/healthRequestList/ healthRequest/{healthRequestId}	32
2.9.3.1 Παράμετροι εισόδου	32
2.9.3.2 Μοντέλο δεδομένων εξόδου	32
2.10 Πόρος FetchList (Αλγοριθμικός)	33
2.10.1 Τερματικό GET /patient/{patientId}/fetchList	33
2.10.1.1 Παράμετροι εισόδου	33
2.10.1.2 Μοντέλο δεδομένων εξόδου	33
2.11 Πόρος UnfinishedHealthRequest	34
2.11.1 Μοντέλο Δεδομένων UnfinishedHealthRequest	35
2.11.2 Τερματικό POST /patient/{patientId}/healthRequestList/ healthRequest/{healthRequestId}/unfinishedHealthRequest	35
2.11.2.1 Παράμετροι εισόδου	35



2.11.2.2 Μοντέλο δεδομένων εξόδου	36
2.12 Πόρος Diagnosis	37
2.12.1 Μοντέλο Δεδομένων Diagnosis	38
2.12.2 Τερματικό POST πόρου Diagnosis για συγκεκριμένο DoctorID, PatientID και HealthRequestID	38
2.12.2.1 Παράμετροι εισόδου	38
2.12.2.2 Μοντέλο δεδομένων εξόδου	39
2.12.3 Τερματικό GET πόρου Diagnosis PatientID και HealthRequestID	40
2.12.3.1 Παράμετροι εισόδου	40
2.12.3.2 Μοντέλο δεδομένων εξόδου	40
2.13 Πόρος GenerateAiEstimation (Αλγοριθμικός)	41
2.13.1 Τερματικό GET πόρου GenerateAiEstimation για συγκεκριμένο HealthRequestID και συγκεκριμένο PatientID	41
2.13.1.1 Παράμετροι εισόδου	41
2.13.1.2 Μοντέλο δεδομένων εξόδου	42



## 1. Αναγνώριση Πόρων (Resources) Συστήματος

### 1.1 Πόροι One-Off

Κλάση BEC	Πόρος REST	Endpoints (HTTP Verbs)
Patient(patient)	/deleteAccount	DELETE

### 1.2 Πόροι Δεδομένων

Κλάση BEC	Πόρος REST	Endpoints (HTTP Verbs)
Patient (doctor)	/patient/{patientId}	GET
HealthRequest (patient)	/patient/{patientId}/healthRequest	POST
HealthRequest (patient)	/healthRequest/{healthRequestId}	PUT
UnfinishedHealthRequest(patient)	/patient/{patientId}/healthRequestList/healthRequest/{healthRequestId}/unfinishedHealthRequest	POST
MedicalHistory Document(patient)	/patient/{patientId}/medicalHistoryDocument	POST
Appointment (patient)	/patient/{patientId}/appointment	POST
Diagnosis(doctor)	/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis	POST
Diagnosis(patient)	/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis	GET
Notification(patient)	/patient/{patientId}/notification	POST
Notification(patient)	/patient/{patientId}/notification/{notificationId}	GET,PUT,



		DELETE
HealthRequestList (patient)	/patient/{patientId}/healthRequestList	POST, GET
HealthRequest (patient)	/patient/{patientId}/healthRequestList/healthRequest/ {healthRequestId}	GET
Diagnosis (patient)	/patient/{patientId}/healthRequestList/healthRequest/ {healthRequestId}/diagnosis	GET

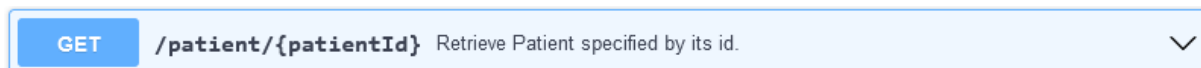
### 1.3 Αλγοριθμικοί Πόροι

Κλάση BEC	Πόρος REST	Endpoints (HTTP Verbs)
(patient)	/healthRequest/generateAiEstimation	GET
(patient)	/healthRequest/doctorChoice/recommendDoctors	GET
(patient)	/healthRequest/doctorChoice/requestSpecificDoctor	GET
(patient)	/patient/{patientId}/fetchList	GET



## 2. Τεκμηρίωση REST API

### 2.1 Πόρος Patient



Εικόνα 1: Περιγραφή του API του πόρου Patient (από swagger-editor).

Endpoint	User Story	FR
/patient/{patientId}	-	-

Ο πόρος Patient είναι ένας από τους βασικότερους πόρους δεδομένων, που αφορούν το υπό ανάπτυξη σύστημα \gnosis. Ο πόρος αυτός αντιστοιχεί σε έναν “ασθενή” και χρησιμοποιείται για να δημιουργήσει, να διαγράψει και να ενημερώσει τον λογαριασμό ενός χρήστη “ασθενή” στο σύστημα. Παρά τη σημαντικότητα αυτού του πόρου σε επίπεδο ολοκληρωμένης υλοποίησης του συστήματος, εφόσον στο συγκεκριμένο παραδοτέο θα υλοποιήσουμε μόνο τους πόρους, με τα αντίστοιχα endpoints τους που αφορούν τα User Stories (Features) και αντίστοιχα τα Functional Requirements του πρώτου παραδοτέου, για τα οποία υλοποιήθηκαν οι αντίστοιχες κλάσεις στο δεύτερο παραδοτέο, τότε για τον πόρο Patient αυτόν καθεαυτόν δεν έχουμε τερματικά που να σχετίζονται με κάποιο User Story και αντίστοιχα κάποιο FR. Παραθέτουμε, μόνο ενδεικτικά το endpoint GET του πόρου Patient για συγκεκριμένο patientID (που αφορούν, δηλαδή συγκεκριμένο χρήστη “ασθενή” στο σύστημα), λόγω της σημαντικότητας του πόρου αυτού, χωρίς να το συσχετίζουμε με κάποιο User Story και FR.

#### 2.1.1 Μοντέλο Δεδομένων Patient

Ο πόρος Patient είναι, ουσιαστικά, η αντιστοίχιση της κλάσης (BEC) Patient, που είχε οριστεί στη “μονολιθική” σχεδίαση του συστήματος. Όλα του τα χαρακτηριστικά/ιδιότητες αντιπροσωπεύουν τα αντίστοιχα attributes της κλάσης Patient και φαίνονται στο παρακάτω μοντέλο δεδομένων. Παρατίθενται, ακόμη, το μοντέλο δεδομένων του πόρου HealthRequest, του πόρου MedicalHistoryDocument και του πόρου Diagnosis, προκειμένου να παρουσιαστούν τα χαρακτηριστικά/ιδιότητες των στοιχείων που περιλαμβάνονται στους πίνακες patientRequests, patientMedicalHistoryDocuments, patientReceivedDiagnoses, αντίστοιχα, οι οποίοι ορίζονται ως χαρακτηριστικά του πόρου Patient.



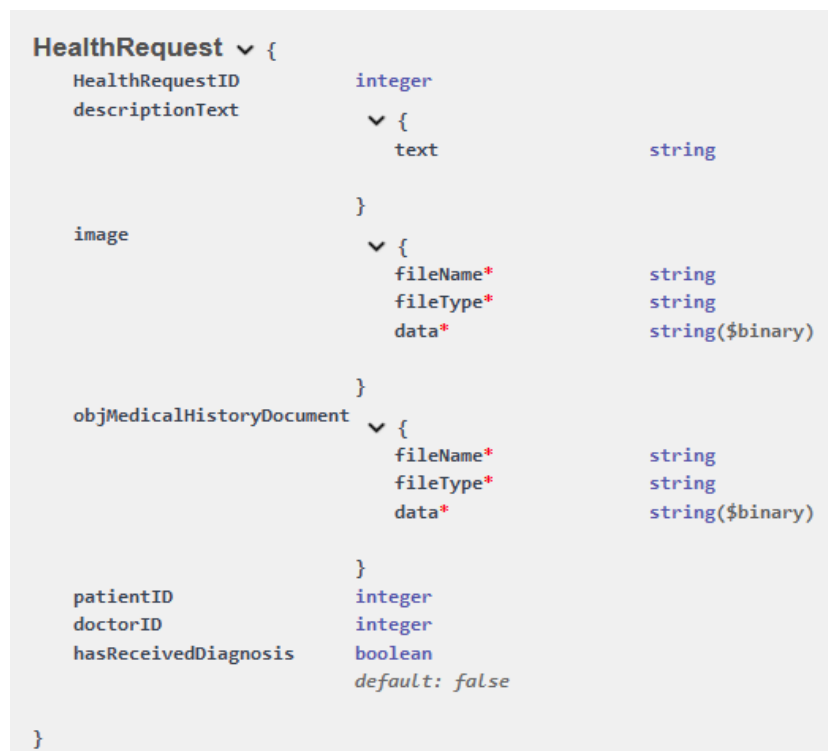


```
Patient {  
  patientID          integer  
  patientName        string  
  patientAge         integer  
  patientGender      string  
  patientRequests    [HealthRequest > {...}]  
  patientMedicalHistoryDocuments [MedicalHistoryDocument > {...}]  
  patientReceivedDiagnoses [Diagnosis > {...}]  
}
```

Εικόνα 2: Μοντέλο δεδομένων πόρου Patient.

```
MedicalHistoryDocument {  
  MedHistoryDocID    integer  
  document            {  
    fileName*        string  
    fileType*        string  
    data*            string($binary)  
  }  
  documentName       string  
  patientID          integer  
}
```

Εικόνα 3: Μοντέλο δεδομένων πόρου MedicalHistoryDocument.



Εικόνα 4: Μοντέλο δεδομένων πόρου HealthRequest.



Εικόνα 5: Μοντέλο δεδομένων πόρου Diagnosis.

### 2.1.2 Τερματικό GET πόρου Patient για συγκεκριμένο patientID

Αυτή η τερματική υπηρεσία (GET) του πόρου Patient χρησιμοποιείται για ανάκτηση όλης της πληροφορίας που σχετίζεται με έναν συγκεκριμένο χρήστη “ασθενή”, δεδομένου του ID του (που δηλώνεται ως path parameter), καθώς ο κάθε χρήστης “ασθενής” είναι ένα “αντικείμενο”, στο οποίο αντιστοιχεί ο πόρος Patient για συγκεκριμένο patientID.

#### 2.1.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου είναι μόνο το ID του χρήστη “ασθενή”, για τον οποίο ζητάται να ανακτηθεί η πληροφορία που περιέχει (ως μια πραγματοποίηση του πόρου Patient).



### Parameters

Name	Description
<b>patientId</b> * required string (path)	The id of the Patient. <input type="text" value="patientId"/>

Εικόνα 6: Παράθεση παραμέτρων του τερματικού GET /patient/{patientId} του πόρου Patient.

### 2.1.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία GET /patient/{patientId} αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση αφορά όλη την πληροφορία που σχετίζεται με τον χρήστη “ασθενή”, ο οποίος αντιστοιχεί στο συγκεκριμένο ID (όπως αυτή περιγράφηκε παραπάνω).

Responses		
Code	Description	Links
200	Successful operation. Patient's account information is retrieved <small>Media type</small> <input type="text" value="application/json"/> <small>Controls Accept header.</small> <small>Example Value   Schema</small>	No links
<pre>{   "patientID": 0,   "patientName": "string",   "patientAge": 0,   "patientGender": "string",   "patientRequests": [     {       "HealthRequestID": 0,       "descriptionText": {         "text": "string"       },       "image": {         "fileName": "string",         "fileType": "string",         "data": "string"       },       "objMedicalHistoryDocument": {         "fileName": "string",         "fileType": "string",         "data": "string"       },       "patientID": 0,       "doctorID": 0,     }   ] }</pre>		

Εικόνα 7α: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού GET /patient/{patientId} του πόρου Patient.



Responses

Code	Description	Links
200	Successful operation. Patient's account information is retrieved	No links

Media type  
application/json

Controls Accept header.

Example Value | Schema

```
{  "hasReceivedDiagnosis": false,
},
"patientMedicalHistoryDocuments": [
  {
    "MedHistoryDocID": 0,
    "document": {
      "fileName": "string",
      "fileType": "string",
      "data": "string"
    },
    "documentName": "string",
    "patientID": 0
  }
],
"patientReceivedDiagnoses": [
  {
    "diagnosisText": {
      "text": "string"
    }
  }
]
}
```

Εικόνα 7β (συμπληρωματική εικόνα): Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού GET /patient/{patientId} του πόρου Patient.

## 2.2 Πόρος HealthRequest

POST	/patient/{patientId}/healthRequest	Create a Health Request and add it to Patient's account	✓
PUT	/healthRequest/{healthRequestId}	Update a specific Health Request for which the Patient has received a diagnosis	✓

Εικόνα 8: Περιγραφή του API του πόρου HealthRequest (από swagger-editor).

Endpoint	User Story	FR
/patient/{patientId}/healthRequest	1, 2	1, 2, 5
/healthRequest/{healthRequestId}	4	10

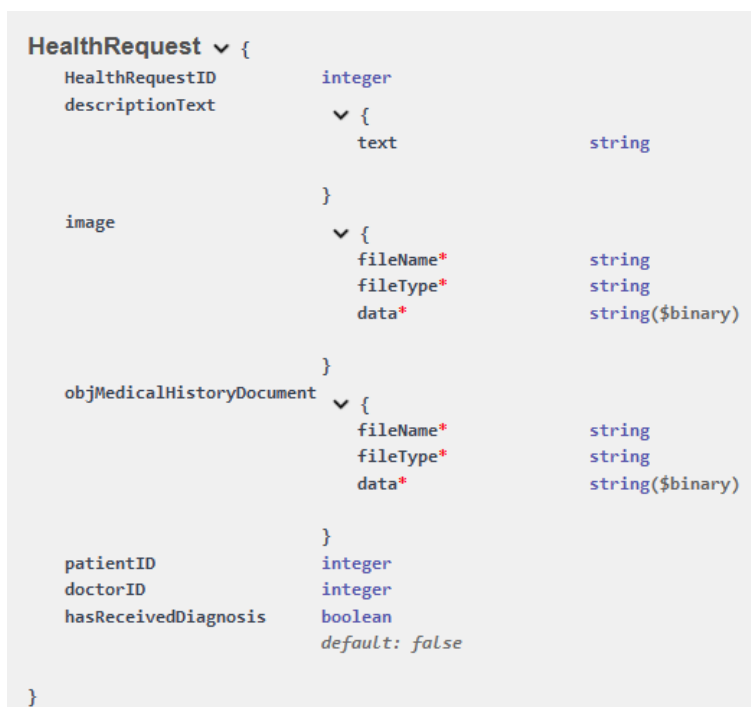
Ο πόρος HealthRequest αντιστοιχεί σε ένα “ιατρικό αίτημα” ενός “ασθενή” και χρησιμοποιείται για να δημιουργήσει, να διαγράψει, να ενημερώσει και να δώσει πληροφορίες για ένα ή



περισσότερα “ιατρικά αιτήματα” ενός ή περισσότερων “ασθενών”, ανάλογα με το τερματικό που χρησιμοποιείται. Στο συγκεκριμένο παραδοτέο θα υλοποιήσουμε μόνο τα αντίστοιχα endpoints του πόρου που αφορούν τα User Stories (Features) και αντίστοιχα τα Functional Requirements του πρώτου παραδοτέου, για τα οποία υλοποιήθηκαν οι αντίστοιχες κλάσεις στο δεύτερο παραδοτέο.

### 2.2.1 Μοντέλο Δεδομένων HealthRequest

Ο πόρος HealthRequest είναι, ουσιαστικά, η αντιστοίχιση της κλάσης (BEC) HealthRequest, που είχε οριστεί στη “μονολιθική” σχεδίαση του συστήματος. Όλα του τα χαρακτηριστικά/ιδιότητες αντιπροσωπεύουν τα αντίστοιχα attributes της κλάσης HealthRequest και φαίνονται στο παρακάτω μοντέλο δεδομένων.



Εικόνα 4: Μοντέλο δεδομένων πόρου HealthRequest.

### 2.2.2 Τερματικό POST πόρου HealthRequest για συγκεκριμένο patientID

Αυτή η τερματική υπηρεσία (POST) του πόρου HealthRequest χρησιμοποιείται για την δημιουργία ενός “ιατρικού αιτήματος” ενός συγκεκριμένου χρήστη “ασθενή”, δεδομένου του ID του (που δηλώνεται ως path parameter). Γίνεται εισαγωγή του “ιατρικού αιτήματος” στο λογαριασμό του “ασθενή”.

#### 2.2.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του χρήστη “ασθενή”, το οποίο δηλώνεται ως path parameter, καθώς και ένα Request Body που περιλαμβάνει όλη την απαραίτητη πληροφορία για την δημιουργία ενός “ιατρικού αιτήματος”.



Parameters

Try it out

Name	Description
<b>patientId</b> * required string (path)	The id of the Patient by whom the Health Request is made. <input type="text" value="patientId"/>

Request body required

application/json

HealthRequest model

Example Value | Schema

```
{
  "HealthRequestID": 0,
  "descriptionText": {
    "text": "string"
  },
  "image": {
    "fileName": "string",
    "fileType": "string",
    "data": "string"
  },
  "objMedicalHistoryDocument": {
    "fileName": "string",
    "fileType": "string",
    "data": "string"
  },
  "patientID": 0,
  "doctorID": 0,
  "hasReceivedDiagnosis": false
}
```

Εικόνα 9: Παράθεση παραμέτρων του τερματικού POST /patient/{patientId}/healthRequest του πόρου HealthRequest.

### 2.2.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία POST /patient/{patientId}/healthRequest αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση τους αφορά έναν κωδικό και ένα μήνυμα, όπως ορίζεται από την επιτυχή δημιουργία του “ιατρικού αιτήματος” που αφορά συγκεκριμένο “ασθενή”.

Responses

Code	Description	Links
201	Successful operation. Patient has made his/her Health Request	No links

Media type  
application/json

Controls Accept header.

Example Value | Schema

```
{
  "message": "Successfully Created!"
}
```



Εικόνα 10: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού POST /patient/{patientId}/healthRequest του πόρου HealthRequest.

### 2.2.3 Τερματικό PUT πόρου HealthRequest για συγκεκριμένο HealthRequestId

Αυτή η τερματική υπηρεσία (PUT) του πόρου HealthRequest χρησιμοποιείται για την ενημέρωση της πληροφορίας που περιέχει ένα συγκεκριμένο “ιατρικό αίτημα” στο σύστημα, όταν για το συγκεκριμένο “ιατρικό αίτημα” ληφθεί κάποια διάγνωση. Ενημερώνεται το αντίστοιχο χαρακτηριστικό του.

#### 2.2.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του “ιατρικού αιτήματος”, το οποίο δηλώνεται ως path parameter, μια boolean τιμή hasReceivedDiagnosis που δηλώνει αν έχει ληφθεί “διάγνωση” για το “ιατρικό αίτημα”, καθώς και ένα Request Body που περιλαμβάνει το κείμενο της διάγνωσης του “ιατρικού αιτήματος”

Parameters Try it out

Name	Description
<b>healthRequestId</b> * required string (path)	The id of the Health Request <input type="text" value="healthRequestId"/>
<b>hasReceivedDiagnosis</b> * required boolean (query)	The boolean value that indicates if a Health Request has received a diagnosis <input type="text" value="--"/>

**Request body** required application/json

A request (body parameters) to update the descriptionText property of a HealthRequest

Example Value | Schema

```
{  
  "text": "string"  
}
```

Εικόνα 11: Παράθεση παραμέτρων του τερματικού PUT /healthRequest/{healthRequestId} του πόρου HealthRequest.

#### 2.2.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία PUT /healthRequest/{healthRequestId} αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση τους αφορά έναν κωδικό και ένα μήνυμα, όπως ορίζεται από την επιτυχή ενημέρωση του “ιατρικού αιτήματος”, ενώ επίσης περιλαμβάνεται και το μοντέλο που φέρει την ανανεωμένη πληροφορία του “ιατρικού αιτήματος”.



Responses		
Code	Description	Links
200	Successful operation. The Health Request is marked as checked by a doctor, with diagnosis	No links
Media type		
application/json		
Controls Accept header.		
Example Value   Schema		
<pre>{   "HealthRequestID": 0,   "descriptionText": {     "text": "string"   },   "image": {     "fileName": "string",     "fileType": "string",     "data": "string"   },   "objMedicalHistoryDocument": {     "fileName": "string",     "fileType": "string",     "data": "string"   },   "patientID": 0,   "doctorID": 0,   "hasReceivedDiagnosis": false }</pre>		

Εικόνα 12: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού PUT /healthRequest/{healthRequestId} του πόρου HealthRequest.

## 2.3 Πόρος MedicalHistoryDocument

POST	/patient/{patientId}/medicalHistoryDocument	Create a Medical History Document and add it to Patient's account		←	✓
------	---	---	--	---	---

Εικόνα 13: Περιγραφή του API του πόρου MedicalHistoryDocument (από swagger-editor).

Endpoint	User Story	FR
/patient/{patientId}/medicalHistoryDocument	2	5

Ο πόρος MedicalHistoryDocument αντιστοιχεί σε ένα “αρχείο ιατρικού ιστορικού” ενός “ασθενή” και χρησιμοποιείται για να δημιουργήσει, να διαγράψει, να ενημερώσει και να δώσει πληροφορίες για ένα ή περισσότερα “αρχεία ιατρικού ιστορικού” ενός ή περισσότερων “ασθενών”, ανάλογα με το τερματικό που χρησιμοποιείται. Στο συγκεκριμένο παραδοτέο θα υλοποιήσουμε μόνο τα αντίστοιχα endpoints του πόρου που αφορούν τα User Stories (Features) και αντίστοιχα τα Functional Requirements του πρώτου παραδοτέου, για τα οποία υλοποιήθηκαν οι αντίστοιχες κλάσεις στο δεύτερο παραδοτέο. Τότε, για τον πόρο στον οποίο αντιστοιχεί κάθε “αρχείο ιατρικού ιστορικού” στο σύστημα, σε επίπεδο υλοποίησης API, θα ορίσουμε το τερματικό POST του πόρου /patient/{patientId}/medicalHistoryDocument, για την κάλυψη της FR3

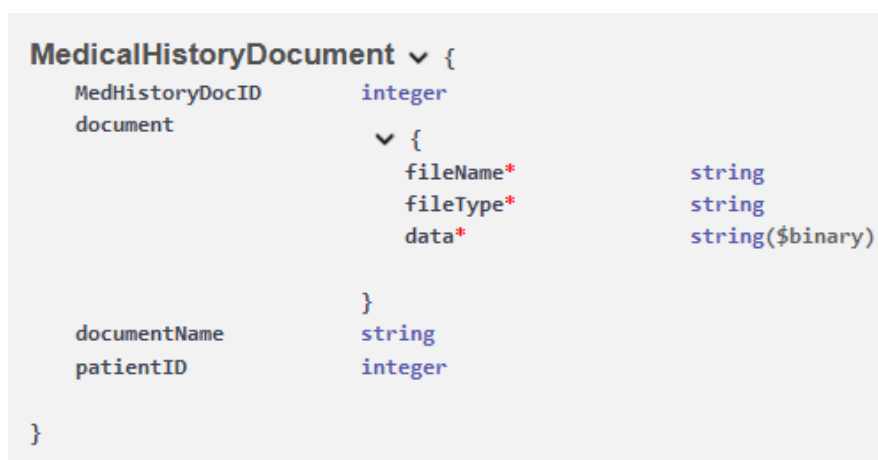




και κατά συνέπεια την ικανοποίηση τμήματος που αφορά το User Story 2.

### 2.3.1 Μοντέλο Δεδομένων MedicalHistoryDocument

Ο πόρος MedicalHistoryDocument είναι, ουσιαστικά, η αντιστοίχιση της κλάσης (BEC) MedicalHistoryDocument, που είχε οριστεί στη “μονολιθική” σχεδίαση του συστήματος. Όλα του τα χαρακτηριστικά/ιδιότητες αντιπροσωπεύουν τα αντίστοιχα attributes της κλάσης MedicalHistoryDocument και φαίνονται στο παρακάτω μοντέλο δεδομένων.



Εικόνα 3: Μοντέλο δεδομένων πόρου MedicalHistoryDocument.

### 2.3.2 Τερματικό POST πόρου MedicalHistoryDocument για συγκεκριμένο patientID

Αυτή η τερματική υπηρεσία (POST) του πόρου MedicalHistoryDocument χρησιμοποιείται για την δημιουργία ενός “αρχείου ιατρικού ιστορικού” ενός συγκεκριμένου χρήστη “ασθενή”, δεδομένου του ID του (που δηλώνεται ως path parameter) και εισαγωγή αυτού του “αρχείου ιατρικού ιστορικού” στο λογαριασμό του “ασθενή”, στο σύστημα..

#### 2.3.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του χρήστη “ασθενή”, το οποίο δηλώνεται ως path parameter, καθώς και ένα Request Body που περιλαμβάνει όλη την απαραίτητη πληροφορία για την δημιουργία ενός “αρχείου ιατρικού ιστορικού”.



Parameters Try it out

Name	Description
<b>patientId</b> * required	The id of the Patient who imports a Medical History Document to Database.
string (path)	<input type="text" value="patientId"/>

Request body **required** application/json

MedicalHistoryDocument model

Example Value | Schema

```
{  "MedHistoryDocID": 0,  "document": {    "fileName": "string",    "fileType": "string",    "data": "string"  },  "documentName": "string",  "patientID": 0}
```

Εικόνα 14: Παράθεση παραμέτρων του τερματικού POST /patient/{patientId}/medicalHistoryDocument του πόρου MedicalHistoryDocument.

### 2.2.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία POST /patient/{patientId}/medicalHistoryDocument αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση τους αφορά έναν κωδικό και ένα μήνυμα, όπως ορίζεται από την επιτυχή δημιουργία του “ιατρικού αιτήματος” που αφορά συγκεκριμένο “ασθενή”.

Responses

Code	Description	Links
201	Successful operation. Patient has uploaded a Medical History Document	No links

Media type application/json

Controls Accept header.

Example Value | Schema

```
{  "message": "Successfully Created!"}
```

Εικόνα 15: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού POST /patient/{patientId}/medicalHistoryDocument του πόρου MedicalHistoryDocument.

## 2.4 Πόρος Appointment

**POST** /patient/{patientId}/appointment ✓

Create an Appointment (medical appointment) that Patient requests to a doctor

Εικόνα 16: Περιγραφή του API του πόρου Appointment (από swagger-editor).



Endpoint	User Story	FR
/patient/{patientId}/appointment	3	4

Ο πόρος Appointment αντιστοιχεί σε ένα “ιατρικό ραντεβού” -το οποίο αιτείται ένας “ασθενής” από έναν συγκεκριμένο γιατρό- και χρησιμοποιείται για να δημιουργήσει, να διαγράψει, να ενημερώσει και να δώσει πληροφορίες για ένα ή περισσότερα “ιατρικά ραντεβού” που σχετίζονται με έναν ή περισσότερους “ασθενείς”, έναν ή περισσότερους γιατρούς, κάποια συγκεκριμένη ώρα ή μέρα, ανάλογα με το τερματικό που χρησιμοποιείται. Στο συγκεκριμένο παραδοτέο θα υλοποιήσουμε μόνο τα αντίστοιχα endpoints του πόρου που αφορούν τα User Stories (Features) και αντίστοιχα τα Functional Requirements του πρώτου παραδοτέου, για τα οποία υλοποιήθηκαν οι αντίστοιχες κλάσεις στο δεύτερο παραδοτέο. Τότε, για τον πόρο στον οποίο αντιστοιχεί κάθε “αρχείο ιατρικού ιστορικού” στο σύστημα, σε επίπεδο υλοποίησης API, θα ορίσουμε το τερματικό POST του πόρου /patient/{patientId}/appointment, για την κάλυψη της FR4 και κατά συνέπεια την ικανοποίηση του User Story 3.

#### 2.4.1 Μοντέλο Δεδομένων Appointment

Ο πόρος Appointment είναι, ουσιαστικά, η αντιστοίχιση της κλάσης (BEC) Appointment, που είχε οριστεί στη “μονολιθική” σχεδίαση του συστήματος. Όλα του τα χαρακτηριστικά/ιδιότητες αντιπροσωπεύουν τα αντίστοιχα attributes της κλάσης Appointment και φαίνονται στο παρακάτω μοντέλο δεδομένων.

```
Appointment {  
  doctorID      integer  
  date          integer  
  time          integer  
  patientID     integer  
  appointmentStatus string  
  appointmentID integer  
}
```

Εικόνα 17: Μοντέλο δεδομένων πόρου Appointment.

#### 2.4.2 Τερματικό POST πόρου Appointment για συγκεκριμένο patientID

Αυτή η τερματική υπηρεσία (POST) του πόρου Appointment χρησιμοποιείται για την δημιουργία ενός “ιατρικού ραντεβού”, το οποίο αιτείται ένας συγκεκριμένος χρήστης “ασθενής”, δεδομένου του ID του (που δηλώνεται ως path parameter) από έναν συγκεκριμένο γιατρό για συγκεκριμένη ώρα και μέρα (αναμένοντας την απάντηση του γιατρού), με τα στοιχεία αυτά να δίνονται ως πληροφορία στο Request Body της μεθόδου.

##### 2.4.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του χρήστη “ασθενή”, το οποίο δηλώνεται ως path parameter, καθώς και ένα Request Body που περιλαμβάνει όλη την απαραίτητη πληροφορία για



την δημιουργία ενός “ιατρικού ραντεβού”.

Parameters

Try it out

Name	Description
<b>patientId</b> * required string (path)	The id of the Patient who requests the Appointment. <input type="text" value="patientId"/>

Request body application/json

Appointment model

Example Value | Schema

```
{
  "doctorID": 0,
  "date": 0,
  "time": 0,
  "patientID": 0,
  "appointmentStatus": "string",
  "appointmentID": 0
}
```

Εικόνα 18: Παράθεση παραμέτρων του τερματικού POST /patient/{patientId}/appointment του πόρου Appointment.

#### 2.2.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία POST /patient/{patientId}/appointment αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση τους αφορά έναν κωδικό και ένα μήνυμα, όπως ορίζεται από την επιτυχή δημιουργία του “ιατρικού ραντεβού”, ενώ επίσης περιλαμβάνεται και το μοντέλο που φέρει την πληροφορία του “ιατρικού ραντεβού”.

Responses

Code	Description	Links
201	Successful operation. Patient requests an Appointment.	No links

Media type application/json

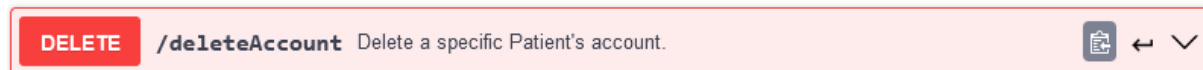
Controls Accept header.

Example Value | Schema

```
{
  "doctorID": 0,
  "date": 0,
  "time": 0,
  "patientID": 0,
  "appointmentStatus": "string",
  "appointmentID": 0
}
```

Εικόνα 19: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού POST /patient/{patientId}/medicalHistoryDocument του πόρου MedicalHistoryDocument.

## 2.5 Πόρος DeleteAccount (One-Off)



Εικόνα 20: Περιγραφή του API του πόρου DeleteAccount (από swagger-editor).

Endpoint	User Story	FR
/deleteAccount	-	-

Ο πόρος DeleteAccount είναι ένας one-off πόρος, που χρησιμοποιείται για να ορίσει μια σημαντική λειτουργία της εφαρμογής, αυτή της διαγραφής του λογαριασμού ενός χρήστη “ασθενή”. Για τον πόρο DeleteAccount ορίζεται το endpoint DELETE, το οποίο χρησιμοποιεί ο “ασθενής” για να εκτελέσει ένα request διαγραφής του λογαριασμού του. Το endpoint εκτελεί τη διαδικασία διαγραφής και επιστρέφει ένα μήνυμα στον χρήστη.

Ο πόρος DeleteAccount είναι ένας από τους βασικότερους one-off πόρους, που αφορούν το υπό ανάπτυξη σύστημα \gnosis. Σε επίπεδο υλοποίησης των User Stories (Features) και αντίστοιχα των Functional Requirements του πρώτου παραδοτέου, για τα οποία υλοποιήθηκαν οι αντίστοιχες κλάσεις στο δεύτερο παραδοτέο, δεν χρειάζεται να οριστεί ένας τέτοιος one-off πόρος, όμως τον παραθέτουμε ενδεικτικά για το σύστημα μας, λόγω της σημαντικότητας του, ακόμη και αν δεν σχετίζεται με κάποιο User Story.

### 2.5.1 Τερματικό DELETE πόρου DeleteAccount για συγκεκριμένο patientId

Αυτή η τερματική υπηρεσία (DELETE) του πόρου DeleteAccount χρησιμοποιείται για την διαγραφή ενός συγκεκριμένου χρήστη “ασθενή”, δεδομένου του ID του (που δηλώνεται ως query parameter).

#### 2.5.1.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου είναι μόνο το ID του χρήστη “ασθενή”, για τον οποίο εκτελείται η διαδικασία της διαγραφής του από το σύστημα.

Parameters	
Name	Description
<b>patientId</b> * required	The id of the Patient.
string (query)	<input type="text" value="patientId"/>

Εικόνα 21: Παράθεση παραμέτρων του τερματικού DELETE /deleteAccount του πόρου DeleteAccount.

#### 2.5.1.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία DELETE /deleteAccount αναπαρίστανται στην



παρακάτω φωτογραφία και η αναπαράσταση τους αφορά έναν κωδικό και ένα μήνυμα, όπως ορίζεται από την επιτυχή διαγραφή του “ασθενή”.

Responses		
Code	Description	Links
200	Successful account deletion	No links
Media type		
application/json		
Controls Accept header.		
Example Value   Schema		
{ "message": "Successfully Deleted!" }		

Εικόνα 22: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού DELETE /deleteAccount του πόρου DeleteAccount.

## 2.6 Πόρος RecommendDoctors (Αλγοριθμικός)

GET	/healthRequest/doctorChoice/requestSpecificDoctor	Retrieve the doctor resulting from the algorithm that executes the doctor selection and sets the name of the doctor chosen by the Patient.	▼
-----	---	--	---

Εικόνα 23: Περιγραφή του API του πόρου RecommendDoctors (από swagger-editor).

Endpoint	User Story	FR
/healthRequest/doctorChoice/recommendDoctors	1, 2, 3, 8	7

Ο πόρος RecommendDoctors είναι ένας αλγοριθμικός πόρος, ο οποίος χρησιμοποιείται για να αναπαραστήσει το αποτέλεσμα ενός αλγορίθμου. Ο αλγόριθμος αυτός, λαμβάνει υπόψη την επιλογή του χρήστη “ασθενή” για “ειδικότητα γιατρού” (παράμετρος εισόδου, δηλαδή, για τον αλγόριθμο) και επιστρέφει μια λίστα σχετικών “γιατρών”, που προτείνονται στο χρήστη “ασθενή” ως επιλογές. Όταν ο χρήστης “ασθενής” βρίσκεται στη διαδικασία επιλογής του “γιατρού” που επιθυμεί, είτε να εξετάσει το “ιατρικό αίτημα” του, είτε να του αιτηθεί “ιατρικό ραντεβού”, μπορεί να επιλέξει τον “γιατρό” αυτόν μέσα από μια λίστα “γιατρών”, οι οποίοι έχουν ως κοινό χαρακτηριστικό την ειδικότητα (doctorSpecialty). Αυτή η λίστα είναι, προφανώς, αποτέλεσμα ενός αλγορίθμου συλλογής δεδομένων και συγκεκριμένα της παραμέτρου doctorSpecialty που αντιστοιχεί στην (επιθυμητή) ειδικότητα “γιατρών”.

Για τον πόρο αυτό δεν υπάρχει μοντέλο δεδομένων, καθώς είναι αλγοριθμικός. Ορίζουμε το τερματικό GET /healthRequest/doctorChoice/recommendDoctors , το οποίο υλοποιεί την αναπαράσταση της λίστας σχετικών “γιατρών” προς επιλογή στον χρήστη “ασθενή”.

### 2.6.1 Τερματικό GET πόρου RecommendDoctors για συγκεκριμένη ειδικότητα



### γιατρού doctorSpecialty

Αυτή η τερματική υπηρεσία (GET) του πόρου RecommendDoctors χρησιμοποιείται για την ανάκτηση μιας λίστας σχετικών “γιατρών”, σύμφωνα με την ειδικότητα γιατρού, την οποία επέλεξε ο χρήστης “ασθενής”. Χρησιμοποιείται για την ανάκτηση της πληροφορίας που προκύπτει ως αποτέλεσμα του αλγορίθμου, ο οποίος λαμβάνει ως παράμετρο εισόδου αυτή την ειδικότητα “γιατρού” που επιλέχθηκε από τον “ασθενή” και δημιουργεί την προτεινόμενη λίστα σχετικών “γιατρών” βάσει αυτής, την οποία επιστρέφει.

#### 2.6.1.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν την ειδικότητα “γιατρού” (doctorSpecialty) την οποία επιλέγει ο χρήστης “ασθενής” ως επιθυμητή και η οποία δίνεται ως query parameter στην υλοποίηση του τερματικού.

Parameters	
Name	Description
<b>doctorSpecialty</b> * required	The doctor specialty chosen by the Patient
string	
(query)	<input type="text" value="doctorSpecialty"/>

Εικόνα 24: Παράθεση παραμέτρων του τερματικού GET /healthRequest/doctorChoice/recommendDoctors του πόρου RecommendDoctors.

#### 2.6.1.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία GET /healthRequest/doctorChoice/recommendDoctors αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση αφορά όλη την πληροφορία που σχετίζεται με το αποτέλεσμα του αλγορίθμου δημιουργίας (“πρότασης”) μια λίστας σχετικών “γιατρών” με την επιλογή της ειδικότητας “γιατρού”, από τον χρήστη “ασθενή”.

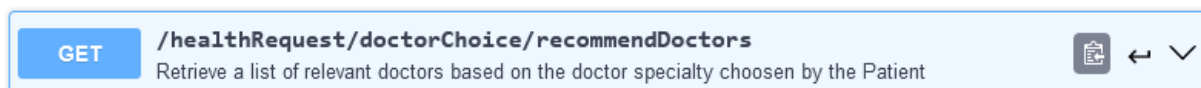
Responses		
Code	Description	Links
200	A list of relevant doctors appears	No links
Media type		
<input type="text" value="application/json"/>		
Controls Accept header.		
Example Value   Schema		
<pre>[   {     "name": "string",     "specialty": "string",     "doctorID": 0,     "certificate": {       "fileName": "string",       "fileType": "string",       "data": "string"     }   } ]</pre>		

Εικόνα 25: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού



GET /healthRequest/doctorChoice/recommendDoctors του πόρου RecommendDoctors.

## 2.7 Πόρος RequestSpecificDoctor (Αλγοριθμικός)



Εικόνα 26: Περιγραφή του API του πόρου RequestSpecificDoctor (από swagger-editor).

Endpoint	User Story	FR
/healthRequest/doctorChoice/requestSpecificDoctor	1, 2, 3	8

Ο πόρος RequestSpecificDoctor είναι ένας αλγοριθμικός πόρος, ο οποίος χρησιμοποιείται για να αναπαράσχει το αποτέλεσμα ενός αλγορίθμου. Ο αλγόριθμος αυτός, λαμβάνει υπόψη την επιλογή του χρήστη “ασθενή” για το “όνομα γιατρού” (παράμετρος εισόδου, δηλαδή, για τον αλγόριθμο) και επιστρέφει τον συγκεκριμένο “γιατρό” και τις πληροφορίες που αντιστοιχούν στο λογαριασμό του (όνομα, ειδικότητα, id, πιστοποιητικό/πτυχίο). Ο “γιατρός” αυτός είναι ο “γιατρός” που επιλέγεται από τον χρήστη “ασθενή”, είτε να εξετάσει το “ιατρικό αίτημα” του, είτε να του αιτηθεί “ιατρικό ραντεβού”. Αυτή η αντιστοίχιση του ονόματος ενός “γιατρού” στην επιστρεφόμενη πληροφορία για τον “γιατρό” επιλογής είναι, προφανώς, αποτέλεσμα ενός αλγορίθμου συλλογής δεδομένων και συγκεκριμένα της παραμέτρου doctorName που αντιστοιχεί στο όνομα του “γιατρού” επιλογής.

Για τον πόρο αυτό δεν υπάρχει μοντέλο δεδομένων, καθώς είναι αλγοριθμικός. Ορίζουμε το τερματικό GET /healthRequest/doctorChoice/requestSpecificDoctor, το οποίο υλοποιεί την αναπαράσταση της σχετιζόμενης με τον “γιατρό” επιλογής (ο οποίος προκύπτει ως αντιστοίχιση του ονόματος που πληκτρολογεί ο “ασθενής” στο id του “γιατρού”) πληροφορίας.

### 2.7.1 Τερματικό GET πόρου RequestSpecificDoctor για συγκεκριμένο όνομα γιατρού doctorName

Αυτή η τερματική υπηρεσία (GET) του πόρου RequestSpecificDoctor χρησιμοποιείται για την ανάκτηση του “γιατρού”, σύμφωνα με το όνομα γιατρού, που επέλεξε ο χρήστης “ασθενής”. Χρησιμοποιείται για την ανάκτηση της πληροφορίας που προκύπτει ως αποτέλεσμα του αλγορίθμου, ο οποίος λαμβάνει ως παράμετρο εισόδου αυτή τη συμβολοσειρά του ονόματος του “γιατρού” που επιλέχθηκε από τον “ασθενή” και την πληροφορία εισόδου αυτή στο λογαριασμό ενός συγκεκριμένου “γιατρού”, την πληροφορία του οποίου επιστρέφει.

#### 2.7.1.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το όνομα “γιατρού” (doctorName), το οποίο δίνεται ως query parameter στην υλοποίηση του τερματικού.





Parameters	
Name	Description
<b>doctorName</b> * required	The name of the doctor chosen by the Patient
string	
(query)	<input type="text" value="doctorName"/>

Εικόνα 27: Παράθεση παραμέτρων του τερματικού GET /healthRequest/doctorChoice/requestSpecificDoctor του πόρου RequestSpecificDoctor.

### 2.7.1.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία GET /healthRequest/doctorChoice/requestSpecificDoctor αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση αφορά όλη την πληροφορία που σχετίζεται με το αποτέλεσμα του αλγορίθμου “εύρεσης” του “γιατρού” επιλογής από τον χρήστη “ασθενή”.

Responses		
Code	Description	Links
200	Default response	No links
Media type		
<input type="text" value="application/json"/>		
Controls Accept header.		
Example Value   Schema		
<pre>{   "name": "string",   "specialty": "string",   "doctorID": 0,   "certificate": {     "fileName": "string",     "fileType": "string",     "data": "string"   } }</pre>		

Εικόνα 28: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού GET /healthRequest/doctorChoice/requestSpecificDoctor του πόρου RequestSpecificDoctor.



## 2.8 Πόρος Notification

POST	/patient/{patientId}/notification	Creates a notification template	▼
GET	/patient/{patientId}/notification/{notificationId}	Returns the Notification based on a single ID	▼
PUT	/patient/{patientId}/notification/{notificationId}	Returns the Notification based on a single ID	▼
DELETE	/patient/{patientId}/notification/{notificationId}	Deletes a single Notification based on the ID supplied	▼

Εικόνα 29: notification

Endpoint	User Story	FR
/notification	1,6	13

Είναι ο πόρος με τον οποίο ενημερώνουμε τον χρήστη ότι έλαβε διάγνωση για κάποιο από τα HealthRequests του.

### 2.8.1 Μοντέλο δεδομένων

Notification	▼	{
NotificationText*	string	
NotificationId*	integer(\$int64)	
	}	

Εικόνα 30: notification data type

Μια ειδοποίηση στην παρούσα φάση της εφαρμογής μας έχει id για να ξεχωρίζουμε διαφορετικά notifications ανα μεταξύ τους και κείμενο το οποίο προβάλλεται στον χρήστη όταν λαμβάνει την ειδοποίηση.

### 2.8.2 Τερματικό POST /patient/{patientId}/notification

Εδώ δημιουργούμε ένα καινούριο notification.

POST	/patient/{patientId}/notification	Creates a notification template	▼
------	-----------------------------------	---------------------------------	---

Εικόνα 31: post notification

#### 2.8.2.1 Παράμετροι εισόδου

Ως είσοδο δεν λαμβάνει κάποια παράμετρο.



Parameters Try it out

Name	Description
<b>patientId</b> * required	The patient's ID
integer(\$int64) (path)	<input type="text" value="patientId"/>

Request body required application/json

Notification to be displayed to the patient

Example Value | Schema

```
{  
  "NotificationText": "string",  
  "NotificationId": 0  
}
```

Εικόνα 32: post notification parameters

### 2.8.2.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε ένα αντικείμενο notification.

Responses

Code	Description	Links
201	notification response	No links
	Media type <span>application/json</span>	
	Controls Accept Header	
	Example Value   Schema	
	<pre>{   "NotificationText": "string",   "NotificationId": 0 }</pre>	
default	unexpected error	No links
	Media type <span>application/json</span>	
	Example Value   Schema	
	<pre>{   "code": 0,   "message": "string" }</pre>	

Εικόνα 33: post notification response

### 2.8.3 Τερματικό GET /patient/{patientId}/notification/{notificationId}

Εδώ ζητάμε το notification.

GET /patient/{patientId}/notification/{notificationId} ✓

Returns the Notification based on a single ID

Εικόνα 34: get notificaiton



### 2.8.3.1 Παράμετροι εισόδου

Ως είσοδο έχουμε το id του notification.

Parameters	
Name	Description
<b>notificationId</b> * required	ID of Notification to fetch
integer(\$int64)	
(path)	notificationId
<b>patientId</b> * required	The patient's ID
integer(\$int64)	
(path)	patientId

Εικόνα 35: get notificaiton parameters

### 2.8.3.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε ένα αντικείμενο notification.

Responses	
Code	Description
200	notification response
	Media type
	application/json
	Controls Accepts Header:
	Example Value   Schema
	<pre>{   "NotificationText": "string",   "NotificationId": 0 }</pre>
default	unexpected error
	Media type
	application/json
	Example Value   Schema
	<pre>{   "code": 0,   "message": "string" }</pre>

Εικόνα 36: get notificaiton response

## 2.8.4 Τερματικό PUT /patient/{patientId}/notification/{notificationId}

Εδώ ενημερώνουμε το notification δίνοντάς του το κείμενο που θα προβάλει στον χρήστη.

PUT	/patient/{patientId}/notification/{notificationId}	Returns the Notification based on a single ID	✓
-----	--	---	---

Εικόνα 37: put notification

### 2.8.4.1 Παράμετροι εισόδου

Ως είσοδο έχουμε το id του notification.



Parameters

Try it out

Name	Description
<b>notificationId</b> * required integer(\$int64) (path)	ID of Notification to update
<b>patientId</b> * required integer(\$int64) (path)	The patient's ID

Request body required

application/json

Updated Notification text

Example Value | Schema

```
{
  "text": "string"
}
```

Εικόνα 38: put notification parameters

#### 2.8.4.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε ένα αντικείμενο notification.

Responses

Code	Description
200	notification response
default	unexpected error

Media type

application/json

Controls Accept: header.

Example Value | Schema

```
{
  "NotificationText": "string",
  "NotificationId": 0
}
```

Media type

application/json

Example Value | Schema

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

Εικόνα 39: put notification response

#### 2.8.5 Τερματικό DELETE patient/{patientId}/notification/{notificationId}

Διαγράφουμε το notification.

**DELETE** /patient/{patientId}/notification/{notificationId}

Deletes a single Notification based on the ID supplied

Εικόνα 40: delete notification



### 2.8.5.1 Παράμετροι εισόδου

Ως είσοδο έχουμε το id του notification.

Parameters	
Name	Description
<b>notificationId</b> * required integer(\$int64) (path)	ID of Notification to delete <input type="text" value="notificationId"/>
<b>patientId</b> * required integer(\$int64) (path)	The patient's ID <input type="text" value="patientId"/>

Εικόνα 41: delete notification parameters

### 2.8.5.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε το μήνυμα ότι το notification διαγράφηκε.

Responses	
Code	Description
204	Notification deleted
default	unexpected error
Media type <input type="text" value="application/json"/>	
Controls Accept header.	
Example Value   Schema	
<pre>{   "code": 0,   "message": "string" }</pre>	

Εικόνα 42: delete notification response

## 2.9 Πόρος HealthRequestList

GET	/patient/{patientId}/healthRequestList	Retrieve health requests, their diagnoses and their sequence number.	▼
GET	/patient/{patientId}/healthRequestList/healthRequest/{healthRequestId}	Retrieve health request and its diagnosis	▼

Εικόνα 43: health request list

Endpoint	User Story	FR
----------	------------	----



/healthRequestList	1	11
--------------------	---	----

Η λίστα HealthRequestList χρησιμεύει στο να συγκεντρώνει όλα τα HealthRequests και Diagnoses συνδεδεμένα με έναν ασθενή. Έτσι είναι εύκολα προσβάσιμα από τον χρήστη.

### 2.9.1 Μοντέλο δεδομένων



Εικόνα 44: health request list data type

Το μοντέλο περιλαμβάνει τριών ειδών δεδομένα. HealthRequest, Diagnosis, και έναν αριθμό ο οποίος αριθμεί τα HealthRequests βάσει του Id τους. Έτσι καθίσταται εφικτή η χρονολογική ταξινόμηση τους.

### 2.9.2 Τερματικό GET /patient/{patientId}/healthRequestList

Φαίρνει από το HealthRequestList, που κάναμε fetch από τον σέρβερ, όλα τα χαρακτηριστικά της λίστας προς προβολή στον χρήστη.

GET	/patient/{patientId}/healthRequestList	Retrieve health requests, their diagnoses and their sequence number.
-----	--	--

Εικόνα 45: get health request list



### 2.9.2.1 Παράμετροι εισόδου

Ως είσοδο λαμβάνει το Id του ασθενή.

Parameters	
Name	Description
<b>patientId</b> * required	The patient's ID
integer(\$int64)	
(path)	

Εικόνα 46: health request list parameters

### 2.9.2.2 Μοντέλο δεδομένων εξόδου

Η έξοδος του είναι ένα array με τα εξής στοιχεία: HealthRequest, Diagnosis, SeqId.

Responses	
Code	Description
200	A list of health requests, their diagnoses and their sequence number.  Media type application/json  Controls Accept header.  Example Value   Schema <pre>[   {     "SeqId": 0,     "HealthRequest": [       {         "HealthRequestID": 0,         "descriptionText": {           "text": "string"         },         "image": {           "fileName": "string",           "fileType": "string",           "data": "string"         },         "objMedicalHistoryDocument": {           "fileName": "string",           "fileType": "string",           "data": "string"         },         "patientID": 0,         "doctorID": 0,         "hasReceivedDiagnosis": false       }     ]   } ]</pre>
default	unexpected error  Media type application/json  Example Value   Schema <pre>{   "code": 0,   "message": "string" }</pre>

```
]
  "Diagnosis": [
    {
      "diagnosisText": {
        "text": "string"
      }
    }
  ]
}
```

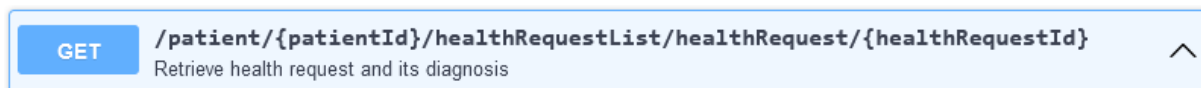
Εικόνα 47: health request list response





### 2.9.3 Τερματικό GET /patient/{patientId}/healthRequestList/healthRequest/{healthRequestId}

Δίνει ένα συγκεκριμένο HealthRequest από την λίστα μαζί με την διάγνωση του, αν αυτή υπάρχει.



Εικόνα 48: get health request

#### 2.9.3.1 Παράμετροι εισόδου

Ως είσοδο έχουμε το συγκεκριμένο το Id του HealthRequest που θέλουμε να δούμε.

Parameters	
Name	Description
<b>healthRequestId</b> * required integer(\$int64) (path)	The health request's ID <input type="text" value="healthRequestId"/>
<b>patientId</b> * required integer(\$int64) (path)	The patient's ID <input type="text" value="patientId"/>

Εικόνα 49: get health request parameters

#### 2.9.3.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε το ζευγάρι HealthRequest - Diagnosis που αντιστοιχεί στο HealthRequestId μας.

Responses

Code	Description
200	A health request with its diagnosis. <div>Media type <input type="text" value="application/json"/></div> <div>Controls Accept header</div> <div>Example Value   Schema</div> <pre>{   "HealthRequest": {     "HealthRequestID": 0,     "descriptionText": {       "text": "string"     },     "image": {       "fileName": "string",       "fileType": "string",       "data": "string"     },     "objMedicalHistoryDocument": {       "fileName": "string",       "fileType": "string",       "data": "string"     },     "patientID": 0,     "doctorID": 0,     "hasReceivedDiagnosis": false   },   "Diagnosis": {     "diagnosisText": {       "text": "string"     }   } }</pre>

default

unexpected error

Media type

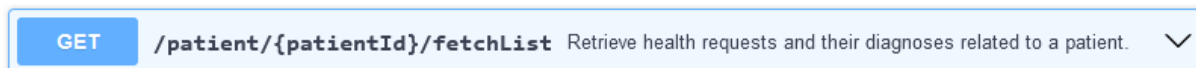
Example Value | Schema

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



Εικόνα 50: get health request response

## 2.10 Πόρος FetchList (Αλγοριθμικός)



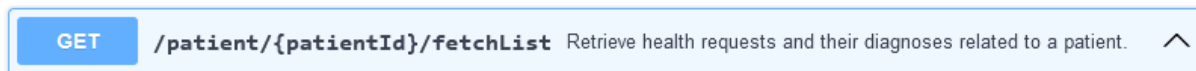
Εικόνα 51: fetch list

Endpoint	User Story	FR
/patient/{patientId}/fetchList	1	11

Ο σκοπός του πόρου αυτού είναι να εξαγάγει από την βάση δεδομένων τα συνδεδεμένα με τον χρήστη HealthRequests και Diagnoses τα οποία έπειτα με get /healthRequestList παίρνουν σωστή μορφή χάρη στην ταξινόμηση τους με το SeqId.

### 2.10.1 Τερματικό GET /patient/{patientId}/fetchList

Μας επιστρέφει την λίστα μας.



Εικόνα 52: get fetch list

#### 2.10.1.1 Παράμετροι εισόδου

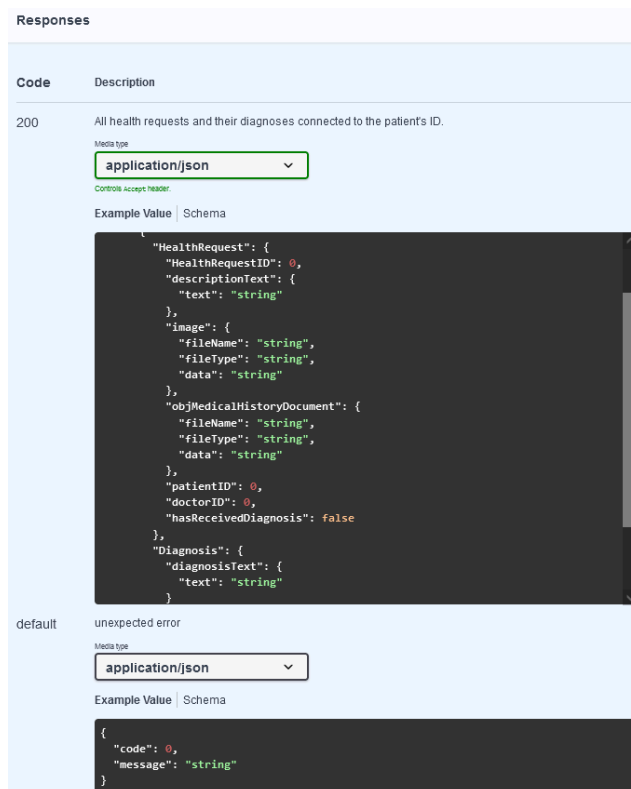
Ως είσοδο έχουμε το Id του ασθενή για τον οποίον θέλουμε να καλέσουμε τα συνδεδεμένα HealthRequests και Diagnoses.

Parameters	
Name	Description
<b>patientId</b> * required	The patient's ID
integer(\$int64)	
(path)	

Εικόνα 53: get fetch list parameters

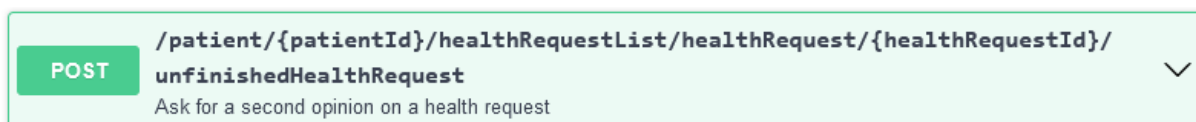
#### 2.10.1.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε τα HealthRequests και Diagnoses.



Εικόνα 54: get fetch list response

## 2.11 Πόρος UnfinishedHealthRequest



Εικόνα 55: unfinished health request

Endpoint	User Story	FR
/patient/{patientId}/healthRequestList/healthRequest/{healthRequestId}/unfinishedHealthRequest	2	12

Αυτός ο πόρος προωθεί ένα ημιτελές HealthRequest, όσον αφορά τα πεδία που έχουν να κάνουν με τον γιατρό και το HealthRequestId, στην σελίδα συμπλήρωσης του HealthRequest εντός της εφαρμογής. Έτσι έχουμε κοινές όλες τις παραμέτρους του original HealthRequest και μπορούμε να επιλέξουμε εκ νέου έναν γιατρό ο οποίος θα μας δώσει μια δεύτερη γνώμη.

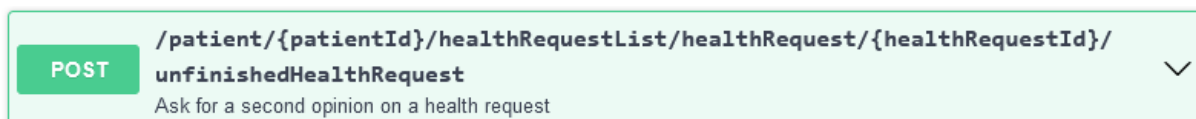


### 2.11.1 Μοντέλο Δεδομένων UnfinishedHealthRequest



### 2.11.2 Τερματικό POST /patient/{patientId}/healthRequestList/healthRequest/{healthRequestId}/unfinishedHealthRequest

Ζητάμε μια δεύτερη γνώμη για ένα συγκεκριμένο HealthRequest.



Εικόνα 56: post unfinished health request

#### 2.11.2.1 Παράμετροι εισόδου

Ως είσοδο έχουμε το Id του HealthRequest που θέλουμε να επαναλάβουμε.



Parameters

Try it out

Name	Description
<b>healthRequestId</b> * required	The health request ID we want a second opinion on
integer(\$int64) (path)	<input type="text" value="healthRequestId"/>
<b>patientId</b> * required	The patient's ID
integer(\$int64) (path)	<input type="text" value="patientId"/>

Request body \* required

application/json

The health request we want a second opinion on

Example Value | Schema

```
{
  "HealthRequestID": 0,
  "descriptionText": {
    "text": "string"
  },
  "image": {
    "fileName": "string",
    "fileType": "string",
    "data": "string"
  },
  "objMedicalHistoryDocument": {
    "fileName": "string",
    "fileType": "string",
    "data": "string"
  },
  "patientID": 0,
  "doctorID": 0,
  "hasReceivedDiagnosis": false
}
```

Εικόνα 57: post unfinished health request parameters

### 2.11.2.2 Μοντέλο δεδομένων εξόδου

Ως έξοδο έχουμε το HealthRequest αυτό χωρίς τις παραμέτρους που έχουν να κάνουν με τον γιατρό και το Id του HealthRequest.



Responses

Code	Description
201	UnfinishedHealthRequest response <div>Media type application/json <small>Controls Accept Header.</small> Example Value   Schema <pre>{  "descriptionText": {    "text": "string"  },  "image": {    "fileName": "string",    "fileType": "string",    "data": "string"  },  "objMedicalHistoryDocument": {    "fileName": "string",    "fileType": "string",    "data": "string"  },  "patientID": 0  }</pre></div>
default	unexpected error <div>Media type application/json  Example Value   Schema <pre>{  "code": 0,  "message": "string" }</pre></div>

Εικόνα 58: post unfinished health request response

## 2.12 Πόρος Diagnosis

POST

**/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis**  
Create a medical Diagnosis related to a specified Health Request. (The doctor acts)

✓

GET

**/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis**  
Retrieve a list of Diagnoses related to the specified (by its id) Health Request that is made by a Patient specified by its id.

✓

Εικόνα 59: Περιγραφή του API του πόρου Diagnosis (από swagger-editor).

Endpoint	User Story	FR
/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis	6	10



/patient/{patientId}/healthRequest/{healthRequestId} /diagnosis	4	11
--	---	----

Ο πόρος Diagnosis αντιστοιχεί σε μια “διάγνωση” για ένα “ιατρικό αίτημα” ενός “ασθενή”.

### 2.12.1 Μοντέλο Δεδομένων Diagnosis

Ο πόρος Diagnosis είναι, ουσιαστικά, η αντιστοίχιση της κλάσης (BEC) Diagnosis, που είχε οριστεί στη “μονολιθική” σχεδίαση του συστήματος. Όλα του τα χαρακτηριστικά/ιδιότητες αντιπροσωπεύουν τα αντίστοιχα attributes της κλάσης Diagnosis και φαίνονται στο παρακάτω μοντέλο δεδομένων.



Εικόνα 5: Μοντέλο δεδομένων πόρου Diagnosis.

### 2.12.2 Τερματικό POST πόρου Diagnosis για συγκεκριμένο DoctorID, PatientID και HealthRequestID

Αυτή η τερματική υπηρεσία (POST) του πόρου Diagnosis χρησιμοποιείται για την δημιουργία μιας “διάγνωσης” από έναν γιατρό, δεδομένου του ID του (που δηλώνεται ως path parameter), για ένα “ιατρικό αίτημα”, δεδομένου του ID του (που δηλώνεται ως path parameter) που αντιστοιχεί σε έναν συγκεκριμένο χρήστη “ασθενή”, δεδομένου του ID του (που δηλώνεται ως path parameter). Γίνεται εισαγωγή της “διάγνωσης” αυτής στο λογαριασμό του “ασθενή”.

#### 2.12.2.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του “γιατρού”, του “ασθενή” και του “ιατρικού αιτήματος”, τα οποία δηλώνονται ως path parameters, καθώς και ένα Request Body που περιλαμβάνει όλη την απαραίτητη πληροφορία για την δημιουργία μιας διάγνωσης.



Parameters Try it out

Name	Description
<b>doctorId</b> * required string (path)	The id of the Doctor who writes the Diagnosis
<b>patientId</b> * required string (path)	The id of the Patient who made the Health Request
<b>healthRequestId</b> * required string (path)	The id of the Health Request

Request body application/json

Diagnosis model

Example Value | Schema

```
{
  "diagnosisText": {
    "text": "string"
  }
}
```

Εικόνα 60: Παράθεση παραμέτρων του τερματικού POST `/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis` του πόρου Diagnosis.

### 2.12.2.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία POST `/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis` από την επιτυχή δημιουργία της “διάγνωσης” αναπαρίστανται στην παρακάτω φωτογραφία.

Responses

Code	Description	Links
201	Successful operation. Doctor sends a Diagnosis related to Health Request	No links

Media type application/json

Controls Accept header.

Example Value | Schema

```
{
  "diagnosisText": {
    "text": "string"
  }
}
```

Εικόνα 61: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού POST `/doctor/{doctorId}/patient/{patientId}/healthRequest/{healthRequestId}/diagnosis` του πόρου Diagnosis.





### 2.12.3 Τερματικό GET πόρου Diagnosis PatientID και HealthRequestID

Αυτή η τερματική υπηρεσία (PUT) του πόρου Diagnosis χρησιμοποιείται για ανάκτηση όλης της πληροφορίας που σχετίζεται με τις “διαγνώσεις” που αφορούν ένα συγκεκριμένο “ιατρικό αίτημα”, δεδομένου του ID του (που δηλώνεται ως path parameter), en;ow συγκεκριμένου “ασθενή”, δεδομένου του ID του (που δηλώνεται ως path parameter).

#### 2.12.3.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου είναι το ID του χρήστη “ασθενή” και το ID του “ιατρικού αιτήματος” στο οποίο αφορά η πληροφορία ζητάται να ανακτηθεί.

Name	Description
<b>patientId</b> * required string (path)	The id of the Patient <input type="text" value="patientId"/>
<b>healthRequestId</b> * required string (path)	The id of the Health Request <input type="text" value="healthRequestId"/>

Εικόνα 62: Παράθεση παραμέτρων του τερματικού GET /patient/{patientId}/healthRequest/{healthRequestId}/diagnosis του πόρου Diagnosis.

#### 2.12.3.2 Μοντέλο δεδομένων εξόδου

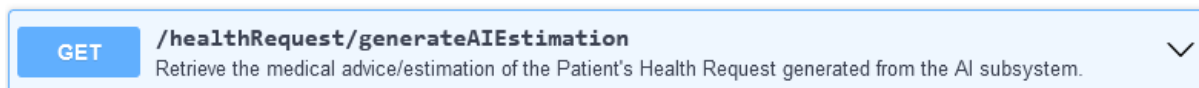
Τα responses της κλήσης στην τερματική υπηρεσία GET /patient/{patientId}/healthRequest/{healthRequestId}/diagnosis αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση αφορά όλη την πληροφορία, δηλαδή τη λίστα “διαγνώσεων” που σχετίζεται με το συγκεκριμένο “ιατρικό αίτημα” του συγκεκριμένου “ασθενή”.

Code	Description	Links
200	Successful operation. A list of Diagnoses that refer to the specified Health Request is retrieved.  Media type <input type="text" value="application/json"/> Controls Accept header.  Example Value   Schema <pre>[   {     "diagnosisText": {       "text": "string"     }   } ]</pre>	No links

Εικόνα 63: Παράθεση παραμέτρων του τερματικού GET /patient/{patientId}/healthRequest/{healthRequestId}/diagnosis του πόρου Diagnosis.



## 2.13 Πόρος GenerateAiEstimation (Αλγοριθμικός)



Εικόνα 64: Περιγραφή του API του πόρου GenerateAiEstimation (από swagger-editor).

Endpoint	User Story	FR
/healthRequest/generateAiEstimation	1	3

Ο πόρος GenerateAiEstimation είναι ένας αλγοριθμικός πόρος, ο οποίος χρησιμοποιείται για να αναπαραστήσει το αποτέλεσμα ενός αλγορίθμου. Ο αλγόριθμος αυτός, εκτελείται από το υποσύστημα AI παράγει την εκτίμηση του AI για ένα συγκεκριμένο “ιατρικό αίτημα” ενός συγκεκριμένου “ασθενή”, ο οποίος αιτήθηκε συμβουλή από το υποσύστημα AI. Το AI παράγει αυτοματοποιημένα την εκτίμηση, μέσω του αλγορίθμου αυτού που δέχεται ως είσοδο την πληροφορία που φέρει ένα “ιατρικό αίτημα”. Η έξοδος του υποσυστήματος AI είναι η έξοδος αυτού του αλγορίθμου, δηλαδή η εκτίμηση (αντίστοιχη της “διάγνωσης” του “γιατρού”) του AI για το συγκεκριμένο “ιατρικό αίτημα”.

Για τον πόρο αυτό δεν υπάρχει μοντέλο δεδομένων, καθώς είναι αλγοριθμικός. Ορίζουμε το τερματικό GET /healthRequest/generateAiEstimation, το οποίο υλοποιεί την αναπαράσταση της εκτίμησης του AI για το “ιατρικό αίτημα”.

### 2.13.1 Τερματικό GET πόρου GenerateAiEstimation για συγκεκριμένο HealthRequestID και συγκεκριμένο PatientID

Αυτή η τερματική υπηρεσία (GET) του πόρου GenerateAiEstimation χρησιμοποιείται για την ανάκτηση της εκτίμησης του υποσυστήματος AI για ένα συγκεκριμένο “ιατρικό αίτημα” ενός “ασθενή”.

#### 2.13.1.1 Παράμετροι εισόδου

Οι παράμετροι εισόδου αφορούν το ID του “ιατρικού αιτήματος” και το ID του “ασθενή” που δημιούργησε το “ιατρικό αίτημα”, τα οποία δηλώνονται ως query parameters στην υλοποίηση του τερματικού.



Parameters	
Name	Description
<b>healthRequestId</b> * required	The id of the Health Request made
string (path)	<input type="text" value="healthRequestId"/>
<b>patientId</b> * required	The id of the Patient who made the Health Request
string (path)	<input type="text" value="patientId"/>

Εικόνα 65: Παράθεση παραμέτρων του τερματικού GET /healthRequest/generateAIEstimation του πόρου GenerateAIEstimation.

### 2.13.1.2 Μοντέλο δεδομένων εξόδου

Τα responses της κλήσης στην τερματική υπηρεσία GET /healthRequest/generateAIEstimation αναπαρίστανται στην παρακάτω φωτογραφία και η αναπαράσταση αφορά όλη την πληροφορία που σχετίζεται με το αποτέλεσμα του αλγορίθμου δημιουργίας της εκτίμησης (“διάγνωσης”) από το υποσύστημα ΑΙ για το συγκεκριμένο “ιατρικό αίτημα” του “ασθενή”.

Responses	
Code	Description
200	The Diagnosis made by AI subsystem for the specific Health Request is retrieved
<div>Media type <input type="text" value="application/json"/></div> <div>Controls Accept: header.</div> <div>Example Value   Schema</div> <div><pre>{   "diagnosisText": {     "text": "string"   } }</pre></div>	

Εικόνα 66: Αποκρίσεις συστήματος/Μοντέλο δεδομένων εξόδου του τερματικού GET /healthRequest/generateAIEstimation του πόρου GenerateAIEstimation.