



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
UNIVERSITY OF PIRAEUS

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ  
ΕΠΙΚΟΙΝΩΝΙΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**ΕΡΓΑΣΙΑ ΣΤΟ ΜΑΘΗΜΑ: ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ**

Επί των φοιτητών: Μόσχου Δημήτριου - Π18209

Σιάτρα Απόστολου – Π18215

Πετρίδη Αχιλλέα – Π18211

Φύτρου Ευάγγελου – Π18220

Διδάσκοντες: Βίρβου Μαρία  
Αλέπης Ευθύμιος

Πειραιάς, 2021



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
UNIVERSITY OF PIRAEUS

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ  
ΕΠΙΚΟΙΝΩΝΙΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΡΓΑΣΙΑ ΣΤΟ ΜΑΘΗΜΑ: ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ

*Τελική Εργασία Μαθήματος*

Επί των φοιτητών: Μόσχου Δημήτριου - Π18209

Σιάτρα Απόστολου – Π18215

Πετρίδη Αχιλλέα – Π18211

Φύτρου Ευάγγελου – Π18220

Διδάσκοντες: Βίρβου Μαρία  
Αλέπης Ευθύμιος

Πειραιάς, 2021

## Πίνακας περιεχομένων

### 1. ΕΙΣΑΓΩΓΗ

- 1.1 ΣΤΟΧΟΙ ΤΗΣ ΕΡΓΑΣΙΑΣ
- 1.2 Ορισμός του προβλήματος προς επίλυση
- 2. Σύντομη παρουσίαση της RUP

### 3. ΦΑΣΗ: ΈΝΑΡΞΗ (INCEPTION)

- 3.1 ΣΥΛΛΗΨΗ ΑΠΑΙΤΗΣΕΩΝ
- 3.2 Ανάλυση-Σχεδιασμός

- 3.2.1 Διαγράμματα Περιπτώσεων Χρήσης
- 3.2.2 Διαγράμματα Τάξεων

### 4. Φάση: Εκπόνηση Μελέτης (Elaboration)

#### 4.1 Ανάλυση-Σχεδιασμός

- 4.1.1 Διαγράμματα Περιπτώσεων Χρήσης (2<sup>η</sup> έκδοση)
- 4.1.2 Διαγράμματα Τάξεων (2<sup>η</sup> έκδοση)
- 4.1.3 Διαγράμματα Αντικειμένων (1<sup>η</sup> έκδοση)
- 4.1.4 Διαγράμματα Συνεργασίας (1<sup>η</sup> έκδοση)
- 4.1.5 Διαγράμματα Σειράς (1<sup>η</sup> έκδοση)
- 4.1.6 Διαγράμματα Δραστηριοτήτων (1<sup>η</sup> έκδοση)
- 4.1.7 Διαγράμματα Καταστάσεων (1<sup>η</sup> έκδοση)
- 4.1.8 Διαγράμματα Εξαρτημάτων (1<sup>η</sup> έκδοση)
- 4.1.9 Διαγράμματα Διανομής (1<sup>η</sup> έκδοση)

#### 4.2 Υλοποίηση-Έλεγχος

- 4.2.1 Υλοποίηση: 1<sup>η</sup> εκτελέσιμη έκδοση
- 4.2.2 Αναφορά ελέγχου για την 1<sup>η</sup> εκτελέσιμη έκδοση

### 5. Φάση: Κατασκευή (Construction)

#### 5.1 Ανάλυση-Σχεδιασμός

- 5.1.1 Διαγράμματα Περιπτώσεων Χρήσης (2<sup>η</sup> έκδοση)
- 5.1.2 Διαγράμματα Τάξεων (2<sup>η</sup> έκδοση)
- 5.1.3 Διαγράμματα Αντικειμένων (1<sup>η</sup> έκδοση)
- 5.1.4 Διαγράμματα Συνεργασίας (1<sup>η</sup> έκδοση)
- 5.1.5 Διαγράμματα Σειράς (1<sup>η</sup> έκδοση)
- 5.1.6 Διαγράμματα Δραστηριοτήτων (1<sup>η</sup> έκδοση)
- 5.1.7 Διαγράμματα Καταστάσεων (1<sup>η</sup> έκδοση)
- 5.1.8 Διαγράμματα Εξαρτημάτων (1<sup>η</sup> έκδοση)
- 5.1.9 Διαγράμματα Διανομής (1<sup>η</sup> έκδοση)

#### 5.2 Υλοποίηση-Έλεγχος

- 5.2.1 Υλοποίηση: 1<sup>η</sup> εκτελέσιμη έκδοση
- 5.2.2 Αναφορά ελέγχου για την 1<sup>η</sup> εκτελέσιμη έκδοση

### 6. Εγχειρίδιο Χρήστη

- 6.1 Σύντομη παρουσίαση του προγράμματος
- 6.2 Παρουσίαση σεναρίων λειτουργίας

## **1. ΕΙΣΑΓΩΓΗ**

### **1.1 Στόχοι της εργασίας**

Η παρούσα εργασία, πραγματεύεται την ανάλυση, σχεδίαση και κατασκευή μιας διαδικτυακής εφαρμογής (web application), η οποία αποσκοπεί στην υποβοήθηση της ασύγχρονης εκπαιδευτικής διαδικασίας, τόσο σε επίπεδο εκπαιδευτή όσο και σε επίπεδο εκπαιδευόμενου.

### **1.2 Ορισμός του προβλήματος προς αναλυση**

Το πρόβλημα που καλούμαστε να υλοποιήσουμε έιναι να μια web εφαρμογή η οποία θα λειτουργήσει α) ως αποθήκη ασκήσεων-δραστηριοτήτων, β) ως αυτοματοποιημένο σύστημα δημιουργίας τεστ αξιολόγησης και γ) ως σύστημα αξιολόγησης. Θα υπάρχουν δύο κατηγορίες χρηστών: ο εκπαιδευτής, ο οποίος θα εισάγει τις ασκήσεις δραστηριότητες στο σύστημα μαζί με ιδιότητες αυτών (π.χ. γνωστικό αντικείμενο, τύπος άσκησης, βαθμός δυσκολίας κ.λ.π.) και θα καθορίζει κάθε φορά τα κριτήρια π.χ. επίπεδο, ηλικία, τύπος ασκήσεων, πλήθος ασκήσεων, βαθμός δυσκολίας άσκησης, με βάση τα οποία το σύστημα θα παράγει το αυτοματοποιημένο τεστ, καθώς και σε ποιους εκπαιδευόμενους θα αφορά το συγκεκριμένο τεστ που θα παραχθεί, και ο εκπαιδευόμενος, ο οποίος θα εισέρχεται στο σύστημα και θα κάνει το τεστ που του δίνεται. Το σύστημα θα κρατάει για κάθε εκπαιδευόμενο την απόδοσή του στο τεστ και θα ενημερώνει σχετικά τον εκπαιδευτή. Και οι δυο τύποι χρηστών θα εισέρχονται στο σύστημα με λογαριασμό που θα πρέπει να δημιουργησουν.

## **2. ΣΥΝΤΟΜΗ ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ RUP**

Η διαδικασία Rational Unified Process είναι η διαδικασία που προτείνουν ο Booch, Rumbaugh και Jacobson για την ανάπτυξη λογισμικού.

Ο κύκλος ζωής λογισμικού προτείνεται να είναι επαναληπτικός. Η ανάπτυξη δηλαδή να προχωρεί σε μια σειρά επαναλήψεων μέχρινα εξελιχθεί το τελικό προϊόν.

Η διαδικασία Rational Unified Process αποτελείται από ένα σύνολο οδηγιών σχετικά με τις τεχνικές και οργανωτικές απόψεις της ανάπτυξης λογισμικού. Η διαδικασία αυτή αφορά κυρίως στην Ανάλυση Απαιτήσεων και στο Σχεδιασμό.

Ο κύκλος ζωής λογισμικού όπως προτείνεται από τη Rational Unified Process φαίνεται στο Σχήμα 2.1.

Η διαδικασία Rational Unified Process είναι δομημένη σε δύο διαστάσεις:

- 1) Χρόνο - Χωρισμός του κύκλου ζωής σε φάσεις και επαναλήψεις.
- 2) Τμήματα διαδικασίας - Καλά ορισμένες εργασίες.

Η δόμηση ενός έργου σε σχέση με το χρόνο ακολουθεί τις εξής φάσεις που έχουν σχέση με το χρόνο:

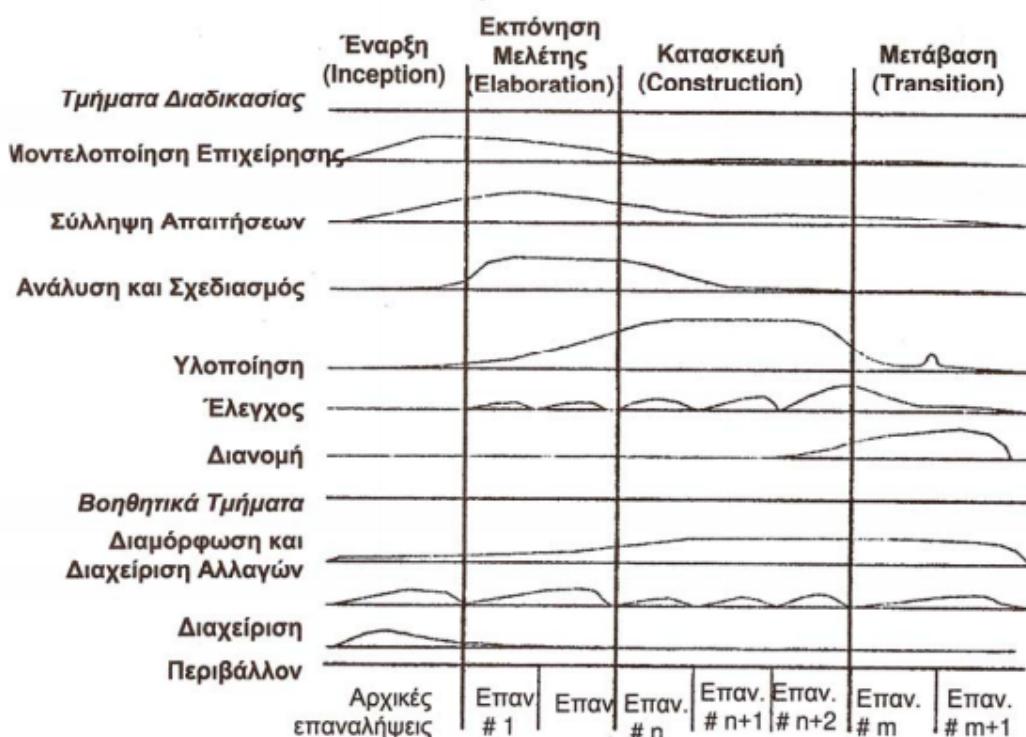
- 1) Έναρξη (Inception): Καθορίζει την προοπτική του έργου.
- 2) Εκπόνηση μελέτης (Elaboration): Σχεδιασμός των απαιτούμενων δραστηριοτήτων και πόρων. Καθορισμός των χαρακτηριστικών και σχεδιασμός της αρχιτεκτονικής.
- 3) Κατασκευή (Construction): Ανάπτυξη του προϊόντος σε μια σειρά βηματικών επαναλήψεων.
- 4) Μετάβαση (Transition): Προμήθευση του προϊόντος στην κοινότητα χρηστών (παραγωγή, διανομή, εκπαίδευση).

Η δόμηση έργου σύμφωνα με τη διάσταση των τμημάτων διαδικασίας περιλαμβάνει τις ακόλουθες δραστηριότητες:

- 1) Σύλληψη απαιτήσεων (Requirements capture): Μια αφήγηση του τι πρέπει να κάνει το σύστημα.
- 2) Ανάλυση και σχεδιασμός (Analysis and design): Μια περιγραφή του πως θα υλοποιηθεί το σύστημα.
- 3) Υλοποίηση (Implementation): Η παραγωγή των κώδικα.
- 4) Έλεγχος (Test): Η επαλήθευση του συστήματος.

Σχήμα 2.1: Κύκλος Ζωής Ανάπτυξης Λογισμικού

### Φάσεις



## **3. ΦΑΣΗ: ΕΝΑΡΞΗ (INCEPTION)**

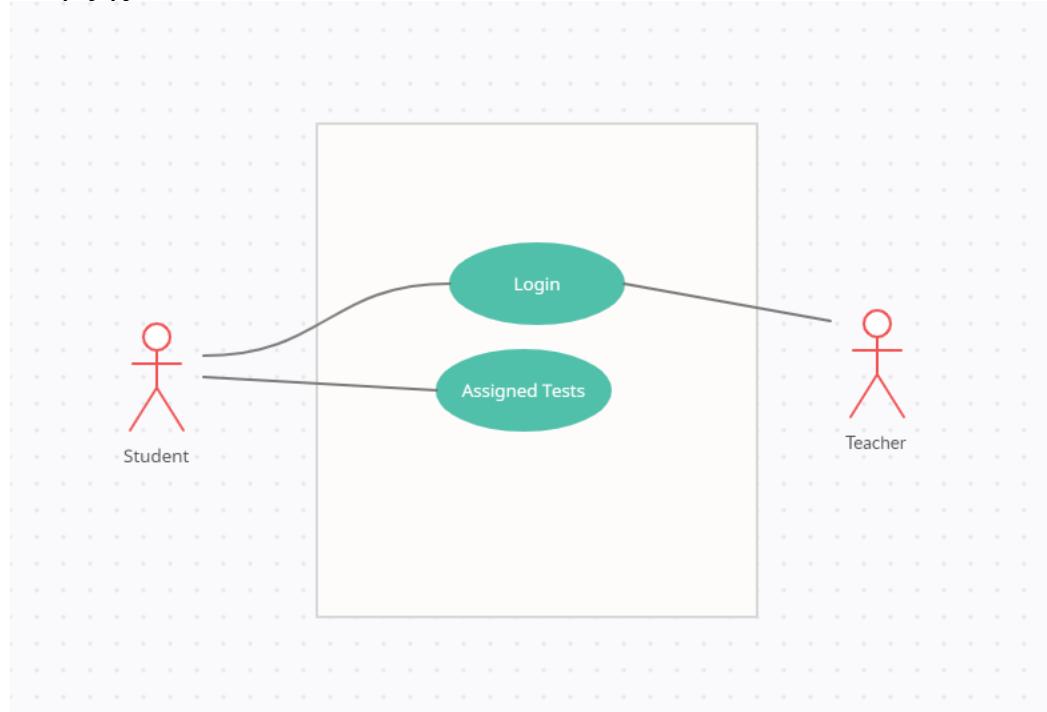
### **3.1 Σύλληψη απαιτήσεων**

Για την παρούσα εργασία, θα χρειαστούν: α) η γλώσσα προγραμματισμού Java, β) το προγραμματιστικό εργαλείο IntelliJ IDEA, γ) η χρήση του Java Servlet Programming, δηλαδή η χρήση των servlet για την δημιουργία και την εκτέλεση της εφαρμογής, δ) η γλώσσα βάσεων δεδομένων PostgreSQL, η οποία θα χρησιμοποιηθεί για την κατασκευή της βάσης δεδομένων και τέλος ε) ο server, Apache Tomcat.

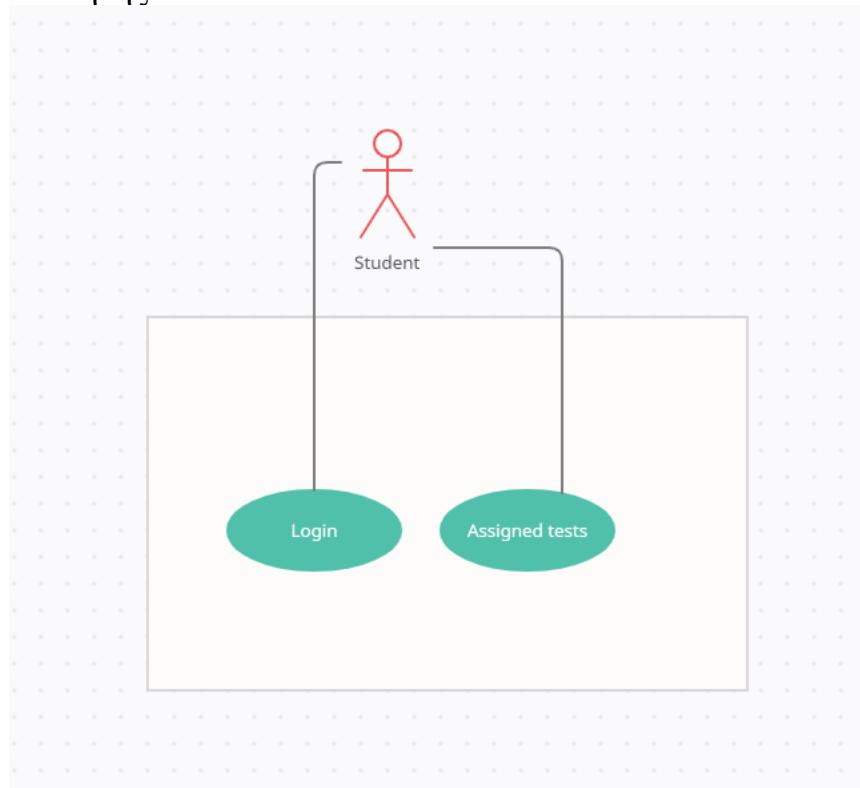
### **3.2 Ανάλυση-Σχεδιασμός**

#### **3.2.1 Διαγράμματα περιπτώσεων χρήσης**

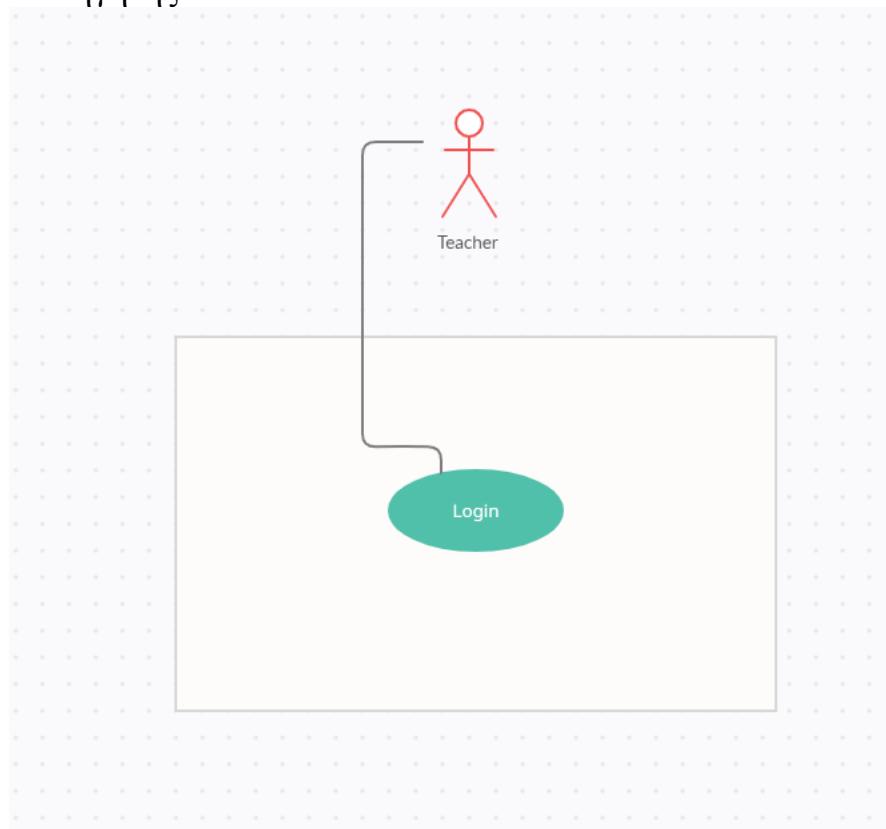
Παρακάτω, παρουσιάζονται τα διαγράμματα περιπτώσεων χρήσης και γενικά και ξεχωριστά για τον μαθητή και τον καθηγητή, στην φάση έναρξης:



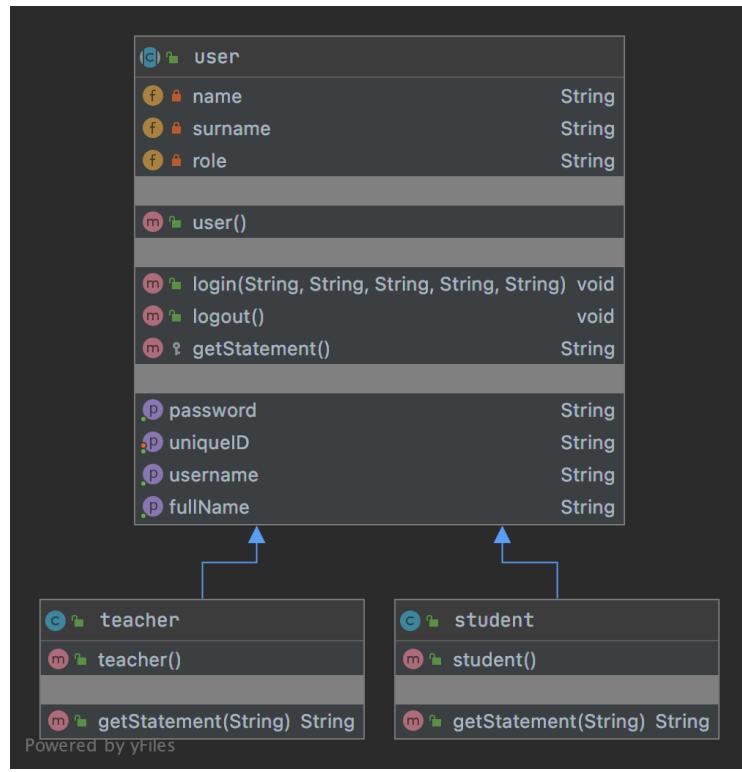
Μαθητής:



Καθηγητής:



### 3.2.2 Διαγράμματα τάξεων



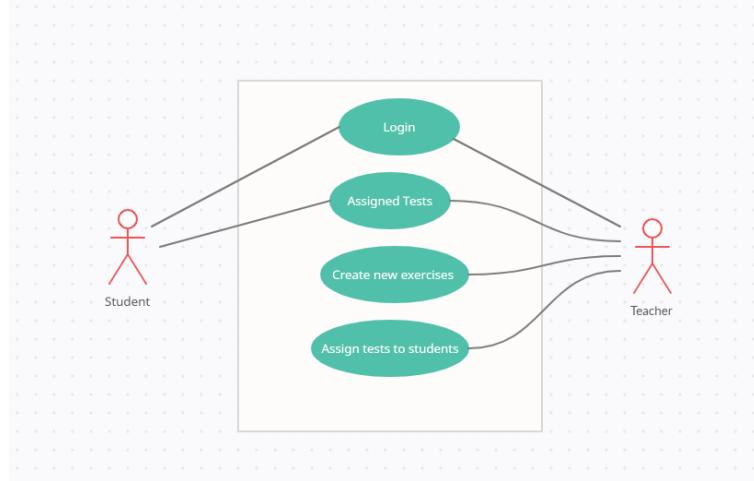
## **4. ΦΑΣΗ: ΕΚΠΟΝΗΣΗ ΜΕΛΕΤΗΣ (ELABORATION)**

Η εκπόνηση μελέτης είναι ο σχεδιασμός των απαιτούμενων δραστηριοτήτων και πόρων και καθορισμός των χαρακτηριστικών και σχεδιασμός της αρχιτεκτονικής.

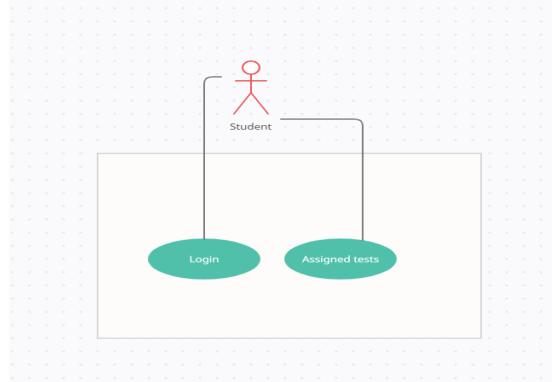
### **4.1 Ανάλυση-Σχεδιασμός**

#### **4.1.1 Διαγράμματα περιπτώσεων χρήσης (2<sup>η</sup> έκδοση)**

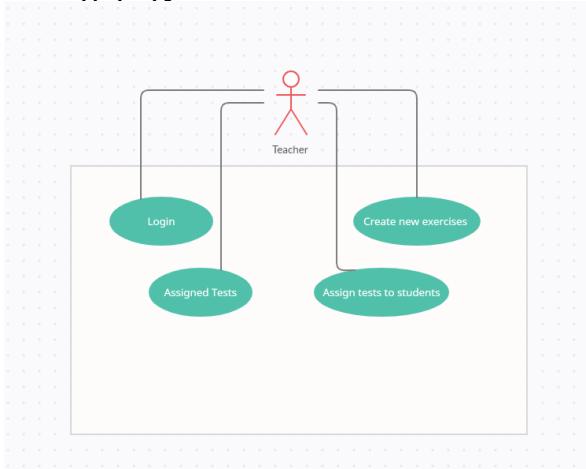
Παρακάτω φαίνεται η δεύτερη έκδοση του διαγράμματος περιπτώσεων χρήσης. Αυτή η έκδοση δεν είναι η τελική. Προστέθηκαν παραπάνω ενέργειες από αυτές που περιείχε το αρχικό διάγραμμα.



Μαθητής:

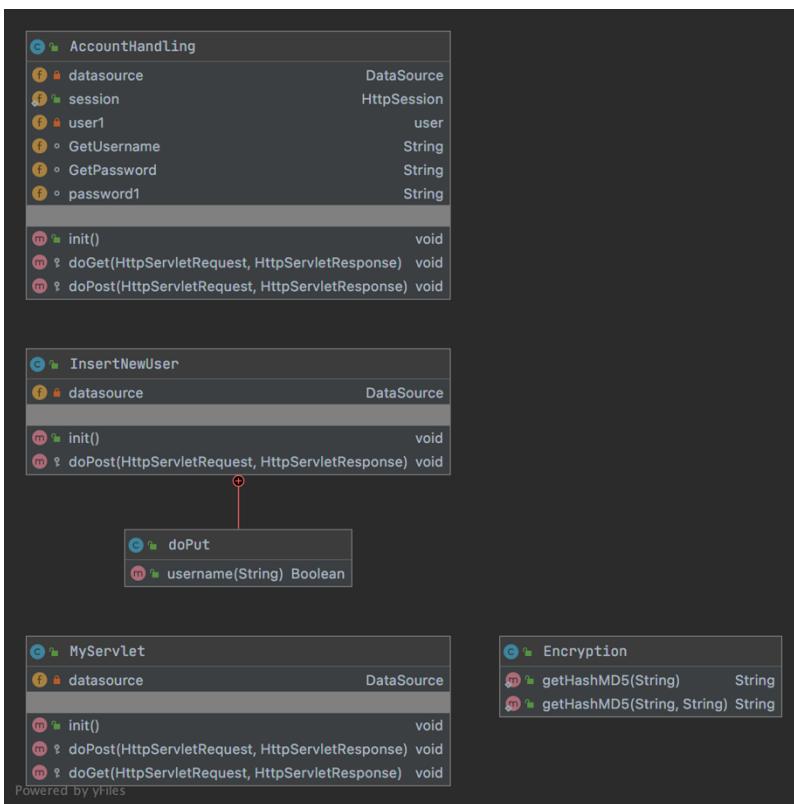


Καθηγητής:



#### 4.1.2 Διαγράμματα τάξεων (2<sup>η</sup> έκδοση)

Παρακάτω φαίνεται η δεύτερη έκδοση του διαγράμματος τάξεων.



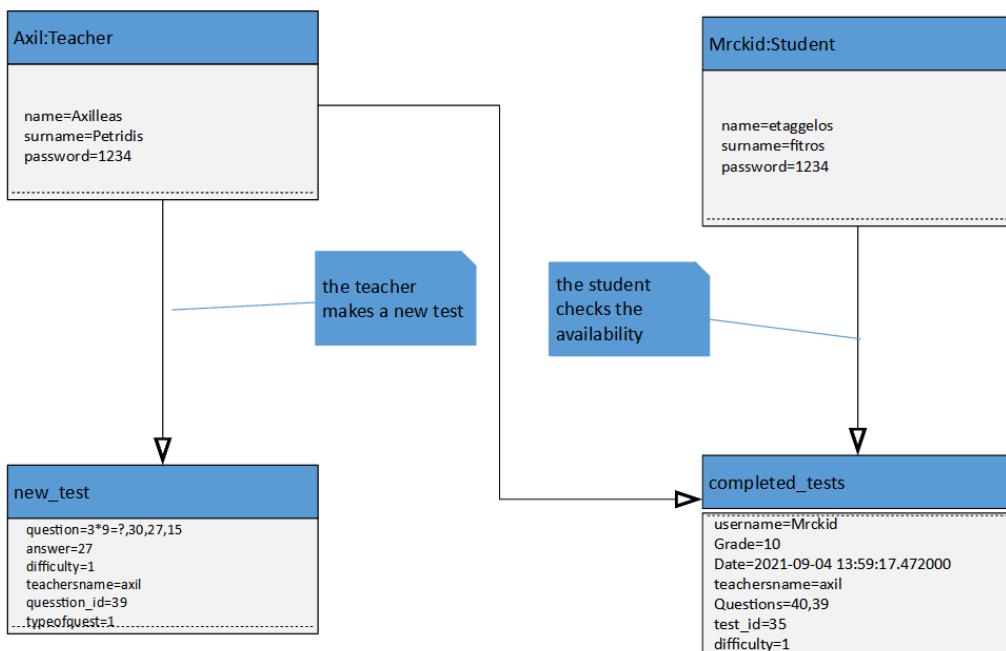
#### 4.1.3 Διάγραμμα Αντικειμένων – Object Diagram (1η έκδοση)

Το Διάγραμμα Αντικειμένων είναι ένα στιγμιότυπο των αντικειμένων του συστήματος κάποια συγκεκριμένη χρονική στιγμή. Χρησιμοποιείται για τον σχεδιασμό στατικής δομής της εφαρμογής και απεικονίζει τα αντικείμενα μαζί με τους συνδέσμους τους.

Κάθε αντικείμενο αναπαρίσταται με ένα ορθογώνιο, το οποίο περιέχει είτε το όνομα του αντικειμένου, είτε το όνομα και την τάξη του αντικειμένου (χωρισμένα με άνω και κάτω τελεία), ή μόνο την τάξη του αντικειμένου (σε αυτή την περίπτωση το αντικείμενο θεωρείται ότι είναι ανώνυμο). Το όνομα από μόνο του αντιστοιχεί σε ένα ελλιπές μοντέλο, στο οποίο η τάξη του αντικειμένου δεν έχει ακόμα καθοριστεί.

Τα ορθογώνια που συμβολίζουν αντικείμενα μπορούν επίσης να περιλαμβάνουν ένα δεύτερο τμήμα που περιέχει τις τιμές των χαρακτηριστικών. Ο τύπος του χαρακτηριστικού είναι ήδη καθορισμένος στην τάξη, έτσι δεν είναι απαραίτητο να εμφανίζεται στην αναπαράσταση των αντικειμένων. Τα αντικείμενα συνδέονται μέσω συνδέσμων, οι οποίοι είναι στιγμιότυπα των συσχετισμών μεταξύ των τάξεων των αντικειμένων που μελετώνται. Οι περισσότεροι σύνδεσμοι είναι δυαδικοί.

Παρακάτω φαίνεται η πρώτη έκδοση του διαγράμματος αντικειμένων:



#### 4.1.4 Διάγραμμα Συνεργασίας – Collaboration Diagram (1η έκδοση)

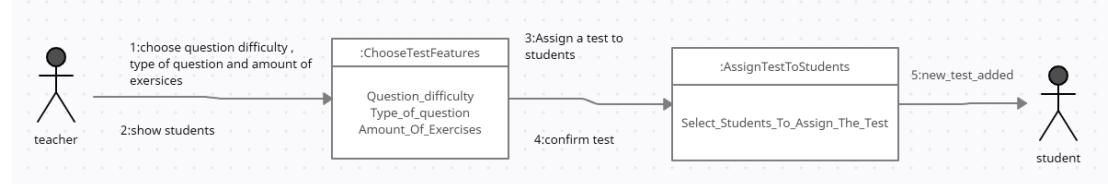
Τα διαγράμματα συνεργασίας απεικονίζουν τις αλληλεπιδράσεις ανάμεσα στα αντικείμενα, χρησιμοποιώντας μία στατική χωρική δομή, που διευκολύνει την αναπαράσταση της συνεργασίας ενός συνόλου αντικειμένων. Τα διαγράμματα συνεργασίας εκφράζουν τόσο το περιβάλλον ενός συνόλου αντικειμένων (μέσω των αντικειμένων και των συνδέσμων) όσο και την αλληλεπίδραση ανά μεσα σε αυτά τα

αντικείμενα (αναπαριστώντας εκπομπές μηνυμάτων). Αυτά τα διαγράμματα είναι μία επέκταση των διαγραμμάτων αντικειμένων.

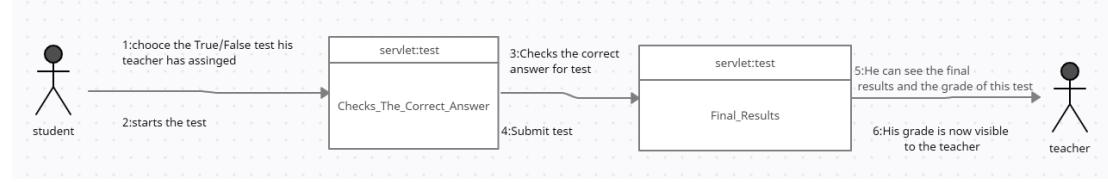
Το περιεχόμενο μίας αλληλεπίδρασης περιλαμβάνει τα ορίσματα, τις τοπικές μεταβλητές που δημιουργήθηκαν κατά την διάρκεια της εκτέλεσης, και τους συνδέσμους ανά μεσα στα αντικείμενα που συμμετέχουν στην αλληλεπίδραση. Μία αλληλεπίδραση εκτελείται από ένα σύνολο αντικειμένων που συνεργάζονται ανταλλάσσοντας μηνύματα. Αυτά τα μηνύματα εμφανίζονται κατά μήκος των συνδέσμων που συνδέουν τα αντικείμενα, χρησιμοποιώντας βέλη που δείχνουν προς τον παραλήπτη του μηνύματος.

Παρακάτω φαίνεται η πρώτη έκδοση ων διαγραμμάτων συνεργασίας:

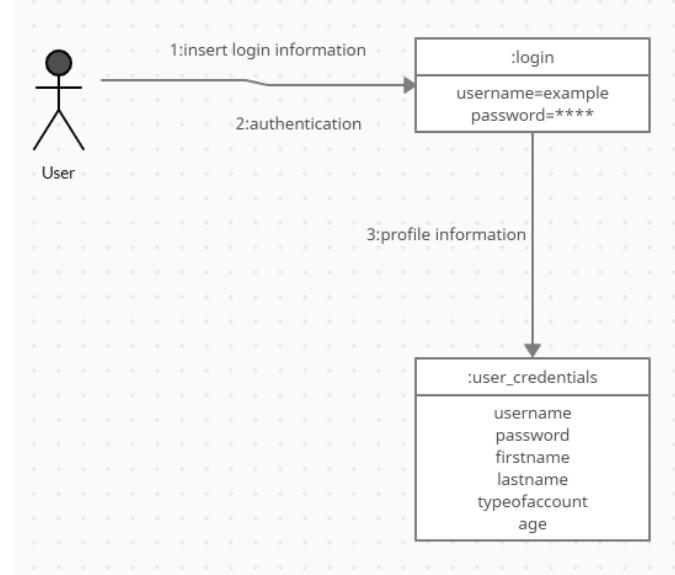
Από τον καθηγητή στον μαθητή:



Από τον μαθητή στον καθηγητή:



User Login:



#### 4.1.5 Διάγραμμα Σειράς – Sequence Diagram (1η έκδοση)

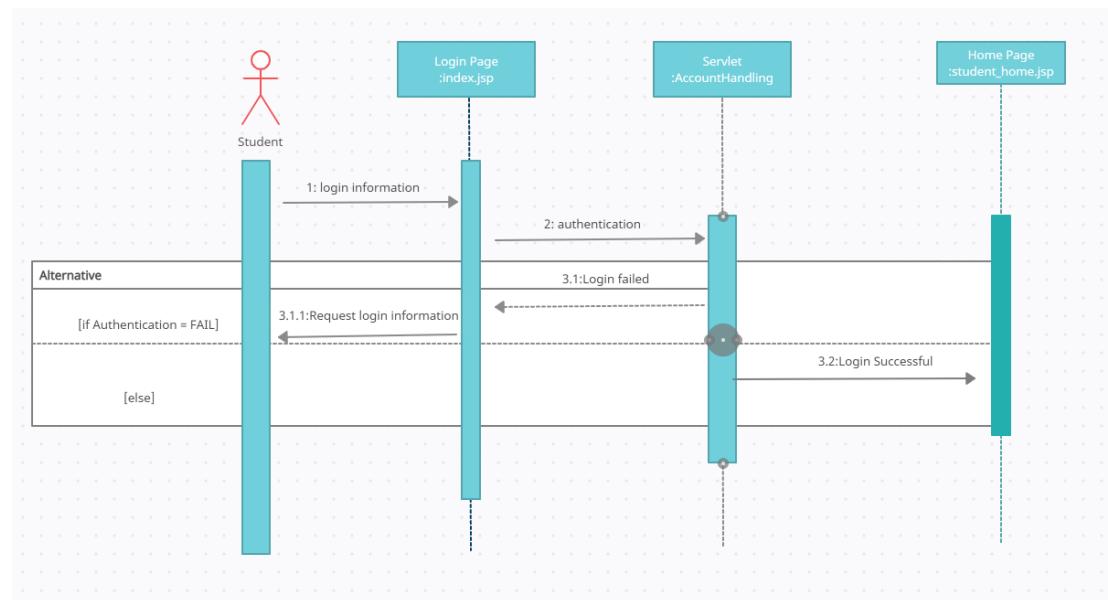
Τα διαγράμματα σειράς αναπαριστούν αλληλεπιδράσεις ανάμεσα στα αντικείμενα από μία χρονική άποψη. Σε αντίθεση με τα διαγράμματα συνεργασίας, το περιβάλλον των αντικειμένων δεν αναπαρίσταται σαφώς. Η αναπαράσταση επικεντρώνεται στην έκφραση των αλληλεπιδράσεων.

Σε ένα διάγραμμα σειράς αναπαρίσταται μία αλληλεπίδραση ανάμεσα σε αντικείμενα, που επικεντρώνεται στη χρονολογία της εκπομπής μηνυμάτων. Ένα αντικείμενο αναπαρίσταται με ένα ορθογώνιο και μία κάθετη γραμμή, που καλείται γραμμή ζωής του αντικειμένου.

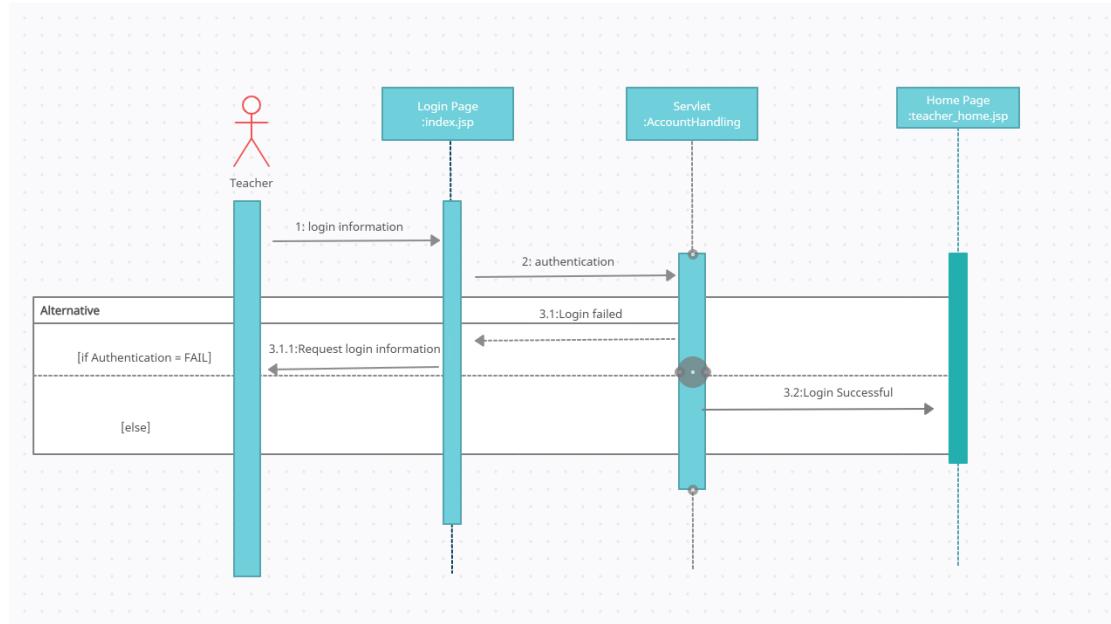
Τα αντικείμενα επικοινωνούν ανταλλάσσοντας μηνύματα, τα οποία αναπαριστώνται με οριζόντια βέλη σχεδιασμένα από τον αποστολέα του μηνύματος προς τον παραλήπτη του μηνύματος. Η σειρά αποστολής μηνύματος καθορίζεται από τη θέση του μηνύματος στον κάθετο άξονα. Ο κάθετος άξονας μπορεί να περιγράφεται με ετικέτες για να εκφράζει ακριβώς τους χρονικούς περιορισμούς.

Παρακάτω φαίνεται η πρώτη έκδοση των διαγραμμάτων σειράς:

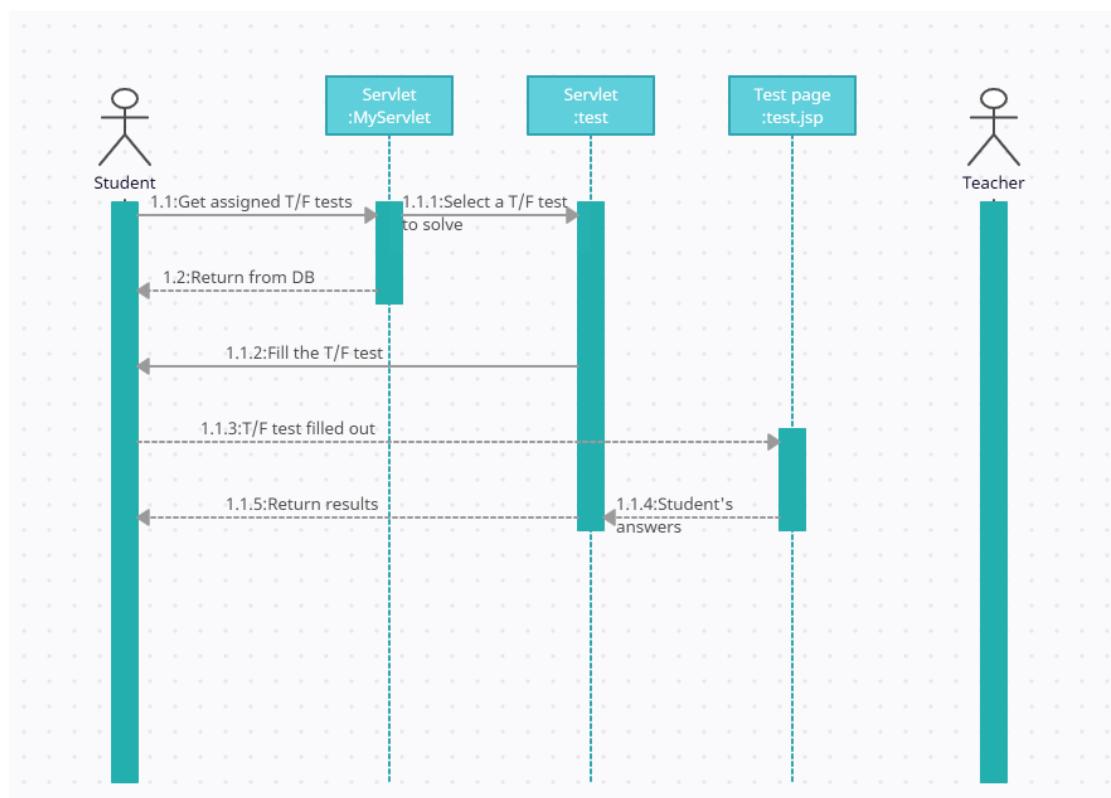
Login Student:



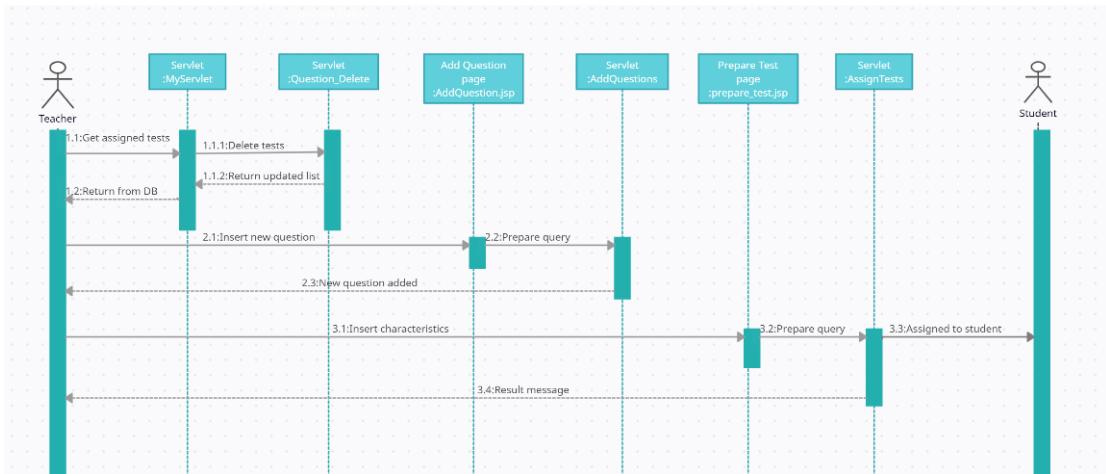
## Login Teacher:



## Student-to-Teacher:



## Teacher-to-Student:



### 4.1.6 Διάγραμμα Δραστηριοτήτων – Activity Diagram (1η έκδοση)

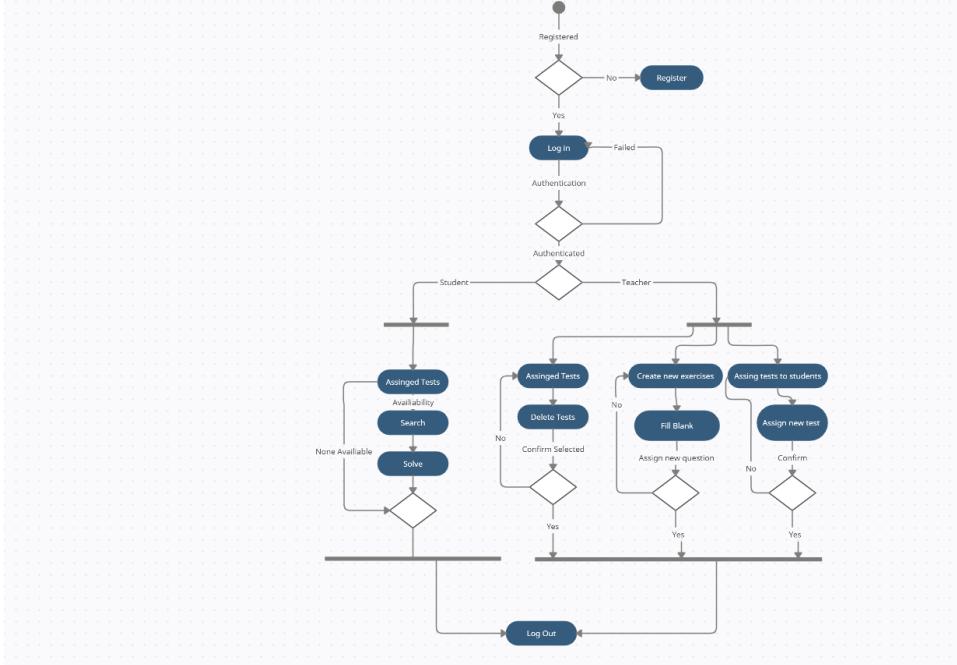
Ένα διάγραμμα δραστηριοτήτων είναι μία παραλλαγή των διαγραμμάτων καταστάσεων οργανωμένο σύμφωνα με ενέργειες, και στοχεύει κυρίως στην αναπαράσταση της εσωτερικής συμπεριφοράς μίας μεθόδου (την υλοποίηση μίας λειτουργίας) ή μίας περίπτωσης χρήσης. Αναπαριστά την κατάσταση εκτέλεσης ενός μηχανισμού σαν μία σειρά βημάτων που ομαδοποιούνται σειριακά σαν παράλληλες διακλαδώσεις ροής ελέγχου.

Κάθε δραστηριότητα αναπαριστά μία συγκεκριμένη κατάσταση στην εκτέλεση της περικλείουσας μεθόδου. Οι δραστηριότητες συνδέονται με αυτόματες μεταβάσεις, που αναπαριστώνται με βέλη, με τον ίδιο τρόπο όπως οι μεταβάσεις στα διαγράμματα καταστάσεων. Όταν μία δραστηριότητα τερματίζει, η μετάβαση ενεργοποιείται και η επόμενη δραστηριότητα αρχίζει. Οι δραστηριότητες δεν έχουν εσωτερικές μεταβάσεις ή μεταβάσεις που ενεργοποιούνται από γεγονότα.

Οι μεταβάσεις ανάμεσα στις δραστηριότητες μπορούν να φυλαχτούν με αμοιβαία αποκλειόμενες συνθήκες που λαμβάνουν λογικές τιμές 0 ή 1. Οι φύλακες φαίνονται δίπλα στις μεταβάσεις την ενεργοποίηση των οποίων επικυρώνουν.

Τα διαγράμματα δραστηριοτήτων δείχνουν συγχρονισμούς ανάμεσα στις ροές ελέγχου χρησιμοποιώντας ράβδους συγχρονισμού. Μία ράβδος συγχρονισμού κάνει δυνατό το άνοιγμα και το κλείσιμο παράλληλων διακλαδώσεων μέσα στη ροή εκτέλεσης μίας μεθόδου ή μίας περίπτωσης χρήσης. Οι μεταβάσεις που αρχίζουν από τη μπάρα συγχρονισμού ενεργοποιούνται ταυτόχρονα.

Παρακάτω φαίνεται η πρώτη έκδοση του διαγράμματος δραστηριοτήτων:



#### 4.1.7 Διάγραμμα Καταστάσεων –State Chart Diagram (1η έκδοση)

Τα διαγράμματα καταστάσεων αναπαριστούν μηχανές καταστάσεων από την άποψη των καταστάσεων και των μεταβάσεων.

Η συμπεριφορά των αντικειμένων μίας τάξης μπορεί να περιγραφεί τυπικά με όρους καταστάσεων και γεγονότων, χρησιμοποιώντας μία μηχανή καταστάσεων συνδεδεμένη με την τάξη που μελετάται. Αντικείμενα που δεν παρουσιάζουν μία πολύ αντιδραστική σε μπεριφορά μπορεί να θεωρηθεί ότι μένουν πάντα στην ίδια κατάσταση. Σε αυτή την περίπτωση οι τάξεις τους δεν κατέχουν μία μηχανή καταστάσεων.

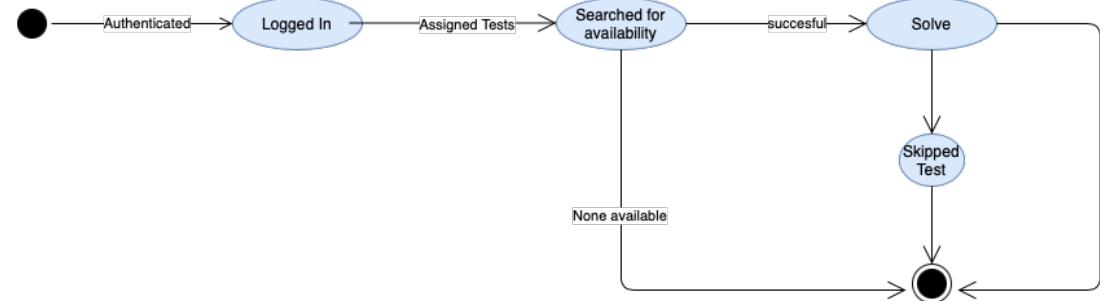
Μία μηχανή καταστάσεων είναι μία αφαίρεση όλων των πιθανών συ μπεριφορών, όμοια με τον τρόπο που τα διαγράμματα τάξεων είναι αφαιρέσεις της στατικής δομής. Κάθε αντικείμενο ακολουθεί τη συ μπεριφορά που περιγράφεται στη μηχανή καταστάσεων και συσχετίζεται με την τάξη του και είναι, σε μία δεδομένη χρονική στιγμή, σε μία κατάσταση που χαρακτηρίζει τις δυναμικές του καταστάσεις.

Οι μηχανές καταστάσεων και τα σενάρια είναι συμπληρωματικά. Τα σενάρια αναπαριστώνται με μία συνεργασία μεταξύ αντικειμένων. Ο τύπος της αλληλεπίδρασης που συμβαίνει μεταξύ αντικειμένων που συνεργάζονται μέσα σε ένα σενάριο καθορίζεται από τις αντίστοιχες

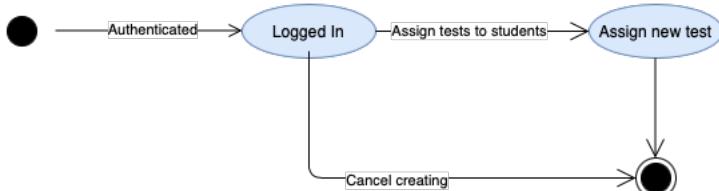
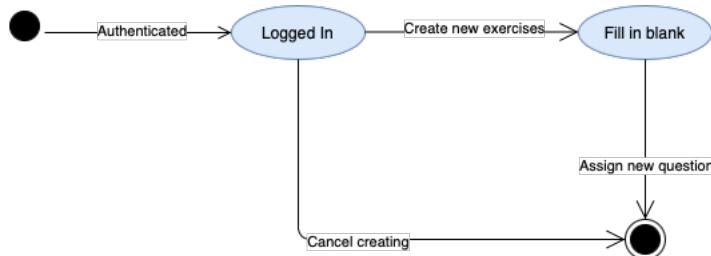
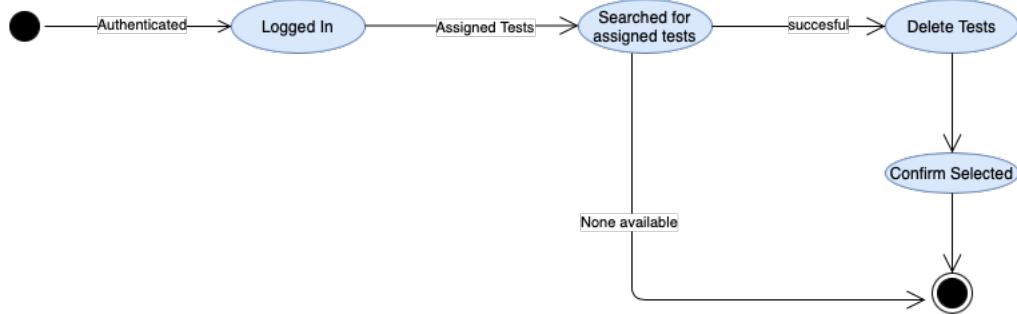
καταστάσεις των διάφορων αντικειμένων. Οι μηχανές καταστάσεων μπορούν να χρησιμοποιηθούν για να περιγράψουν τη συμπεριφορά συνόλων αντικειμένων συσχετίζοντας μία μηχανή καταστάσεων με ένα σύνθετο αντικείμενο, ή ακόμα και με μία περίπτωση χρήσης (use case).

Παρακάτω φαίνεται η πρώτη έκδοση του διαγράμματος καταστάσεων για τον μαθητή και τον καθηγητή ξεχωριστά:

Μαθητής:



Καθηγητής:



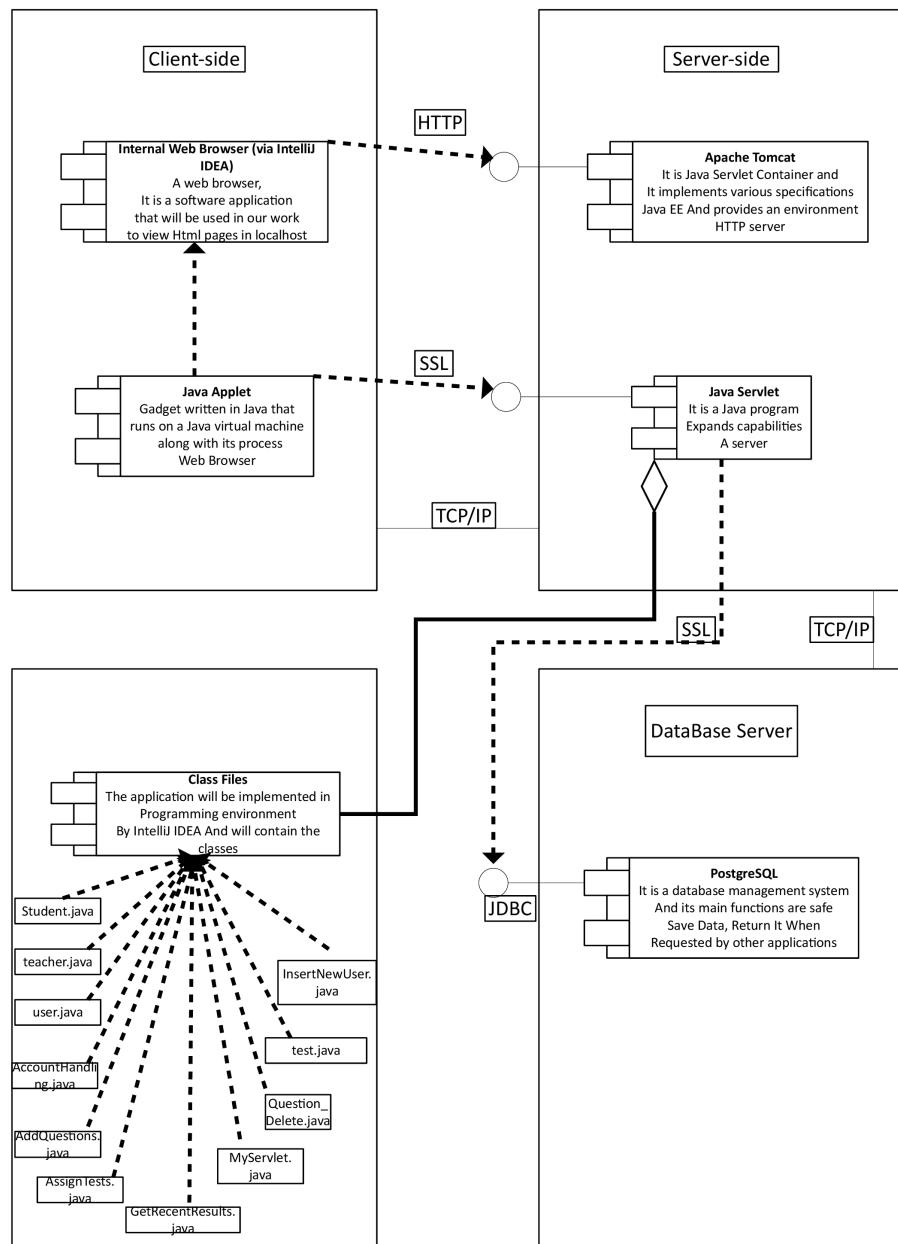
#### 4.1.8 Διάγραμμα Εξαρτημάτων –Component Diagram (1η έκδοση)

Τα εξαρτήματα αναπαριστούν όλα τα είδη των στοιχείων που αναφέρονται στην τμηματοποίηση των εφαρμογών λογισμικού. Μεταξύ άλλων, αυτά μπορεί να είναι απλά αρχεία ή βιβλιοθήκες που «διαβάζονται» και χρησιμοποιούνται δυναμικά.

Εξ' ορισμού, κάθε τάξη στο λογικό μοντέλο έχει έναν προσδιορισμό και ένα κυρίως σώμα. Ο προσδιορισμός περιέχει το σύστημα διεπαφής της τάξης, η οποία αναπαρίσταται με το συμβολισμό της UML με ένα μικρό κύκλο που συνδέεται μέσω μιας γραμμής με το σώμα της τάξης. Το σώμα περιέχει την υλοποίηση της ίδιας της τάξης και αναπαρίσταται με το ίδιο το σύμβολο του εξαρτήματος.

Οι σχέσεις εξάρτησης χρησιμοποιούνται στα διαγράμματα εξαρτημάτων για να δηλώσουν ότι ένα εξάρτημα αναφέρεται σε υπηρεσίες που προσφέρονται από άλλα εξαρτήματα. Αυτός ο τύπος εξάρτησης ανακλά επιλογές υλοποίησης. Μία σχέση εξάρτησης αναπαρίσταται με ένα βέλος με διακεκομένη γραμμή σχεδιασμένο από τον πελάτη προς τον προμηθευτή.

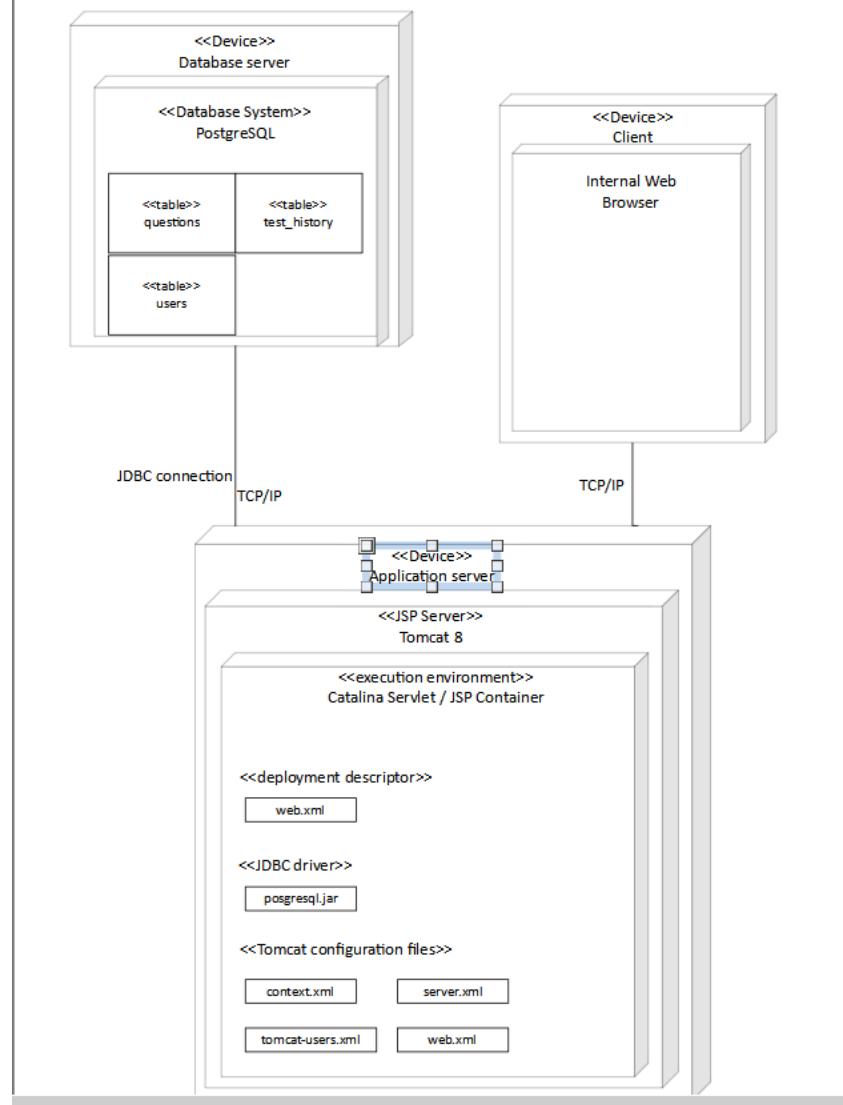
Παρακάτω φαίνεται η πρώτη έκδοση του διαγράμματος εξαρτημάτων:



#### 4.1.9 Διάγραμμα Διανομής- Deployment Diagram (1η έκδοση)

Τα διαγράμματα διανομής δείχνουν τη φυσική διάταξη των διάφορων εξαρτημάτων (κόμβων) υλικού, που αποτελούν ένα σύστημα, όπως και τη διανομή των εκτελέσιμων προγραμμάτων σε αυτό το υλικό.

Κάθε πόρος υλικού αναπαρίσταται με ένα κύβο. Κάθε σύστημα μπορεί να περιγραφεί με ένα μικρό αριθμό διαγραμμάτων διανομής. Πολλές φορές, ένα μόνο διάγραμμα είναι αρκετό.



## 4.2 Υλοποίηση- Έλεγχος

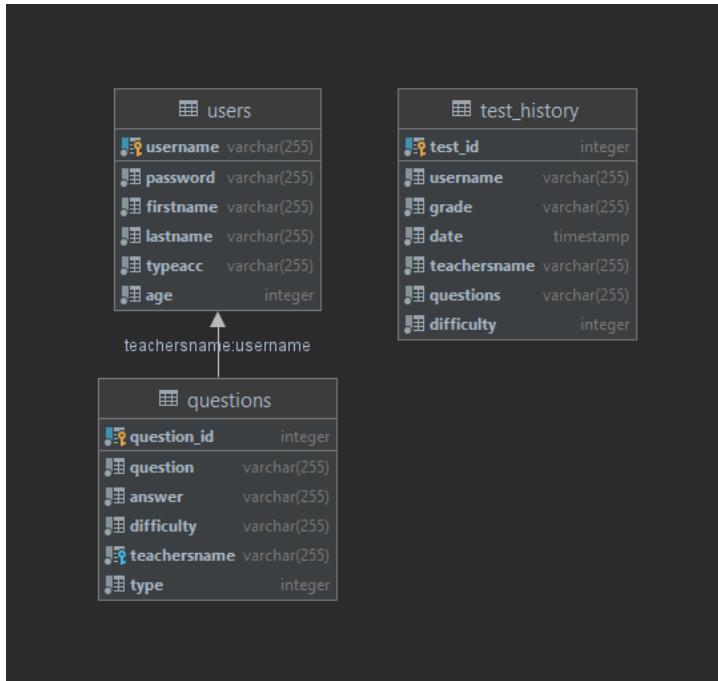
### 4.2.1 Υλοποίηση: 1η εκτελέσιμη έκδοση

Με γνώμονα τα προαναφερθέντα διαγράμματα, πραγματοποιήθηκε η πρώτη εκτελέσιμη έκδοση της εφαρμογής, λαμβάνοντας υπόψιν και την ανάλυση

απαιτήσεων.

Για την εύρυθμη λειτουργία της εφαρμογής, δημιουργήθηκε Βάση Δεδομένων PostgreSQL.

Παρακάτω φαίνεται το διάγραμμα σχέσεων της Βάσης (Relationship Diagram) το οποίο απεικονίζει τις σχέσεις μεταξύ των πινάκων. Δίνει πληροφορίες σχετικά με τα πρωτεύοντα κλειδιά, τα ξένα κλειδιά και τα πεδία του κάθε πίνακα.



Η Βάση Δεδομένων αποτελείται από τέσσερις πίνακες:

- Users: ο πίνακας αυτός περιέχει τις πληροφορίες για όλους τους χρήστες της εφαρμογής. Αποθηκεύει το username, το ονοματεπώνυμο (firstname, lastname), το κωδικό (password), την ιδιότητα (typeacc) ενός χρήστη και την ηλικία του (age). Η ιδιότητα παίρνει τιμές Student ή Teacher.
- Test History: ο πίνακας αυτός περιέχει το αρχείο των διαγωνισμάτων (ολοκληρωμένων και μη).
- Questions: αυτός ο πίνακας χρησιμοποιείται σαν τράπεζα ερωτήσεων, για τα διαγωνίσματα που ορίζει ο κάθε καθηγητής.

Χρησιμοποιήθηκε 3-tier αρχιτεκτονίκη για την υλοποίηση του προγράμματος.

Τεχνολογίες που εφαρμόστηκαν για την 3-tier ανάπτυξη αυτής της εφαρμογής στο Διαδίκτυο είναι:

- Application Server: Apache Tomcat, παρέχει την δυνατότητα δημιουργίας δυναμικών ιστοσελίδων μέσω Servlets.
- Βάση Δεδομένων PostgreSQL.

- Προγραμματιστικό περιβάλλον ανάπτυξης εφαρμογών 3-tier: JetBrains IntelliJ IDEA.

Η υλοποίηση της διαδικτυακής διεπαφής έγινε και με την δημιουργία .jsp σελίδων. Οι σελίδες αυτές βρίσκονται μέσα στον φάκελο με όνομα <Web Content>.

Αυτά τα αρχεία είναι:

- login.jsp: Χρησιμοποιείται για την σύνδεση των χρηστών στην εφαρμογή.
- logout.jsp: Αντίστοιχο αρχείο για την αποσύνδεση των χρηστών.
- student\_home.jsp: Η αρχική σελίδα των μαθητών.
- teacher\_home.jsp: Η αρχική σελίδα των καθηγητών.
- student.jsp: Σελίδα για την επιλογή διαγωνισμάτων από τον μαθητή.
- test.jsp: Σελίδα για την επίλυση του διαγωνίσματος.
- results.jsp: Σελίδα για την εμφάνιση των αποτελεσμάτων μετά την επίλυση του διαγωνίσματος.
- teacher.jsp: Η σελίδα των ήδη αναθετημένων διαγωνισμάτων από τους καθηγητές.
- AddQuestion.jsp: Η σελίδα για την δημιουργία ασκήσεων.
- prepare\_tesst.jsp: Σελίδα για τον ορισμό των προϋποθέσεων για την ανάθεση του διαγωνίσματος.
- AssignTests.jsp: Η σελίδα για την ανάθεση διαγωνισμάτων στους μαθητές.

#### 4.2.2 Αναφορά ελέγχου για την 1η εκτελέσιμη έκδοση

Παρόλο που η πρώτη εκτελέσιμη έκδοση περιέχει όλα τα βασικά στοιχεία της εφαρμογής, χρειάζεται βελτίωση και προσθήκη επιπρόσθετων λειτουργιών.

Οι λειτουργίες που οφείλουν να υλοποιηθούν στην τελική εκτελέσιμη έκδοση είναι:

- Σε σημεία κλειδία της εφαρμογής θα προστεθεί κουμπί online help με οδηγίες για την σωστή περιήγηση στην εφαρμογή.
- Σελίδα, για την δημιουργία νέου χρήστη (μαθητής/καθηγητής).
- Λίστα για το ιστορικό των διαγωνισμάτων του μαθητή.
- Λίστα για το ιστορικό των διαγωνισμάτων που έχουν αναθέσει οι καθηγητές.

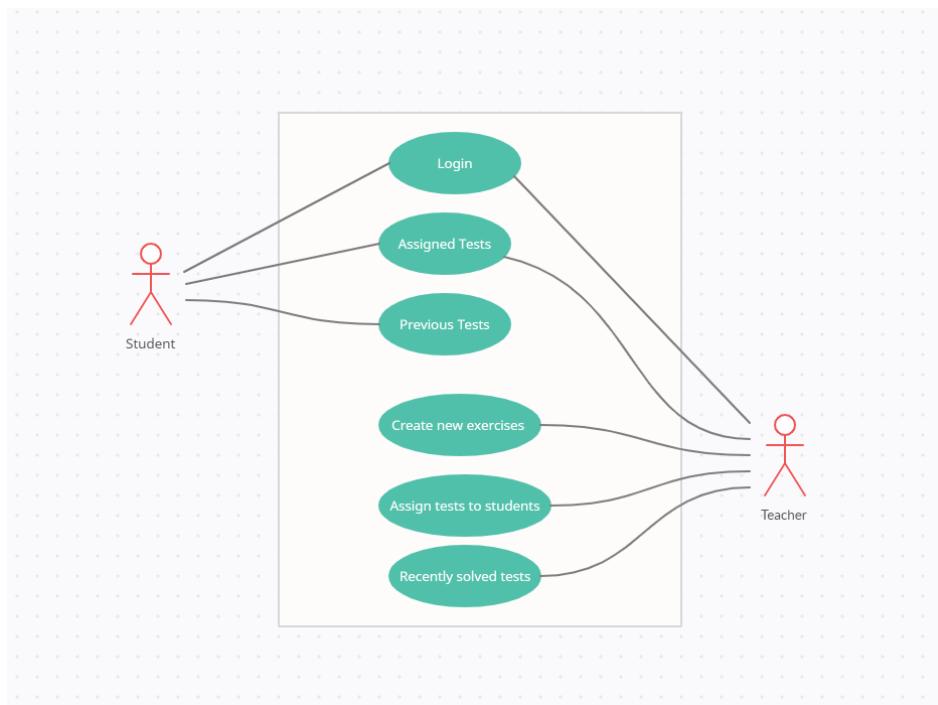
## **5. ΦΆΣΗ: ΚΑΤΑΣΚΕΥΗ (CONSTRUCTION)**

### **5.1 Ανάλυση-Σχεδιασμός**

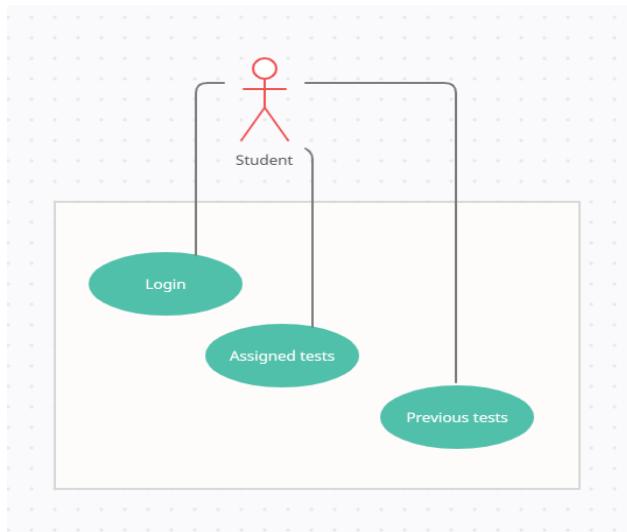
Σε αυτή τη φάση της εφαρμογής, στόχος είναι η καλύτερη οργάνωση του κώδικα και η προσθήκη περαιτέρω ενεργειών που δεν κατάφεραν να υλοποιηθούν στην πρώτη εκτελέσιμη έκδοση.

#### 5.1.1 Διάγραμμα Περιπτώσεων Χρήσης- Use Case Diagram (3η έκδοση)

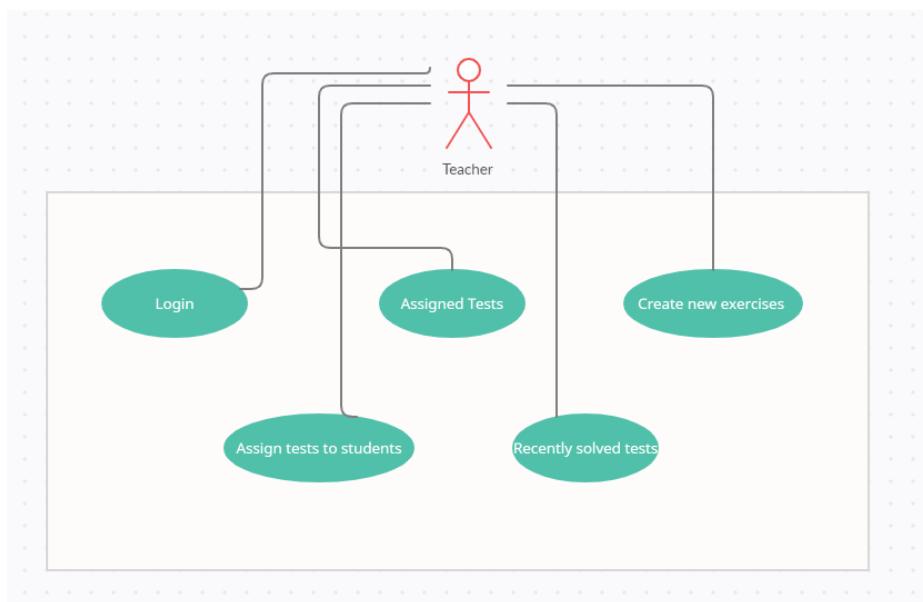
Παρακάτω φαίνεται η 3η και τελική έκδοση του διαγράμματος περιπτώσεων χρήσης. Περιέχει όλες τις πιθανές ενέργειες που μπορούν να τελέσουν οι χρήστες τις εφαρμογής:



Μαθητής:

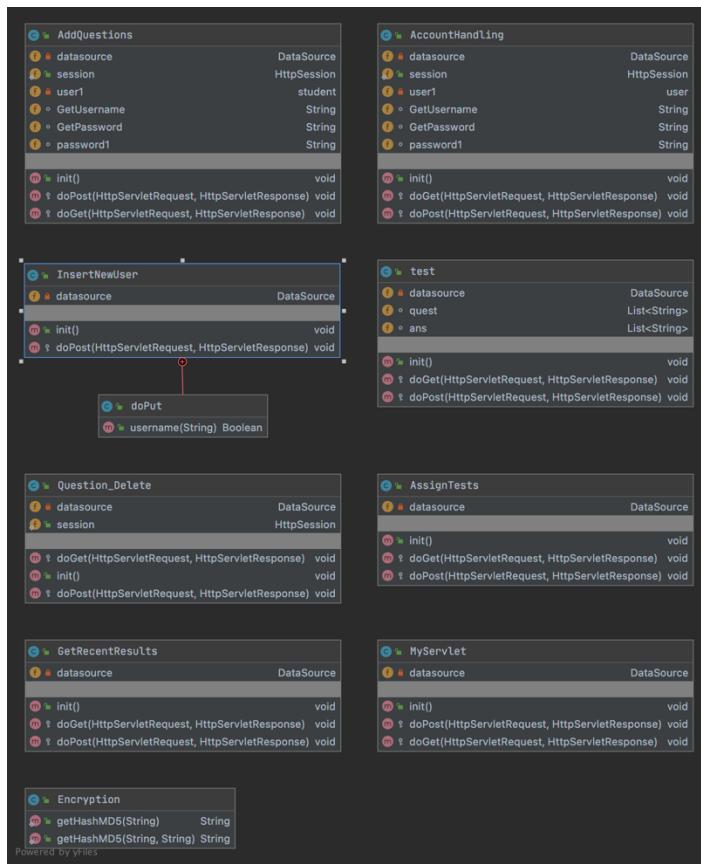


Καθηγητής:



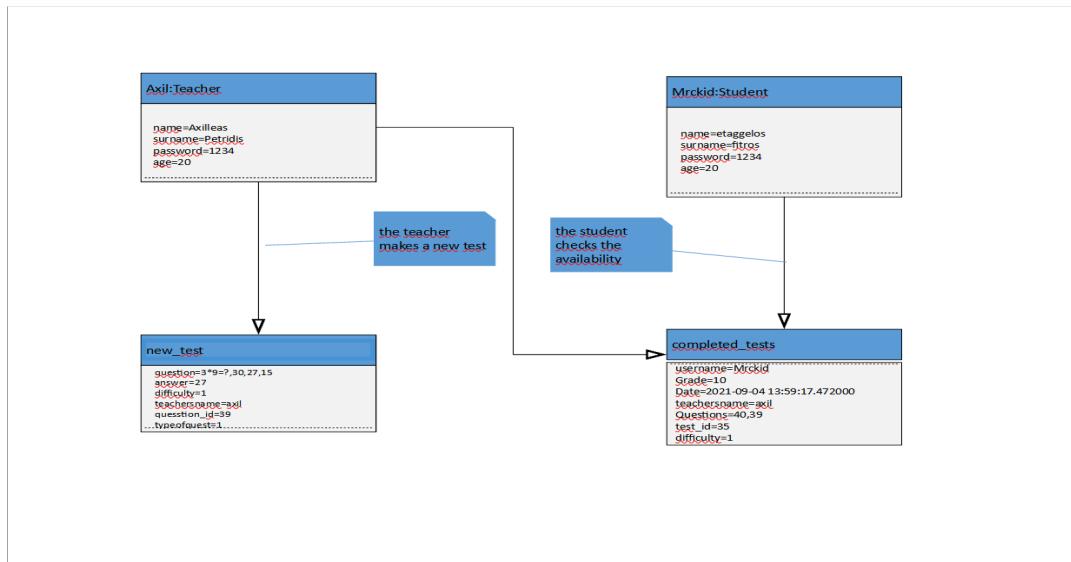
### 5.1.2 Λιάγραμμα Τάξεων – Class Diagram (3η έκδοση)

Παρακάτω παρατίθεται η τελική έκδοση του διαγράμματος τάξεων:



### 5.1.3 Διάγραμμα Αντικειμένων – Object Diagram (2η έκδοση)

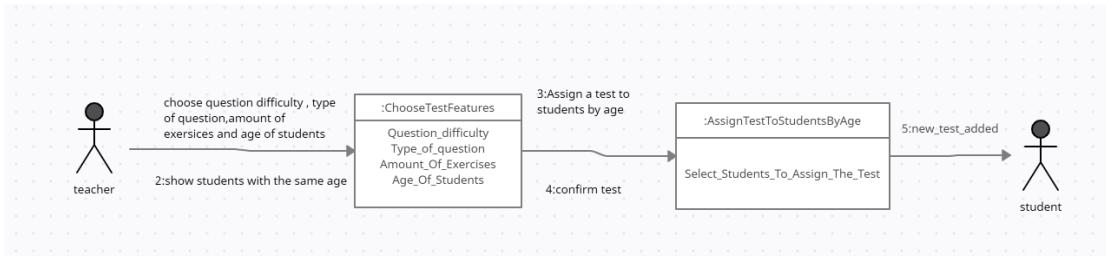
Παρακάτω φαίνεται η δεύτερη και τελική έκδοση του διαγράμματος αντικειμένων:



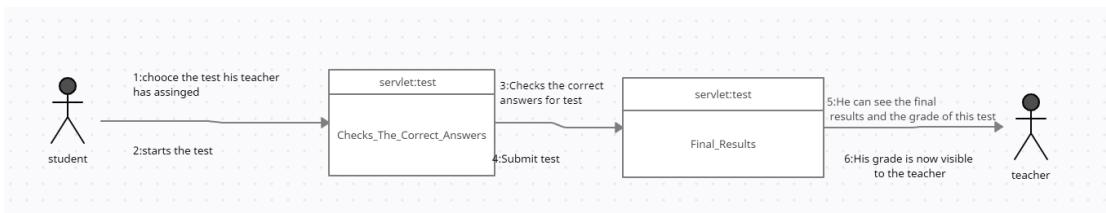
### 5.1.4 Διάγραμμα Συνεργασίας – Collaboration Diagram (2η έκδοση)

Παρακάτω φαίνεται η δεύτερη και τελική έκδοση του διαγράμματος συνεργασίας:

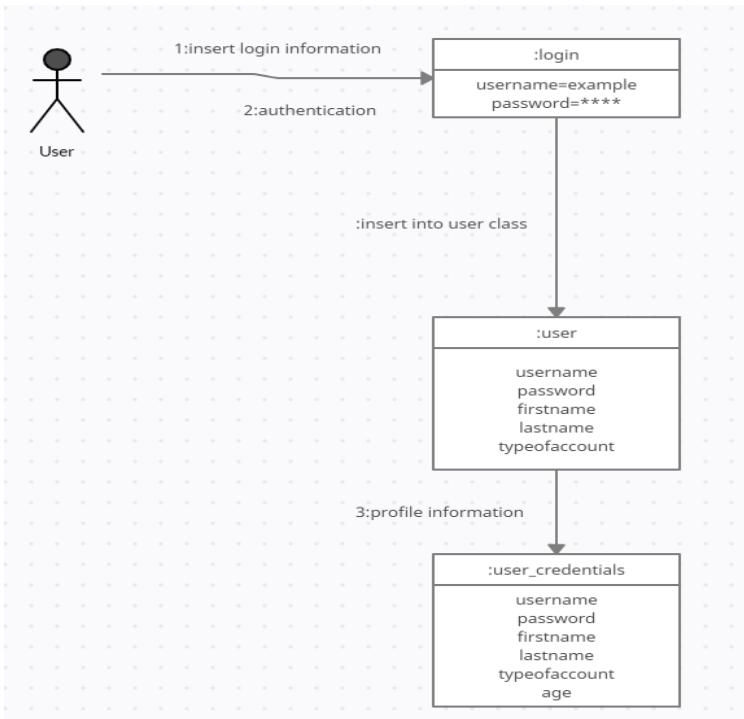
Από τον καθηγητή στον μαθητή:



Από τον μαθητή στον καθηγητή:



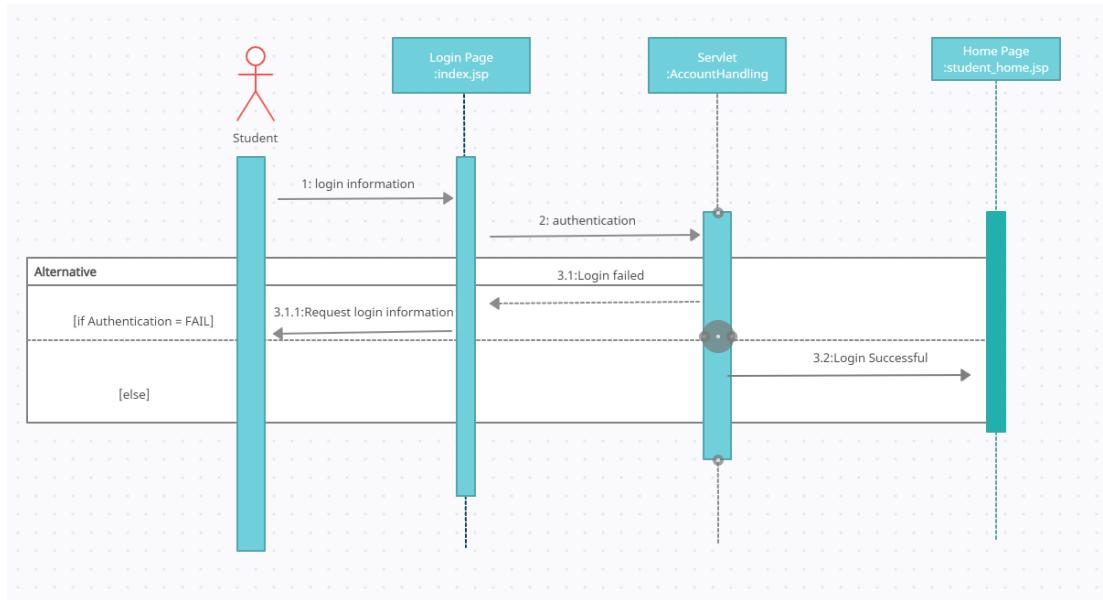
User Login:



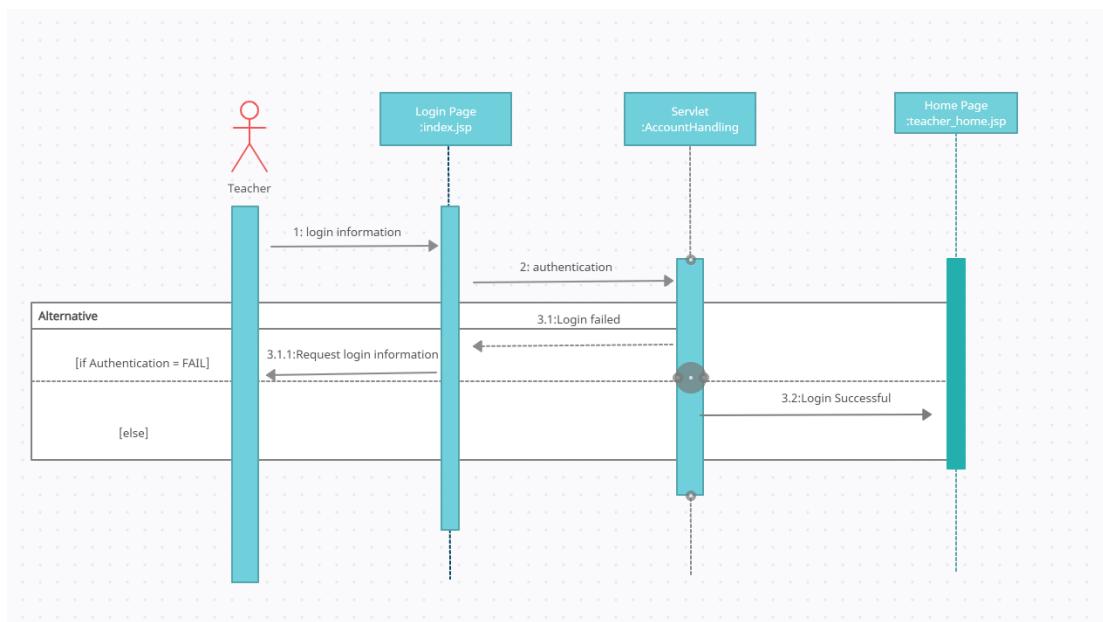
### 5.1.5 Διάγραμμα Σειράς – Sequence Diagram (2η έκδοση)

Παρακάτω φαίνεται η δεύτερη και τελική έκδοση του διαγράμματος σειράς για την αλληλεπίδραση μεταξύ των αντικειμένων:

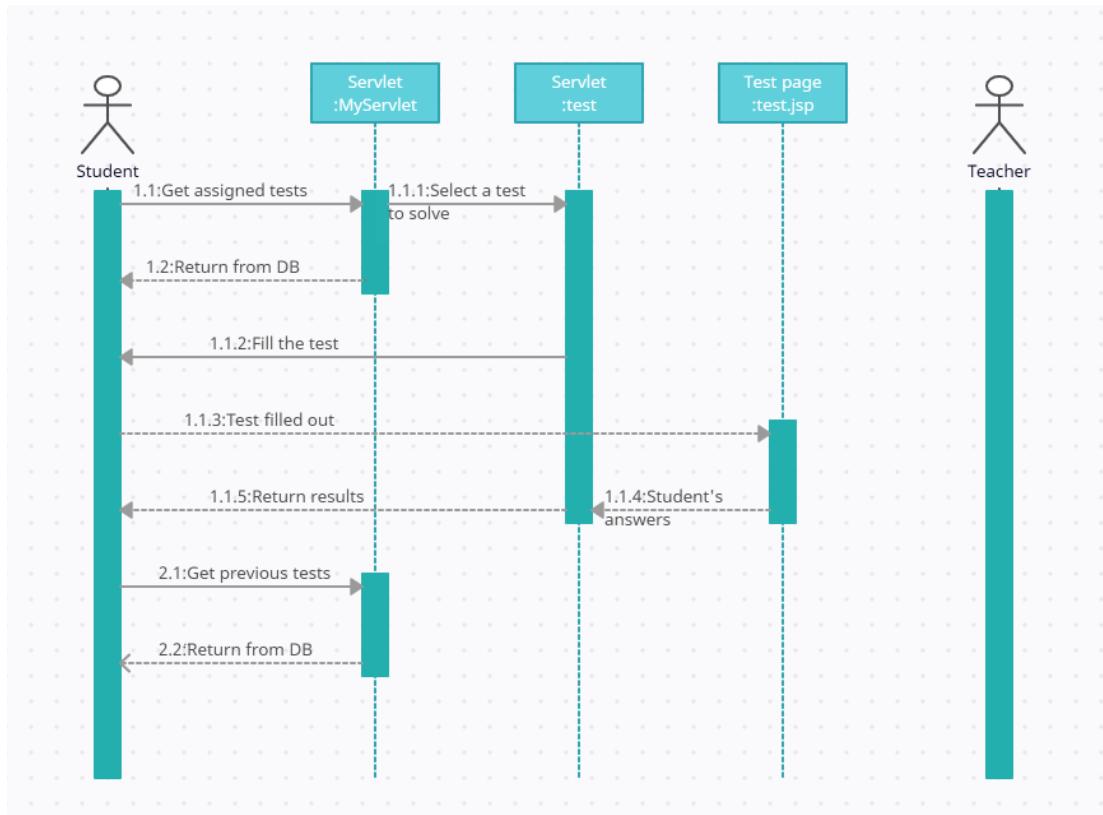
Login Student:



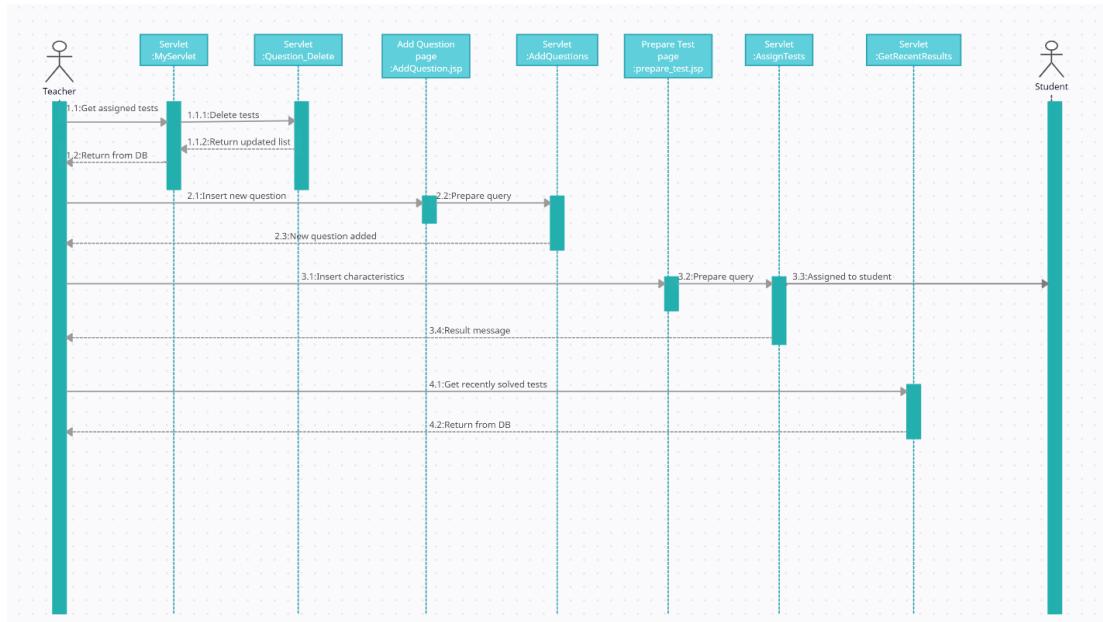
Login Teacher:



## Student-to-Teacher:

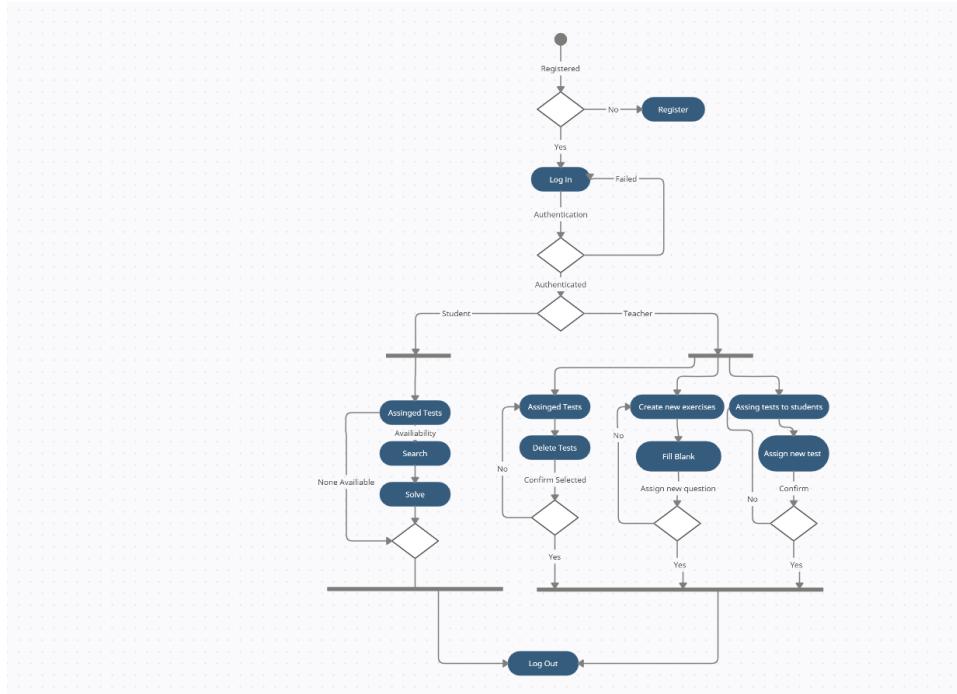


## Teacher-to-Student:



### 5.1.6 Διάγραμμα Δραστηριοτήτων – Activity Diagram (1η έκδοση)

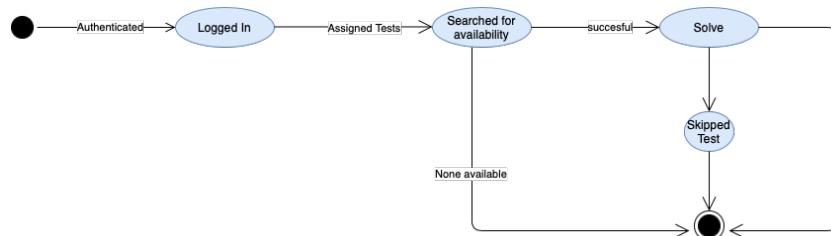
Παρακάτω φαίνεται η 2η και τελική έκδοση του διαγράμματος δραστηριοτήτων.



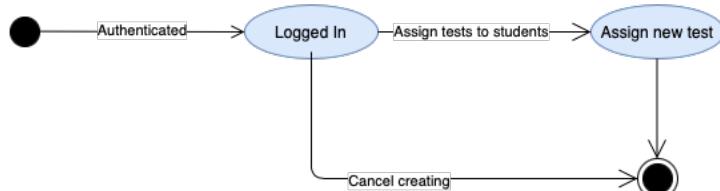
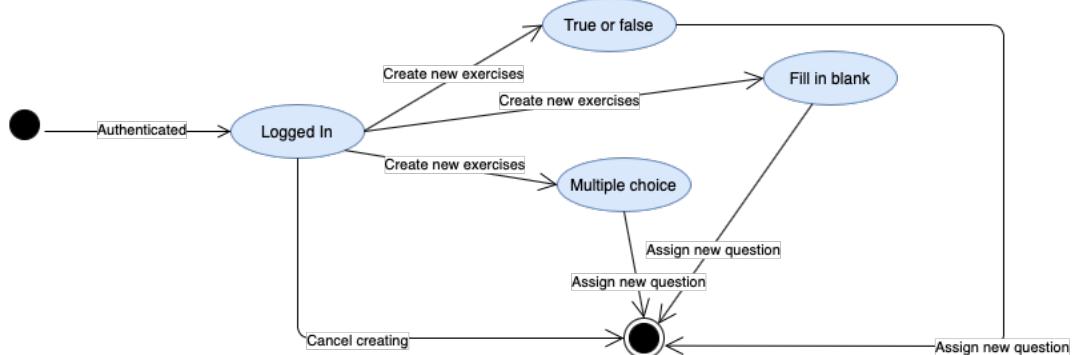
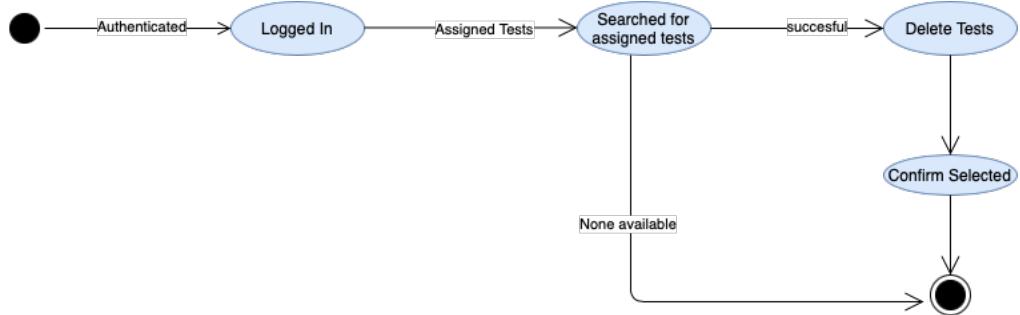
### 5.1.7 Διάγραμμα Καταστάσεων – State Chart Diagram (2η έκδοση)

Παρακάτω φαίνεται η δεύτερη και τελική έκδοση του διαγράμματος καταστάσεων για τον μαθητή και τον καθηγητή:

Μαθητής:

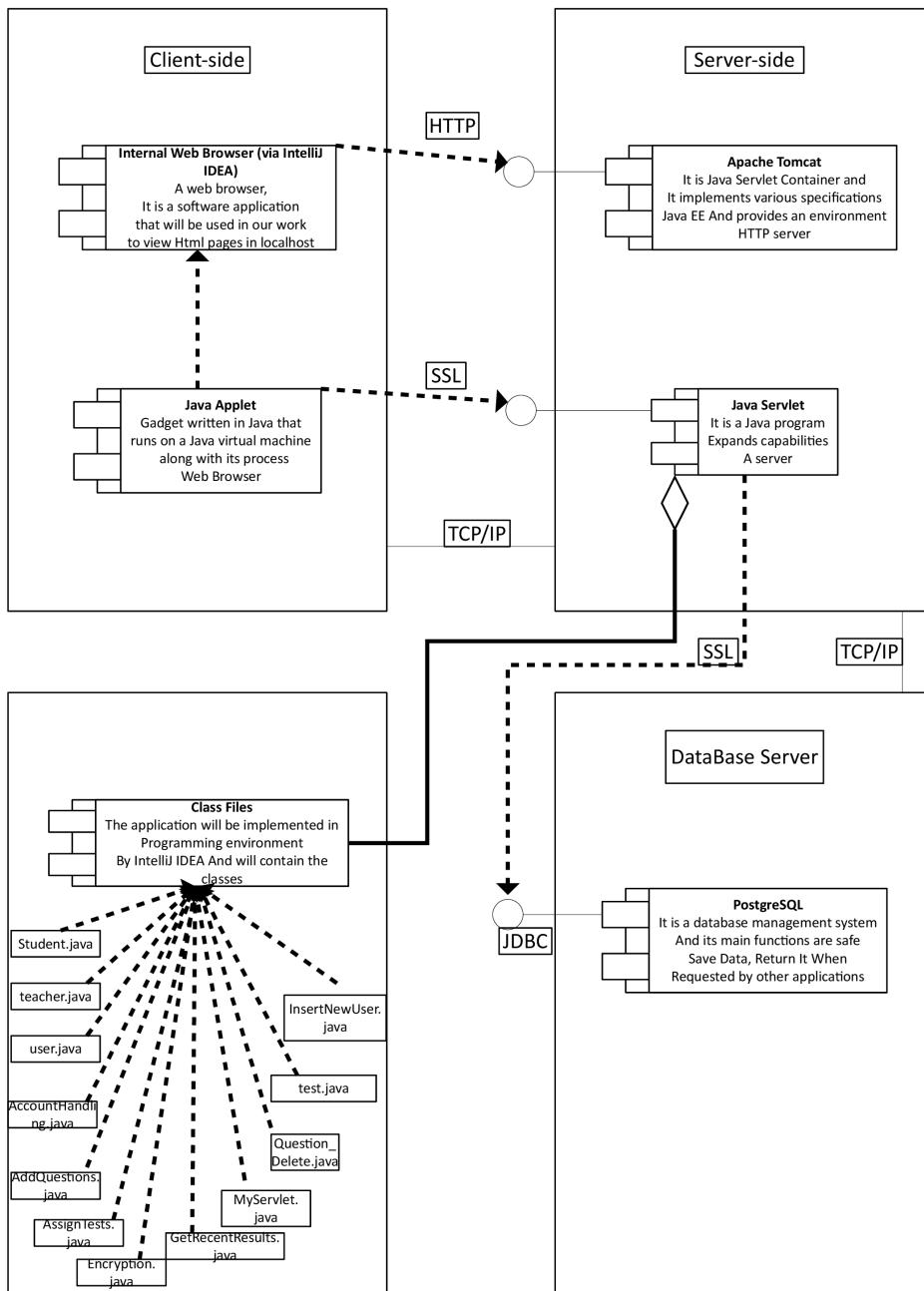


Καθηγητής:



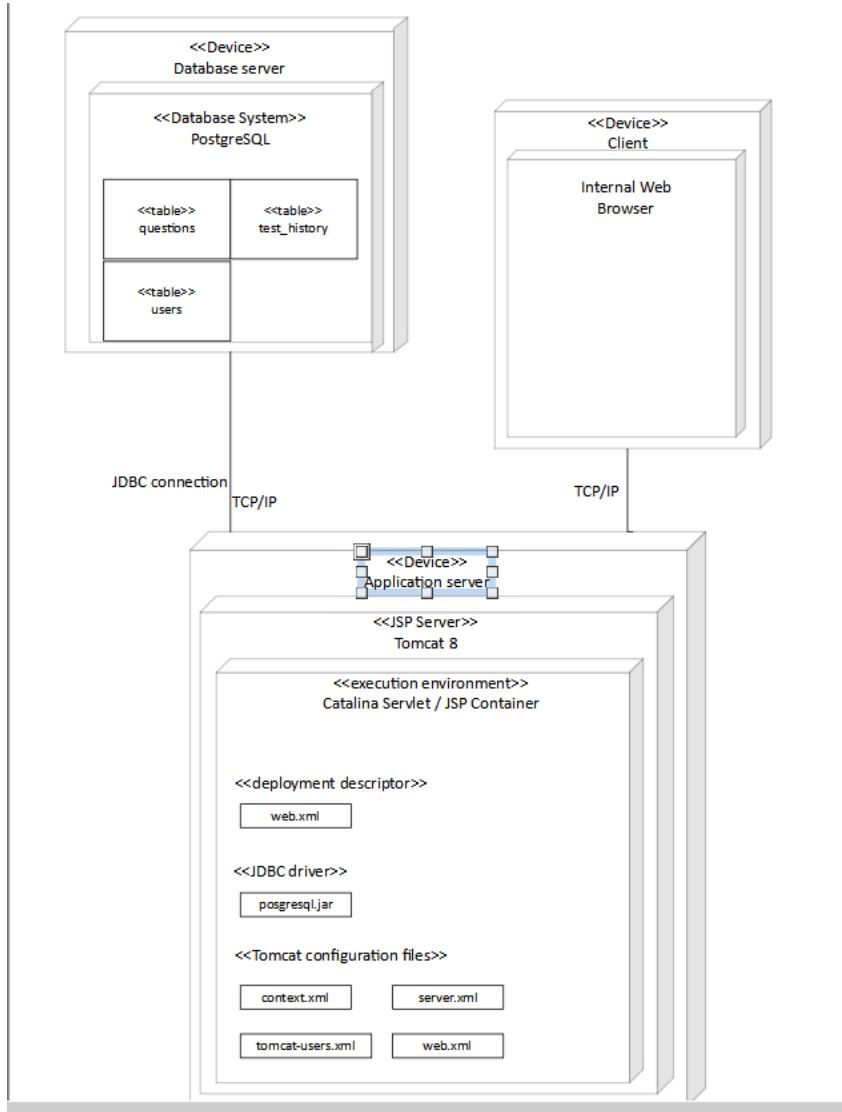
### 5.1.8 Λιάγραμμα Εξαρτημάτων –Component Diagram (2η έκδοση)

Παρακάτω φαίνεται η δεύτερη και τελική έκδοση του διαγράμματος εξαρτημάτων:



### 5.1.9 Διάγραμμα Διανομής- Deployment Diagram (2η έκδοση)

Το διάγραμμα διανομής δεν έχει αλλάξει σε σχέση με την πρώτη έκδοση αφού τα τμήματα της φυσικής διάταξης του λογισμικού παραμένουν τα ίδια:



## 5.2 Υλοποίηση- Έλεγχος

### 5.2.1 Υλοποίηση: Τελική εκτελέσιμη έκδοση

Με γνώμονα τις τελικές εκδόσεις των διαγραμμάτων, πραγματοποιήθηκε η τελική εκτελέσιμη έκδοση της εφαρμογής, λαμβάνοντας υπόψιν και την πρώτη εκτελέσιμη έκδοση.

Αλλαγές στην Βάση Δεδομένων δεν πραγματοποιήθηκαν και έτσι η δομή της παρέμεινε ίδια με αυτήν που παρουσιάστηκε στην πρώτη εκτελέσιμη έκδοση.

Στην υλοποίηση της διαδικτυακής διεπαφής προστέθηκαν οι παρακάτω καινούργιες σελίδες JSP:

- previous.jsp: Σελίδα για την προβολή του ιστορικού των ολοκληρωμένων διαγωνισμάτων από τον μαθητή.
- recent\_solved.jsp: Η σελίδα για την προβολή του ιστορικού των αναθετημένων διαγωνισμάτων από τον καθηγητή.
- createuser.jsp: Σελίδα για την δημιουργία νέου χρήστη (μαθητή/καθηγητή).

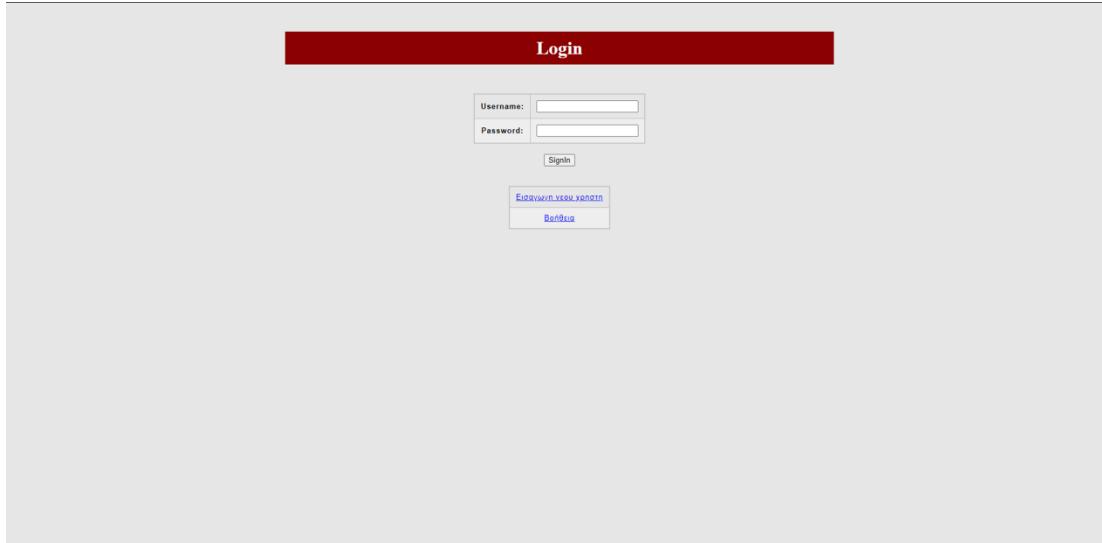
### 5.2.2 Αναφορά για την τελική εκτελέσιμη έκδοση

Μέσα από την διαδικασία της RUP, δημιουργήθηκε το τελικό αποτέλεσμα της web εφαρμογής καταχώρησης ραντεβού ασθενών-ιατρών. Όλες οι εκκρεμότητες που αναλύθηκαν παραπάνω, διευθετήθηκαν. Το πρόγραμμα είναι έτοιμο προς χρήση. Παρέχει όλα τα βασικά ζητούμενα που αναφέρθηκαν κατά την φάση της έναρξης, καθώς και επιπλέον λειτουργίες οι οποίες προέκυψαν κατά την φάση της ανάλυσης απαιτήσεων.

## **6. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ**

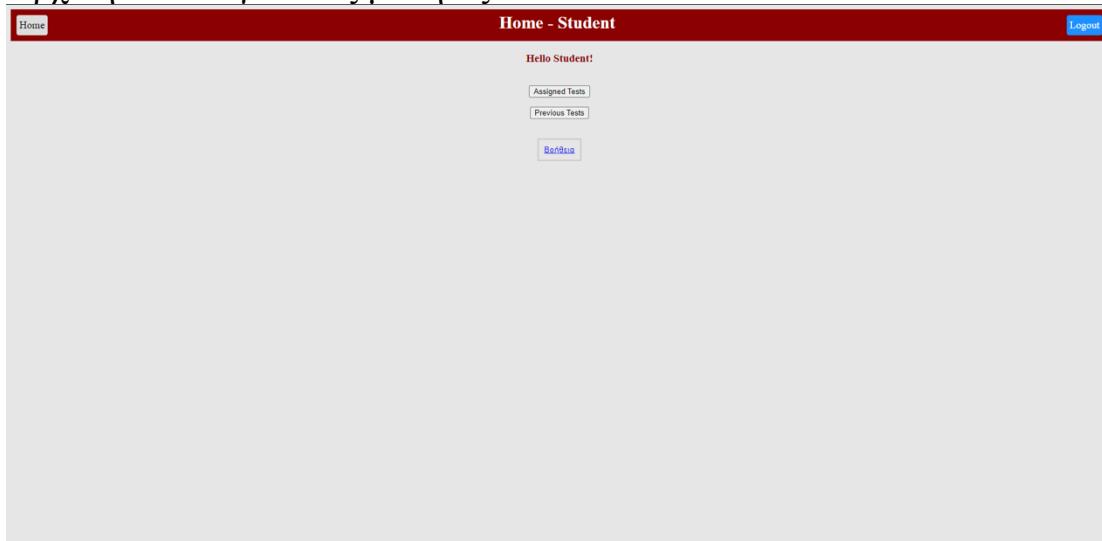
### **6.1 Σύντομη παρουσίαση του προγράμματος**

Αρχική σελίδα για το σύνολο των χρηστών:



The screenshot shows a login interface with a dark red header bar containing the word "Login". Below the header is a form with two input fields: "Username:" and "Password:", each with its own "Signin" button below it. At the bottom of the form are two links: "Επαναφέρειν όντας χρήστη" and "Βοήθεια".

Αρχική σελίδα για τους μαθητές:



The screenshot shows the "Home - Student" page. It features a dark red header bar with "Home", "Hello Student!", and "Logout" buttons. The main content area displays three buttons: "Assigned Tests", "Previous Tests", and "Βοήθεια".

## Σελίδα για την επιλογή διαγωνίσματος από τον μαθητή:

Home      Select Test      Logout

Please select one of the tests your teacher has assigned to you, in order to solve it!

Teacher	Questions	Difficulty	Date Assigned	Select
Babas	6	1	2021-07-13 10:49:37 251	<input type="radio"/>
Babas	6	1	2021-07-13 10:48:14 553	<input type="radio"/>
Babas	6	1	2021-07-13 10:47:55 706	<input type="radio"/>
akii	5	1	2021-07-13 10:41:50 445	<input type="radio"/>
Alepis	7	3	2021-07-08 14:59:36 967	<input type="radio"/>

## Σελίδα διαγωνίσματος:

Home      Test      Logout

Please fill out the test before submitting it!

1. Madrid is the capital of Spain  
 True  False  
The right answer was: true

2. Madrid is the capital of Sweden  
 True  False  
The right answer was: false

3. Athens is the capital of Greece  
 True  False  
The right answer was: true

4. The earth is flat  
 True  False  
The right answer was: false

5. Rome is the capital of Spain  
 True  False  
The right answer was: false

6. Paris is the capital of France  
 True  False  
The right answer was: true

## Σελίδα αποτελεσμάτων:

Home      Results      Logout

Mrekid, here is your final results and your grade:

1. Madrid is the capital of Spain  
Your answer was: **true**  
The right answer was: **true**

2. Madrid is the capital of Sweden  
Your answer was: **false**  
The right answer was: **true**

3. Athens is the capital of Greece  
Your answer was: **true**  
The right answer was: **true**

4. The earth is flat  
Your answer was: **false**  
The right answer was: **true**

5. Rome is the capital of Spain  
Your answer was: **false**  
The right answer was: **false**

6. Paris is the capital of France  
Your answer was: **true**  
The right answer was: **true**

Your final grade is : 6.67 / 10

## Σελίδα προηγουμένων διαγωνισμάτων:

Home      Previous Tests      Logout

Here you can see the previous tests you have filled out!

Teacher	Questions	Difficulty	Date Solved	Grade
Babas	6	1	2021-09-07 15:49:53.351	6.67
kappatolis	2	1	2021-07-20 20:37:21.473	10.0
Alepis	8	2	2021-07-11 17:40:25.607	7.5

## Αρχική σελίδα για τον καθηγητή:

Home      Home - Teacher      Logout

Hello Teacher!

Assigned Tests  
Create new exercises  
Assign tests to students  
Recently solved tests

## Σελίδα αναθετημένων διαγωνισμάτων στους μαθητές:

Home      Assigned Tests      Logout

Here you can see the tests you have assigned to your students!

Teacher	Student's username	Questions	Difficulty	Date Assigned/Solved	Grade	DELETE
kappatolis	AxillesSmall	3	1	2021-07-21 14:23:04.036	10.0	<input type="radio"/>
kappatolis	Mrckid	2	1	2021-07-20 20:37:21.473	10.0	<input type="radio"/>

## Σελίδα δημιουργίας ερωτήσεων:

Home      kappatolis, assign new questions      Logout

Question difficulty: 1 ▾

Multiple choice  Fill the blank  True or false

Question: Athens is the \_\_\_\_\_ of greece. Answer: capital

Question: Answer:

## Σελίδα ορισμού προϋποθέσεων για δημιουργία του διαγωνίσματος:

Home      Preparing Test Assignment      Logout

Please select the required features in order to assign a test to your students.

Question difficulty: 1 ▾

Type of question: Multiple choice  Fill the blank  True or false

Amount of exercises:

Age:

## Σελίδα ανάθεσης διαγωνίσματος σε μαθητή:

Home      Assign Tests      Logout

Here you can assign tests to students!

Username	Name	Surname	Age	SELECT
AxilleasSmall	Axilleas1	Peridis1	20	<input type="checkbox"/>
Mrcrid	etaggelos	fittros	20	<input type="checkbox"/>
Daddy	Marios	Xyainos	20	<input type="checkbox"/>

Σελίδα πρόσφατων επιλυμένων διαγωνισμάτων από μαθητή:

The screenshot shows a web interface titled "Recently Solved Tests". At the top, there are navigation links for "Home" and "Logout". Below the title, a message reads: "Here you can see the tests your students have filled out in the past 24hrs!". A table header row includes columns for "Username", "Questions", "Difficulty", "Date Solved", and "Grade". A note below the table states: "NO TEST HAS BEEN RECENTLY SUBMITTED".

Σελίδα δημιουργίας νέου χρήστη (μαθητή/καθηγητή):

The screenshot shows a registration form with a red header bar containing the text "Create an account". The form fields include: Username, Password, Firstname, Lastname, Age (with a dropdown menu), Type of account (with a dropdown menu set to "Student"), and Βιογραφικός κωδικός. At the bottom right of the form area is a "Create User!" button.

## 6.2 Ανάλυση σεναρίων λειτουργείας

Αρχικα, εμφανίζεται η αρχική σελίδα, στην οποία υπάρχει η φόρμα για σύνδεση στην εφαρμογή (login), καθώς και η επιλογή για την εγγραφή ενός νέου χρήστη στην εφαρμογή. Στη συνέχεια, ανάλογα με τον τύπο του χρήστη εμφανίζονται και οι αντίστοιχες επιλογές. Στον τύπο χρήστη καθηγητή πχ, εμφανίζονται οι επιλογές για δημιουργία ερωτήσεων και καταχώρηση σε μαθητή, ενώ αντίστοιχα στον μαθητή εμφανίζονται οι επιλογές απάντησης ερωτήσεων και εμφάνιση βαθμολογιών .