



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ**

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΙΤΛΟΣ: *Ευφυείς και προσαρμοστικοί παιδαγωγικοί
πράκτορες*

ΜΠΟΜΠΑ ΕΣΤΙΝΑ

**ΕΠΙΒΛΕΠΟΥΣΑ ΚΑΘΗΓΗΤΡΙΑ:
Δασκαλοπούλου Ασπασία**

Βόλος 2012

Περιεχόμενα

Κεφάλαιο 1.Εισαγωγή.....	5
1.1 Ιστορική Ανασκόπηση.....	5
1.1.1 Πρώτα Συστήματα Υπολογιστών.....	5
1.1.2 Η εμφάνιση των ITS.....	6
1.2 Αντικείμενο της διπλωματικής.....	6
1.3 Οργάνωση κειμένου.....	7
Κεφάλαιο 2.Ευφύες Σύστημα Διδασκαλίας.....	8
2.1 Ορισμός ITS.....	8
2.2 Η δομή ενός συστήματος ITS.....	8
2.3 Γενική Αρχιτεκτονική ITS.....	9
2.3.1 Η μονάδα εξειδικευμένης γνώσης.....	10
2.3.2 Η ενότητα μοντέλο μαθητή.....	11
2.3.3 Η ενότητα διδασκαλία.....	13
2.3.4 Η μονάδα user interface.....	14
2.4 Μια βασιζόμενη στην ηθοποιία αρχιτεκτονική για τα ITS.....	15
2.4.1 Ευφύης πράκτορες.....	16
2.4.2 Ένα νέο πρότυπο πράκτορα : ο ηθοποιός.....	18
2.4.2.1 Ιδιότητες.....	18
2.4.2.2 Καθολική Αρχιτεκτονική ενός ηθοποιού.....	18
2.4.3 Αρχιτεκτονική πολλαπλών-στρατηγικών για ένα ITS.....	21
2.5 Ένα βασιζόμενο σε πράκτορα Ευφύης Σύστημα Διδασκαλίας για εξ αποστάσεως εκπαίδευση – ABITS.....	25
2.5.1 Εισαγωγή.....	25
2.5.2 Ευρετηρίαση της γνώσης.....	27
2.5.2.1 Μεταδεδομένα.....	27
2.5.2.2 Εννοιολογικά διαγράμματα.....	28
2.5.3 Μοντελοποίηση Μαθητή.....	30
2.5.3.1 Γνωστική κατάσταση.....	31
2.5.3.2 Προτιμήσεις μάθησης.....	31
2.5.4 Αυτόματη παραγωγή διδακτέας ύλης.....	33
2.5.5 Αρχιτεκτονική ABITS.....	34

Κεφάλαιο 3.Σχεδιασμός και Υλοποίηση της Εφαρμογής μας.....	37
3.1 Ανάγκες και στόχους που καλύπτει η εφαρμογή	37
3.2 Τεχνολογίες.....	37
3.3 Εισαγωγή στην εφαρμογή.....	41
3.3.1 Αρχική σελίδα.....	42
3.3.2 Υλη του μαθήματος.....	43
3.3.3 Κεφάλαιο 1.....	43
3.4 Δομή και περιεχόμενα της εφαρμογής.....	46
3.4.1 Βάση δεδομένων.....	48
3.4.2 Αρχεία-files της εφαρμογής μας.....	52
3.4.3 Ροή της εφαρμογής.....	55
3.4.4 Αρχιτεκτονική της εφαρμογής.....	58
3.5 Ειδικά σημεία.....	58
Κεφάλαιο 4.Συμπεράσματα και μελλοντική δουλειά	59
4.1 Συμπεράσματα.....	59
4.2 Μελλοντική δουλειά.....	59
References.....	60

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά την επιβλέπουσα καθηγήτρια μου, κ. Δασκαλοπούλου Ασπασία για την καθοδήγηση και την βοήθεια της κατά την εκπόνηση της διπλωματικής μου εργασίας .

Ευχαριστώ από καρδιάς την οικογένεια μου για την συμπαράσταση και την οικονομική στήριξη όλα αυτά τα χρόνια.

Επιπλέον ένα μεγάλο ευχαριστώ στους φίλους και τους συμφοιτητές για τις ωραίες στιγμές που περάσαμε μαζί.

Κεφάλαιο 1. Εισαγωγή

Στις μέρες μας η επιστήμη των υπολογιστών αποτελεί πλέον αναγκαίο κομμάτι σε κάθε τομέα της ζωής μας. Ένας από αυτούς τους τομείς είναι και η εκπαίδευση. Παρακάτω παρουσιάζουμε τους λόγους για τους οποίους οι υπολογιστές άρχισαν να χρησιμοποιούνται στην εκπαίδευση.

1.1 Ιστορική Ανασκόπηση

Το 1984, ο Benjamin Bloom παρουσιάζει το «two-sigma problem», το οποίο ορίζει ότι οι φοιτητές που λαμβάνουν one-on-one διδασκαλία παρουσιάζουν καλύτερη κατανόηση - απόδοση από τους μαθητές που λαμβάνουν τη παραδοσιακή διδασκαλία. Καθώς είναι αδύνατο για ένα ίδρυμα να παρέχει έναν εκπαιδευτικό για κάθε μαθητή, ξεκίνησε σθεναρά η χρήση των ηλεκτρονικών υπολογιστών, ως υποκατάστατο των δασκάλων. Παρακινήμένοι από την παραπάνω αιτία, πολλές ερευνητικές ομάδες άρχισαν να εργάζονται στον τομέα αυτό και ανέπτυξαν συστήματα υπολογιστών με διάφορα χαρακτηριστικά.

1.1.1 Πρώτα Συστήματα Υπολογιστών

Η πρώτη γενιά εργαλείων εκπαίδευσης που χρησιμοποιούσαν υπολογιστές ονομάστηκαν , Συστήματα Διδασκαλίας βασισμένοι στους Υπολογιστές (Computer-Aided Instruction CAI).

Ένα από τα πρώτα παραδείγματα ενός τέτοιου συστήματος είναι το σύστημα από τον Uhr το 1969, το οποίο απλά δείχνει προβλήματα αριθμητικής και ερωτήσεις λεξιλογίου στον χρήστη. Το κύριο μειονέκτημα αυτού του συστήματος ήταν ότι δεν είχε μοντέλο χρήστη ή τεχνικές προσαρμογής.

Αργότερα εμφανίστηκαν ορισμένα σύγχρονα συστήματα όπως το Suppes (1967) και το σύστημα από το Woods και Hartley (1971) που θα μπορούσαν να χαρακτηριστούν προσαρμοστικά, επειδή εδώ τα προβλήματα ήταν ανάλογα με τις αποδόσεις του χρήστη. Ωστόσο το μοντέλο χρήστη που χρησιμοποιούσαν ήταν αρκετά πρωτόγονο. Το μοντέλο ήταν μόνο μια παραμετρική περίληψη , δεν αποθήκευε την πραγματική κατάσταση της γνώσης του χρήστη. Τα συστήματα αυτά μπορούν να χαρακτηριστούν ως προάγγελος για τα Ευφυής Συστήματα Διδασκαλίας (Intelligent Tutoring Systems ITS).

Εν τω μεταξύ, ένα άλλο είδος συστήματος διδασκαλίας ανακαλύφθηκε . Αυτός ο τύπος συστημάτων ονομάστηκε Εξάσκηση και Εξέταση (Drill and Test). Εδώ τα προβλήματα παρουσιάζονταν στους μαθητές με μορφή τεστ και στο τέλος της διαδικασίας παρέχονταν τα αποτελέσματα του τεστ. Μια απλή παραλλαγή αυτού

του συστήματος ήταν το Προσαρμοστική Εξάσκηση και Εξέταση (Adaptive Drill and Test). Σε αυτή την έκδοση δεν είχαμε μόνο παρουσίαση προβλημάτων, αλλά αποθήκευση της απόδοσης και ανταπόκρισης του μαθητή σε μορφή πινάκων, τα οποία στη συνέχεια χρησιμοποιούταν για την επιλογή των προβλημάτων. Έτσι σε αυτό το σημείο, θεωρήθηκε ότι η γνώση του μαθητή έπρεπε να είναι ένας σημαντικός παράγοντας για την υλοποίηση των συστημάτων.

Το κεντρικό πρόβλημα με τα παραπάνω συστήματα είναι ότι δεν ήταν σε θέση να παρέχουν πλούσια πληροφορία για τον χρήστη (feedback) ή εξατομίκευση εκπαίδευση (individualization), επειδή δεν είχαν σχεδιαστεί για να ξέρουν τι δίδασκαν, ποιόν δίδασκαν ή πώς να διδάξουν. Για να λυθεί αυτό το πρόβλημα, τα CAI συστήματα έχουν εξελιχθεί κατά τα τελευταία τριετήμια δεκαετίες σε αυτό που είναι τώρα συνήθως ονομάζεται «Ευφυή Συστήματα Διδασκαλίας» (ITS).

1.1.2 Η εμφάνιση των ITS (Intelligent Tutoring System)

Το 1982, ο Sleeman και Brown επινόησαν τον όρο Ευφυή Συστήματα Διδασκαλίας (ITS) για να περιγράψουν τα εξελισσόμενα συστήματα διδασκαλίας και να τα διακρίνουν από τα προηγούμενα συστήματα CAI. Τα συστήματα αυτά ορίζονται ως Ευφυής επειδή περιέχουν λειτουργίες της τεχνητής νοημοσύνης (Artificial Intelligence AI) που κάνουν τα συστήματα αυτά «έξυπνα». Με τις νέες AI λειτουργίες φάνηκε ότι οι υπολογιστές ήταν σχεδόν σε θέση να "Σκέπτονται" όπως τους ανθρώπους, με αποτέλεσμα να είναι πιο εύκολη η επίτευξη των στόχων υλοποίησης παιδαγωγικών συστημάτων. Άλλοι λόγοι που προκάλεσαν την χρήση της AI στα ITS ήταν οι εξής:

- Δομημένο και κομψό πρόγραμμα σπουδών .
- Προσαρμοστικότητα για διαφορετικούς τύπους μαθητών.
- Ατομική παρουσίαση και αξιολόγηση του περιεχομένου.
- Συλλογή στοιχείων τα οποία εκπαιδευτές θα μπορούσαν να χρησιμοποιήσουν για να διδάσκουν και αποκαταστήσουν τα κενά των φοιτητών.

1.2 Αντικείμενο της Διπλωματικής

Όπως αναφέρθηκε πιο πάνω λόγω της ταχύτατης εξέλιξης της επιστήμης των υπολογιστών στην εκπαίδευση υπάρχει ραγδαία αλλαγή στον τρόπο μάθησης. Ένας από τους συγχρόνους αυτούς τρόπους είναι και τα Web-Based Intelligent Tutoring Systems πάνω στην οποία θα στηριχτεί η ανάπτυξη της Διπλωματικής αυτής εργασίας. Το αντικείμενο το οποίο υλοποιείται είναι τα Μαθηματικά Α Δημοτικού.

1.3 Οργάνωση Κειμένου

Η εργασία έχει οργανωθεί ως εξής : Στο 1^ο κεφάλαιο κάνουμε γενική αναφορά των λόγων που οι υπολογιστές άρχισαν να χρησιμοποιούνται στην εκπαίδευση. Στο 2^ο κεφάλαιο ορίζουμε τι είναι τα Ευφυής Συστήματα Διδασκαλίας (ITS). Πιο συγκεκριμένα αναλύουμε μια γενική αρχιτεκτονική ITS και ύστερα παρουσιάζουμε δύο γνωστά συστήματα ITS, με απώτερο σκοπό να κατανοήσουμε πως λειτουργεί ένα Ευφυής Σύστημα Διδασκαλίας. Στο 3^ο κεφάλαιο δίνουμε μία εκτενέστερη αναφορά για την σχεδίαση και υλοποίηση του δικού μας Συστήματος Διδασκαλίας. Τέλος στο 4^ο κεφάλαιο παρουσιάζουμε τα συμπεράσματα και τις μελλοντική δουλεία που μπορεί να γίνει στην εφαρμογή μας.

Κεφάλαιο 2

Ευφυές σύστημα διδασκαλίας -Intelligent tutoring system (ITS)

2.1 Ορισμός ITS

Ένα **ευφυές σύστημα διδασκαλίας (ITS)** είναι κάθε σύστημα υπολογιστή που παρέχει άμεση εξατομικευμένη διδασκαλία στους μαθητές, χωρίς την παρέμβαση του ανθρώπου. Έτσι, τα ITS θέτουν σε εφαρμογή τη θεωρία μάθησης μέσω της πράξης και χρησιμοποιούν μια σειρά από διαφορετικές τεχνολογίες. Ωστόσο, συνήθως τα συστήματα αυτά θεωρούνται συστήματα τεχνητής νοημοσύνης, πιο συγκεκριμένα έμπειρα συστήματα που προσομοιώνουν τις πτυχές ενός δασκάλου. Τα Ευφυή Συστήματα Διδασκαλίας είναι γνωστά από τα τέλη της δεκαετίας του 1970, αλλά αυξήθηκαν σε δημοτικότητα στη δεκαετία του 1990.

2.2 Η δομή ενός συστήματος ITS

Τα Ευφυή συστήματα διδασκαλίας αποτελούνται από τέσσερα διαφορετικά υποσυστήματα ή ενότητες : την ενότητα διεπαφή (interface module), την ενότητα *εμπειρογνώμων* (expert module), την ενότητα των μαθητών (student module) και την ενότητα εκπαιδευτή (tutor module).

Η *ενότητα διεπαφή* παρέχει τα μέσα στον μαθητή να αλληλεπιδράσει με τα ITS, συνήθως μέσω μιας γραφικής διεπαφής χρήστη (graphical user interface) και μερικές φορές μέσω μιας πλούσιας εξομίωσης του τομέα εργασίας του μαθητή. Η *ενότητα εμπειρογνώμων* αναφέρεται σε ένα έμπειρο σύστημα ή γνωστικό μοντέλο που περιέχει περιγραφή των γνώσεων ή συμπεριφορών που αντιπροσωπεύουν τεχνογνωσία στον αντικείμενο που το ITS διδάσκει. Η *ενότητα μαθητή* χρησιμοποιεί ένα *μοντέλο μαθητή* που περιέχει περιγραφές των γνώσεων ή συμπεριφορών του μαθητή, συμπεριλαμβανομένων των παρανοήσεων του και τα κενά γνώσης. Η αναντιστοιχία μεταξύ της συμπεριφοράς ή των γνώσεων του μαθητή και την υποτιθέμενη συμπεριφορά ή γνώση του εμπειρογνώμονα σηματοδοτείται με την *ενότητα εκπαιδευτή*, το οποίο στη συνέχεια παίρνει διορθωτικά μέτρα, όπως αποκατάσταση διδασκαλίας. Για να πραγματοποιηθεί αυτό, χρειάζονται πληροφορίες σχετικά με το τι ένας δάσκαλος θα έκανε σε τέτοιες καταστάσεις: το μοντέλο δασκάλου .

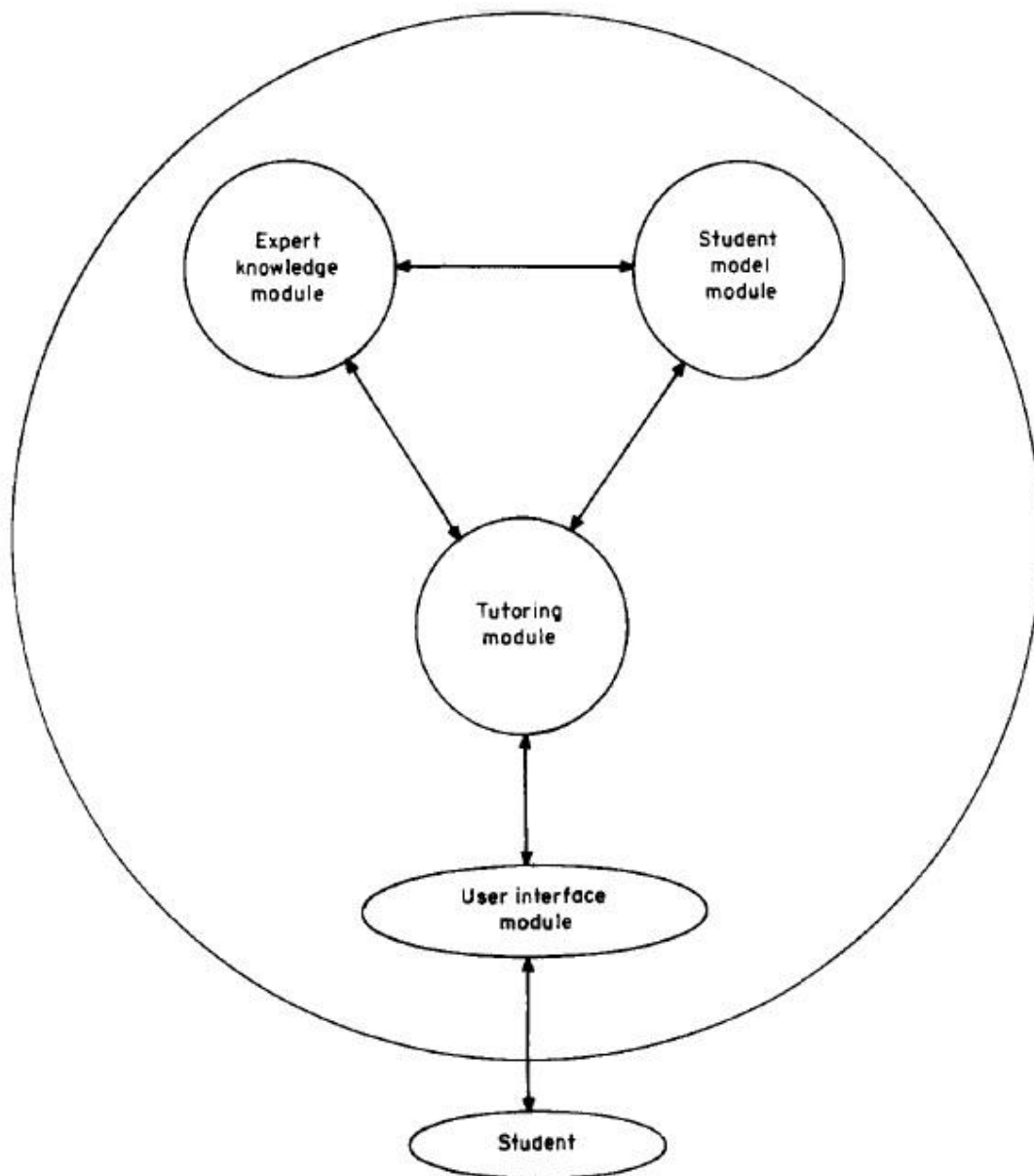
Ένα έξυπνο σύστημα διδασκαλία είναι τόσο αποτελεσματικό όσο τα διάφορα μοντέλα βασίζονται επαρκώς στη γνώση και τη συμπεριφορά των εμπειρογνώμων, των φοιτητών και του καθηγητή. Έτσι, η οικοδόμηση μιας ITS χρειάζεται προσεκτική προετοιμασία όσον αφορά την περιγραφή των γνώσεων και των πιθανών συμπεριφορών των εμπειρογνώμων, των φοιτητές και του καθηγητή. Η περιγραφή αυτή πρέπει να γίνει σε μια *επίσημη γλώσσα* ώστε η ITS να μπορεί να επεξεργάζεται τις πληροφορίες και να εξάγει συμπεράσματα, πληροφορίες για τον μαθητή ή τη διδασκαλία (student or tutorial feedback) . Κατά συνέπεια, μια απλή περιγραφή δεν είναι αρκετή. Η γνώση που περιέχεται στα μοντέλα θα πρέπει να οργανωθεί και να συνδέεται με μια *μηχανή εξαγωγής συμπερασμάτων*.

Όλα τα υποσυστήματα και τα μοντέλα που προαναφέραμε θα πρέπει να είναι οργανωμένες με κάποια αρχιτεκτονική. Παρακάτω θα μελετήσουμε τη γενική αρχιτεκτονική.

2.3 Γενική Αρχιτεκτονική ITS

Τα υπάρχοντα Ευφυή Συστήματα Διδασκαλίας διαφέρουν παρά πολύ στην αρχιτεκτονική. Στην πραγματικότητα, είναι σχεδόν σπάνιο να βρεθούν δύο ITS που βασίζονται στην ίδια αρχιτεκτονική. Ωστόσο, στη βιβλιογραφία αναφέρετε ότι τα Ευφυή Συστήματα Διδασκαλίας αποτελούνται από τουλάχιστον τέσσερα βασικά συστατικά :

- Η ενότητα εξειδικευμένης γνώσης (expert knowledge module).
- Η ενότητα μοντέλο μαθητή (student model module).
- Η ενότητα διδασκαλίας (tutoring module).
- Η ενότητα διεπαφή χρήστη (user interface module).



Εικόνα 1: Γενική Αρχιτεκτονική Ενός Συστήματος ITS

2.3.1 Η μονάδα εξειδικευμένης γνώσης ή μονάδα εμπειρογνώμονα

Η μονάδα εξειδικευμένης γνώσης ή μονάδα εμπειρογνώμονα, επιτελεί ένα διπλό ρόλο. Πρώτον, χρησιμεύει ως πηγή της γνώσης η οποία θα παρουσιαστεί στο μαθητή και περιλαμβάνει συστήματα παραγωγής θεμάτων, επεξηγήσεων και απαντήσεων. Δεύτερον, παρέχει ένα πρότυπο για την αξιολόγηση των επιδόσεων των φοιτητών. Γι 'αυτό, πρέπει να είναι σε θέση να παράγει λύσεις στα προβλήματα, έτσι ώστε οι λύσεις αυτές να συγκριθούν με αυτά του μαθητή . Η μονάδα πρέπει επίσης να είναι σε θέση να ανιχνεύει κοινά συστηματικά λάθη και αν είναι δυνατό να εντοπίσει τυχόν κενά στις γνώσεις του μαθητή που μπορεί να είναι η αιτία αυτού του γεγονότος. Επίσης, η μονάδα εξειδικευμένης γνώσης

μπορεί να χρησιμοποιηθεί για να αξιολογήσει τη συνολική πρόοδο του μαθητή. Για να επιτευχθεί αυτό απαιτείται η δημιουργία κάποιων κριτηρίων για την αξιολόγηση της γνώσης. Η αξιολόγηση είναι δυνατή μόνο αν η γνώση παρουσιάζεται ρητά-σαφώς. Ως εκ τούτου, τα Ευφυή Συστήματα Διδασκαλίας διαφέρουν σημαντικά από παραδοσιακά προγράμματα CAI (computer – assisted instruction) στο ότι η γνώση του τελευταίου σιωπηρά εκπροσωπείται μέσα στον κώδικα του. Τέλος αξίζει να σημειωθεί ότι η μονάδα εξειδικευμένη γνώσης κατά ανάγκη ενσαρκώνει μια συγκεκριμένη άποψη του τομέα, αυτό του σχεδιαστή. Έτσι, η διδασκαλία μπορεί να τεθεί σε κίνδυνο, αν ο μαθητής δεν καταλαβαίνει τις οδηγίες του συστήματος ή επειδή το σύστημα δεν μπορεί να ερμηνεύσει τη συμπεριφορά του μαθητή λαμβάνοντας υπόψη τη δική του όψη της γνώσης (Wenger, 1987). Φυσικά, το ανθρώπινο εκπαιδευτικό έχει τις δικές του απόψεις, αλλά έχει επίσης την απίστευτη ικανότητα να τα προσαρμόσει αναλόγως με την αντίληψη του κάθε φοιτητή. Το θέμα αυτό αποτελεί το κεντρικό πρόβλημα της αναπαράσταση γνώσης σε AI (Artificial Intelligence) και η λύση του είναι ιδιαίτερα σημαντική για τα Ευφυή Συστήματα Διδασκαλίας, λόγω των σημαντικών παιδαγωγικών επιπτώσεων του.

2.3.2 Η ενότητα μοντέλο μαθητή

Η ενότητα μοντέλο μαθητή αναφέρεται στη δυναμική παρουσίαση των αναδυομένων γνώσεων και δεξιοτήτων του μαθητή. Δεν μπορεί να υπάρξει ευφυής διδασκαλία χωρίς την κατανόηση του μαθητή. Έτσι, προέκυψε η ανάγκη να παρουσιάζονται ρητά-σαφώς οι γνώσεις –δεξιότητες του μαθητή, με τη μορφή ενός μοντέλου μαθητή. Στην ιδανική περίπτωση, αυτό το μοντέλο πρέπει να περιλαμβάνει όλες εκείνες τις πτυχές της συμπεριφοράς και των γνώσεων του μαθητή που έχουν πιθανές επιπτώσεις στην επίδοση και μάθηση του. Ωστόσο, το έργο της δημιουργίας ενός πλήρους μοντέλου είναι όχι μόνο μη-τετριμμένο, αλλά κατά πάσα πιθανότητα αδύνατο, θεωρώντας ότι το κανάλι επικοινωνίας, το οποίο είναι συνήθως το πληκτρολόγιο, είναι τόσο περιοριστικό. Αντίθετα οι καθηγητές μπορούν να χρησιμοποιήσουν δεδομένα από μια ποικιλία πηγών, όπως φωνή ή εκφράσεις προσώπου αλλά και άλλους φαινομενολογικούς παράγοντες, όπως η ανία ή το κίνητρο που είναι ζωτικής σημασίας για την μάθηση. Οι λειτουργίες του μοντέλου μαθητή θα μπορούσαν γενικά να ταξινομηθούν σε έξι τύπους.

- (1) Διορθωτική: να εξαλείψει σφάλματα στις γνώσεις του μαθητή.
- (2) Επεξεργάσιμη: να διορθώσει ελλείψεις γνώσεις του μαθητή.
- (3) Στρατηγική: να αρχικοποιήσει σημαντικές αλλαγές στη στρατηγική εκμάθησης.
- (4) Διαγνωστική: να διαγνώσει σφάλματα στις γνώσεις του μαθητή.
- (5) Προβλεπτική: να προσδιορίσει πιθανές απαντήσεις του μαθητή στις ενέργειες

του μαθήματος .

(6) Αξιολογική: να αξιολογήσει τον μαθητή ή το ITS.

Το μοντέλο του μαθητή θα μπορούσε να θεωρηθεί ότι πληροί διπλή λειτουργία. Από τη μία πλευρά, λειτουργεί ως πηγή πληροφοριών σχετικά με τον φοιτητή. Από την άλλη πλευρά, χρησιμεύει ως μια αναπαράσταση της γνώσης του μαθητή, Wenger (1987). Οι επόμενοι παράγραφοι θα αναλύσουν τη παραπάνω άποψη. Με τη λειτουργία του ως πηγή πληροφοριών, το σύστημα συμπεραίνει μη προφανής πτυχές της συμπεριφοράς του μαθητή από το μοντέλο. Ένα τέτοιο συμπέρασμα θα μπορούσε να παράγει μια ερμηνεία των ενεργειών του μαθητή και επίσης να οδηγήσει σε μια ανακατασκευή της γνώσης που οδήγησε σε αυτές τις ενέργειες. Αυτή η γνώση είναι ζωτικής σημασίας για την παιδαγωγική συνιστώσα της ITS. Το μοντέλο μαθητή είναι επίσης πιθανό να εμπεριέχεται μέσα στην ενότητα εξειδικευμένης γνώσης. Κατά συνέπεια, το μοντέλο του μαθητή μπορεί να περιλαμβάνει μια σαφή αξιολόγηση της κάθε μονάδας της γνώσης στην μονάδα εμπειρογνομόνων (η λειτουργία του μοντέλου μαθητή εδώ είναι αξιολογική). Αυτό επιτρέπει η κατάσταση γνώσης του μαθητή να συγκριθεί με την ενότητα εξειδικευμένης γνώσης, και η εκπαίδευση στη συνέχεια θα κλίνει προς τα τμήματα του μοντέλου που έχουν αποδειχθεί ότι είναι αδύνατα (ως εκ τούτου, η λειτουργία του μοντέλου μαθητή εδώ είναι επεξεργάσιμη). Αυτή η μορφή δείχνει ότι η κατάσταση γνώσης του μαθητή θεωρείται ένα υποσύνολο του εμπειρογνώμονα, το οποίο αποδεικνύει πως το μοντέλο μαθητή λειτουργεί ως πηγή πληροφόρησης. Ωστόσο, εσφαλμένες ή μη ιδανικές συμπεριφορές δεν προκαλούνται πάντα από την ελλιπή γνώση. Θα μπορούσε επίσης να οφείλεται σε εσφαλμένες εκδοχές της γνώσης. Ως εκ τούτου ένα πιο διαμορφωτικό μοντέλο μαθητή πρέπει επίσης να παρέχει σαφείς αναπαραστάσεις εσφαλμένων εκδοχών της γνώσης του μαθητή με σκοπό την αποκατάσταση (προφανώς, η λειτουργία του μοντέλου μαθητή εδώ είναι διαγνωστική και διορθωτική). Σε αυτή την ιδιότητα, το μοντέλο του μαθητή χρησιμεύει ως μια αναπαράσταση της γνώσης του μαθητή. Το μοντέλο μαθητή, χρειάζεται επίσης να είναι εκτελέσιμο. Αυτό επιτρέπει την ακριβής πρόβλεψη σχετικά με ένα συγκεκριμένο μαθητή σε ένα συγκεκριμένο πλαίσιο (κατά συνέπεια, η λειτουργία του μοντέλου μαθητή εδώ είναι προβλέψιμη). Επίσης η μονάδα διδασκαλία μπορεί να χρησιμοποιήσει τις εν λόγω εκτελέσιμες παραστάσεις για παιδαγωγικούς σκοπούς (π.χ. η γνώση που παρουσιάζεται στο μοντέλο του μαθητή χρησιμοποιείται με στρατηγικό τρόπο). Το μοντέλο μαθητή υπηρετεί και εδώ ως αναπαράσταση της γνώσης του μαθητή.

Εν κατακλείδι, το μοντέλο του φοιτητή θα μπορούσε να θεωρηθεί για την εκτέλεση δύο «σούπερ» λειτουργιών: ενεργώντας ως πηγή πληροφοριών σχετικά με το μαθητή και λειτουργώντας ως μια αναπαράσταση της γνώσης του μαθητή. Για την επίτευξη αυτών των λειτουργιών, το μοντέλο δρα σε διάφορους ρόλους, όπως διορθωτική, επεξεργάσιμη, στρατηγική, διάγνωση, αξιολόγηση και πρόβλεψη.

2.3.3 Η ενότητα διδασκαλία

Η ενότητα διδασκαλία είναι το μέρος του ITS που σχεδιάζει και ρυθμίζει τις εκπαιδευτικές αλληλεπιδράσεις με το μαθητή. Σε άλλες αρχιτεκτονικές, αυτή η ενότητα αναφέρεται ως τη στρατηγική της διδασκαλίας ή την παιδαγωγική μονάδα . Συνδέεται στενά με το μοντέλο του φοιτητή και χρησιμοποιώντας τη γνώση που κατέχει ο μαθητής και τους δικούς του εκπαιδευτικούς στόχους , αποφασίζει ποιές παιδαγωγικές δραστηριότητες θα παρουσιαστούν. Για παράδειγμα παρουσιάζει συμβουλές για να ξεπεραστούν αδιέξοδα στην επίδοση του μαθητή . Επίσης προσφέρει υποστήριξη, επεξήγηση ,διάφορες πρακτικές εργασίες, καθώς και δοκιμές για την επιβεβαίωση υποθέσεων που υπάρχουν στο μοντέλο του μαθητή . Έτσι, η ενότητα διδασκαλία είναι η πηγή και ο ενορχηστρωτής όλων των παιδαγωγικών ενοτήτων που μεσολαβούν .Η σειρά και ο τρόπος με τον οποίο αντιμετωπίζονται τα θέματα μπορεί να παράγουν πολύ διαφορετικές εμπειρίες μάθησης.

Στη διδασκαλία, μερικές φορές είναι πιο αποτελεσματικό να αφήσουμε τον μαθητή να δράσει ανεπιτυχώς για λίγο, πριν τον διακόψουμε : μερικές φορές, ο φοιτητής θα κολλήσει ή θα χαθεί εάν αφεθεί να δράσει εντελώς αυτόβουλα (όμως δεν είναι καλό να "καταστρέψουμε" τα προσωπικά κίνητρα του μαθητή ή την αίσθηση της ανακάλυψης). Ως εκ τούτοις η διδασκαλία στα υπάρχοντα ITS μπορεί να ταξινομηθεί σε δύο κατηγορίες . Στην πρώτη που αποτελείται από συστήματα που παρακολουθούν το μαθητή από πολύ κοντά σε κάθε δραστηριότητα, προσαρμόζοντας τις ενέργειες τους σύμφωνα με τις απαντήσεις των μαθητών, αλλά ποτέ δεν παραδίδουν τον έλεγχο στον μαθητή και στη δεύτερη που αποτελείται από συστήματα μάθησης καθοδηγούμενες από ανακάλυψη (guided-discovery learning systems), όπου ο φοιτητής έχει σχεδόν τον πλήρη έλεγχο της δραστηριότητας και ο μόνος τρόπος που το σύστημα μπορεί να κατευθύνει την πορεία της διδασκαλίας είναι με την τροποποίηση του περιβάλλοντος . Στη μέση αυτών των δύο είναι τα μικτά συστήματα- πρωτοβουλίας (mixed-initiative systems) στα οποία ο έλεγχος γίνεται από κοινού από το φοιτητή και το σύστημα, καθώς ανταλλάσσουν ερωτήσεις και απαντήσεις. Η ύπαρξη των εν λόγω κατηγοριών αναδεικνύει με σαφήνεια το γεγονός ότι η μάθηση είναι μια τέχνη που απαιτεί μεγάλη ευελιξία , η οποία εξακολουθεί να είναι εξαιρετικά δύσκολη να πραγματοποιηθεί στα συστήματα ITS. Παρ 'όλα αυτά , υπάρχει κάποια πρόοδος, αν και λίγο περιορισμένη.

Τα Ευφυή Συστήματα Διδασκαλίας επίσης αποσκοπούν στην σαφή παρουσίαση της γνώσης που βρίσκεται στην μονάδα διδασκαλίας. Για άλλη μια φορά αυτό έρχεται σε αντίθεση με τα παραδοσιακά συστήματα διδασκαλίας (computer – assisted instruction CAI) όπου η παιδαγωγική γνώση είναι βαθιά ενσωματωμένη

στα διάφορα κομμάτια του κώδικα που ελέγχουν την αλληλεπίδραση της διδασκαλίας.

2.3.4 Η μονάδα user interface

Η μονάδα user interface είναι το συστατικό επικοινωνίας του ITS που ελέγχει την αλληλεπίδραση μεταξύ του φοιτητή και του συστήματος, όπως απεικονίζεται στο εικόνα 1. Το user interface και προς τις δύο κατευθύνσεις, μεταφράζει μεταξύ της εσωτερικής αναπαράστασης του συστήματος και της γλώσσα του περιβάλλοντος εργασίας που είναι κατανοητή στον μαθητή.

Το user interface, ανεξάρτητα από το πόσο «ευφυής» είναι το εσωτερικό σύστημα, χαρακτηρίζεται συνήθως ως μία διακριτή συνιστώσα από μόνη της ,για δύο λόγους . Πρώτον, όταν το ITS παρουσιάζει ένα θέμα, το user interface μπορεί να ενισχύσει ή να μειώσει τη παρουσίαση. Δεδομένου ότι το interface είναι η τελική μορφή μέσω της οποίας το ITS “παρουσιάζεται”, ιδιότητες όπως η ευκολία χρήσης και η ελκυστικότητα μπορεί να είναι ζωτικής σημασίας για το μαθητή ώστε να κατανοήσει το σύστημα. Δεύτερον, η πρόοδος στην τεχνολογία των μέσων ενημέρωσης εξελίσσεται ραγδαία παρέχοντας περισσότερα και πιο εξελιγμένα εργαλείων των οποίων η επικοινωνιακή δύναμη επηρεάζει σε μεγάλο βαθμό το σχεδιασμό του ITS.

Τα υπάρχοντα Ευφυή Συστήματα Διδασκαλίας παρέχουν διεπαφές χρηστών τα οποία ,για μία συγκεκριμένη είσοδο, κυμαίνονται από σταθερό μενού με πολλαπλής- επιλογής απαντήσεις μέχρι μια αρκετά ελεύθερη μεταχείριση μιας ψευδο-γλώσσας (pseudonatural language). Για μία συγκεκριμένη έξοδο, κυμαίνονται από την απλή απεικόνιση των προ-αποθηκευμένων κειμένων ,που χαρακτηρίζει τα CAI, μέχρι τη χρήση αρκετά περίπλοκων γενικών πλαισίων. Μερικά ITS κάνουν την αλληλεπίδρασή τους με το μαθητή πιο «φιλική προς το χρήστη» χρησιμοποιώντας εικόνες και γραφικά , τα οποία είναι πιο κατανοητά και εύκολα στην χρήση. Παρακάτω θα δούμε μερικά είδη user interface τα οποία είναι τα πιο ευρέως χρησιμοποιημένα.

Τύποι user interface :

- *Γραφικές διεπαφές χρήστη (Graphical user interfaces _GUI)* που δέχονται είσοδο μέσω συσκευών, όπως πληκτρολόγιο και το ποντίκι του υπολογιστή και παρέχουν γραφική έξοδο στην οθόνη του υπολογιστή. Υπάρχουν τουλάχιστον δύο διαφορετικές αρχές που χρησιμοποιούνται ευρέως στο σχεδιασμό GUI: Αντικειμενοστραφής διεπαφή χρήστη (Object-oriented user interfaces OOUIs) και application oriented interfaces.
- *Web-based διεπαφές χρήστη (Web-based user interfaces WUI)* είναι μια υποκατηγορία των GUIs που δέχονται είσοδο και παρέχουν έξοδο

δημιουργώντας ιστοσελίδες, οι οποίες μεταδίδονται μέσω του Διαδικτύου και είναι ορατά στον χρήστη χρησιμοποιώντας ένα πρόγραμμα περιήγησης στο Web (web browser program: Mozilla ,Explorer κτλ). Νεότερες υλοποιήσεις χρησιμοποιούν Java, AJAX, Adobe Flex, το Microsoft .NET, ή παρόμοιες τεχνολογίες για να παρέχουν σε πραγματικό χρόνο έλεγχο σε ένα ξεχωριστό πρόγραμμα, εξαλείφοντας την ανάγκη να ανανεώνετε μια παραδοσιακή HTML που βασίζεται web browser. Administrative web interfaces για web-servers, servers and networked computers συχνά ονομάζονται πίνακες ελέγχου.

- *Διεπαφή χρήστη αφής* (Touch user interface) είναι γραφικές διεπαφές χρήστη που χρησιμοποιούν ένα touchpad ή οθόνη αφής ως μια συνδυασμένη input και output συσκευή. Συμπληρώνει ή αναπληρώνει την εξόδο με μεθόδους απτικής ανάδρασης (haptic feedback methods). Χρησιμοποιείται ολοένα και περισσότερο σε κινητές συσκευές ,βιομηχανικές διεργασίες και μηχανήματα, self-service μηχανήματα, κλπ.
- *Διεπαφές χρήστη Κειμένου* (Text user interfaces) είναι οι διεπαφές χρήστη που έχουν σαν έξοδο κείμενο, αλλά δέχονται σαν είσοδο και άλλες μορφές εκτός των δακτυλογραφημένων σειρών εντολών.
- *Διεπαφές χρήστη Voice* (Voice user interfaces), που δέχονται είσοδο και παρέχουν έξοδο δημιουργώντας φωνητικά μηνύματα. Η παρέμβαση του χρήστη γίνεται με το πάτημα πλήκτρων ή κουμπιών ή με το να ανταποκρίνεται προφορικά στη διασύνδεση.
- *Φυσικής γλώσσας διεπαφές* (Natural-Language interfaces). Χρησιμοποιείται για τις μηχανές αναζήτησης και στις ιστοσελίδες. Ο χρήστης δακτυλογραφεί σε μια ερώτηση και περιμένει μια απάντηση.

Η γενική αρχιτεκτονική ,που μόλις αναλύσαμε, δε εφαρμόζεται στην πράξη σε κάποιο σύστημα ITS. Για αυτό θα δούμε παρακάτω δύο παραδείγματα αρχιτεκτονικών ITS, που χρησιμοποιούνται στην πράξη και είναι από τα πιο αντιπροσωπευτικά παιδαγωγικά συστήματα ITS. Πιο συγκεκριμένα θα αναλύσουμε στην ενότητα 2.4 μια αρχιτεκτονική βασιζόμενη στην ηθοποιία και στην ενότητα 2.5 μια αρχιτεκτονική βασιζόμενη σε πράκτορα. Με αυτόν τον τρόπο θα κατανοήσουμε καλύτερα τον τρόπο λειτουργίας ενός ITS.

2.4 Μία βασιζόμενη στην ηθοποιία αρχιτεκτονική για τα ITS - An Actor-Based architecture for Intelligent Tutoring Systems

Η μάθηση στα ευφυή συστήματα διδασκαλίας (ITS) έχει εξελιχθεί κατά τη διάρκεια των τελευταίων δύο δεκαετιών. Κύριος στόχος ήταν να αναπαράγουν την συμπεριφορά ενός ευφυούς δασκάλου, ο οποίος μπορεί να προσαρμόσει τη διδασκαλία του στο ρυθμό μάθησης του μαθητή, δηλαδή να υπάρχει ευελιξία. Αρχικά, ο έλεγχος της εκπαίδευσης γινόταν από τον δάσκαλο, όχι τον μαθητή. Αργότερα όμως τα ITS άρχισαν να χρησιμοποιούν την προσέγγιση συνεργασίας μεταξύ του μαθητή και του συστήματος. Το σύστημα συνεργάζεται με τον μαθητή στη διαδικασία της μάθησης και διευκολύνει την απόκτηση γνώσεων από τις αλληλεπιδράσεις υπό τον έλεγχο του μαθητή.

Για να υπάρχει ευελιξία στη μάθηση, τα ITS πρέπει να βασίζονται σε πολλαπλές στρατηγικές. Για να υπάρχουν πολλαπλές στρατηγικές, πρέπει να καθοριστεί μια πολλαπλών-πρακτόρων αρχιτεκτονική (multi-agent architecture) το οποίο χρησιμοποιεί έξυπνους πράκτορες (intelligent agents) με συγκεκριμένα χαρακτηριστικά. Οι πράκτορες αυτοί θα «παίζουν» ένα ρόλο σε συγκεκριμένες συνθήκες, σε συνεργασία με άλλους πράκτορες. Παρακάτω θα καθορίσουμε τα στοιχεία μιας τέτοιας αρχιτεκτονικής και θα δείξουμε πώς λειτουργεί. Πρώτα θα ορίσουμε τι είναι ένας πράκτορας, ύστερα θα ορίσουμε τον ηθοποιό πράκτορα (actor agent) και την αρχιτεκτονική του και τέλος θα αναλύσουμε την multi-strategic, multi-agent αρχιτεκτονική.

2.4.1 Ευφυής πράκτορες

Υπάρχουν πολλοί ορισμοί για το τι είναι ένας πράκτορας και ο λόγος είναι ότι ο πράκτορας θα πρέπει να αντιπροσωπεύει όλα τα χαρακτηριστικά ενός ανθρώπου και συνεπώς πρέπει να έχει πολλές ιδιότητες –ορισμούς. Ας εξετάσουμε ορισμένες από αυτές τις ιδιότητες.

Οι πράκτορες έχουν γενικά *αυτονομία*. Μπορούν να λειτουργήσουν χωρίς τον ανθρώπινο έλεγχο και αλληλεπιδρούν με άλλους πράκτορες (που έχουν κοινωνική ικανότητα) χρησιμοποιώντας μια γλώσσα agent-communication. Δείχνουν αντίδραση όταν υπάρχουν αλλαγές στο περιβάλλον και ενεργούν σύμφωνα με μία κατευθυνόμενη από στόχους συμπεριφορά (goal-oriented behavior). Το τελευταίο σημείο συγκλίνει προς ένα σύνολο ιδιοτήτων, όπως επιθυμίες, πεποιθήσεις και προθέσεις, το οποίο είναι ίδιο με εκείνων των ανθρώπων.

Αν λάβουμε υπόψη την εξέλιξη των ευφυών πρακτόρων μπορούμε να διακρίνουμε μια πρώτη κατηγορία στην οποία πράκτορες ορίζονται με τις βασικές ικανότητες, όπως η *αντιδραστικότητα*, *σχεδιασμός*, *πρόβλεψη* και *διάγνωση*.

Αντιδραστικοί πράκτορες έχουν μόνο άμεση αντίδραση σε ερεθίσματα, χωρίς αιτιολογία. Σχεδιαστικοί πράκτορες χρησιμοποιούνται στην εξεύρεση μιας σειράς ενεργειών που θα πετύχουν τον επιθυμητό στόχο. Οι προβλεπτικοί πράκτορες χρησιμοποιούνται στον σχεδιασμό με υποθέσεις . Πράκτορες με δυνατότητες διάγνωσης προορίζονται για την αναζήτηση ενός συμπεράσματος μέσα από ένα σύνολο υποθέσεων . Ωστόσο, αυτές οι ικανότητες δεν επαρκούν για να αναλάβουν οι πράκτορες καθήκοντα διδασκαλίας. Χρειαζόμαστε μία αιτιολογική και δυναμική βελτίωση των γνώσεων του πράκτορα.

Η δεύτερη κατηγορία ονομάζεται *instructable πράκτορες* , οι οποίοι μπορούν να λαμβάνουν δυναμικά οδηγίες ή νέους αλγορίθμους και μπορούν να μάθουν ή να βελτιώσουν τη μάθησή τους σύμφωνα με την ιστορία των ενεργειών τους. Instructable πράκτορες μπορούν να λάβουν τις οδηγίες για την εκτέλεση των νέων καθηκόντων, άμεσα από τους χρήστες ή από το σύστημα .

Η τρίτη κατηγορία ονομάζεται *προσαρμοστικοί πράκτορες* (adaptive agents). Οι πράκτορες αυτοί μπορούν να προσαρμόσουν την αντίληψή τους για τις καταστάσεις και να τροποποιήσουν τις αποφάσεις τους, επιλέγοντας νέες μεθόδους συλλογισμού. Ειδικότερα, προσαρμόζουν τη στρατηγική τους στις δυναμικές απαιτήσεις πληροφόρησης, προσαρμόζουν τον έλεγχό τους σε δυναμικούς , βασιζομένους σε στόχους (goal-oriented), περιορισμούς και τέλος προσαρμόζουν τις μεθόδους συλλογισμού με τα ήδη υπάρχον κριτήρια πληροφοριών και επιδόσεων . Αυτές οι ιδιότητες είναι πολύ σημαντικές για εκπαιδευτικούς σκοπούς, αλλά εξακολουθούν να είναι ανεπαρκή για τις απαιτήσεις του ITS.

Η τέταρτη κατηγορία περιλαμβάνει πράκτορες που μοντελοποιούν την ανθρώπινη συμπεριφορά σε συνθήκες μάθησης. Αυτή η κατηγορία είναι οι *γνωστικοί πράκτορες* (cognitive agents) και είναι η πιο κατάλληλη για τα προβλήματα των ITS. Η γνωστική πτυχή ενός πράκτορα βασίζεται στην ικανότητά του να μαθαίνει και να ανακαλύπτει νέα στοιχεία ή να βελτιώνει τις γνώσεις του για καλύτερη χρήση. Μάθησης μπορεί να επιτευχθεί με ποικιλία μεθόδων.

Λαμβάνοντας υπόψη τα στοιχεία που πρέπει να έχει ένα ITS, χρειαζόμαστε μια αρχιτεκτονική με πολλαπλούς γνωστικούς πράκτορες οι οποίοι διαμορφώνουν - μοντελοποιούν όχι μόνο το μαθητή αλλά και διαφορετικές συμπεριφορές που αντιστοιχούν σε διάφορες παιδαγωγικές καταστάσεις που μπορεί να συμβούν με έναν δάσκαλο, συμμαθητή, κτλ. Αυτή η εκτεταμένη έννοια του Intelligent Agent θεωρεί τον πράκτορα ως έναν ηθοποιό που είναι σε θέση να παίξει διαφορετικούς ρόλους ανάλογα με τις καταστάσεις μάθησης (κατάσταση της γνώσης που αποκτήθηκε από το μαθητή, τις συνθήκες μάθησης, επιθυμητή στρατηγική, ...) σε ένα συνεργατικό περιβάλλον. Έτσι καταλήγουμε σε ένα νέο πρότυπο πράκτορα ,τον ηθοποιό (actor) πράκτορα .

2.4.2 Ένα νέο πρότυπο πράκτορα : ο ηθοποιός

2.4.2.1 Ιδιότητες

Οι ιδιότητες που χρειαζόμαστε σε ένα ITS περιβάλλον οδηγούν στο εξής ορισμό του ηθοποιού : Ένας ηθοποιός είναι ένας έξυπνος πράκτορας ο οποίος είναι αντιδραστικός (βασική ιδιότητα), προσαρμοστικός, instructable και γνωστικός.

- Αντιδραστική ιδιότητα. Το πρώτο επίπεδο των δράσεων που χρειάζεται το ITS απαιτεί αντιδραστικές δυνατότητες όπως: προσδιορισμό της αλληλουχίας γνώσεων που θα παρουσιαστούν , τη διάγνωση των παρανοήσεων ή των ελλείπων αντιλήψεων και αντιστοιχία κατάστασης του μαθητή με τις τυπικές καταστάσεις σύμφωνα με μία case-based προσέγγιση.
- Προσαρμοστική ιδιότητα. Η προσαρμογή της αντίληψης και του ελέγχου είναι απαραίτητη όταν, για παράδειγμα, μια νέα στρατηγική μάθησης πρέπει να ενεργοποιηθεί για να αντιμετωπιστεί μια νέα κατάσταση του μαθητή.
- Γνωστική ιδιότητα. Για να βελτιωθεί η συμπεριφορά του ITS το σύστημα πρέπει να μάθει από την εμπειρία ή την απόκτηση νέων στρατηγικών.
- Instructable ιδιότητα. Τέλος, το σύστημα πρέπει να είναι σε θέση να δημιουργήσει νέες διεργασίες έλεγχου ή να προσαρμόσει τη στρατηγική συλλογισμού , λαμβάνοντας υπόψη μία βαθύτατη ανάλυση της συμπεριφοράς των μαθητών.

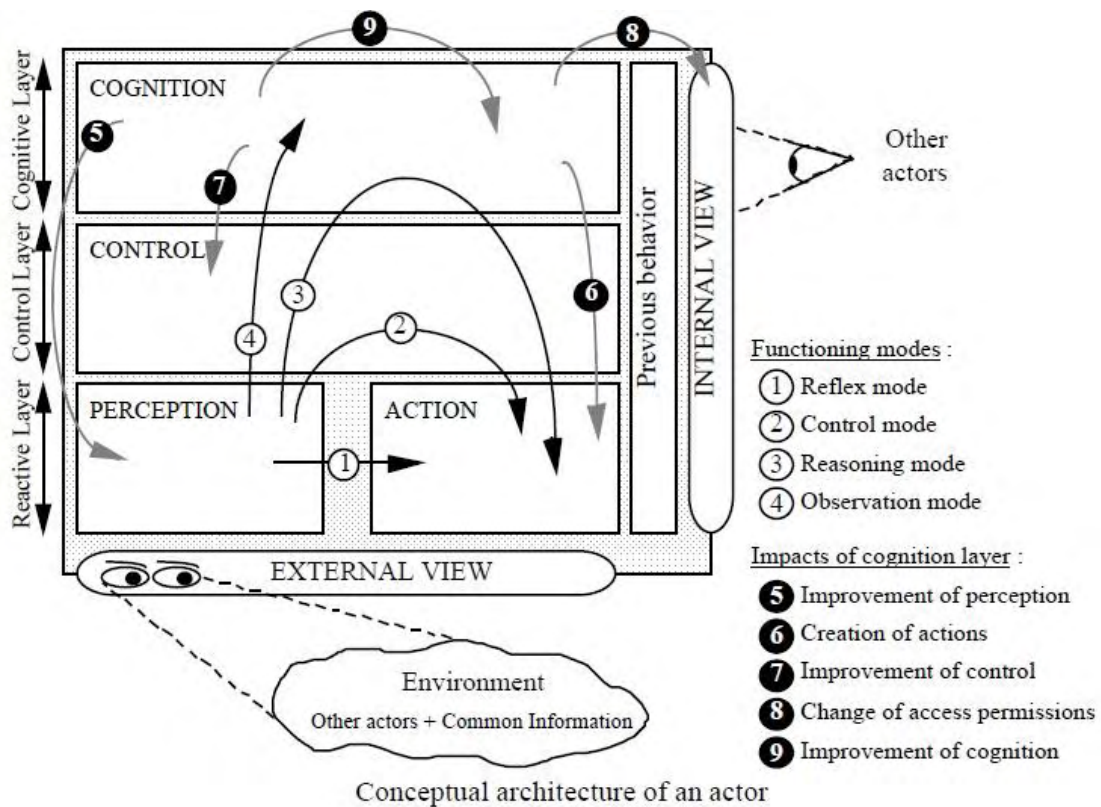
Η χρήση πολλαπλών στρατηγικών επιτρέπει την ενεργή συμμετοχή του μαθητή με έναν συμμαθητή , έναν δάσκαλο, έναν troublemaker , ... Με τον ίδιο τρόπο, ένας actor πρέπει να καλύπτει μια συμπεριφορά βασιζόμενη σε πολλαπλές στρατηγικές (multistrategy-based behavior) .Υπό αυτή την έννοια, ένας ηθοποιός πρέπει να έχει την δυνατότητα να συνεργαστεί σε πολλαπλές στρατηγικές μάθησης και να μάθει από αυτή τη συνεργασία αντί να στοχεύει σε πολλαπλούς στόχους για να ικανοποιήσει την πρόθεση του κάθε πράκτορα. Έτσι βελτιώνουν τη συμπεριφορά τους σε συνεργασία με άλλους πράκτορες .

2.4.2.2 Καθολική αρχιτεκτονική ενός ηθοποιού

Η αρχιτεκτονική (εικόνα 2) περιλαμβάνει τέσσερις ενότητες Αντίληψη, Δράση, Έλεγχο και Γνώση (perception, action, control and cognition module) και διανέμεται σε τρία επίπεδα: Αντιδραστική, Ελέγχου και Γνωστική (reactive, control and cognitive layer). Η αρχιτεκτονική μας περιέχει ένα αντιδραστικό επίπεδο, με άμεση συσχέτιση μεταξύ των εννοιών της αντίληψης και της δράσης και ένα επίπεδο ελέγχου που αντιστοιχεί στη ενότητα του σχεδιασμού. Το τρίτο επίπεδο (γνωστικό

επίπεδο) υποστηρίζει τη δυνατότητα οι πράκτορες να μάθουν από την εμπειρία ,το οποίο αποτελεί σημαντικό στοιχείο μίας ευφυούς οντότητα .

Ένα αρχικό σημείο αυτής της αρχιτεκτονικής είναι ότι κάθε actor έχει τη δυνατότητα να παρατηρεί την προηγούμενη συμπεριφορά των άλλων actors , τα βήματα των ενεργειών τους (όχι μόνο τα αποτελέσματα της δράσης τους). Για να επιτραπεί αυτή η δυνατότητα κάθε ηθοποιός έχει μια εξωτερική άποψη (external view) σχετικά με τους άλλους ηθοποιούς και μια εσωτερική άποψη (internal view) επιτρέποντας άλλους ηθοποιούς να συμβουλευούνται τη συμπεριφορά του. Επιπλέον, ένας ηθοποιός μπορεί να αποφασίσει να κρύψει μέρος της συμπεριφοράς του προς τους άλλους ηθοποιούς. Ο ρόλος της κάθε ενότητας είναι η ακόλουθη.



Εικόνα 2: Θεμελιώδη Αρχιτεκτονική Ενός Ηθοποιού

Αντίληψη(perception)

Ανιχνεύει την αλλαγή του περιβάλλοντος του ηθοποιού και προσδιορίζει τις περιπτώσεις στις οποίες ο ηθοποιός μπορεί να παρέμβει. Το περιβάλλον ενός ηθοποιού αποτελείται και από τους άλλους ηθοποιούς και τοποθετείται σε μια κοινή μνήμη. Οι αλλαγές του περιβάλλοντος προκύπτουν και από τη δραστηριότητα των άλλων ηθοποιών (το γεγονός ότι ο ταραχοποιός έχει δώσει

μόλις ένα λάθος πληροφορίες ή ότι η απάντηση του μαθητή είναι διαθέσιμη σε κοινή μνήμη).

Δράση(action)

Συγκεντρώνει όλες τις δράσεις (λειτουργίες) που επιτρέπουν τον ηθοποιό να λειτουργεί στο περιβάλλον. Για παράδειγμα εμφανίζει μια απάντηση ή υποβάλλει ένα πρόβλημα ή συγχαρεί τον μαθητή ή τον παραπλανείκτλ.

Έλεγχος(control)

Χειρίζεται τις καταστάσεις που συνεπάγονται μια πτυχή σχεδιασμού για τον προσδιορισμό δράσεων που θα ενεργοποιηθούν. Για παράδειγμα, ο δάσκαλος μπορεί να αποφασίσει να σταματήσει είτε να συνεχίσει τη διδασκαλία, ο troublemaker μπορεί να δώσει τη σωστή ή λάθος λύση, ...

Γνώση/Νόηση(cognition)

Επιτρέπει τον ηθοποιό να βελτιώσει τις επιδόσεις του, σύμφωνα με διάφορες πτυχές όπως : βελτίωση της αντίληψης του ηθοποιού (5) , επέκταση του έλεγχου(7) , συμπλήρωση του συνόλου ενεργειών (6) ,τροποποίηση της άποψης των άλλων ηθοποιών στη εσωτερική δραστηριότητά του (8) και τέλος βελτίωση των δικών του στρατηγικών συλλογισμού , βελτίωση της γνωστικής λειτουργίας,(9).

Σύμφωνα με το εικόνα 2 οι (5)(6)(7)(8)(9) : είναι οι επιδράσεις του γνωστικού επιπέδου.

Οι διασυνδέσεις μεταξύ αυτών των διαφορετικών ενοτήτων επιτρέπουν τέσσερις τρόπους λειτουργίας (functioning modes).

- **Ανακλαστική Λειτουργία** (reflex mode ,1): αφορά την ενότητα αντίληψης και δράσης. Στην περίπτωση αυτή έχουμε άμεση συσχέτιση μεταξύ μίας τυπικής κατάστασης και μιας συγκεκριμένης δράσης (αυθόρμητη δράση) χωρίς να υπάρχει κάποια αιτιολογία. Υπάρχει μόνο ένα είδος ταύτισης συμπεριφορών, που ελέγχει τις συνθήκες στην οποία κάθε δράση (action) πρέπει να εκτελεστεί. Αυτή η λειτουργία δεν καλεί το υψηλότερο επίπεδο για την ανάπτυξη ενεργειών ελέγχου.

- **Λειτουργία έλεγχου** (control mode ,2): αφορά την ενότητα αντίληψης, δράσης και έλεγχου. Ξεκινώντας από μια συγκεκριμένη κατάσταση, η μονάδα ελέγχου λαμβάνει μια απόφαση μεταξύ των πιθανών εναλλακτικών λύσεων και κατασκευάζει μία διαδικασία και δράσεις που έχουν επιλεχθεί και πρέπει να εκτελεστούν.

- **Συλλογιστική λειτουργία** (reasoning mode ,3): αφορά και τις τέσσερις ενότητες .Αυτή η λειτουργία μεσολαβεί όταν ο ηθοποιός δεν είναι σε θέση να λάβει κάποια απόφαση στο επίπεδο ελέγχου (όταν υπάρχουν αντιφάσεις, συγκρούσεις, αβεβαιότητα μεταξύ διαφορετικών αποφάσεων). Η γνωστική ενότητα θα

παρακάμψει τις γνώσεις των άλλων ενοτήτων προκειμένου να βελτιώσει ο ηθοποιός τη τρέχουσα συμπεριφορά. Ενώ ο πρωταρχικός στόχος αυτών των λειτουργιών είναι να έχει ο ηθοποιός αλληλεπίδραση με το περιβάλλον του, με αυτήν την λειτουργία ο ηθοποιός μπορεί επιπλέον να μάθει (δεδομένου ότι οι γνωστικές εργασίες/ενότητες είναι πάντα ενεργές). Με αυτόν τον τρόπο η επίδραση του επιπέδου γνώσης βελτιώνει το επίπεδο ελέγχου (7).

- **Λειτουργία παρατήρησης/μάθησης** (observation/learning mode ,4): αφορά την ενότητα αντίληψης και γνώσης. Στην περίπτωση αυτή, στόχος του ηθοποιού είναι να μάθει και να τροποποιήσει τον τρόπο που σκέφτεται από την παρατήρηση των άλλων ηθοποιών. Σε αυτή την κατάσταση ο ηθοποιός δεν έχει καμία κοινωνική συμπεριφορά (αλληλεπίδραση με άλλους ηθοποιούς), αλλά μόνο εσωτερική τροποποίηση των στόχων του .

Σύμφωνα με το εικόνα 2 οι (1)(2)(3)(4) : είναι οι λειτουργίες (functioning modes).

2.4.3 Αρχιτεκτονική πολλαπλών-στρατηγικών για ένα ITS (Architecture of a Multi-strategic Intelligent Tutoring System)

Στις πρώτες εκδόσεις των ITS, η παιδαγωγική λειτουργική μονάδα ρύθμιζε τις αλληλεπιδράσεις μεταξύ του εκπαιδευόμενου/μαθητή και του εκπαιδευτή/δάσκαλο. Σε αυτήν την αρχιτεκτονική υπάρχει επιπλέον μια συνεργατική προσέγγιση μεταξύ του μαθητή και του συστήματος , χρησιμοποιώντας έναν συμμαθητή (co-learner or a learning companion) που προσομοιώνει τη συμπεριφορά ενός δεύτερου μαθητή ο οποίος θα μάθει μαζί με τον μαθητή. Υπάρχουν δύο εναλλακτικές στρατηγικές σε αυτή τη συνεργατική προσέγγιση που μας οδηγούν στα πρότυπα που ονομάζονται «Μαθαίνω από τη διδασκαλία» (learning by teaching) και «Μάθηση από ενόχληση» (learning by disturbing). Στο πρώτο ο μαθητής διδάσκει το συμμαθητή του , δίνοντας του εξηγήσεις. Στο δεύτερο ο υπολογιστής προσομοιώνει δύο πράκτορες : ένα δάσκαλο και ένα ταραχοποιό (troublemaker). Ο ταραχοποιός έχει μια συγκεκριμένη συμπεριφορά : θα προσπαθήσει σκόπιμα να παραπλανήσει τον μαθητή ώστε να δοκιμάσει συστηματικά την αυτοπεποίθηση και τις γνώσεις του. Ο μαθητής συζητά τη λύση ενός προβλήματος με τον ταραχοποιό σε μια διαδικασία που ελέγχεται από τον δάσκαλο. Όταν υπάρχει αδιέξοδο ο δάσκαλος μπορεί να παρέμβει, δίνοντας μια υπόδειξη, μία σωστή λύση, κλπ.

Άρα καταλήγουμε σε 3 στρατηγικές :

- 1) Απλή στρατηγική: συμμετέχει μόνο ο μαθητής και ο δάσκαλος.
- 2) Learning by teaching: συμμετέχει ο μαθητής, ο δάσκαλος και ο συμμαθητής.

3) Learning by disturbing: συμμετέχει ο μαθητής, ο δάσκαλος και ο troublemaker.

Όπως είδαμε παραπάνω μια σύγχρονη ITS πρέπει να ενσωματώσει αρκετές στρατηγικές μάθησης και ο λόγος είναι ότι μία στρατηγική δε είναι κατάλληλη για όλα τα είδη των φοιτητών . Κάθε στρατηγική έχει τα ιδιαίτερα πλεονεκτήματα του και η επιλογή της εξαρτάται από διάφορους παράγοντες:

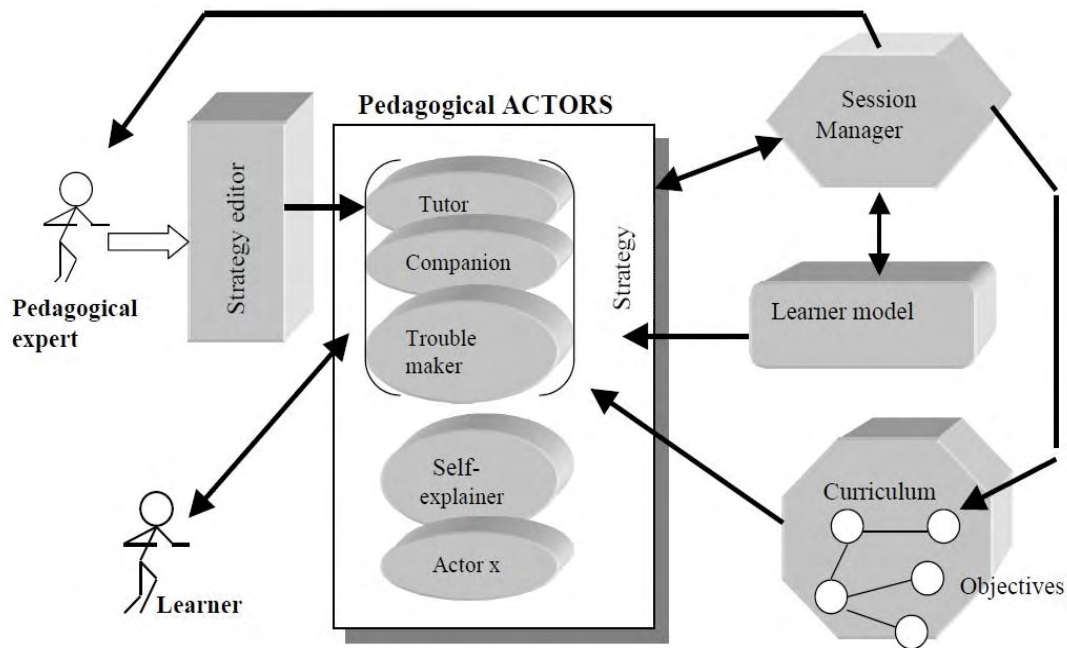
- *Το επίπεδο γνώσης του μαθητή:* Για παράδειγμα, η συνεργασία με έναν συμμαθητή (learning companion) θα ταιριάζει σε έναν μαθητή του οποίου το γνωστικό επίπεδο είναι χαμηλό ,επομένως χρειάζεται βοήθεια και πρέπει να υποκινηθεί από τη συνεργασία. Από την άλλη πλευρά, στη προσέγγιση learning by disturbing ο μαθητής πρέπει να υπερασπιστεί την άποψή του. Θα πρέπει να γνωρίζει πώς να ανακτήσει τις πληροφορίες που έχουν αποθηκευτεί για αυτόν προηγουμένως. Η στρατηγική αυτή ενισχύει τους καλούς σπουδαστές, αλλά αποθαρρύνει τους αδύναμους .

- *Τον τομέα της διδασκαλίας :* μερικές στρατηγικές είναι περισσότερο χρήσιμες - προσαρμόσιμες για έναν τομέα σε αντίθεση με άλλους τομείς . Για παράδειγμα, για να διδαχτούν έννοιες/στόχοι του μαθήματος μπορεί κανείς να χρησιμοποιήσει τη προσέγγιση οδηγίων (directive approach), ενώ κατά τη διδασκαλία επίλυσης προβλημάτων ο μαθητής μπορεί να επωφεληθεί από τη προσέγγιση «μάθηση μέσω της πράξης» (learning by doing) με έναν συμμαθητή.

- *Το κίνητρο:* τα κίνητρα του μαθητή μπορεί να εξελίσσονται με το χρόνο. Μεταβάλλοντας τη στρατηγική που χρησιμοποιείται με την πάροδο του χρόνου μπορεί να διατηρήσουμε το ενδιαφέρον του μαθητή, ιδίως όταν εναλλάσσονται μεταξύ ατομικής μάθησης στη ομαδική μάθησης .

- *Τα συναισθηματικά χαρακτηριστικά:* πρέπει κανείς να λάβει υπόψη τις διάφορες πτυχές του μαθητή όπως η αυτοπεποίθηση, το στυλ μάθησης, συναισθηματική κατάσταση (όλα αυτά, επίσης, μεταβάλλονται με την πάροδο του χρόνου).

Για να λάβουμε υπόψη όλες αυτές τις πτυχές του μαθητή θα πρέπει να καθορίσουμε μια multi- strategic αρχιτεκτονική ITS (εικόνα 3).



Architecture of a Multi-strategic Intelligent Tutoring System

Εικόνα 3: Αρχιτεκτονική ενός Ευφυούς Συστήματος Διδασκαλίας με Πολλαπλές-Στρατηγικές

Η διδακτέα ύλη (Curriculum) είναι μια οργανωμένη εκπροσώπηση των θεμάτων που θα διδαχθούν. Το curriculum αποτελείται από τους αντικειμενικούς/βασικούς στόχους (objectives) που πρέπει να πετύχει ο μαθητής (όπως η μάθηση μιας συγκεκριμένης έννοιας ή απόκτηση συγκεκριμένης δεξιότητας) και από τις δραστηριότητες που το σύστημα επιλέγει, παρουσιάζει και διαχειρίζεται, για να βοηθήσει τον μαθητή στην επίτευξη των στόχων αυτών. Οι δραστηριότητες χαρακτηρίζονται από τους στόχους που υποστηρίζουν και από το ρόλο που παίζουν (πχ μάθημα, επίδειξη, άσκηση κλπ.).

Ο διαχειριστής συνεδρίας (Session Manager) επιλέγει τι θα διδάξει (στόχοι της διδακτέας ύλης που πρέπει να επιτευχθούν) και ποία στρατηγική θα χρησιμοποιηθεί στη δεδομένη στιγμή, με βάση την περιγραφή της εφαρμοσιμότητας /καταλληλότητας της στρατηγικής και το μοντέλο μαθητή. Για να επιλέξει την καλύτερη στρατηγική, το session manager χρησιμοποιεί μια βάση γνώσεων που εμπεριέχει (επίπεδο γνώσης μαθητή, κίνητρο,...). Αφού επιλεγεί, η στρατηγική ενεργοποιείται με βάση τη συνεργασία μερικών παιδαγωγικών πρακτόρων (pedagogical actors).

Ο επεξεργαστής στρατηγικής (strategy editor) είναι ένα εργαλείο εξαγωγής γνώσης που βοηθάει τον παιδαγωγικό εμπειρογνώμονα/ειδικό να ορίσει λεπτομερώς τα χαρακτηριστικά των ηθοποιών/πρακτόρων και να καθορίσει νέες στρατηγικές με το συνδυασμό νέων ή υπαρχόντων ηθοποιών. Έτσι επιτρέπουμε στους ηθοποιούς να αποκτήσουν την ικανότητα να διδάσκουν (actor instructability).

Έχουμε επιλέξει να μοντελοποιήσουμε τη πλειοψηφία αυτών των στοιχείων του ITS σαν ηθοποιούς, επειδή έχουν σχεδιαστεί για τη σταδιακή απόκτηση της απαραίτητης τεχνογνωσίας. Ειδικότερα, στη παραπάνω αρχιτεκτονική οι παιδαγωγικοί ηθοποιοί (pedagogical actors) και ο session manager είναι ηθοποιοί που περιέχουν περίπλοκη παιδαγωγική εμπειρογνωμοσύνη για την βελτίωση του συστήματος. Για παράδειγμα, ο session manager πρέπει να λαμβάνει υπόψη πολλούς παράγοντες στην διαδικασία επιλογής μιας στρατηγικής, της οποίας η πολυπλοκότητα απαγορεύει ένα επιτυχημένο μοντέλο το οποίο θα είναι αποτελεσματικό με τη πρώτη προσπάθεια. Το γεγονός ότι οι ηθοποιοί μπορούν να εξελιχθούν με τη πάροδο του χρόνου, μας δίνει τη δυνατότητα οι ηθοποιοί να ξεκινήσουν με μικρή τεχνογνωσία, πχ ο strategy editor actor, η οποία εμπλουτίζεται συνεχώς.

Η προηγούμενη αρχιτεκτονική και οι διαφορετικές ιδιότητες των ηθοποιών οδηγούν σε μία κυκλική διαδικασία λειτουργίας, ώστε να επιτευχθούν οι παιδαγωγικοί στόχοι του ITS. Το γνωστικό επίπεδο των ηθοποιών τους επιτρέπει να ξεκινήσουν με ένα μικρό σύνολο παιδαγωγικών γνώσεων και σταδιακά να εμπλουτίσουν την εμπειρία τους, χρησιμοποιώντας μια δυναμική διαδικασία διδασκαλίας.

Ο κύκλος λειτουργίας είναι ο εξής:

Αρχικά ο pedagogical expert χρησιμοποιεί τον strategy editor για να τροφοδοτήσει τους ηθοποιούς, οι οποίοι εμπλέκονται στη στρατηγική που θέλει να αναπτύξει, με μια αρχική εξειδίκευση/γνώση (κλασική διδασκαλία). Η στρατηγική τίθεται σε λειτουργία μέσω ενός συγκεκριμένου παραδείγματος με έναν μαθητή.

Στη συνέχεια, ο session manager είναι σε θέση να δοκιμάσει την αποδοτικότητα της στρατηγικής χάρη στις μετά-συνθήκες (post-conditions) και το μοντέλο του μαθητή. Όταν ο session manager μάθει κάτι (έχει αλλάξει τη συμπεριφορά του) θα προτείνει στον expert, ένα νέο σύνολο κανόνων για την επιλογή των στρατηγικών.

Τελικά ο expert θα αποφασίζει για την εγκυρότητα της πρότασης και θα εμπλουτίσει την τεχνογνωσία των άλλων ηθοποιών, χρησιμοποιώντας το strategy editor.

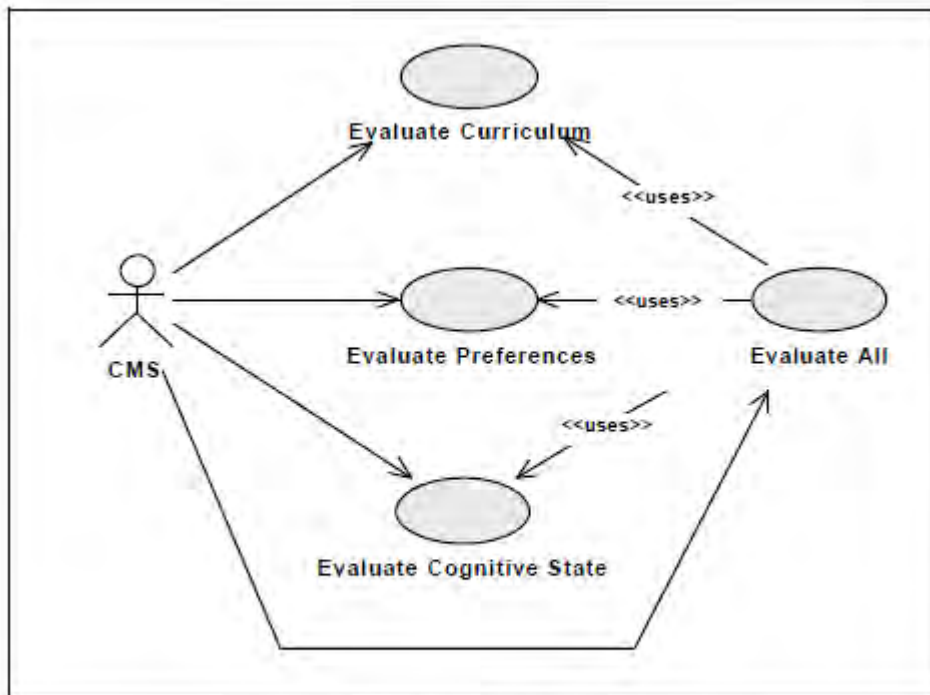
2.5 Ένα βασισμένο σε πράκτορα Ευφυής Σύστημα Διδασκαλίας για εξ αποστάσεως εκπαίδευση - ABITS: An Agent-Based Tutoring System for distance learning.

2.5.1 Εισαγωγή

Ένα αποτελεσματικό σύστημα εκπαίδευσης θα πρέπει να επιτρέπουν στους χρήστες να κάνουν το μάθημα τους, χωρίς να τους περιορίζει ο χρόνος και ο χώρος. Για να εκπληρωθούν αυτές οι απαιτήσεις, η καλύτερη λύση είναι ένα σύστημα βασισμένο στο Web (Διαδίκτυο). Το σύστημα αυτό έχει μηδέν κόστος εγκατάστασης και παρέχει στους τελικούς χρήστες τη μέγιστη ευελιξία χρόνου και τόπου.

Τρία είδη Συστημάτων Διδασκαλίας που βασίζονται στο Web (Web-Based Tutoring WBT) υπάρχουν διαθέσιμες αυτή τη στιγμή :

- *Στατική WBT*: ο δάσκαλος οργανώνει το εκπαιδευτικό υλικό, ώστε να καλύπτει ένα ή περισσότερα θέματα και τα μετατρέπει σε συνδεδεμένες διαδραστικές σελίδες HTML (interactive linked HTML pages or different kinds of Web-deliverable objects). Έπειτα το υλικό τοποθετείται on-line έτσι ώστε να είναι ορατό σε όλους. Οι μαθητές μπορούν να παρακολουθήσουν το μάθημα όπως καθορίστηκε από τον δάσκαλο.
- *Εξατομικευμένη WBT*: οι εκπαιδευτικοί, χρησιμοποιώντας ένα συγκεκριμένο είδος λογισμικού που ονομάζεται Σύστημα Διαχείρισης Μαθήματος (Course Management System i.e. Macromedia Attain), είναι σε θέση να εκτελέσουν ένα σύνολο καθηκόντων. Μπορούν να παρακολουθούν τις γνώσεις των μαθητών, εξετάζοντας τους, να ορίζουν διαφορετικά μονοπάτια για τα αντικείμενα μάθησης (learning objects) ώστε να καλύψουν διαφορετικούς μαθησιακούς στόχους, κλπ.
- *Προσαρμοστική WBT*: περιλαμβάνει όλα τα χαρακτηριστικά της εξατομικευμένης WBT, αλλά ο δάσκαλος προσομοιώνει στη δραστηριότητά του τεχνικές τεχνητής νοημοσύνης (artificial intelligence techniques).



ABITS Use Case Diagram

Εικόνα 4: Διάγραμμα ροής ενός ABITS

Στην υποενότητα αυτή θα παρουσιάσουμε το ABITS: ένα άκρως επαναχρησιμοποιήσιμο Έξυπνο Σύστημα Διδασκαλίας το οποίο είναι σε θέση να επεκτείνει ένα παραδοσιακό Σύστημα Διαχείρισης Μαθημάτων (Course Management System CMS) με ένα σύνολο «έξυπνων» λειτουργιών που επιτρέπουν τη μοντελοποίηση των φοιτητών και τη αυτόματη παραγωγή της διδακτέας ύλης. Προσθέτοντας το ABITS ως μια ενότητα, κάθε εξατομικευμένο WBT θα μπορεί να γίνει προσαρμοστικό WBT. Το ABITS έχει αναπτυχθεί στα πλαίσια του InTraSys ESPRIT έργου. Το InTraSys (Intelligent Training System in Technical Assistance) είναι ένα πολύ σύνθετο σύστημα εκπαίδευσης/ μάθησης που απευθύνεται προς υψηλής τεχνολογίας οργανώσεις , οι στόχοι των οποίων είναι βελτίωση της αποτελεσματικότητας εκπαίδευσης και μάθησης, μείωση του κόστους εκπαίδευσης και μείωση του χρόνου εκπαίδευσης των εργαζομένων .

Οι ευφυείς λειτουργίες του ABITS συνοψίζονται στο διάγραμμα της εικόνας 4 και θα περιγράφουν παρακάτω. Ειδικότερα, τη "Αξιολόγηση της Διδακτέας Ύλης» (Evaluate Curriculum) λειτουργία θα την μελετήσουμε λεπτομερώς στη ενότητα 2.5.4 . Τις "Αξιολόγηση των Προτιμήσεων" (Evaluate Preferences) και "Αξιολόγηση της Γνωστικής Κατάστασης" (Evaluate Cognitive State) λειτουργίες, θα τις περιγράψουμε πλήρως στη ενότητα 2.5.3.

Οι λειτουργίες αυτές έχουν αποτελεσματικότητα αν χρησιμοποιήσουμε ένα σύνολο κανόνων για την ευρετηρίαση της γνώσης (knowledge indexing), το οποίο βασίζεται σε μετά-δεδομένα και εννοιολογικά γραφήματα. Παρακάτω θα αναλύσουμε τα θέματα αυτά.

2.5.2 Ευρετηρίαση της Γνώσης (Knowledge Indexing)

Μέσα στο ABITS, όλο το διδακτικό υλικό είναι οργανωμένο σε διάφορα αντικείμενα μάθησης (Learning Objects) και αποθηκεύονται σε ένα σύστημα αρχείων (Course Material File System). Στην περίπτωση μας, ένα μαθησιακό αντικείμενο είναι ένα "λογικό δοχείο" που αντιπροσωπεύει/περιέχει ένα ατομικό Web-deliverable πόρο, όπως ένα μάθημα (μια σελίδα HTML), μια προσομοίωση (μια εφαρμογή Java applet), ένα τεστ (μια σελίδα HTML με φόρμα αξιολόγησης) και κάθε είδους Web-deliverable αντικείμενο.

Τα αντικείμενα μάθησης πρέπει να είναι κατηγοριοποιημένα, ώστε το ABITS να ξέρει τι είναι το κάθε ένα από αυτά και πώς μπορεί να χρησιμοποιηθούν κατά τη διάρκεια της μαθησιακής διαδικασίας. Κάποιο είδος πληροφοριών σχετικά με τα αντικείμενα αυτά είναι επίσης απαραίτητο. Καλούμε αυτό το είδος των πληροφοριών μεταδεδομένων (Metadata).

2.5.2.1 Μεταδεδομένα

"Τα μεταδεδομένα είναι πληροφορίες για ένα αντικείμενο μάθησης και κύριος στόχος τους είναι να τοποθετούν, με αποτελεσματικό και αποδοτικό τρόπο, τους πόρους σε ένα σύστημα ή δίκτυο υπολογιστών".

Στο ABITS χρησιμοποιείται το Learning Object Metadata (LOM) πρότυπο. Το πρότυπο αυτό καθορίζει το ελάχιστο σύνολο ιδιοτήτων που χρειάζονται για να επιτρέψει στο σύστημα να διαχειριστεί, να βρει, και να αξιολογήσει τα αντικείμενα μάθησης. Στη LOM τα μεταδεδομένα είναι δομημένα με ιεραρχικό τρόπο: οι κατηγορίες αποτελούνται από υποκατηγορίες, τα οποία περιέχουν τα στοιχεία δεδομένων. Τα στοιχεία δεδομένων μπορεί να είναι "Υποχρεωτικά" (πρέπει να είναι παρόντες) ή "προαιρετικά" (μπορεί να υπάρχουν). Ο IEEE ορισμός για τα μεταδεδομένα σημαίνει ότι οι περιγραφές μαθησιακών πόρων έχουν ομαδοποιηθεί σε ουσιαστικές κατηγορίες. Παρακάτω φαίνονται έξι κατηγορίες που ορίζονται από το IEEE και τα χρησιμοποιεί το ABITS:

·Γενικές(**General**): ομαδοποίηση όλων των αντικειμένων ανεξαρτήτως χαρακτηριστικών και σημασιολογικών περιγραφών.

·Κύκλος Ζωής (**Life Cycle**): ομαδοποίηση βάση χαρακτηριστικών που συνδέονται με τον κύκλο ζωής των πόρων.

·Μέτα Μεταδεδομένα (**Meta Metadata**): ομαδοποίηση βάση χαρακτηριστικών της ίδιας της περιγραφής .

·Τεχνική (**Technical**): ομαδοποίηση βάση των τεχνικών χαρακτηριστικών των πόρων.

·Εκπαιδευτικά (**Educational**): ομαδοποίηση βάση των εκπαιδευτικών και παιδαγωγικών χαρακτηριστικών των πόρων.

·Διαχείριση Δικαιωμάτων (**Rights Management**): ομαδοποίηση βάση χαρακτηριστικών που εξαρτώνται από το είδος της χρήσης που προβλέπονται για το πόρο.

Στο Πίνακα 1 ανακεφαλαιώνονται οι κατηγορίες και τα στοιχεία δεδομένων που αποτελούν το metadata scheme (σχήμα μεταδεδομένων) του ABITS.

General	MetaMetaData	Educational	RightsManagmnt
Identifier	Create	PedagogicalType	Role
Title	MetadataScheme	CoursewareGenre	Description
Language	Technical	Format	Conditions
Description	Format	Approach	Reciprocity
Domain	Size	InteractivityLevel	Attribution
Idea	LocSpec	SemanticDensity	Prize
Structure	Requirements	EducationalUse	MonetaryUnit
LifeCycle	Type	Role	Amount
Version	Name	Difficulty	UnitOfPricing
Create	MinimumVersion	Level	
	MaximumVersion	Duration	

ABITS Learning Object Metadata Scheme

Πίνακας 1: Σχήμα Μεταδεδομένων του ABITS

2.5.2.2 Εννοιολογικά Διαγράμματα (Conceptual Graphs)

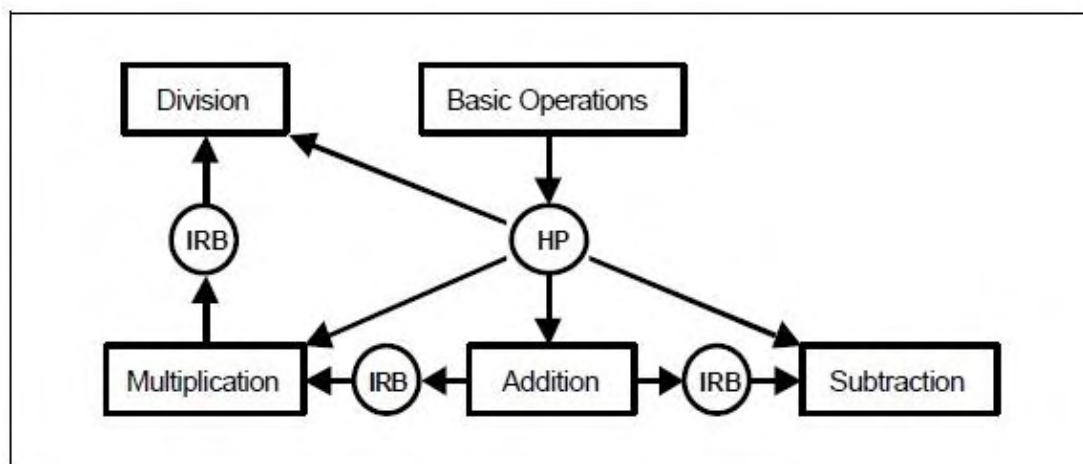
Το σχήμα μεταδεδομένων όχι μόνο πρέπει να παρέχουν πληροφορίες σχετικά με ένα Αντικείμενο Μάθησης αλλά πρέπει επίσης να παρέχουν πληροφορίες σχετικά με τις σχέσεις και αλληλεξαρτήσεις των αντικειμένων αυτών . Για το σκοπό αυτό το LOM πρότυπο παρέχει ένα σημαντικό στοιχείο δεδομένων που ονομάζεται «Ιδέα» (ανήκει στη κατηγορία "Γενική") και επιτρέπει τη λειτουργία του τομέα «Αντίληψη» (Conceptualization).

Ο τομέας αυτός είναι μία αφηρημένη, απλοποιημένη άποψη του κόσμου που θέλουμε να αντιπροσωπεύσουμε για κάποιο σκοπό. Για να το πετύχουμε αυτό χρησιμοποιούμε ένα εννοιολογικό γράφο (Conceptual Graph). Ο εννοιολογικός γράφος αποτελείται από έννοιες και εννοιολογικές σχέσεις και κάθε γραμμή συνδέει κάποια εννοιολογική σχέση r με κάποια έννοια c . Στο πλαίσιο του ABITS, τα εννοιολογικά γραφήματα χρησιμοποιούνται για να συνδέσουν έννοιες με τρία είδη σχέσεων: προαπαιτούμενη (prerequisite) σχέση, υπό-έννοια (sub-concept) σχέση και τη γενική (general) σχέση (βλέπε πίνακα 2 για περισσότερες λεπτομέρειες).

Kind of Relation	Relation Name	Abbreviation
Prerequisite	... "IsRequiredBy" ...	IRB
	... "Requires" ...	R
Sub-Concept	... "IsPartOf" ...	IPO
	... "HasPart" ...	HP
General Relation	... "IsRelatedTo" ...	IRT

Πίνακας 2: ABITS Σχέσεις

Παρακάτω στη εικόνα 5 θα δούμε ένα παράδειγμα εννοιολογικού γράφου που εξετάζει τις έννοιες της πρόσθεσης, αφαίρεσης, διαίρεσης, του πολλαπλασιασμού και "βασικές λειτουργίες" μέσα στον τομέα των μαθηματικών πράξεων. Μπορούμε να παρατηρήσουμε πώς συνδέονται όλες οι έννοιες με τις εννοιολογικές σχέσεις, βασιζόμενοι στο πεδίο abbreviation του πίνακα 2.



Εικόνα 5: Παράδειγμα Εννοιολογικού Γράφου

Το ABITS χρησιμοποιεί το Conceptual Graph Interchange Form (CGIF) σα γλώσσα ενός εννοιολογικού σχήματος.

Για να ανακεφαλαιώσουμε, τα πεδία μεταδεδομένων δίνουν πληροφορίες για τα Αντικείμενα Μάθησης και για τις Έννοιες, ενώ τα Εννοιολογικά Γραφήματα δίνουν πληροφορίες σχετικά με το πώς οι Έννοιες σχετίζονται μεταξύ τους. Στα επόμενα κεφάλαια θα δούμε πώς οι πληροφορίες αυτές χρησιμοποιούνται για το μοντέλο του μαθητή και για την αυτόματη παραγωγή της διδακτέας ύλης (curriculum) .

2.5.3 Μοντελοποίηση Μαθητή (Student Modeling)

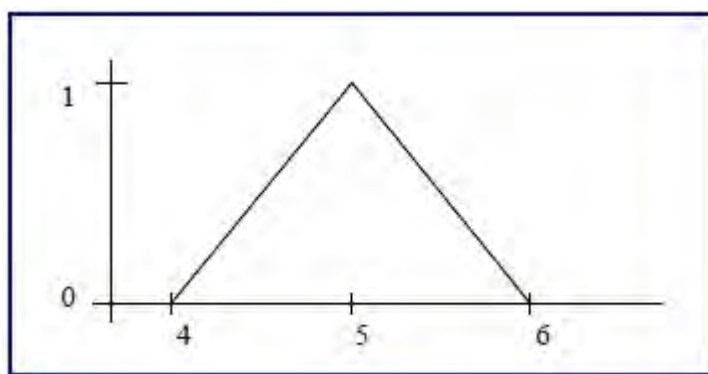
Κάθε στιγμή το ABITS πρέπει να είναι σε θέση να εκτιμήσει τη γνωστική κατάσταση και τις προτιμήσεις μάθησης κάθε μαθητή , βασιζόμενος στις δραστηριότητες που ανέπτυξε ο μαθητής . Οι εν λόγω πληροφορίες αποτελούν το λεγόμενο μοντέλο του μαθητή. Οι πληροφορίες αυτές αποτελούνται από πολλά "ασαφής" πεδία για αυτό είναι χρήσιμο να ορίσουμε τους Fuzzy Numbers (Ασαφή νούμερα) .

Αν X είναι ένα σύνολο από αντικείμενα, ένα fuzzy («ασαφή») υποσύνολο A του X είναι ένα σύνολο που χαρακτηρίζεται από μια συνάρτηση συμμετοχής του τύπου:

$$\mu(x | A): X \rightarrow [0,1]$$

όπου x είναι ένα γενικό στοιχείο του X και $\mu(x | A)$ είναι ο βαθμός συμμετοχής του x στο A . Το A υποσύνολο μπορεί να θεωρηθεί μια ειδική περίπτωση ασαφούς υποσύνολου με σχέσεις , που έχουν τιμές στο $[0,1]$ όπου το 1 υποδεικνύει την πλήρη ένταξη/σχέση στο A και 0 μη ένταξη/σχέση στο A .

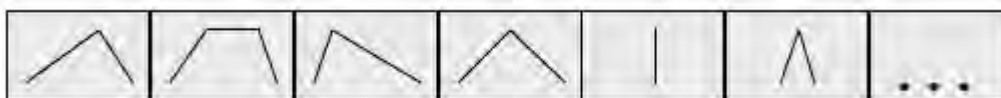
Ένας ασαφής αριθμός μπορεί να παρουσιαστεί γραφικά, μέσω της λεγόμενης belief graphs (γράφημα πεποίθησης), που αντιστοιχεί τις σχέσεις με όλες τις υποστηριζόμενες τιμές. Η εικόνα 6 παρουσιάζει ένα belief graphs για ένα τριγωνικό ασαφής αριθμό, το οποίο έχει μόνο μια τιμή στο μέγιστο βαθμό συμμετοχής, τον αριθμό 5.



Εικόνα 6: Τριγωνικός Ασαφής Αριθμός

2.5.3.1 Γνωστική Κατάσταση (Cognition State)

Στη γνωστική κατάσταση καταγράφουμε το βαθμό γνώσης, κάθε έννοιας/αρχής, που επιτεύχθηκε από έναν συγκεκριμένο μαθητή. Αυτές οι πληροφορίες αντιπροσωπεύονται από μία συμβολοσειρά/σειρά ασαφών αριθμών (ένα για κάθε έννοια). Γραφικά μία τέτοια συμβολοσειρά φαίνεται στην εικόνα 7. Αυτή η συμβολοσειρά ενημερώνεται (update) δυναμικά από το ABITS κατά τη διάρκεια εκπαιδευτικών δραστηριοτήτων και τεστ.



Εικόνα 7: Γραφική Αναπαράσταση Γνωστικής Κατάστασης

Η απόφαση για τη χρήση fuzzy αριθμών στη Γνωστική Κατάσταση προκύπτει από την ανάγκη να διαχειριστούμε την αβεβαιότητα στην αξιολόγηση των φοιτητών. Με τον τρόπο αυτό, στην πραγματικότητα, μπορούμε να χρησιμοποιήσουμε διάφορα είδη αξιολογήσεων με διαφορετικό βαθμό αξιοπιστίας. Για παράδειγμα, όταν ένας μαθητής διαβάζει την επεξήγηση ενός αντικειμένου μάθησης, που σχετίζεται με ένα συγκεκριμένο σύνολο εννοιών, το ABITS μπορεί να συμπεράνει ότι υπάρχει μια μικρή αύξηση στην γνωστική κατάσταση του φοιτητή για αυτές τις έννοιες, αλλά με πολύ μεγάλο βαθμό αβεβαιότητας. Όταν ο μαθητής απαντάει σωστά σε ένα τεστ, τότε υπάρχει επίσης μια αύξηση στη γνωστική κατάσταση του μαθητή για τις έννοιες που εμπλέκονται στο τεστ αλλά με πιο χαμηλό βαθμό αβεβαιότητας. Μια άλλη σημαντική πτυχή που πρέπει να ληφθεί υπόψη είναι ότι ένας μαθητής μπορεί να ξεχάσει ό, τι έχει μάθει στο παρελθόν. Για να διαμορφώσει αυτήν την ιδιότητα του ανθρώπου, το ABITS εφαρμόζει μία Forgetting Function για την Γνωστική Κατάσταση. Επειδή όλοι οι άνθρωποι δε ξεχνούν με τον ίδιο τρόπο, αυτός ο αλγόριθμος δεν μπορεί να μειώσει το βαθμό γνώσης του μαθητή, αλλά ενισχύουν την ανάγκη για χρήση των fuzzy αριθμών, που υποδεικνύουν ότι οι αξιολογήσεις είναι όλο και πιο αναξιόπιστες.

2.5.3.2 Προτιμήσεις μάθησης (Learning Preferences)

Οι μαθησιακές προτιμήσεις περιέχουν όλες τις πληροφορίες για τις αντιληπτικές ικανότητες του μαθητή, πχ σε ποίο είδος πόρων ο συγκεκριμένος μαθητής φαίνεται να είναι πιο δεκτικός. Για να αξιολογήσουμε τις προτιμήσεις των μαθητών εκμεταλλευόμαστε τη κατηγορία μεταδεδομένων "Educational".

Τα στοιχεία δεδομένων που περιέχονται στη κατηγορία αυτή φαίνονται στον πίνακα 3, όλα τα πεδία μαθησιακών προτιμήσεων με όλες τις πιθανές τιμές.

Fields	Possible Values
Format	Text, Image, Slide, Hypertext, Video Clip, Simulation, Virtual Reality
Approach	Inductive, Deductive, Explorative
Interactivity Level	Very Low, Low, Medium, High, Very High
Semantic Density	Very Low, Low, Medium, High, Very High
Difficulty	Very Low, Low, Medium, High, Very High

Πίνακας 3: Πιθανές Τιμές των Προτιμήσεων Μάθησης

Για τη μορφή (*Format*) των στοιχείων δεδομένων, το ABITS διατηρεί μια λίστα από ασαφούς αριθμούς, που αξιολογούν την δεκτικότητα ενός συγκεκριμένου μαθητή στα μέσα ενημέρωσης που συμμετέχει (κείμενο, υπερκείμενο (hypertext), βίντεο, προσομοίωση, διαφάνεια (slide)). Η ίδια ιδέα χρησιμοποιείται και στο παιδαγωγικό τομέα προσέγγιση (*Approach*), για να αξιολογήσουμε εάν ο φοιτητής δείχνει πιο ενδιαφέρον για τη εισαγωγική (inductive), συμπερασματική (deductive) ή εξερευνητική (exploratory) προσέγγιση μάθησης. Οι περιπτώσεις της *Interactivity Level* (Επίπεδο Διαδραστικότητας), *Semantic Density* (Σημασιολογική Πυκνότητα) και *Difficulty* (Δυσκολία), εκπροσωπούνται από τις αξίες (πολύ χαμηλή, χαμηλή, μέση, υψηλή, πολύ υψηλή), το ABITS διατηρεί έναν ασαφή αριθμό (για κάθε παράμετρο) που αντιπροσωπεύει το καλύτερο δεκτικό επίπεδο για ένα συγκεκριμένο μαθητή. Για την αξιολόγηση των προτιμήσεων των σπουδαστών, το ABITS εκμεταλλεύεται την ιδέα αυτή: κατά τη διάρκεια της διαδικασίας μάθησης υπάρχουν κάποια Milestones, όταν η Γνωστική Κατάσταση του μαθητή ενημερώνεται. Αυτό σημαίνει ότι για κάθε Έννοια που συμμετέχει στις δραστηριότητες που εκτελούνται από τον μαθητή, υπάρχει μια νέα αξιολόγηση. Με αυτόν τον τρόπο το ABITS μπορεί να αξιολογήσει την παιδαγωγική αποτελεσματικότητα των αντικειμένων μάθησης. Για παράδειγμα, εάν το επίπεδο γνώσης μιας συγκεκριμένης Έννοιας έχει αυξηθεί κατά δύο milestones και τα Αντικείμενα Μάθησης σχετικά με αυτή τη συγκεκριμένη Έννοια ήταν προσομοιώσεις, το ABITS συμπεραίνει ότι ο φοιτητής είναι δεκτικός σε προσομοιώσεις. Το σύστημα θα αυξήσει, κατά συνέπεια, στο πεδίο Format, τη τιμή που αναφέρεται στις προσομοιώσεις.

Οι πληροφορίες που υπολογίζονται από το ABITS για το Μοντέλο Μαθητή, μπορούν να αξιοποιηθούν άμεσα από δασκάλους ή μπορούν να χρησιμοποιηθούν στην αυτόματη παραγωγή διδακτέας ύλης (curriculum). Στο επόμενο κεφάλαιο θα εξηγήσουμε πώς ένα ABITS μπορεί να το κάνει αυτό.

2.5.4 Αυτόματη παραγωγή διδακτέας ύλης (curriculum)

Σε κάθε φοιτητή μπορεί να ανατεθεί ένα ή περισσότερα διαφορετικά μαθήματα (course). Ένα Μάθημα ABITS αποτελείται κυρίως από ένα σύνολο μαθησιακών στόχων (Learning Goals) και ένα curriculum .

Με τη λέξη μαθησιακούς στόχους (που είναι εντελώς διαφορετικό από Learning Objects) εννοούμε ένα σύνολο βασικών εννοιών/αρχών (Concepts) ,τα οποία είναι αναγκαία να τα μάθει ο μαθητής ώστε να ολοκληρωθεί επιτυχώς ένα συγκεκριμένο μάθημα. Οι έννοιες/αρχές είναι μέρος ενός τομέα και εκπροσωπούνται στο εσωτερικό ενός Conceptual Graph.

Με τη λέξη curriculum εννοούμε μία διατεταγμένη λίστα μαθησιακών αντικειμένων (Learning Objects), που μπορεί να χρησιμοποιηθεί για να παρέχει σε ένα συγκεκριμένο φοιτητή όλες τις αναγκαίες γνώσεις για να συμπληρώσει ένα συγκεκριμένο μάθημα. Ενώ οι μαθησιακοί στόχοι δείχνουν τι πρέπει να μάθει ένας φοιτητής (δηλαδή ποιες έννοιες/αρχές), το curriculum διευκρινίζει πώς μπορεί ο μαθητής να μάθει αυτές τις έννοιες.

Διαφορετικοί μαθητές μπορούν να απαιτούν διαφορετικά curriculum για να μάθουν τον ίδιο μαθησιακό στόχο, ανάλογα με την γνωστική τους κατάσταση και τις μαθησιακές προτιμήσεις. Για το λόγο αυτό το curriculum παρέχεται ως εξής: Ξεκινώντας από τη λίστα των μαθησιακών στόχων που πρέπει να καλυφθούν, το curriculum παρουσιάζεται με μια διαδικασία τριών βημάτων.

1. Εξετάζεται η λίστα ΜΣ των Μαθησιακών Στόχων, προκειμένου να λάβουμε τη λίστα E το οποίο περιέχει όλες τις Έννοιες, που πρέπει ο φοιτητής να μάθει για να φτάσει αυτούς τους στόχους. Για να λάβουμε τη λίστα E χρησιμοποιούμε μία αναδρομική συνάρτηση: για κάθε ϵ έννοια που περιέχεται στην ΜΣ γίνεται μια δοκιμή: αν ο μαθητής ξέρει ήδη το ϵ (εξετάζουμε την Γνωστική του Κατάσταση) τότε το ϵ απορρίπτεται , ειδάλλως το ϵ προστίθεται στο E όπως επίσης και όλες οι προαπαιτούμενες έννοιες του ϵ (εξετάζουμε το Εννοιολογικό Σχήμα του ϵ). Η διαδικασία αυτή επαναλαμβάνεται μέχρι να μην υπάρχουν άλλα ϵ μέσα στη λίστα ΜΣ.

2. Η λίστα των εννοιών E μετατρέπεται σε μια λίστα των Μαθησιακών Αντικειμένων MA ,τα οποία λαμβάνονται από τη Βάση Δεδομένων των Μαθημάτων (Course Material Database) έχοντας υπόψη ποιες είναι και οι μαθησιακές προτιμήσεις του μαθητή (εξετάζουμε τη «Educational » κατηγορία στα μεταδεδομένα). Η σειρά των μαθησιακών αντικειμένων μέσα στη MA λίστα ακολουθεί τον εξής κανόνα: αν ϵ_1 έχει σα προαπαιτούμενη την ϵ_2 έννοια τότε η επεξήγηση (το μαθησιακό αντικείμενο που αντιστοιχεί στην έννοια αυτή) του ϵ_2 θα τοποθετηθεί στο curriculum πριν την ϵ_1 .

3. Τελικά η λίστα ΜΣ των Μαθησιακών Αντικειμένων μετατρέπεται σε μία Σειρά Μαθημάτων (curriculum) CURR, με την προσθήκη Testing Material και Milestones. Το Testing material είναι Testing Learning Objects (δηλαδή σελίδες HTML με φόρμες αξιολόγησης) και τα milestones είναι ενσωματωμένες ABITS κλήσεις. Τα Milestones τοποθετούνται μετά από κάθε μπλοκ τεστ και στο τέλος του curriculum , ώστε να συμβουλευτεί το Σύστημα Διαχείρισης Μαθημάτων , που το ABITS πρέπει να καλέσει σε αυτό το σημείο , για να ενημερώσει το μοντέλο του μαθητή ή και την ακολουθία curriculum για το συγκεκριμένο μαθητή.

Είναι σημαντικό να σημειωθεί ότι όλες οι λειτουργίες ABITS χρησιμοποιούνται από το Σύστημα Διαχείρισης Μαθήματος , με συστηματικό ή περιστασιακό τρόπο, σύμφωνα με τις παιδαγωγικές και διαχειριστικές πολιτικές .

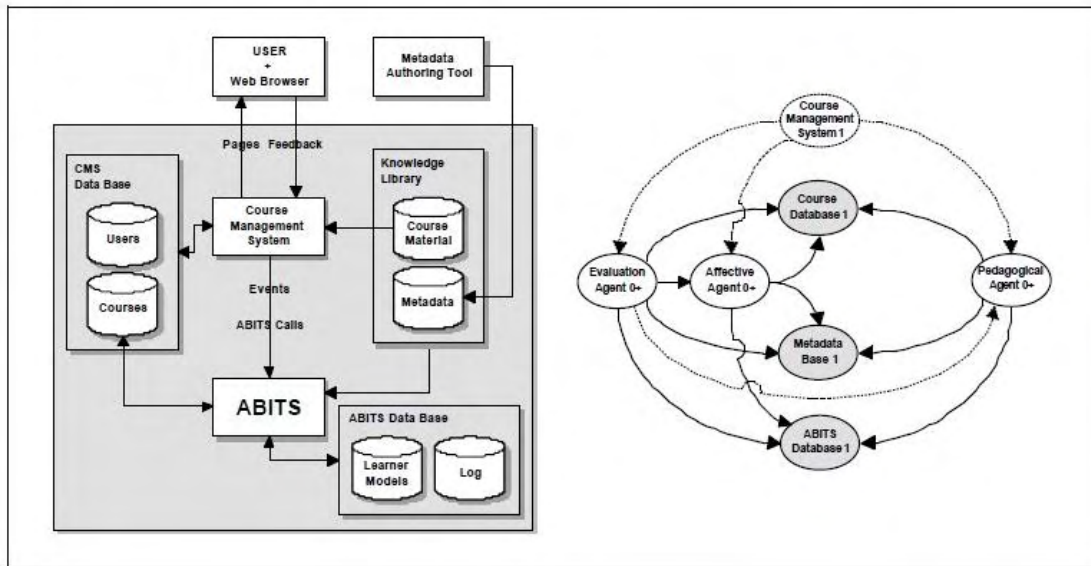
Το μοντέλο του μαθητή μπορεί να χρησιμοποιηθεί άμεσα από το ABITS στη διαδικασία δημιουργίας curriculum ή από τους καθηγητές για τη λήψη αποφάσεων σχετικά με τα μαθήματα και τις δραστηριότητες που θα αναθέσουν στους μαθητές .

2.5.5 Αρχιτεκτονική ABITS

Ενώ στα προηγούμενα κεφάλαια εξετάσαμε το ABITS από την λειτουργική άποψη (λειτουργίες που παρέχονται από ABITS και πώς λειτουργούν), στο κεφάλαιο αυτό θα δούμε την αρχιτεκτονική του ABITS (πώς συγκροτείται το ABITS).

Στη εικόνα 8α απεικονίζονται σχέσεις και αλληλεξαρτήσεις μεταξύ ABITS και ενός εξωτερικού Συστήματος Διαχείρισης Μαθημάτων (Course Management System). Και οι δύο ενότητες έχουν τη δυνατότητα πρόσβασης σε μία κοινόχρηστη βάση δεδομένων μαθημάτων (CMS Data Base). Το Σύστημα Διαχείρισης επιτρέπει κλήσεις και πρόσβαση στις βάσεις δεδομένων, μέσω μιας περιγραφικής γλώσσας που υποστηρίζεται από το ABITS. Οι βάσεις δεδομένων είναι οι εξής :

- *Course Material Database* περιέχει όλα τα μαθησιακά αντικείμενα, με τη μορφή web-deliverable αρχείων (HTML, VRML, CLASS, κλπ.).
- *Metadata database* περιέχει όλα τα σχήματα μεταδεδομένων σε XML / RDF μορφή (η βάση αυτή μπορεί να προσεγγιστεί και να τροποποιηθεί με Metadata Authoring Tool).
- *Log database* περιέχει όλες τις δραστηριότητες του μαθητή που εκτελούνται κατά τη διάρκεια της μάθησης (σελίδες που έχει επισκεφτεί, αποτελέσματα των τεστ κλπ.).
- *Learner Models Database* περιέχει τη Γνωστική Κατάσταση και τις Μαθησιακές Προτιμήσεις για κάθε μαθητή.



Εικόνα 8: Εξωτερικό(α) και Εσωτερικό (β) Σύστημα Αρχιτεκτονικής ABITS

- *Users Database* περιέχει όλες τις πληροφορίες σχετικά με τους χρήστες του συστήματος (ονόματα, ρόλους, μαθήματα που έχει παρακολουθήσει κλπ.).
- *Courses database* περιέχει όλες τις πληροφορίες σχετικά με τους μαθησιακούς στόχων και το curriculum όλων των μαθημάτων που έχει παρακολουθήσει ο φοιτητής.

Αλληλεπιδράσεις μεταξύ βάσεων δεδομένων και λειτουργικών μονάδων (λειτουργική μονάδα που μπορεί να έχει πρόσβαση στα δεδομένα αυτά) απεικονίζονται στην εικόνα 8α όπου οι γραμμές αντιπροσωπεύουν τη ροή δεδομένων.

Όπως έχουμε ήδη πει, το ABITS εσωτερικά, έχει σχεδιαστεί ως σύστημα πολλαπλών πρακτόρων (Multi Agent System MAS). Το κίνητρο της επιλογής αυτής βασίζεται στην επεκτασιμότητα που μας επιτρέπει αυτή η αρχιτεκτονική. Πιο συγκεκριμένα το ABITS αποτελείται από τρία είδη πρακτόρων :

- *Evaluation agents*: αξιολογούν και ενημερώνουν το Μοντέλο του Μαθητή. Για να γίνει αυτό, αλληλεπιδρούν με τη metadata database, τη ABITS database, το Affective και το Pedagogical Agent.
- *Affective agents*: αξιολογούν και ενημερώνουν τις μαθησιακές προτιμήσεις του μαθητή. Για να γίνει αυτό, αλληλεπιδρούν με τη metadata database και τη ABITS database.
- *Pedagogical Agents*: αξιολογούν και ενημερώνουν τα curriculums. Για να γίνει αυτό, αλληλεπιδρούν με τη metadata database, τη ABITS database και το Courses Database.

Η εικόνα 8β δείχνει πώς οι πράκτορες αλληλεπιδρούν μεταξύ τους και με τις βάσεις δεδομένων και τις λειτουργικές μονάδες. Είναι σημαντικό να σημειωθεί ο αριθμός πολλαπλότητας που υπάρχει κοντά σε κάθε αντικείμενο. Ενώ το CMS και οι πηγές δεδομένων έχουν πολλαπλότητα 1 (μία μόνο συνιστώσα αυτού του είδους πρέπει να είναι παρών), κάθε πράκτορας έχει πολλαπλότητα 0 + (μηδέν ή περισσότεροι πράκτορες αυτού του είδους μπορεί να είναι παρόν). Αυτή είναι η ιδιότητα που προσφέρει επεκτασιμότητα και κλιμάκωση στην αρχιτεκτονική αυτή .

Κεφαλαίο 3

Σχεδιασμός και Υλοποίηση της Εφαρμογής μας

3.1 Ανάγκες και Στόχους που καλύπτει η εφαρμογή

Υπάρχουν πολλές σελίδες στο διαδίκτυο τα οποία προσφέρουν βοηθήματα για τα μαθηματικά Α δημοτικού, για παράδειγμα φύλλα εργασίας , διάφορα on-line ασκήσεις σε μορφή παιχνιδιών, on-line βιβλία κοκ. Ωστόσο δεν υπάρχουν προγράμματα –συστήματα που να υλοποιούν τα μαθηματικά κάθε αυτό . Με αυτό το εγχείρημα προσπάθησα να φτιάξω ένα Web-Based Environment-Application για τα μαθηματικά Α δημοτικού. Την εφαρμογή μπορεί να την κατεβάσουν ο δάσκαλος ή οι γονείς του μαθητή , να την εγκαταστήσουν και να τη παρουσιάσουν στον μαθητή. Έτσι με λίγο καθοδήγηση από τον εκπαιδευτικό ή τους γονείς ο μαθητής μπορεί να χρησιμοποιήσει την εφαρμογή για να διδαχθεί τις απαραίτητες έννοιες που χρειάζεται να γνωρίζει σε θεωρητικό υπόβαθρο για τα μαθηματικά και στη συνέχεια να λύσει τις ασκήσεις όταν του ζητείται . Η παρουσίαση του μαθήματος μπορεί να είναι ποικιλόμορφη , εικόνες , βίντεο και ηχογραφημένα κείμενα , διευκολύνοντας τον κάθε μαθητή ανάλογα αν είναι οπτικός ή ακουστικός τύπος. Απώτερος σκοπός της εφαρμογής αυτής είναι να επιτρέπει στους χρήστες να κάνουν το μάθημα τους , χωρίς να τους περιορίζει ο χρόνος και ο χώρος και να αποθηκεύει τα δεδομένα των μαθητών ώστε να εντοπίζει ποια είναι τα αδύναμα σημεία τους .

Με λίγα λόγια το σύστημα προσφέρει εξατομικευμένη εκπαίδευση στον χρήστη για να καλυφτούν οι προσωπικές του ανάγκες (π.χ. αδύναμα σημεία) στον τομέα των μαθηματικών, κάτι το οποίο δεν είναι εφικτό να καλυφτεί με τη παραδοσιακή εκπαίδευση (δε μπορούμε να προσφέρουμε έναν εκπαιδευτικό ανά μαθητή).

3.2 Τεχνολογίες

Πριν προχωρήσουμε στην ανάλυση του συστήματος μας θα αναφέρουμε τις τεχνολογίες που χρειαστήκαμε για να την υλοποίηση.

1.Netbeans

Το NetBeans είναι ένα open-source ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment IDE) για την Java, PHP, C + + και άλλες γλώσσες προγραμματισμού. Είναι επίσης γνωστό και ως μια πλατφόρμα αποτελούμενη από αρθρωτά εξαρτήματα που χρησιμοποιούνται για την

ανάπτυξη εφαρμογών Java desktop. Επιπλέον είναι γραμμένο σε Java και τρέχει στα περισσότερα συστήματα που λειτουργούν με Java Virtual Machine (JVM), συμπεριλαμβανομένων των Solaris, Mac OS και Linux.

Διαχειρίζεται τα ακόλουθα χαρακτηριστικά και συστατικά πλατφόρμας:

- User settings
- User interface
- Windows (placement, appearance, etc.)
- NetBeans Visual Library
- Storage
- Integrated development tools
- Framework wizard

Επιπλέον η πλατφόρμα NetBeans επιτρέπει στις εφαρμογές να αναπτυχθούν από ένα σύνολο ανεξάρτητων κομματιών λογισμικού που ονομάζεται ενότητες. Οι εφαρμογές που βασίζονται στην πλατφόρμα NetBeans μπορεί να επεκταθούν από τρίτους προγραμματιστές .

Τέλος οι NetBeans IDE ενότητες περιλαμβάνουν NetBeans Profiler, μια γραφική διεπαφή χρήστη (GUI) εργαλείο σχεδιασμού και το NetBeans JavaScript Editor.

2. Τεχνολογίες Web

Μια διαδικτυακή εφαρμογή (Web Application) είναι μια εφαρμογή το οποίο είναι προσβάσιμο μέσω ενός δικτύου όπως το Internet ή ένα intranet. Ο όρος μπορεί επίσης να σημαίνει μια λογισμική εφαρμογή υπολογιστή που φιλοξενείται σε ένα browser-controlled περιβάλλον. Για να δημιουργήσουμε μια web εφαρμογή χρησιμοποιούμε web-τεχνολογίες.

Οι web-τεχνολογίες είναι οι τεχνικές ή τεχνολογίες που χρησιμοποιούνται για στην ανάπτυξη διαδικτύου (web).

Οι τεχνολογίες Web σχετίζονται με τη διασύνδεση μεταξύ των web servers και των πελατών τους. Αυτές οι τεχνολογίες περιλαμβάνουν markup languages (γλώσσες σήμανσης), programming interfaces and languages (διεπαφές και γλώσσες προγραμματισμού) και πρότυπα για την αναγνώριση και εμφάνιση του εγγράφων.

Υπάρχουν τόσες πολλές γλώσσες που σχετίζονται με τις τεχνολογίες του διαδικτύου, που μια πλήρη λίστα τους θα είναι πολλές σελίδες. Παρακάτω παρουσιάζουμε μια λίστα με πολύ δημοφιλείς γλώσσες και πού χρησιμοποιείται το καθένα:

- HTML, XHTML και XML είναι markup γλώσσες, δηλαδή γλώσσες για τον σχεδιασμό της σελίδας.

- JavaScript και AJAX χρησιμοποιούνται συχνά για client-side διεργασίες, οι επιδράσεις και οι επιπτώσεις τους σχετίζονται με κινήσεις του ποντικιού.
- PHP , ASP.Net και JSP είναι δημοφιλείς ως server-side γλώσσες, πράγμα που σημαίνει ότι χρησιμοποιούνται για να οικοδομήσουμε μια σελίδα ανάλογα με τα cookies, τις ρυθμίσεις του browser του χρήστη ή του session status (= είναι συνδεδεμένος ή όχι, κατηγορία χρήστη, προνόμια χρήσης). Χρησιμοποιούνται επίσης για την επικοινωνία μεταξύ ιστοσελίδας και βάσης δεδομένων.
- MySQL και η Oracle είναι γνωστές τεχνολογίες διαχείρισης βάσεων δεδομένων.

Οι web-τεχνολογίες που έχω χρησιμοποιήσει στο πρόγραμμα μου είναι :

- **HyperText Markup Language (HTML)** είναι η επικρατέστερη γλώσσα σήμανσης για τις ιστοσελίδες. Τα HTML στοιχεία είναι οι βασικές δομικές μονάδες της κάθε ιστοσελίδας. Όλα τα HTML στοιχεία είναι γραμμένα μέσα σε ετικέτες, εντός των αγκύλων <html> </html>, στο περιεχόμενο της ιστοσελίδας. Τα HTML tags (ετικέτες) εμφανίζονται πιο συχνά σε ζεύγη, όπως <h1> και </ h1>, αν και μερικές ετικέτες, που είναι γνωστά ως κενά στοιχεία, είναι αταίριαστα, για παράδειγμα . Η πρώτη ετικέτα σε ένα ζευγάρι είναι η ετικέτα έναρξη και η δεύτερη ετικέτα είναι η ετικέτα τέλους (ονομάζονται επίσης ετικέτες ανοίγματος και κλεισίματος). Μεταξύ αυτών των ετικετών σχεδιαστές ιστοσελίδων μπορούν να προσθέσουν κείμενο, ετικέτες, σχόλια και άλλα είδη text-based περιεχόμενο.

Ο σκοπός ενός προγράμματος περιήγησης (web browser) στο Web είναι να διαβάσει τα HTML έγγραφα και να τα συνθέσει σε ορατές ή ακουστικές ιστοσελίδες. Το πρόγραμμα περιήγησης δεν εμφανίζει τις ετικέτες HTML, αλλά χρησιμοποιεί τις ετικέτες για να ερμηνεύσει το περιεχόμενο της σελίδας.

Τα HTML στοιχεία αποτελούν τα δομικά στοιχεία όλων των ιστοσελίδων. Το HTML επιτρέπει τις εικόνες και τα αντικείμενα να είναι ενσωματωμένα και μπορεί να χρησιμοποιηθούν για να δημιουργήσουμε διαδραστικές φόρμες. Επίσης αποτελεί ένα μέσο για να δημιουργήσουμε δομημένα έγγραφα, χρησιμοποιώντας σημασιολογικές δομές για κείμενο, όπως επικεφαλίδες, παράγραφοι, λίστες, συνδέσεις, εισαγωγικά και άλλα στοιχεία. Μπορεί επίσης να ενσωματώσει scripts τα οποία έχουν δημιουργηθεί με JavaScript και επηρεάζουν τη συμπεριφορά των HTML ιστοσελίδων .

Τα προγράμματα περιήγησης στο Web μπορεί επίσης να παραπέμπουν σε Cascading Style Sheets (CSS) για να καθορίσουν την εμφάνιση και τη διάταξη του κειμένου και άλλου υλικού.

- **JavaServer Pages (JSP)** είναι μια τεχνολογία Java που βοηθά τους προγραμματιστές να δημιουργήσουν δυναμικές ιστοσελίδες τα οποία βασίζονται σε HTML, XML, ή άλλους τύπους εγγράφων. Κυκλοφόρησε το 1999 ως απάντηση της Sun στην ASP και PHP. Το JSP σχεδιάστηκε για να παρέχει υποστήριξη Java περιβάλλον προγραμματισμού για το Διαδίκτυο. Αρχιτεκτονικά, το JSP μπορεί να θεωρηθεί ως μια υψηλού επιπέδου αφαίρεση των Java servlets. JSPs φορτώνονται στο server και διευθύνονται από ένα δομημένο ειδικό πακέτο Java, που λέγεται Java EE Web Application. Το JSP επιτρέπει να γράψουμε κώδικα Java και ορισμένες προκαθορισμένες ενέργειες μέσα στο στατικό περιεχόμενο ιστοσελίδων σήμανσης, με αποτέλεσμα η σελίδα να εκτελείται στο server για να παραδώσει ένα HTML ή XML έγγραφο.

- **Servlet** είναι μια [Java programming language class](#), που χρησιμοποιείται για να επεκτείνει τις δυνατότητες των servers που φιλοξενούν τις εφαρμογές που είναι προσβάσιμες μέσω ενός μοντέλου προγραμματισμού request – response (αίτημα-απάντηση). Αν και τα servlets μπορούν να ανταποκριθούν σε κάθε τύπο αιτήματος, τα μέρη που χρησιμοποιούνται συνήθως είναι για την επέκταση των εφαρμογών που φιλοξενείται από τους Web servers. Έτσι, μπορεί να θεωρηθεί ως μια βοηθητική εφαρμογή Java που τρέχει σε ένα server (server-side application) αντί σε ένα πρόγραμμα περιήγησης.

- **Η JavaScript** είναι [γλώσσα προγραμματισμού](#) η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του πελάτη (client-side) σε [ιστοσελίδες](#). Η αρχική έκδοση της Javascript βασίστηκε στη [σύνταξη](#) στη [γλώσσα προγραμματισμού C](#), αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες. Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του [πελάτη](#) (client), που ήταν ο [φυλλομετρητής](#) (browser) του χρήστη και χαρακτηρίστηκε σαν *client-side γλώσσα προγραμματισμού*. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου [HTML](#) δεν πραγματοποιείται στο διακομιστή, αλλά στο [πρόγραμμα περιήγησης](#) των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες [HTML](#). Αντίθετα, άλλες γλώσσες όπως η [PHP](#) εκτελούνται στο [διακομιστή](#) (*server-side γλώσσες προγραμματισμού*).

- **AJAX** (Asynchronous JavaScript και XML) είναι μια ομάδα αλληλοσχετιζόμενων μεθόδων ανάπτυξης ιστοσελίδων που χρησιμοποιούνται στην πλευρά του πελάτη για να δημιουργήσουν ασύγχρονα web εφαρμογές. Με τη Ajax, web εφαρμογές μπορούν να

στέλνουν τα δεδομένα σε, και να ανακτήσετε δεδομένα από, ένα διακομιστή ασύγχρονα, χωρίς να παρεμβαίνει στην εμφάνιση και τη συμπεριφορά της υπάρχουσας σελίδας. Τα δεδομένα συνήθως ανακτώνται με το αντικείμενο XMLHttpRequest.

Το Ajax δεν είναι μια τεχνολογία, αλλά μια ομάδα τεχνολογιών. HTML και CSS μπορεί να χρησιμοποιηθούν σε συνδυασμό για τη σήμανση και το στυλ. Το DOM είναι προσβάσιμο με τη JavaScript, για να εμφανίζει δυναμικά και να επιτρέπει στο χρήστη να αλληλεπιδράσει με τις πληροφορίες που παρουσιάζονται. JavaScript και το αντικείμενο XMLHttpRequest παρέχουν μια μέθοδο για την ανταλλαγή δεδομένων ασύγχρονα μεταξύ του browser και του server, αποφεύγοντας έτσι να ξαναφορτωθεί η σελίδα.

- **MySQL** είναι ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων, που λειτουργεί ως server παρέχοντας multi-user πρόσβαση σε μια σειρά από βάσεις δεδομένων. Το SQL σημαίνει Structured Query Language. Έργα ανοικτού κώδικα που απαιτούν ένα πλήρως εξοπλισμένο σύστημα διαχείρισης βάσεων δεδομένων συχνά χρησιμοποιούν MySQL. Για εμπορική χρήση, αρκετές εκδόσεις είναι διαθέσιμες και προσφέρουν επιπλέον λειτουργικότητα. MySQL χρησιμοποιείται επίσης σε πολλές υψηλού προφίλ, μεγάλης κλίμακας World Wide Web προϊόντα, συμπεριλαμβανομένης της Wikipedia, Google (αν και όχι για τις αναζητήσεις), το Facebook και το Twitter.

3.3 Εισαγωγή στην εφαρμογή

Το web – application μου δεν πλήρη όλες τις προδιαγραφές ενός ITS συστήματος (δε περιέχει όλες τις βασικές ενότητες που πρέπει να έχει ένα ITS : student model module, user interface module, tutoring module και expert knowledge module), αλλά κύριος στόχος της είναι να λειτουργεί προσαρμοστικά για κάθε μαθητή. Η προσαρμοστικότητα υλοποιείται ως εξής:

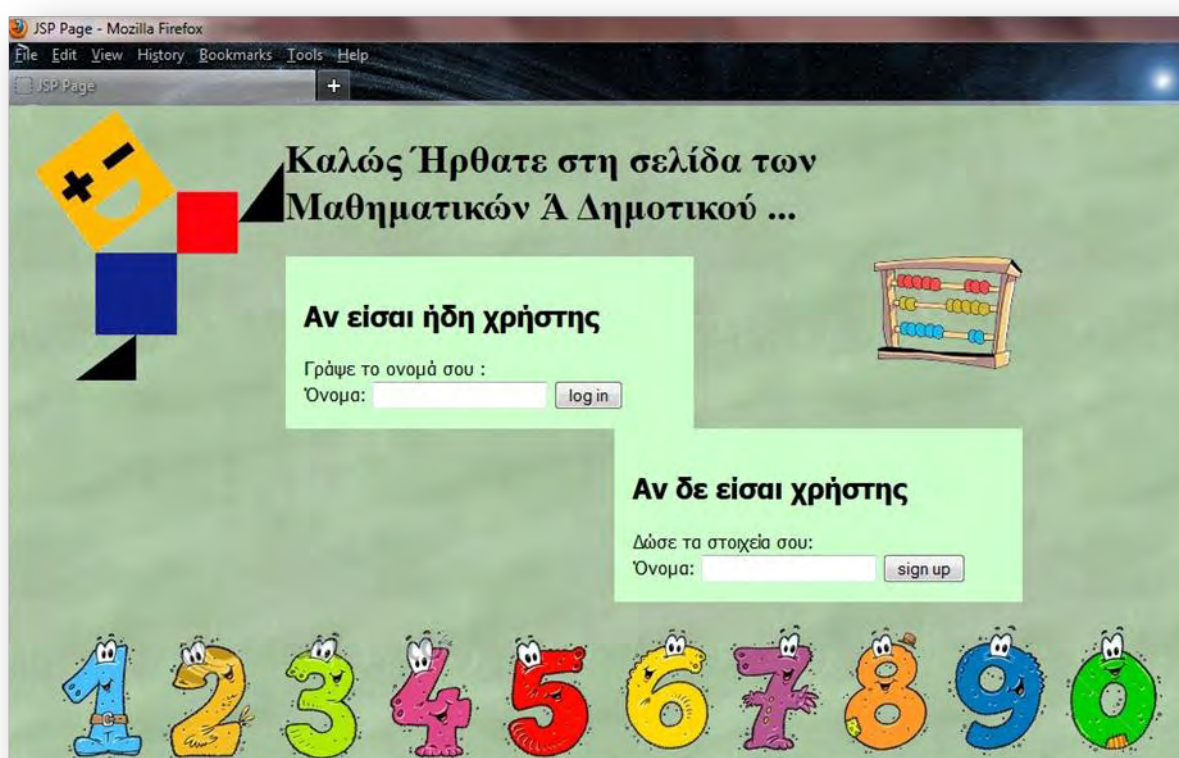
1. Αρχικά το σύστημα δε γνωρίζει τον μαθητή (είναι πρώτη φορά που ο μαθητής χρησιμοποιεί το πρόγραμμα) οπότε του παρουσιάζει τα μαθήματα με στατικό τρόπο (ενότητα πρώτη, ενότητα δεύτερη ... κοκ). Η ύλη του μαθήματος είναι οργανωμένη σε διδακτικές ενότητες, που το καθένα από αυτά καλύπτει μία έννοια – αρχή των μαθηματικών, πχ η πρώτη ενότητα αναφέρεται στον προσανατολισμό στο χώρο, η δεύτερη αναφέρεται στα γεωμετρικά σχήματα ... κοκ.
2. Μετά την ολοκλήρωση της κάθε ενότητας παρουσιάζονται στον χρήστη κάποιες ασκήσεις τις οποίες πρέπει να λύσει, ώστε το σύστημα να

αξιολογήσει και να εξετάσει αν ο μαθητής κατανόησε τη σχετική διδακτική έννοια (αν οι απαντήσεις του είναι σωστές τότε το σύστημα θεωρεί ότι κατανόησε την έννοια). Αφού ολοκληρωθούν οι ασκήσεις τα αποτελέσματα αποθηκεύονται στη βάση δεδομένων στο ανάλογο πίνακα.

3. Την επόμενη φορά που ο μαθητής θα ξανά επισκεφτεί τα σύστημα ,εκείνο θα ελέγξει ποιες διδακτικές (μαθηματικές) έννοιες δε έχουν καλυφτεί από τον μαθητή και θα εμφανίσει μόνο τις ενότητες εκείνες που αφορούν τις συγκεκριμένες έννοιες/αρχές. Αυτήν την φορά οι ασκήσεις που θα πρέπει να λύσει ο μαθητής μπορεί να περιέχουν επεξηγήσεις και βοηθήματα ,ώστε να ξεπεραστούν οι αδυναμίες – τα κενά που χει ο μαθητής και έκανε λάθος σε αυτά.

3.3.1 Αρχική σελίδα

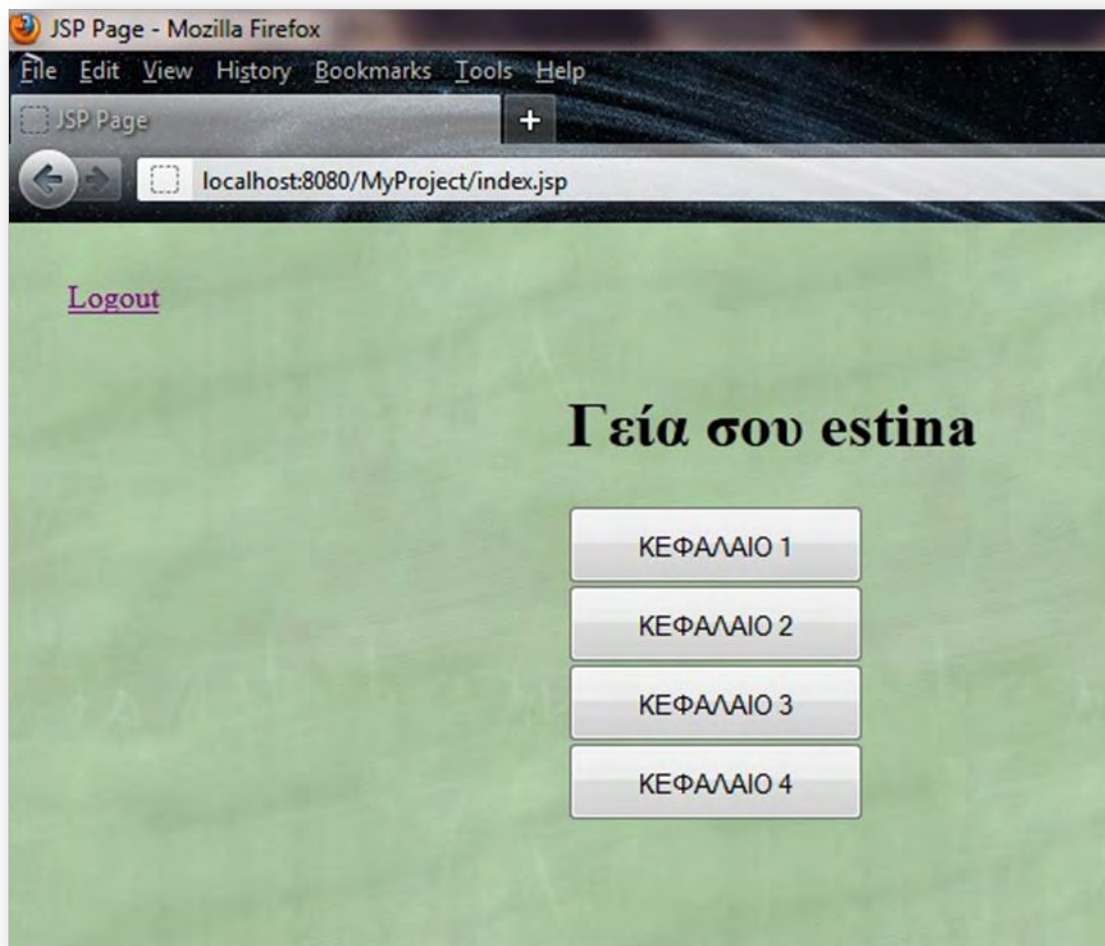
Στην αρχική ιστοσελίδα εμφανίζονται δύο φόρμες , μία που ο μαθητής συμπληρώνει το όνομα του (αν είναι ήδη χρήστης) και μία που θα δώσει τα στοιχεία του για να κάνει εγγραφή (αν δεν είναι χρήστης). Να σημειώσουμε ότι κάθε χρήστης πρέπει να είναι εγγεγραμμένος στο σύστημα , ώστε το σύστημα να είναι σε θέση να κρατήσει τα αποτελέσματα του στην βάση δεδομένων. Η ιστοσελίδα είναι η εξής :



Εικόνα 9: Αρχική Σελίδα

3.3.2 Υλη του μαθήματος

Αφού ο μαθητής πληκτρολογήσει το όνομα του, το σύστημα του παρουσιάζει την ύλη των μαθηματικών που είναι οργανωμένη ανά κεφάλαιο.



Εικόνα 10 : Υλη του μαθήματος

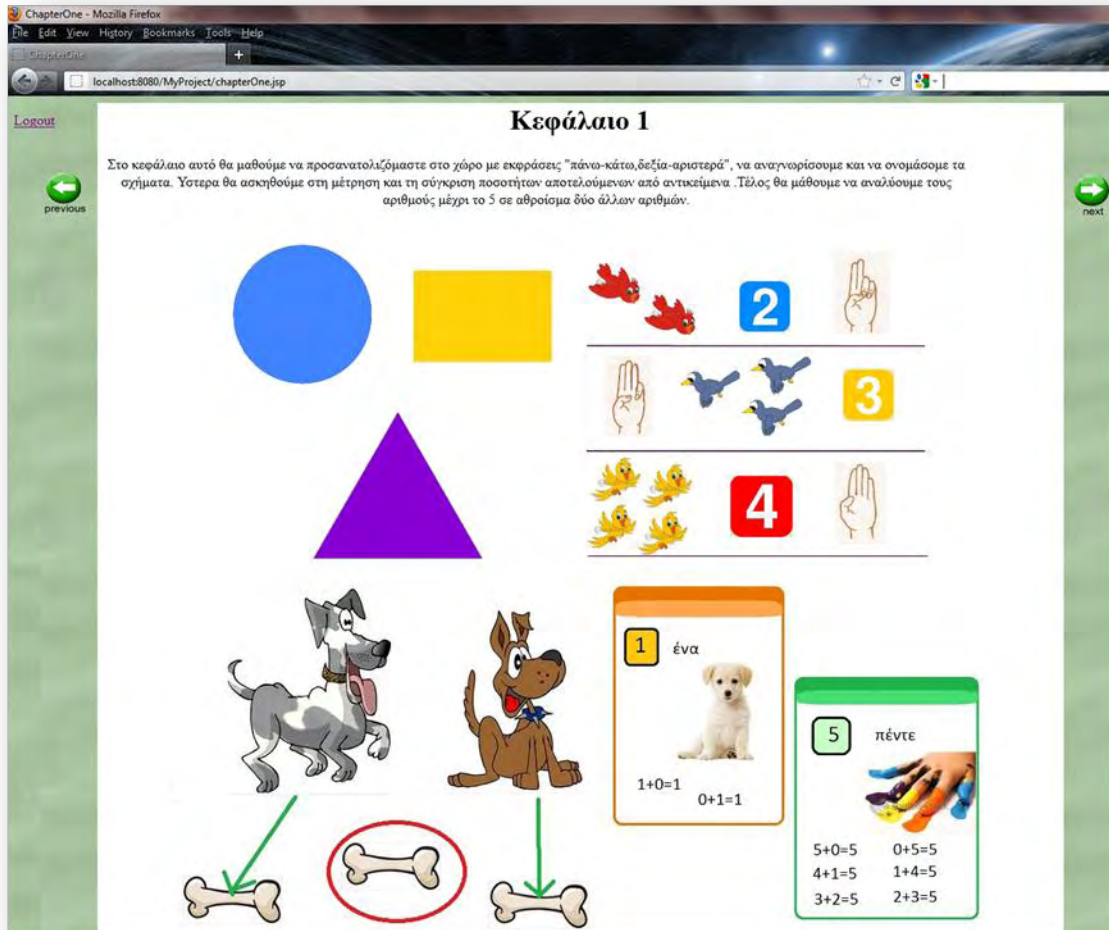
3.3.3 Κεφάλαιο 1

Σε αυτήν την ενότητα θα δούμε κάποιο δείγμα από το κεφάλαιο ένα. Κάθε κεφάλαιο αποτελείται από την θεωρία και τις ασκήσεις. Η ύλη αυτή είναι οργανωμένη σε μαθηματικές έννοιες/αρχές (ή αλλιώς μαθησιακοί στόχοι), τις οποίες και πρέπει να μάθει ο μαθητής. Το κεφάλαιο ένα περιέχει 6 μαθηματικές έννοιες :

1. Προσανατολισμός στο χώρο
2. Γεωμετρικά σχήματα
3. Συγκρίση και εκτίμηση ποσοτήτων
4. Οι αριθμοί από το 1 έως το 5

- 5.Αριθμηση και Ανάγνωση των αριθμών
- 6.Πρόσθεση και Ανάλυση των αριθμών μέχρι το 5

Στην εικόνα 11 φαίνεται η εισαγωγική σελίδα του πρώτου κεφαλαίου.



Εικόνα 11: Εισαγωγική σελίδα 1^{ου} κεφαλαίου

1. Θεωρία

Παρακάτω (εικόνα 12) επέλεξα να δείξω σαν παράδειγμα την έννοια 6 (Πρόσθεση και Ανάλυση των αριθμών μέχρι το 5). Θα δούμε ένα από τα παραδείγματα που προσπαθεί να επεξηγήσει στον μαθητή την έννοια.

6.AdditionAndAnalysis - Mozilla Firefox
 local:localhost:8080/MyProject/6_AdditionAndAnalysis.jsp

Logout previous next

6.Πρόσθεση και ανάλυση των αριθμών μέχρι το 5 .

Ανάλυση των αριθμών μέχρι 5 ..

1 ένα

 $1+0=1$ $0+1=1$

2 δύο

 $2+0=2$ $0+2=2$
 $1+1=2$

3 τρία

 $3+0=3$ $0+3=3$
 $2+1=3$ $1+2=3$

4 τέσσερα

 $4+0=4$ $0+4=4$
 $3+1=4$ $1+3=4$
 $2+2=4$

5 πέντε

 $5+0=5$ $0+5=5$
 $4+1=5$ $1+4=5$
 $3+2=5$ $2+3=5$

Εικόνα 12: Παράδειγμα Θεωρίας

Όπως βλέπουμε κάθε εικόνα δείχνει έναν αριθμό συνοδευόμενο από τα αντίστοιχα αντικείμενα και από κάτω την μαθηματική ανάλυση του.

2. Άσκηση


Αφού ο μαθητής έχει μελετήσει όλα τα παραδείγματα, στο τέλος της διδακτικής ενότητας υπάρχει μία άσκηση(ή ασκήσεις) , την οποία καλείται να λύσει . Συμπεραίνουμε ότι ο μαθητής κάλυψε μία συγκεκριμένη έννοια αν οι απαντήσεις των ασκήσεων που αφορούν την έννοια αυτή είναι σωστές . Ακολουθεί μία άσκηση που αφορά την έννοια την οποία δείξαμε παραπάνω.

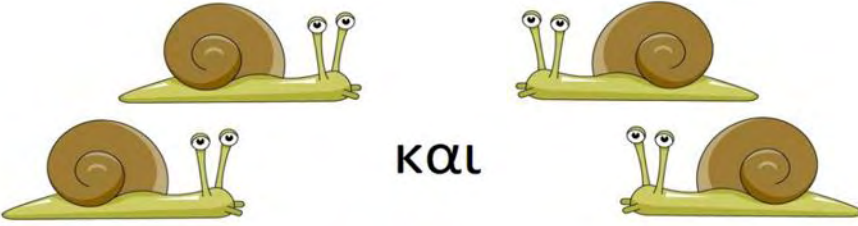




6_AdditionAndAnalysis - Mozilla Firefox
 File Edit View History Bookmarks Tools Help
 localhost:8080/MyProject/6_AdditionAndAnalysis.jsp

Logout

Ωρα για ασκήσεις...

previous next


 Παρατηρώ τις εικόνες και γράφω πόσα είναι όλα κάθε φορά .
 Τους αριθμούς τους γράφουμε στο κόκκινο μικρό κουτάκι .

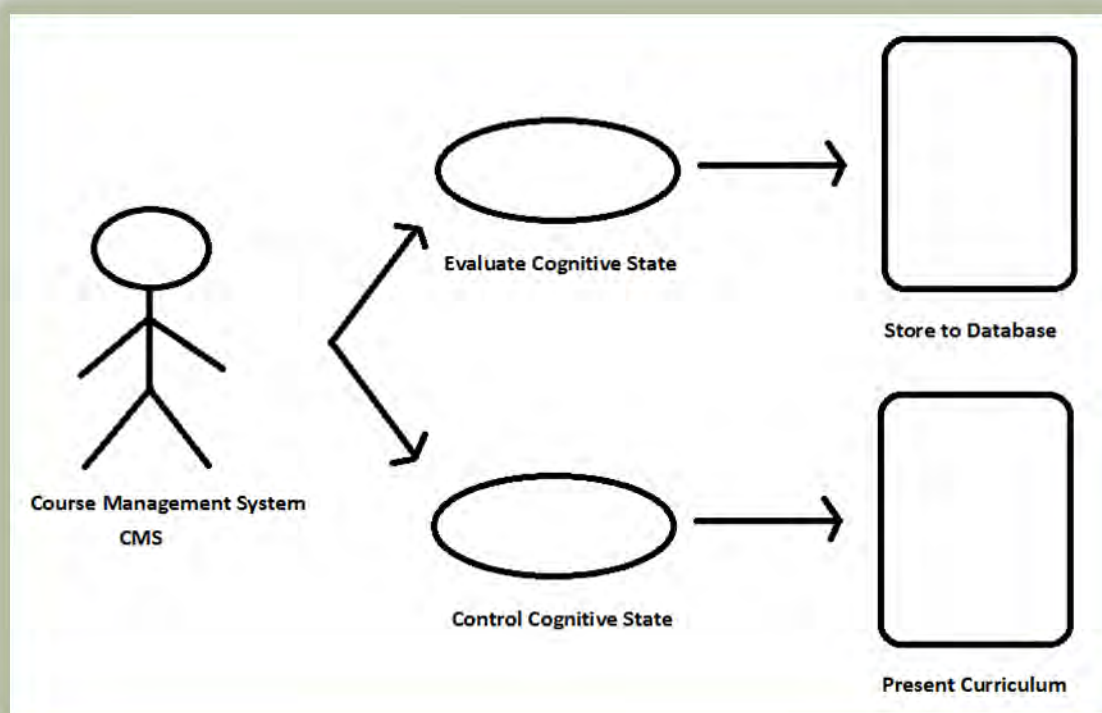
	και <input type="text"/>
	και  <input type="text"/>
	και  <input type="text"/>

Εικόνα 13: Παράδειγμα Άσκησης

3.4 Δομή και περιεχόμενα της εφαρμογής

Όπως αναφέραμε στο κεφάλαιο 2 στην ενότητα 2.5 , όπου αναλύουμε το ABITS, η καλύτερη λύση για ένα αποτελεσματικό σύστημα εκπαίδευσης ,είναι να βασίζεται στο web καθώς παρέχει στους τελικούς χρήστες τη μέγιστη ευελιξία χρόνου και χώρου. Επίσης αναφέραμε ότι υπάρχουν 3 είδη Web-Based Tutoring Systems WBT, στατική ,εξατομικευμένη και προσαρμοστική WBT. Η εφαρμογή μας ανήκει στην κατηγορία των εξατομικευμένων WBT, όπου το σύστημα ,χρησιμοποιώντας ένα ειδικό λογισμικό που ονομάζεται Σύστημα Διαχείρισης μαθημάτων (Course Management System) , εκτελεί δύο κυρίως λειτουργίες. Πρώτον, παρακολουθεί τις γνώσεις των μαθητών, εξετάζοντας τους και δεύτερον ορίζει διαφορετικά μονοπάτια για τα αντικείμενα μάθησης ώστε να καλυφτούν διαφορετικοί μαθησιακοί στόχοι.

Οι ευφυής λειτουργίες της εφαρμογής μας φαίνεται στη εικόνα 14.



Εικόνα 14: Διάγραμμα ροής της εφαρμογής

Για να επεξηγήσουμε τις λειτουργίες αυτές είναι αναγκαίο να μελετήσουμε την δομή της βάσης δεδομένων μας (ενότητα 3.4.1) και τα αρχεία της εφαρμογής μας (ενότητα 3.4.2).

3.4.1 Βάση Δεδομένων

Η ανάγκη δημιουργίας βάσης προέκυψε ώστε να μοντελοποιήσουμε την γνωστική κατάσταση (Cognition State) του μαθητή. Έτσι ο κύριος στόχος της είναι να αποθηκεύει τα δεδομένα του μαθητή, για να καθορίσει τα μονοπάτια των αντικειμένων μάθησης που θα μελετήσει ο συγκεκριμένος μαθητής (εξατομικευμένη μάθηση).

1. Ορισμός

Μια βάση δεδομένων είναι μια οργανωμένη συλλογή των δεδομένων για ένα ή περισσότερους λόγους, συνήθως σε ψηφιακή μορφή. Τα δεδομένα οργανώνονται συνήθως για να μοντελοποιήσουν σχετικές πτυχές της πραγματικότητας (για παράδειγμα, τη διαθεσιμότητα των δωματίων σε ξενοδοχεία), με τρόπο που να υποστηρίζουν τις διαδικασίες που απαιτούν αυτές τις πληροφορίες (για παράδειγμα, η εύρεση ενός ξενοδοχείου με κενές θέσεις). Αυτός ο ορισμός είναι πολύ γενικός και είναι ανεξάρτητος από τη χρησιμοποιούμενη τεχνολογία. Οι περισσότερες βάσεις δεδομένων περιέχουν πολλούς πίνακες, που μπορεί να περιλαμβάνουν πολλά διαφορετικά πεδία. Για παράδειγμα, μια βάση δεδομένων εταιρείας μπορεί να περιλαμβάνει πίνακες για τα προϊόντα, τους εργαζομένους, καθώς και για τα οικονομικά στοιχεία. Κάθε ένας από αυτούς τους πίνακες θα έχει διαφορετικά πεδία που είναι σχετικές με τις πληροφορίες που είναι αποθηκευμένες στον πίνακα αυτό.

Με το σύστημα διαχείρισης βάσεων δεδομένων (ή DBMS) μπορούμε εύκολα να έχουμε πρόσβαση, να διαχειριστούμε και να ενημερώσουμε τις πληροφορίες αυτές.

2. Microsoft Office Access

Το Microsoft Office Access, παλαιότερα γνωστό ως Microsoft Access, είναι ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων της Microsoft που συνδυάζει τους σχεσιακούς μηχανισμούς βάσης δεδομένων με μια γραφική διεπαφή χρήστη και τα εργαλεία λογισμικού ανάπτυξης.

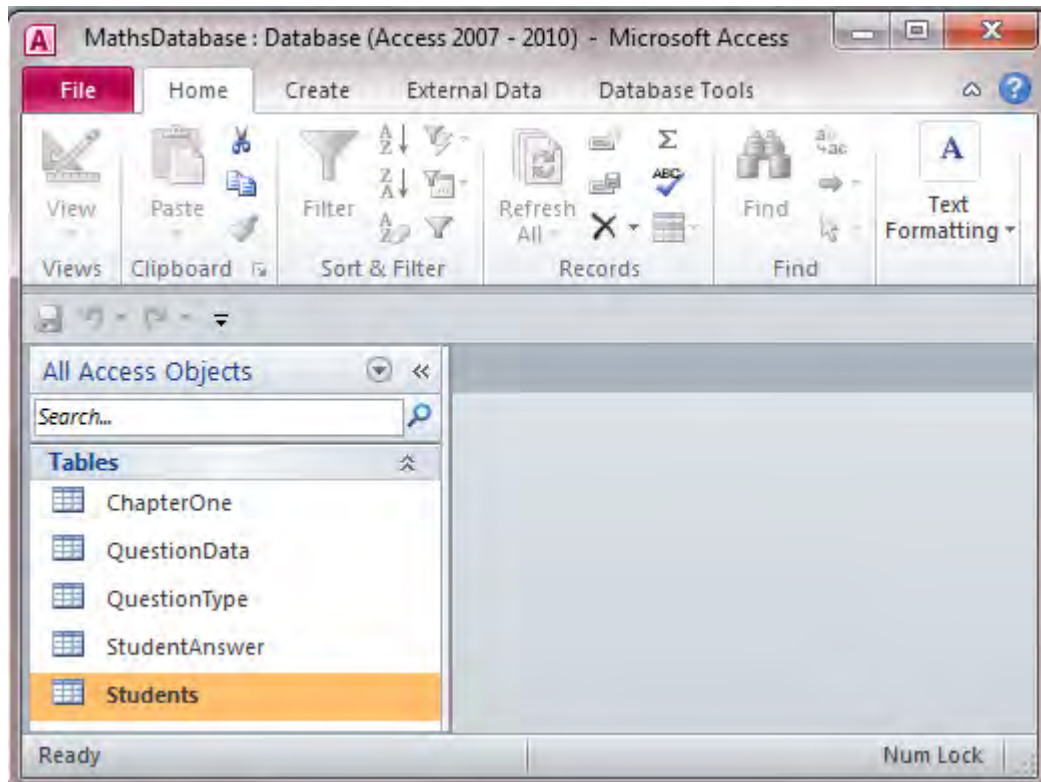
MS Access αποθηκεύει τα δεδομένα στη δική της μορφή με βάση το Access Jet Database Engine. Μπορεί επίσης να εισάγει ή να συνδεθεί απευθείας σε δεδομένα που είναι αποθηκευμένα σε άλλες εφαρμογές και βάσεις δεδομένων.

3. Το δομικό στοιχείο της βάσης δεδομένων

Το δομικό στοιχείο της βάσης δεδομένων είναι ο πίνακας. Ο κάθε πίνακας αποτελείται από στήλες-πεδία, που περιέχουν μια συγκεκριμένη κατηγορία δεδομένων. Παρακάτω θα δείξουμε και θα εξηγήσουμε την χρησιμότητα των πινάκων της δικής μας βάσης.

Όπως παρατηρούμε στη εικόνα 15 η βάση μας έχει 6 πίνακες :

Σε κάθε πίνακα το πρώτο πεδίο (του οποίου το όνομα είναι ID) έχει την ιδιότητα του πρωτεύον κλειδί (primary key). Το πρωτεύον κλειδί ενός πίνακα προσδιορίζει μοναδικά την κάθε εγγραφή στον πίνακα. Μπορεί να είναι είτε ένα κανονικό χαρακτηριστικό που είναι εγγυημένο ότι είναι μοναδικό (όπως ο αριθμός κοινωνικής ασφάλισης) ή μπορεί να παράγεται από το DBMS . Στη δική μας περίπτωση παράγεται από το DBMS (είναι auto number).



Εικόνα 15: Η βάση δεδομένων

1. Students

Ο πίνακας Students έχει 2 πεδία-στήλες, το ID και το StudentName. Το StudentName αποθηκεύει το όνομα του μαθητή. Ο μαθητής καταχωρείται στον πίνακα αυτό μόνο όταν δώσει το όνομα του στη φόρμα της εγγραφής (sing up form) στην αρχική σελίδα (βλέπε 3.2.1 ενότητα). Αφού ο μαθητής κάνει log in, πληκτρολογώντας το όνομα του, το http session αποθηκεύει το όνομα αυτό σε μια δική του μεταβλητή και έτσι το σύστημα ξέρει ποιος εμπλέκεται κάθε φορά.

2. QuestionType

Ο πίνακας QuestionType έχει 2 πεδία το ID (αναγνωριστικό) και το Type. Η ύλη των μαθηματικών είναι οργανωμένη σε μαθησιακούς στόχους (πχ προσανατολισμός, γεωμετρικά σχήματα, πρόσθεση κοκ) , τα οποία

αποθηκεύονται στο πεδίο Type. Τα δεδομένα του πίνακα αυτού δε συμπληρώνονται από την εφαρμογή αλλά από τον administrator.

3. QuestionData

Ο πίνακας QuestionData έχει 4 πεδία το ID, το QuestionTypeID, το Question και το Answer και χρησιμοποιείται για να αποθηκεύσουμε τις ερωτήσεις των ασκήσεων, που θα υποβληθούν στον μαθητή. Το πεδίο QuestionTypeID αναφέρεται στο ID του πίνακα QuestionType και συνδέει την κάθε ερώτηση με κάποιο μαθησιακό στόχο. Το πεδίο Question αποθηκεύει την ερώτηση (πχ Βρες πόσα είναι όλα τα αντικείμενα;) και τέλος το πεδίο Answer αποθηκεύει την απάντηση που αντιστοιχεί σε κάθε ερώτηση. Επίσης τα δεδομένα αυτού του πίνακα δε συμπληρώνονται από την εφαρμογή αλλά από τον administrator.

4. StudentAnswer

Ο πίνακας StudentAnswer έχει 7 πεδία το ID, το StudentID, το QuestionID, το Answer, το QuestionTypeID, το WrongRight και το RightAnswer. Το πεδίο StudentID και το πεδίο QuestionID αναφέρονται στα ID των πινάκων Students και QuestionData αντίστοιχα. Το πεδίο Answer αποθηκεύει την απάντηση που έδωσε ο μαθητής στην ερώτηση-άσκηση που του υποβλήθηκε. Το πεδίο QuestionTypeID αναφέρεται στο ID του πίνακα QuestionType. Το πεδίο WrongRight αποθηκεύει αν ο μαθητής πέτυχε το μαθησιακό στόχο που συνδέεται με τη συγκεκριμένη άσκηση, αν απαντήσει σωστά τότε το πέτυχε (right) αλλιώς όχι (wrong). Πρέπει να σημειώσουμε ότι σε αυτό το σημείο πραγματοποιείται η πρώτη ευφυής λειτουργία αξιολόγησης της γνωστικής κατάστασης (**Evaluate Cognition State**) του μαθητή. Το πεδίο RightAnswer αναφέρεται στο πεδίο Answer του πίνακα QuestionData. Τα δεδομένα του πίνακα αυτού συμπληρώνονται από την εφαρμογή, πιο συγκεκριμένα μία εγγραφή λαμβάνει χώρα κάθε φορά που ο μαθητής πατάει submit στο τέλος της άσκησης.

5. ChapterOne

Ο πίνακας ChapterOne έχει 8 πεδία το ID, το StudentID, το Orientation, το Shapes, το ComparisonAndEstimation, το Numbers1to5, το NumberingAndReading και το AdditionAndAnalysis. Το πεδίο StudentID αναφέρεται στο ID του πίνακα Students. Τα υπόλοιπα πεδία αναφέρονται στους μαθησιακούς στόχους του πρώτου κεφαλαίου. Αυτά τα πεδία αρχικοποιούνται όλα με true όταν ένας μαθητής κάνει εγγραφή στο σύστημα, το οποίο σημαίνει ότι ο μαθητής δε έχει καλύψει κανέναν από αυτούς τους μαθησιακούς στόχους. Όταν ο μαθητής απαντήσει σωστά σε όλες τις ασκήσεις που αφορούν έναν μαθησιακό στόχο τότε το πεδίο του

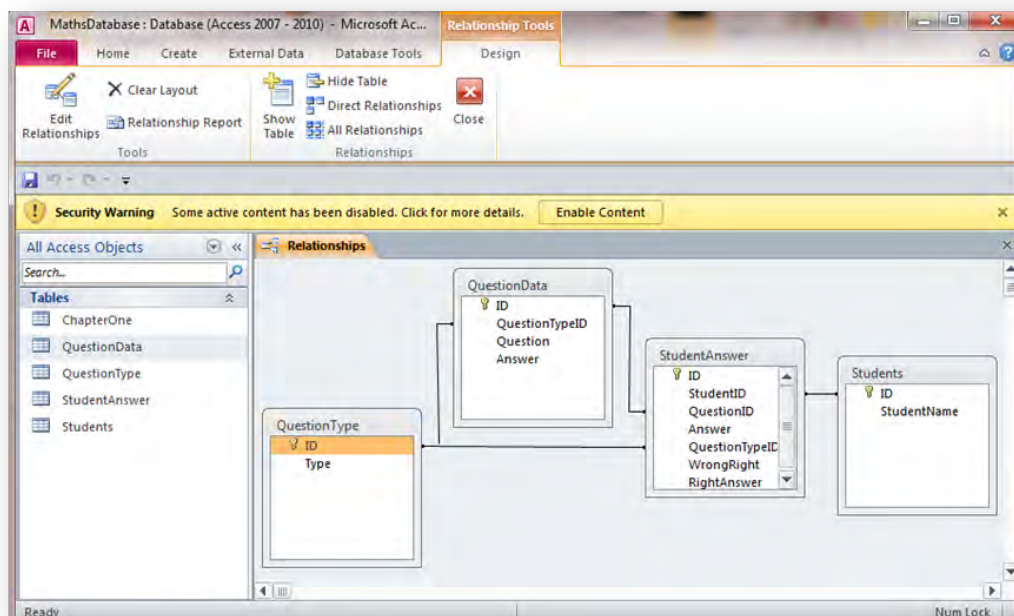
στόχου αυτού γίνεται false.

Ο κάθε στόχος αντιστοιχεί σε ένα μαθησιακό αντικείμενο. Με τον όρο μαθησιακό αντικείμενο εννοούμε μια σελίδα διαδικτύου (στην εφαρμογή μας η σελίδα αυτή είναι JSP Java Server Page). Στόχος του μαθησιακού αντικειμένου είναι να διευκρινίσει πως ο μαθητής θα μάθει τον αντίστοιχο μαθησιακό στόχο. Έτσι η σελίδα του μαθησιακού αντικειμένου αποτελείται από την θεωρία (ενότητα 3.2.3.1) και τις ασκήσεις (ενότητα 3.2.3.2). Τέλος να αναφέρουμε ότι όταν το πεδίο ενός μαθησιακού στόχου είναι true τότε εμφανίζεται και η αντίστοιχη σελίδα μαθησιακού αντικειμένου στον μαθητή, αλλιώς όχι.

4. Σχέσεις μεταξύ πινάκων (Relationships)

Για να υπάρξουν σχέσεις μεταξύ πινάκων, πεδία του ενός πίνακα αναφέρονται σε πεδία άλλων πινάκων, χρειαζόμαστε το ξένο κλειδί (foreign key).

Στο πλαίσιο των σχεσιακών βάσεων δεδομένων, ένα ξένο κλειδί είναι αναφορικός περιορισμός ανάμεσα σε δύο πίνακες. Ένα ξένο κλειδί είναι ένα πεδίο στους σχεσιακούς πίνακες που αντιστοιχεί σε ένα υποψήφιο κλειδί ενός άλλου πίνακα. Το ξένο κλειδί μπορεί να χρησιμοποιηθεί για παραπομπή σε πίνακες. Παρακάτω παρουσιάζουμε τις σχέσεις των πινάκων της βάσης μας.



Εικόνα 16: Σχέσεις μεταξύ πινάκων

Έστω οι δύο πίνακες, ο πίνακας Students που περιλαμβάνει όλους τους μαθητές και ο πίνακας StudentAnswer που περιλαμβάνει όλες τις απαντήσεις των μαθητών.

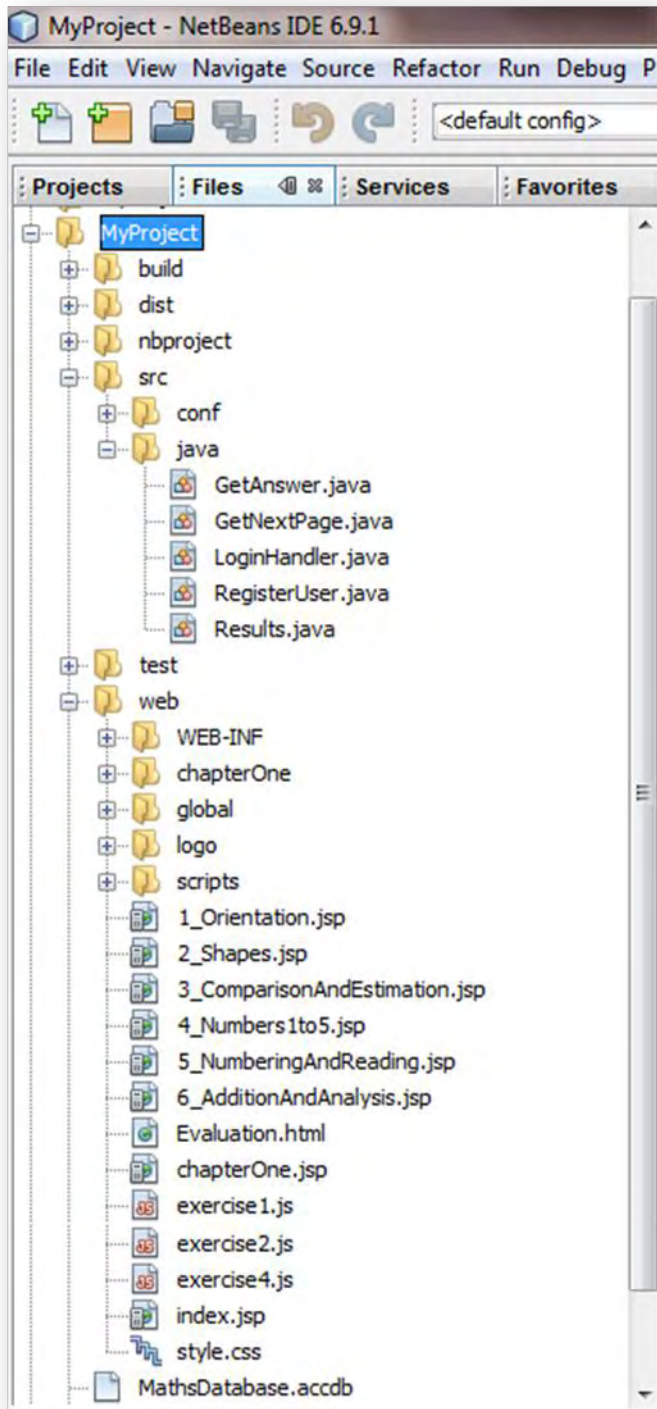
Η πρόθεση εδώ είναι ότι όλες οι απαντήσεις πρέπει να συνδέονται με έναν μαθητή που υπάρχει ήδη στον πίνακα Students. Για να γίνει αυτό, θα τοποθετήσουμε ένα ξένο κλειδί στον πίνακα StudentAnswer το οποίο έχει σχέση με το πρωτεύον κλειδί του πίνακα Students.

3.4.2 Αρχεία-files της εφαρμογής μας

Στη εικόνα 17 παρουσιάζουμε τα αρχεία από τα οποία αποτελείται η εφαρμογή μας.

Όπως παρατηρούμε έχουμε τα .jsp, τα .java, το .html, το .css και το .acddb αρχεία.

- Τα .jsp αρχεία :
Το πρόγραμμα εκτελεί πρώτα το αρχείο index.jsp, το οποίο όπως αναφέραμε ,στην ενότητα 3.2.1, αποτελεί την αρχική σελίδα όπου παρουσιάζονται δύο φόρμες (sign in and sing up form) αν ο χρήστης δε έχει κάνει log in, αλλιώς αποτελεί την ύλη του μαθήματος, ενότητα 3.2.2.
Μετά εκτελείται το chapterOne.jsp αρχείο το οποίο είναι η εισαγωγική σελίδα του πρώτου κεφαλαίου όπως δείξαμε στην ενότητα 3.2.3.
Τέλος θα εκτελεστούν τα μαθησιακά αντικείμενα από το 1_Orientation.jsp ... μέχρι το 6_AdditionAndAnalysis.jsp .
- Το .html αρχείο :
Το Evaluation.html αρχείο εμφανίζεται μετά τα μαθησιακά αντικείμενα και είναι χρήσιμο για τον εκπαιδευτικό η τους γονείς ώστε να δούνε που ακριβώς έκανε λάθος ο μαθητής. Παρουσιάζετε ένα κείμενο που επεξηγεί σε ποίο μαθησιακό στόχο , σε ποία ερώτηση και σε ποίο υπο-στόχο έκανε λάθος ο μαθητής .
- Τα .js αρχεία :
Τα αρχεία αυτά είναι JavaScript και τα έχουμε χρησιμοποιήσει για να υλοποιήσουμε τις ασκήσεις τύπου drag and drop . Επίσης μέσα στα .jsp αρχεία υπάρχουν ενσωματωμένα .js αρχεία τα οποία μας βοηθάνε να στείλουμε request (ασύγχρονα με AJAX ή σύγχρονα) , τα οποία πυροδοτούν την εκτέλεση των .java αρχείων. Τα request αυτά μπορεί να περιέχουν τις απαντήσεις που έδωσε ο μαθητής στις ερωτήσεις.
- Το .css αρχείο :
Το style.css αρχείο χρησιμοποιεί τη γλώσσα Cascading Style Sheets για τον έλεγχο εμφάνισης των αρχείων-ιστοσελίδων .html και .jsp . Με λίγα λόγια αναπτύσσει στιλιστικά (χαρακτηριστικά , χρώματα στοίχιση κοκ) τις ιστοσελίδες αυτές.



Εικόνα 17: Τα αρχεία της εφαρμογής μας

➤ Τα .java αρχεία :

Τα .java αρχεία είναι servlet τα οποία τα χρησιμοποιούμε για να αποθηκεύσουμε δεδομένα στην βάση δεδομένων καθώς και να εξάγουμε δεδομένα από την βάση δεδομένων . Όπως προαναφέραμε ,στην ενότητα τεχνολογίες, τα servlet είναι server-side αρχεία τα οποία εκτελούνται όταν δέχονται ένα request (που περιέχει δεδομένα τα οποία πρέπει να αποθηκευτούν στην βάση ή ζητάει δεδομένα από την βάση) και ύστερα πραγματοποιούν μια σύνδεση με την βάση (MathDatabase) . Η σύνδεση αυτή επιτρέπει να εκτελεστούν οι SQL ερωτήσεις, τα οποία έχει ετοιμάζει το servlet, στη βάση. Οι ερωτήσεις αυτές μπορεί να είναι τύπου INSERT, που καταχωρεί μια εγγραφή στην βάση, τύπου UPDATE, που ενημερώνει κάποιο/κάποια πεδία/στήλες μίας εγγραφής και τύπου QUERY, που επιστρέφει τα δεδομένα πεδίων κάποιας εγγραφής.

- LoginHandler.java

Το αρχείο αυτό χειρίζεται την σύνδεση του μαθητή. Εκτελείται όταν ο χρήστης πατήσει sing in (αρχική σελίδα) με σκοπό να ελέγξει ,μέσω μιας QUERY ερώτησης, αν ο μαθητής είναι εγγεγραμμένος στο σύστημα , δηλαδή ελέγχει αν το όνομα που πληκτρολογήθηκε υπάρχει στον πίνακα Students της βάσης. Αν υπάρχει ο μαθητής μπορεί να χρησιμοποιήσει το σύστημα αλλιώς δεν μπορεί, γίνεται redirect στην αρχική σελίδα.

Επίσης αυτό το αρχείο εκτελείται όταν πατηθεί ένα link το οποίο γράφει log out και σαν αποτέλεσμα έχουμε ο χρήστης να γίνει redirect στην αρχική σελίδα . Το log out link εμφανίζεται σε όλες τις ιστοσελίδες και δίνει την δυνατότητα στον χρήστη να βγει από το σύστημα.

- RegisterUser.java

Το αρχείο αυτό χειρίζεται την εγγραφή του μαθητή. Εκτελείται όταν ο χρήστης πατήσει sing up (αρχική σελίδα) και μέσω μιας ερώτησης τύπου INSERT καταχωρείται ο μαθητής στο πίνακα Students . Να σημειώσουμε ότι αν το όνομα που πληκτρολογήθηκε χρησιμοποιείται από άλλον μαθητή το σύστημα θα του ζητήσει άλλο όνομα.

- GetAnswer.java

Το αρχείο αυτό χειρίζεται από τις ασκήσεις. Εκτελείται όταν ο χρήστης πατήσει submit ,στο τέλος της κάθε άσκησης και μέσω μιας ερώτησης τύπου INSERT καταχωρείται η απάντηση του μαθητή στο πίνακα StudentAnswer. Η απάντηση του μαθητή είναι αποθηκευμένη στο request που λαμβάνει το αρχείο αυτό. Πριν εκτελεστεί η INSERT ερώτηση το GetAnswer.java ελέγχει αν η απάντηση του μαθητή είναι σωστή ή λάθος, αξιολογώντας έτσι τον μαθητή (**Evaluate Cognition**

State λειτουργία). Πρέπει να αναφέρουμε ότι το request αυτό στέλνεται ασύγχρονα με AJAX στο GetAnswer.java.

- Results.java

Το αρχείο αυτό χειρίζεται από το Evaluation.html και εκτελείται όταν ο χρήστης πατήσει submit. Σα αποτέλεσμα εμφανίζονται τα λάθη που έκανε ο μαθητής ,μέσω μιας QUERY ερώτησης που εκτελεί το Results.java.

- GetNextPage.java

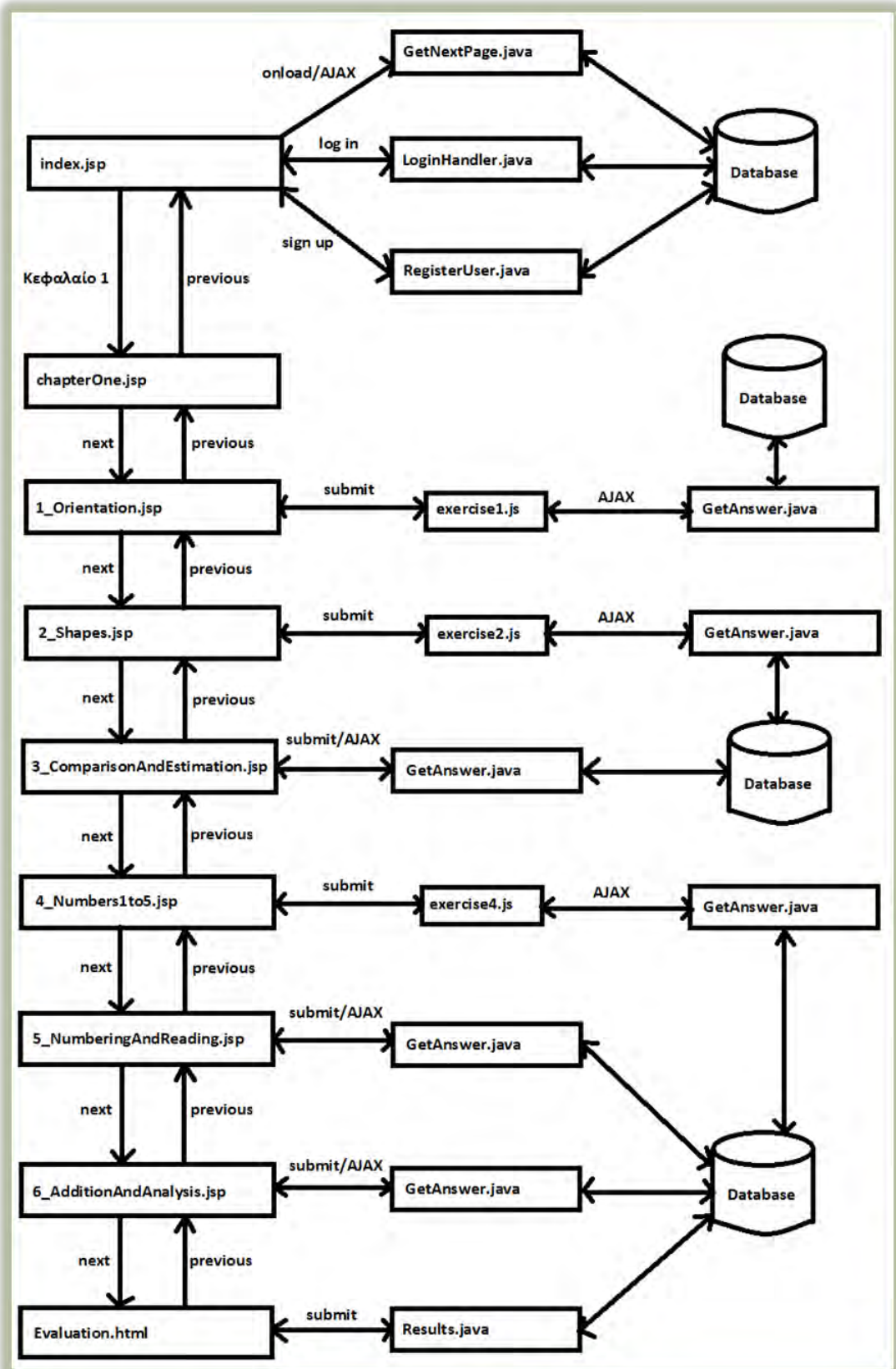
Το αρχείο αυτό εκτελείται μόλις ο χρήστης κάνει log in . Καλείται ασύγχρονα με AJAX όταν φορτώνεται η σελίδα index.jsp. Στόχος της είναι να ανακτήσει τα δεδομένα από τον πίνακα ChapterOne που αφορούν τον συγκεκριμένο μαθητή , μέσω μιας QUERY ερώτησης. Υστέρα αυτά τα δεδομένα αποθηκεύονται σε μεταβλητές του http session ώστε να είναι διαθέσιμα στα μαθησιακά αντικείμενα (.jsp αρχεία). Να σημειώσουμε ότι κάθε πεδίο της εγγραφής του πίνακα ChapterOne αποτελεί έναν μαθησιακό στόχο και συνδέεται με ένα μαθησιακό αντικείμενο .Αν το πεδίο είναι true (ο μαθητής δε έχει καλύψει αυτόν τον μαθησιακό στόχο) τότε θα εμφανιστεί το αντίστοιχο μαθησιακό αντικείμενο (.jsp αρχείο) . Αν το πεδίο είναι false (ο μαθητής έχει καλύψει αυτόν τον μαθησιακό στόχο) τότε δεν εμφανίζεται το αντίστοιχο μαθησιακό αντικείμενο (.jsp αρχείο) . Σε αυτό το σημείο πραγματοποιείται η ευφυής λειτουργία εξέτασης της γνωστικής κατάστασης του μαθητή (**Control Cognitive State**), με σκοπό να έχουμε εξατομικευμένη εκπαίδευση . Παρατηρούμε ότι ο οι τιμές της εγγραφής του πίνακα ChapterOne καθορίζουν την σειρά εμφάνισης των μαθησιακών αντικείμενων (**Show Curriculum λειτουργία**).

3.4.3 Ροή της εφαρμογής μας

Αφού είδαμε τα δομικά στοιχεία της εφαρμογής (βάση δεδομένων και αρχεία) είμαστε πλέον σε θέση να ορίσουμε το διάγραμμα ροής της εφαρμογής μας (Web – Application User Case) στη εικόνα 18 . Η ροή καθορίζεται από τα βελάκια στο σχήμα και πάνω τους έχουμε γράψει από τι πυροδοτείται η εκτέλεση κάθε σελίδας.

- Log in, sing up, next, previous, submit και Κεφαλαίο 1, αποτελούν τα κουμπιά που πατάει ο χρήστης και πυροδοτούν την εκτέλεση.
- Onload/AJAX, submit/AJAX και AJAX πυροδοτούν την εκτέλεση servlets επειδή στέλνουν κάποιο request σε αυτά.

- Τέλος τα βελάκια από τα .java (servlets) αρχεία προς την βάση δεδομένων λαμβάνουν χώρα όταν εκτελείται κάποια ερώτηση τύπου INSERT, UPDATE ή QUERY που έχουν ετοιμάσει τα servlets.

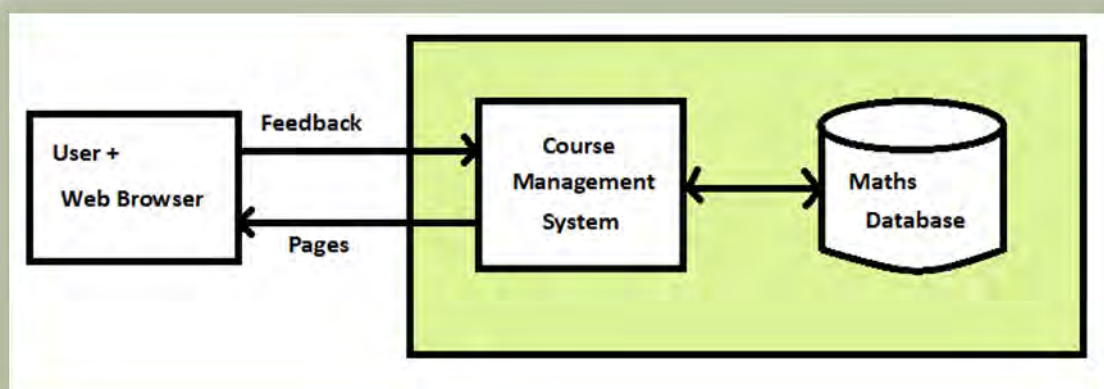


Εικόνα 18 : Διάγραμμα Ροής

Με την ίδια λογική υπάρχουν βελάκια από το 1_Orientation.jsp προς 3ComparisonAndEstimation.jsp, 4_Numbers1to5.jsp, 5NumberingAndEstimation.jsp , 6_AdditionAndAnalysis και αντίστροφα. Το ίδιο ισχύει και τις υπόλοιπες σελίδες.

3.4.4 Αρχιτεκτονική της εφαρμογής

Αφού μελετήσαμε όλα τα παραπάνω τώρα είμαστε σε θέση να ορίσουμε την γενική αρχιτεκτονική της εφαρμογής.



Εικόνα 20: Αρχιτεκτονική του συστήματος μας

Το Course Management System αποτελείται από όλα τα .java αρχεία που αναφέραμε και προσφέρει τις λειτουργίες που περιγράψαμε .

Το MathDatabase είναι η βάση δεδομένων μας και περιέχει τους πίνακες που περιγράψαμε.

Τέλος το μάθημα δηλαδή οι ιστοσελίδες παρουσιάζονται στον μαθητή-χρήστη με τη βοήθεια ενός web browser (Firefox ή Explorer).

3.5 Ειδικά Σημεία

Ο σχεδιασμός και η υλοποίηση της εφαρμογής μας ήταν σχετικά εύκολη με εξαίρεση το σχεδιασμό της Βάσης Δεδομένων . Έπρεπε να γίνει προσεκτική μελέτη των πινάκων και αλληλεπιδράσεων τους , ώστε η εφαρμογή να λειτουργήσει σωστά.

Κεφάλαιο 4

Συμπεράσματα και μελλοντική δουλεία

4.1 Συμπεράσματα

Ξεκινήσαμε να δημιουργήσουμε το δικό μας σύστημα διδασκαλίας για τα μαθηματικά Α δημοτικού. Κρίναμε αναγκαίο ότι το σύστημα αυτό πρέπει να προσφέρει εξατομικευμένη διδασκαλία, έτσι καταλήξαμε στη δημιουργία μοντέλου μαθητή. Το σύστημα «συμβουλευέται» το μοντέλο του μαθητή και ανάλογα προσαρμόζει την διδασκαλία για τον κάθε μαθητή. Σε αυτό το σημείο πρέπει να αναφέρουμε ότι το δικό μας μοντέλο μαθητή που χρησιμοποιήσαμε για την υλοποίηση είναι πολύ απλοϊκό (ένα πλήρες μοντέλο θα πρέπει να περιλαμβάνει μαθησιακές προτιμήσεις και κατάσταση γνώσης). Η μόνη πληροφορία που αποθηκεύει το μοντέλο μας είναι οι απαντήσεις που έδωσε ο μαθητής στις ερωτήσεις που του υποβλήθηκαν. Λαμβάνοντας υπόψη τις απαντήσεις αυτές το σύστημα αποφασίζει ποιους μαθησιακούς στόχους δεν έχει καλύψει ο μαθητής, ώστε να κατευθύνει την διδασκαλία προς αυτούς τους στόχους. Ως εκ τούτου το σύστημα μας είναι ένα σύστημα διδασκαλίας μαθηματικών Α δημοτικού, το οποίο προσαρμόζεται για να καλύψει τις ανάγκες του κάθε μαθητή.

4.2 Μελλοντική δουλεία

Για να βελτιωθεί το σύστημα διδασκαλίας μπορεί να υλοποιηθούν πολλά σημεία ακόμα, κάποια από αυτά είναι :

- Να βαθμολογείται η επίδοση του μαθητή και πιο συγκεκριμένα να χρησιμοποιούνται βάρη στην διαδικασία αυτή. Οι ασκήσεις που έχουν πιο μεγάλη βαρύτητα να χρησιμοποιούν πιο μεγαλύτερα βάρη ώστε να συνδράμουν πιο πολύ στην βαθμολογία.
- Οι μαθησιακοί στόχοι μπορεί να αναπαριστώνται σε γράφους και απώτερος στόχος θα είναι περιέχουν πληροφορίες σχετικά με τις σχέσεις και αλληλεξαρτήσεις των στόχων αυτών. Έτσι αν ο μαθητής δε έχει καλύψει έναν μαθησιακό στόχο σ_2 , το οποίο έχει σα προαπαιτούμενο έναν άλλον στόχο σ_1 , τότε θα αναλυθεί πρώτα το σ_1 και μετά το σ_2 .
- Όταν ξαναπαρουσιάζονται κάποιες διδακτικές ενότητες στον μαθητή μπορεί να υπάρχουν επιπλέον πληροφορίες-επεξηγήσεις που θα βοηθήσουν τον μαθητή να κατανοήσει καλύτερα τις ενότητες αυτές.
- Το σύστημα μπορεί να υλοποιηθεί έτσι ώστε κάθε φορά να φτιάχνει μόνο του τις ασκήσεις ανάλογα με το επίπεδο του μαθητή.

References

- 1) Sunandan Chakraborty, Devshri Roy, Anupam Basu. Development of knowledge Based Intelligent Tutoring System. In Advanced Knowledge Based Systems : Model, Applications & Research (Eds. Sajja & Akerkar) , Vol. 1, pp 74-100, 2010
- 2) Hyacinth S. Nwana (1990).Intelligent Tutoring Systems: an overview. In artificial Intelligent Review, pp. 251-277.
- 3) Frasson,C., Mengelle,T., Aimeur,E., Gouarderes, G.(1996) "An Actor-Based Architecture for Intelligent Tutoring Systems ", ITS'96 Conference, Lecture Notes in Computer Science, No 1086,Springer Verlag , Montreal,pp.57-65.
- 4) Frasson, C., Mengelle, T., Aimeur, E. (1997) "Using pedagogical Agent in a Multistrategic Intelligent System ",AIED Workshop on Pedagogical agents in AI-ED 97, World Conference on Artificial Intelligent and Educational , Japan, pp 40-47.
- 5) Capuano, N., Marsella, M., Salerno, S. ABITS: An Agent Intelligent Tutoring System for Distance Learning. CRMPA, University of Salerno, 84084, Fisciano (SA), Italy.

Τέλος