



# ΠΛΑΝΟ ΕΡΓΟΥ ΛΟΓΙΣΜΙΚΟΥ– FETAL DIAGNOSIS ENHANCEMENT TOOL

ΜΑΘΗΜΑ: ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ

Ονοματεπώνυμο:	Κωνσταντίνος Παπαθανασίου	ΑΜ: 2008
	Ηλίας Σταθάκος	ΑΜ: 2017
	Γεώργιος Κτιστάκης	ΑΜ: 1981

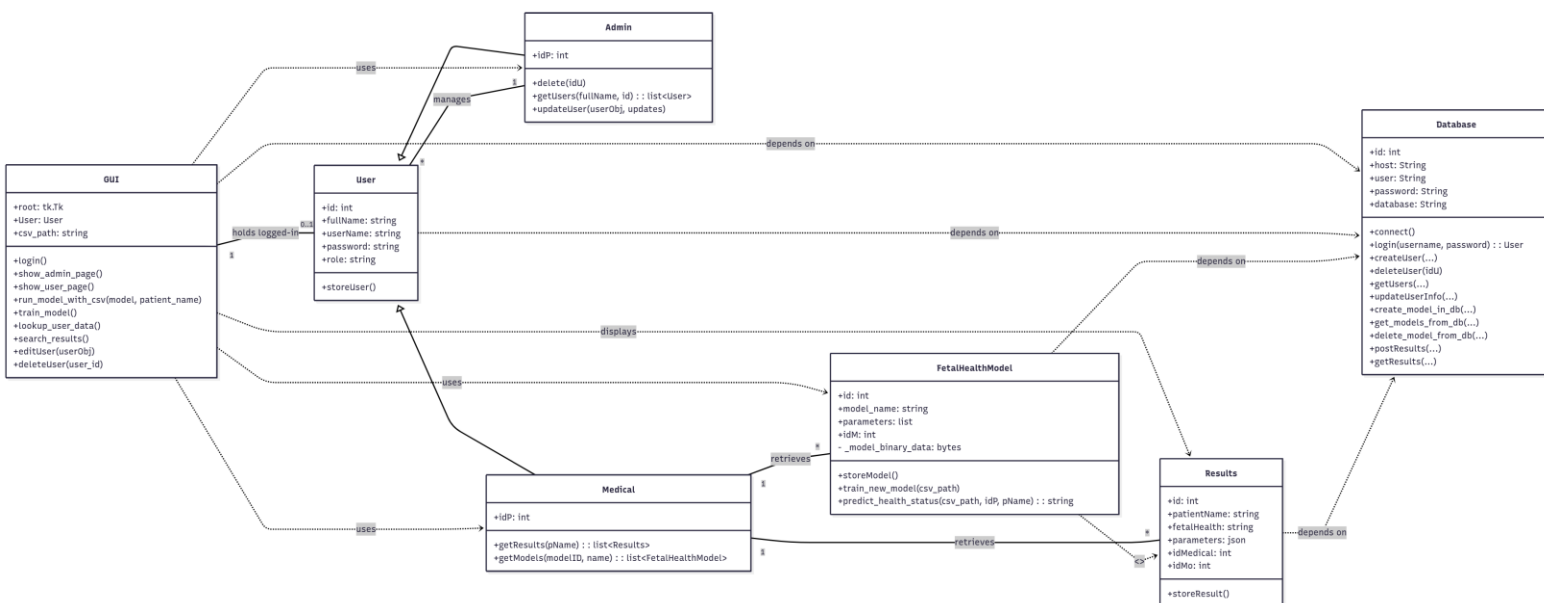


## Περιεχόμενα

1	ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ.....	3
1.1	Επεξήγηση Διαγράμματος.....	3
2	ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ.....	6
3	ΣΗΜΕΙΑ ΕΠΑΝΑΧΡΗΣΙΜΟΠΟΙΗΣΗΣ ΤΟΥ ΚΩΔΙΚΑ.....	9



### 1. ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ



#### 1.1 Επεξήγηση Διαγράμματος Κλάσεων

Το παρόν διάγραμμα κλάσεων UML απεικονίζει την αρχιτεκτονική δομή της εφαρμογής, αναδεικνύοντας τις κεντρικές οντότητες, τις μεταξύ τους σχέσεις και τις εξαρτήσεις τους. Η αρχιτεκτονική ακολουθεί ένα πολυεπίπεδο πρότυπο, διαχωρίζοντας το περιβάλλον χρήστη (GUI), την επιχειρησιακή λογική (κλάσεις User, Admin, Medical, FetalHealthModel) και την πρόσβαση στα δεδομένα (DatabaseFunctions).

#### Κλάση GUI (Graphical User Interface)

Η κλάση GUI αποτελεί το επίπεδο παρουσίασης (Presentation Layer) της εφαρμογής. Είναι υλοποιημένη με τη βιβλιοθήκη tkinter και είναι υπεύθυνη για τη δημιουργία όλων των γραφικών παραθύρων, των πεδίων εισαγωγής και των κουμπιών με τα οποία αλληλεπιδρά ο τελικός χρήστης. Λειτουργεί ως ο κεντρικός "ενορχηστρωτής", λαμβάνοντας τις ενέργειες του χρήστη (π.χ., κλικ σε κουμπί) και καλώντας τις κατάλληλες μεθόδους από τις κλάσεις της επιχειρησιακής λογικής. Μετά την επιτυχή είσοδο ενός χρήστη, η GUI διατηρεί μια ενεργή συσχέτιση (association) με το αντικείμενό του (τύπου Admin ή Medical). Αυτή η σχέση (πολλαπλότητας 1 προς 0..1) είναι θεμελιώδης, καθώς επιτρέπει στη GUI να προσαρμόζει δυναμικά το περιεχόμενο που εμφανίζει ανάλογα με τον ρόλο και τα δικαιώματα του συνδεδεμένου χρήστη.



## Κλάση User (Χρήστης)

Η κλάση User αποτελεί την βάση για κάθε τύπο χρήστη του συστήματος. Δημιουργείται από τους διαχειριστές και καθορίζει τον ρόλο των χρηστών, τον οποίο αποκτούν κατά τη σύνδεσή τους στο σύστημα. Οι κλάσεις Admin και Medical κληρονομούν από την User, υιοθετώντας τη βασική της λειτουργικότητα. Η User περιλαμβάνει τα βασικά χαρακτηριστικά ενός χρήστη, όπως id, fullName, userName, email και telephone. Επιπλέον, διαθέτει τη μέθοδο storeUser() για την αρχική αποθήκευση του χρήστη στη βάση δεδομένων.

## Κλάσεις Admin (Διαχειριστής) & Medical (Ιατρός)

Αυτές οι δύο κλάσεις αποτελούν τις συγκεκριμένες (concrete) υλοποιήσεις της αφηρημένης κλάσης User. Η σχέση τους με την User είναι κληρονομικότητα (inheritance), που σημαίνει ότι ένας Admin είναι ένας User και ένας Medical είναι ένας User, κληρονομώντας όλες τις ιδιότητες και μεθόδους της. Πάνω σε αυτή τη βάση, επεκτείνουν τη λειτουργικότητα με εξειδικευμένες δυνατότητες που αντιστοιχούν στον ρόλο τους.

**Admin:** Ο Admin έχει αυξημένα προνόμια διαχείρισης του συστήματος. Η κλάση αυτή υλοποιεί μεθόδους όπως getUsers(), updateUser() και delete(), επιτρέποντας στον διαχειριστή να επιβλέπει και να τροποποιεί τους λογαριασμούς των άλλων χρηστών. Έτσι, έχει μια σχέση συσχέτισης "ένα-προς-πολλά" (1:\*) με την κλάση User.

**Medical:** Ο Medical αντιπροσωπεύει το ιατρικό προσωπικό και είναι ο κύριος χρήστης των λειτουργιών μηχανικής μάθησης. Η κλάση αυτή παρέχει μεθόδους για την ανάκτηση και διαχείριση μοντέλων (getModels()) και την αναζήτηση προηγούμενων αποτελεσμάτων (getResults()). Έχει, συνεπώς, σχέσεις συσχέτισης "ένα-προς-πολλά" (1:\*) με τις κλάσεις FetalHealthModel και Results.

## Κλάση FetalHealthModel (Μοντέλο Μηχανικής Μάθησης)

Η FetalHealthModel είναι η κεντρική κλάση που ενσωματώνει τη λογική της μηχανικής μάθησης. Είναι υπεύθυνη για δύο κύριες λειτουργίες: την εκπαίδευση νέων μοντέλων από δεδομένα (train\_new\_model) και τη χρήση υπαρχόντων μοντέλων για την πρόβλεψη της εμβρυϊκής υγείας (predict\_health\_status). Ένα αντικείμενο αυτής της κλάσης περιέχει όχι μόνο τα μετα-δεδομένα του μοντέλου (όπως id, model\_name), αλλά και τα δυαδικά δεδομένα του εκπαιδευμένου αλγορίθμου και του scaler που χρησιμοποιήθηκε, εξασφαλίζοντας συνέπεια μεταξύ εκπαίδευσης και πρόβλεψης. Έχει μια κατευθυνόμενη σχέση εξάρτησης δημιουργίας (<<creates>>) με την κλάση Results, καθώς η εκτέλεση της μεθόδου πρόβλεψης οδηγεί στη δημιουργία ενός νέου αντικειμένου Results.



## Κλάση Results (Αποτελέσματα)

Η κλάση Results είναι μια απλή κλάση-δομή (data class) που μοντελοποιεί ένα και μόνο αποτέλεσμα διάγνωσης. Κάθε αντικείμενο Results αποθηκεύει την ταυτότητα του ασθενούς (patientName), το τελικό πόρισμα (fetalHealth), καθώς και κλειδιά αναφοράς (idMedical, idMo) που το συνδέουν με τον επιβλέποντα ιατρό και το μοντέλο που χρησιμοποιήθηκε για την πρόβλεψη. Η σχέση της είναι "πολλά-προς-ένα" (\*:1) τόσο με την κλάση Medical όσο και με την FetalHealthModel.

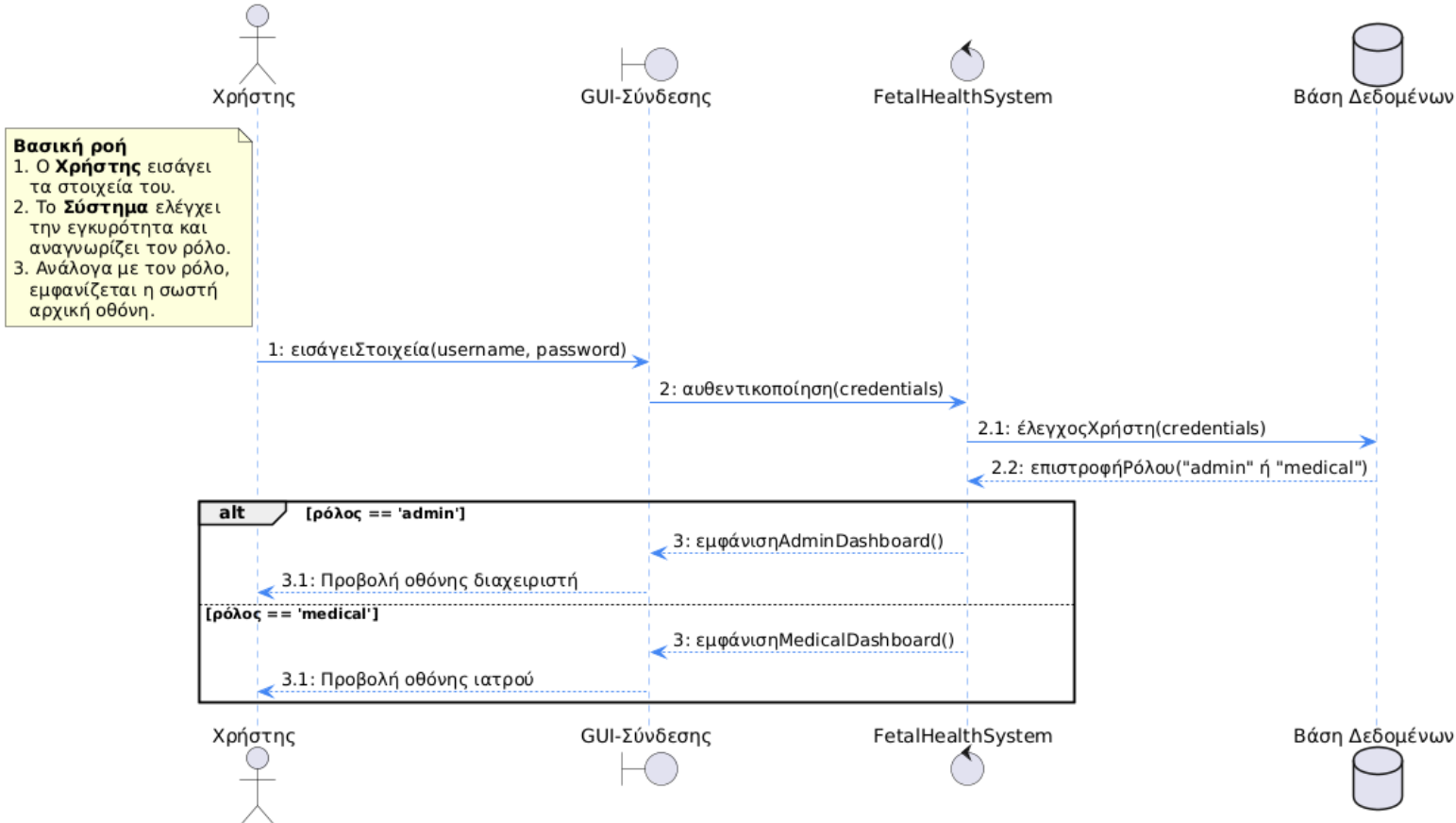
## Κλάση Database (Βάση Δεδομένων)

Η κλάση Database λειτουργεί ως το Επίπεδο Πρόσβασης Δεδομένων (Data Access Layer). Αυτή η ενότητα κώδικα απομονώνει την υπόλοιπη εφαρμογή από τις λεπτομέρειες της επικοινωνίας με τη βάση δεδομένων MySQL. Περιέχει ένα σύνολο από μεθόδους (όπως connect, login, createUser, postResults κ.λπ.) που εκτελούν συγκεκριμένα ερωτήματα SQL (queries). Σχεδόν όλες οι κλάσεις της επιχειρησιακής λογικής εξαρτώνται (depend on) από αυτήν την κλάση για την αποθήκευση και ανάκτηση των δεδομένων τους — μια σχέση που στο διάγραμμα απεικονίζεται με διακεκομμένες γραμμές (..>). Αυτός ο διαχωρισμός απλοποιεί τη συντήρηση και επιτρέπει την πιθανή μελλοντική αλλαγή του συστήματος βάσης δεδομένων με ελάχιστες τροποποιήσεις στον υπόλοιπο κώδικα.



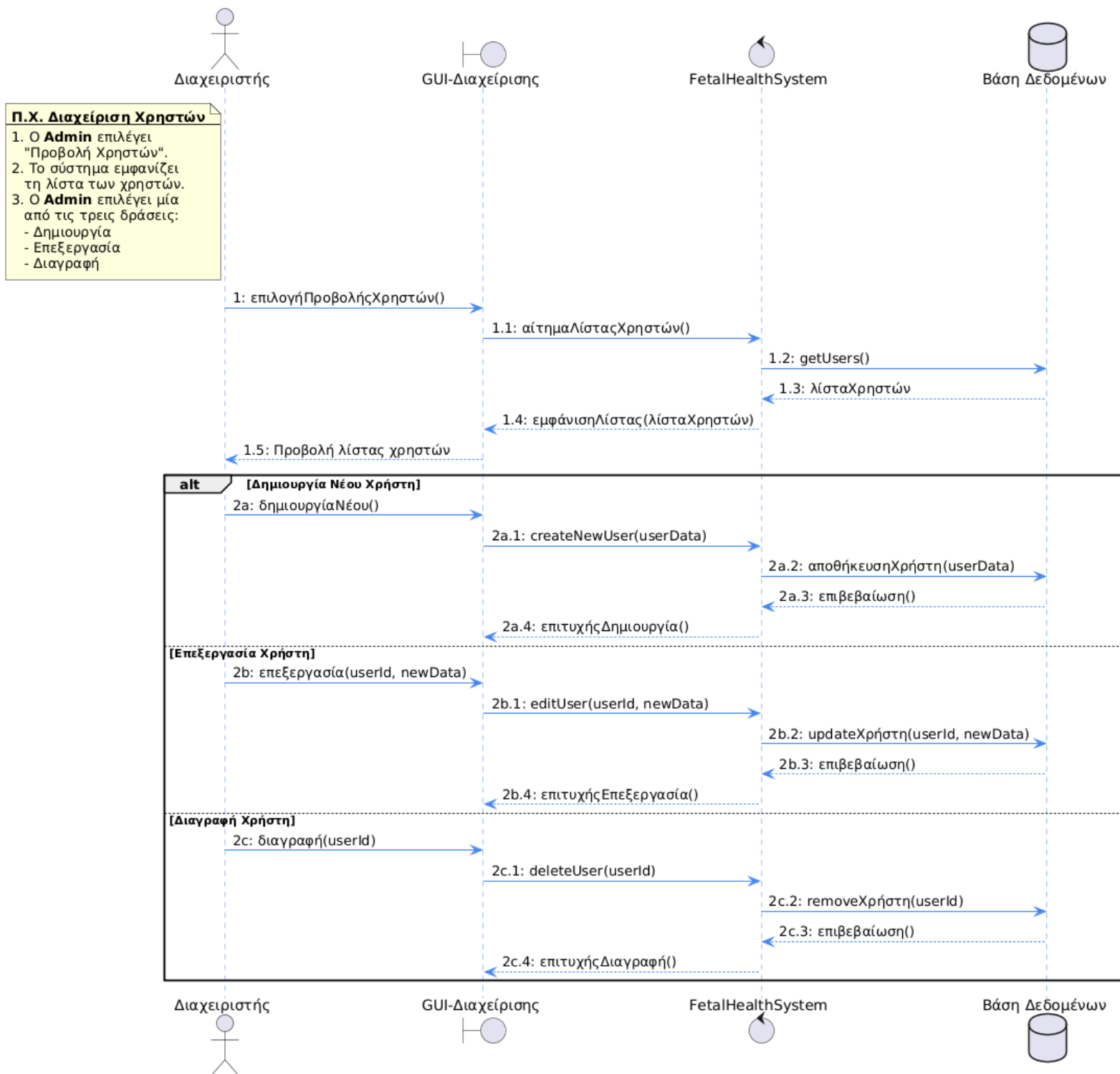
## 2. ΔΙΑΓΡΑΜΜΑΤΑ ΑΚΟΛΟΥΘΙΑΣ

Διάγραμμα 1: Σύνδεση & Επιλογή Ρόλου



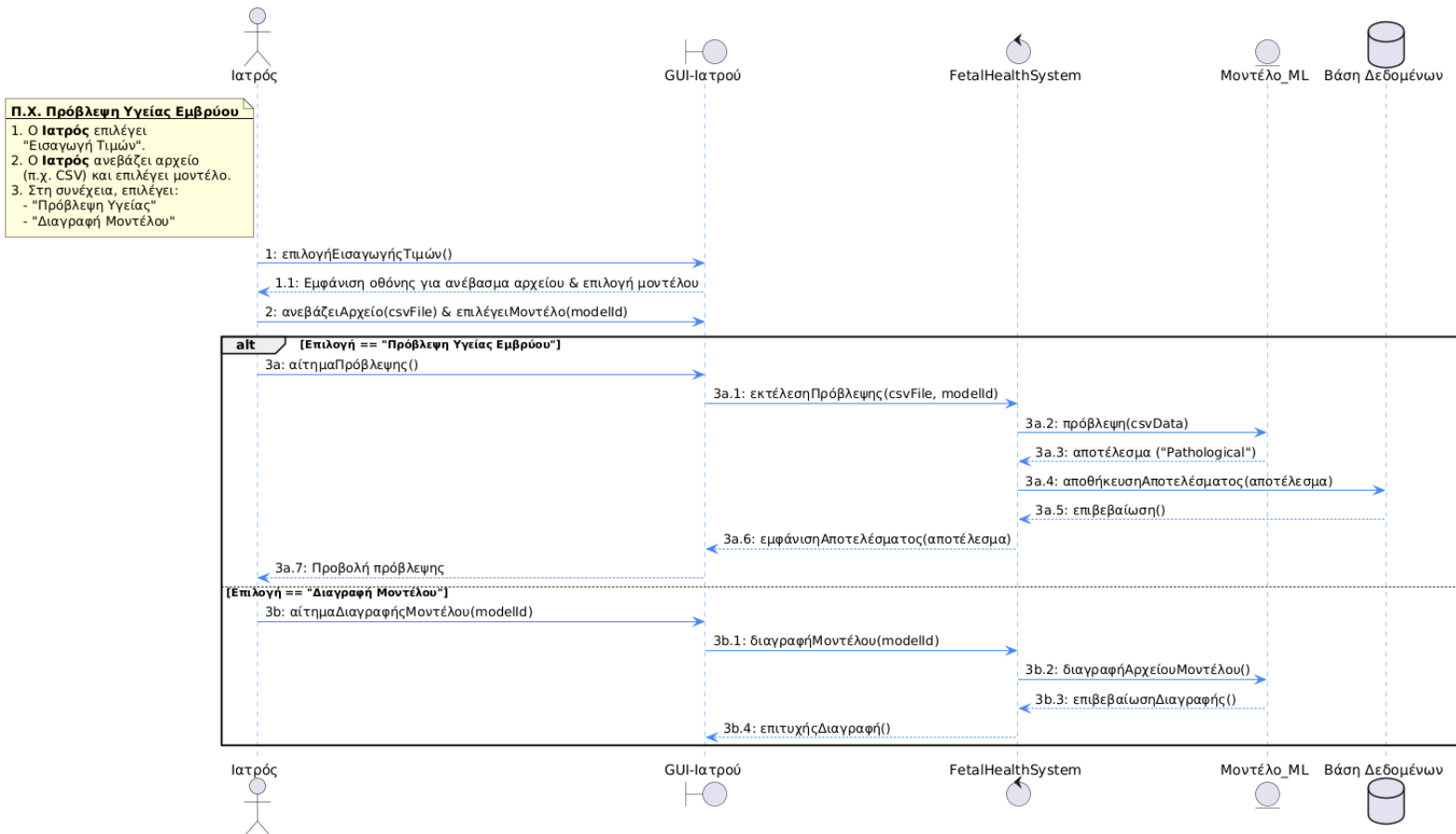


Διάγραμμα 2: Ροή Διαχειριστή - Διαχείριση Χρηστών





Διάγραμμα 3: Ροή Ιατρού - Πρόβλεψη & Διαχείριση Μοντέλου







### 3. ΣΗΜΕΙΑ ΕΠΑΝΑΧΡΗΣΙΜΟΠΟΙΗΣΗΣ ΤΟΥ ΚΩΔΙΚΑ

Αναλύοντας τη δομή και την υλοποίηση του κώδικα, διακρίνονται αρκετά τμήματα που έχουν σχεδιαστεί με γνώμονα την επαναχρησιμοποίηση και την επεκτασιμότητα, καθιστώντας τα ιδανικά για μελλοντική χρήση σε παρόμοια ή εξελιγμένα projects.

Ένα από τα πιο θεμελιώδη και επαναχρησιμοποιήσιμα κομμάτια είναι η κλάση Database. Αυτή η κλάση λειτουργεί ως ένα ολοκληρωμένο Data Access Layer (DAL), απομονώνοντας πλήρως την υπόλοιπη εφαρμογή από τις λεπτομέρειες της επικοινωνίας με τη βάση δεδομένων MySQL. Οι μέθοδοί της, όπως connect, login, createUser, getUsers, getModels και delete\_model\_from\_db, είναι γενικές και καλά ορισμένες. Αυτό σημαίνει ότι ολόκληρη η κλάση θα μπορούσε να μεταφερθεί αυτούσια σε ένα άλλο project που απαιτεί διαχείριση χρηστών και μοντέλων, με ελάχιστες ή καθόλου τροποποιήσεις, αρκεί η δομή της βάσης δεδομένων να παραμένει συνεπής. Η χρήση της κλάσης αυτής αντί για διάσπαρτες κλήσεις SQL στον κώδικα είναι μια κρίσιμη σχεδιαστική απόφαση που προάγει την καθαρότητα και τη συντηρησιμότητα.

Εξίσου σημαντική και άμεσα επαναχρησιμοποιήσιμη είναι η κλάση FetalHealthModel. Αυτή η κλάση ενσωματώνει ολόκληρο τον κύκλο ζωής ενός μοντέλου μηχανικής μάθησης: από την εκπαίδευση (train\_new\_model) και την αξιολόγηση, μέχρι τη σειριοποίηση των αντικειμένων (pickle.dumps) για αποθήκευση και την εκτέλεση προβλέψεων (predict\_health\_status). Επειδή η κλάση είναι αποσυνδεδεμένη από το γραφικό περιβάλλον και δέχεται ως ορίσματα τις διαδρομές των αρχείων δεδομένων, μπορεί να χρησιμοποιηθεί σε οποιοδήποτε περιβάλλον, είτε πρόκειται για ένα διαφορετικό GUI, μια web εφαρμογή, είτε ακόμα και για ένα απλό script που εκτελείται από τη γραμμή εντολών για μαζικές επεξεργασίες. Οι βοηθητικές μέθοδοι που δημιουργήθηκαν για την παραγωγή γραφημάτων, όπως plot\_prediction\_probabilities, είναι επίσης αυτόνομες και μπορούν εύκολα να προσαρμοστούν για να οπτικοποιούν αποτελέσματα από οποιοδήποτε μοντέλο ταξινόμησης.

Πέρα από τις κλάσεις, η ιεραρχική δομή των χρηστών με τις κλάσεις User, Admin και Medical αποτελεί ένα εξαιρετικό παράδειγμα αντικειμενοστραφούς σχεδιασμού. Η βασική κλάση User ορίζει τα κοινά χαρακτηριστικά και μεθόδους, ενώ οι υποκλάσεις Admin και Medical κληρονομούν αυτή τη λειτουργικότητα και την επεκτείνουν με τις δικές τους εξειδικευμένες δυνατότητες. Αυτό το μοντέλο κληρονομικότητας είναι εξαιρετικά επεκτάσιμο. Για παράδειγμα, αν στο μέλλον προστεθεί ένας νέος ρόλος, όπως "Ερευνητής", θα μπορούσε απλά να δημιουργηθεί μια νέα κλάση Researcher(User) που θα κληρονομεί τις βασικές λειτουργίες και θα προσθέτει νέες, χωρίς να επηρεάζεται ο υπάρχων κώδικας.

Τέλος, η ίδια η κλάση GUI στο αρχείο gui.py, παρόλο που είναι στενά συνδεδεμένη με τη συγκεκριμένη εφαρμογή, περιέχει επαναχρησιμοποιήσιμα μοτίβα σχεδιασμού. Οι μέθοδοι clear\_frame για τον καθαρισμό της οθόνης, καθώς και η δομή για τη δυναμική δημιουργία πινάκων με scrollbar (display\_user\_table, display\_model\_table), αποτελούν πρότυπα που μπορούν να αντιγραφούν και να προσαρμοστούν για την εμφάνιση οποιουδήποτε είδους δεδομένων σε πίνακα μέσα σε ένα περιβάλλον Tkinter. Η οργάνωση της διεπαφής σε λογικές σελίδες (show\_admin\_page, show\_user\_page) καθιστά τη διαχείριση της πλοήγησης απλή και εύκολα επεκτάσιμη.