



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΜΣ <<Προηγμένα Συστήματα Πληροφορικής>>
2016-2017

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

(Διαδικτυακή εκπαιδευτική εφαρμογή αξιολόγησης επίδοσης του μαθητή)

Σπουδαστής

Καραγεωργόπουλος Κυριάκος,
Αρ. Μητρώου: ΜΣΠ15032

Επιβλέποντες

Ευθύμιος Αλέπης (Λέκτορας)

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία με θέμα «Ανάπτυξη διαδικτυακής εκπαιδευτικής εφαρμογής», εκπονήθηκε την περίοδο 2017 από το φοιτητές Καραγεωργόπουλο Ιωάννη και Κυριάκο του μεταπτυχιακού τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς. Μετά το πέρας της εκπόνησής της, νιώθω την ανάγκη να εκφράσω τις ευχαριστίες μου προς τα άτομα που συνέβαλαν στην ολοκλήρωση της, παρέχοντάς μου είτε γνώση, είτε ηθική υποστήριξη.

Αρχικά, ευχαριστώ θερμά τον επιβλέπων Λέκτορα Ευθύμιο Αλέπη για την ανάθεση του θέματος, την καθοδήγηση αλλά και την άριστη επικοινωνία που είχαμε αφού η ανταπόκριση ήταν άμεση σε κάθε είδους απορία που έθετα.

Ευχαριστίες θα ήθελα επίσης να απευθύνω σε όλους τους καθηγητές του τμήματος Πληροφορικής, για το γνωστικό υπόβαθρο, τον επαγγελματισμό και την έμπειρη καθοδήγηση που μου πρόσφεραν όλα αυτά τα χρόνια.

Τέλος, δεν θα μπορούσα να μην αναφερθώ στην οικογένειά μου και στους φίλους μου, για την κατανόηση και την ενθάρρυνση που μου παρείχαν καθ' όλη τη διάρκεια διεκπεραίωσης αυτής της εργασίας.

Χωρίς όλους τους παραπάνω είναι δεδομένο πως κάθε εμπόδιο που θα συναντούσα θα ήταν πολύ πιο δύσκολο να ξεπεραστεί και κατά συνέπεια η ακαδημαϊκή μου πορεία δεν θα ήταν η επιθυμητή.

1. ΠΕΡΙΛΗΨΗ

Στόχος της παρούσας εργασίας είναι η εξοικείωση με την ανάπτυξη διαδικτυακών εκπαιδευτικών εφαρμογών. Θα γίνει λοιπόν μία προσπάθεια ανάδειξης των δυνατοτήτων των συγκεκριμένων εφαρμογών, αναπτύσσοντας μία εφαρμογή τύπου 'quiz' στην οποία καθηγητής και μαθητής αλληλεπιδρούν με στόχο την βελτίωση των μαθησιακών επιδόσεων .

Εκτός από την συμμετοχή του μαθητή στα συγκεκριμένα 'quiz' ερωτήσεων η εφαρμογή θα μπορεί να αναγνωρίσει τις κλίσεις του μαθητή βάση των απαντήσεων και των επιλογών του.

Έτσι λοιπόν, στην εργασία που ακολουθεί θα δούμε ορισμένες πληροφορίες που αφορούν το framework asp.net mvc 5 με το οποίο έχει υλοποιηθεί η εφαρμογή μας ,όπως ποια είναι η αρχιτεκτονική του και ποια ήταν η εξέλιξή του, από τη στιγμή της δημιουργίας του μέχρι σήμερα. Ακόμα, θα αναλύσουμε τη διαδικασία που απαιτείται για τη δημιουργία ενός αρχικού project και ποια είναι τα δομικά στοιχεία από τα οποία αποτελείται. Τέλος, θα αναλύσουμε λεπτομερώς την υλοποίηση της διαδικτυακής εκπαιδευτικής εφαρμογής και θα καταλήξουμε σε ορισμένα συμπεράσματα και σε πιθανές βελτιώσεις που θα χρειαστούν μελλοντικά.

1. SUMMARY

The aim of this paper is to familiarize students with the development of online educational applications. An effort will be made to highlight the potential of these applications by developing a quiz-like application in which the teacher and student interact with the aim of improving learning performance.

In addition to the student's participation in specific question queries, the application will be able to recognize the learner's gradients based on his / her answers and choices.

So, in the following work we will see some information regarding the framework asp.net mvc 5 with which our application has been implemented, such as what its architecture is and what its evolution has been since its creation until today . We will also analyze the process required to create an original project and what the building blocks from which it is composed. Finally, we will analyze in detail the implementation of the online educational application and we will come up with some conclusions and possible improvements that will be needed in the future.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	1
1. ΠΕΡΙΛΗΨΗ.....	3
ΠΕΡΙΕΧΟΜΕΝΑ.....	4
ΚΕΦΑΛΑΙΟ 2 - Εισαγωγή.....	6
ΚΕΦΑΛΑΙΟ 3 - Ανασκόπηση πεδίου.....	8
ΦΥΣΙΚΗ – ΠΡΟΣΟΜΟΙΩΣΕΙΣ ΠΕΙΡΑΜΑΤΩΝ.....	13
Περιγραφή	14
Τύπος εκπαιδευτικού λογισμικού	14
Θεματική κατάταξη	14
Εταιρεία ανάπτυξης.....	14
Fuction Probe.....	15
Περιγραφή	15
Τύπος εκπαιδευτικού λογισμικού	15
Θεματική κατάταξη	16
Μαθηματικά	16
Εταιρεία ανάπτυξης.....	16
eCity – Εικονική Πόλη	16
Περιγραφή	16
Τύπος εκπαιδευτικού λογισμικού	17
Θεματική κατάταξη	17
Εταιρεία ανάπτυξης.....	17
Άξονες.....	17
Περιγραφή	17
Τύπος εκπαιδευτικού λογισμικού	18

Θεματική κατάταξη	18
Εταιρεία ανάπτυξης	18
ΚΕΦΑΛΑΙΟ 4 - Παρουσίαση και χρήση εφαρμογής	19
4.1 Εγχειρίδιο χρήστη	19
4.1.1 Αρχική οθόνη	19
4.1.2 Διαχειριστική οθόνη καθηγητή	20
4.1.3 Οθόνη διαχείρισης ερωτήσεων (test).....	21
4.1.4 Δημιουργία νέας ερώτησης συμπλήρωσης κενών.....	22
4.1.5 Δημιουργία ερωτήσεων πολλαπλής επιλογής	23
4.1.6 Επεξεργασία ερωτήσεων πολλαπλής επιλογής.....	24
4.1.7 Οθόνη μαθητών	24
4.1.8 Οθόνη έγκρισης εγγραφής μαθητών.....	25
4.2 Αρχιτεκτονική της εφαρμογής	26
4.2.1 MVC - Αντιστοιχία με την Εφαρμογή.....	28
4.3 Λεπτομέρειες υλοποίησης.....	31
4.3.1 Εργαλεία υλοποίησης	31
4.4 Βάση δεδομένων	32
4.4.1 Πίνακες βάσης δεδομένων	33
Πίνακας QuestionGap	33
Πίνακας QuestionVariants	33
Πίνακας Variant	34
Πίνακας Chapter.....	34
Πίνακας TestPass	34
Πίνακας Account	35
Πίνακας ForApprove.....	35
Διάγραμμα περιπτώσεων χρήσης	36
ΚΕΦΑΛΑΙΟ 5 - Συμπεράσματα	41
5.1 Πλεονεκτήματα της υλοποίησης	40
5.2 Μελλοντικές βελτιώσεις	43
ΒΙΒΛΙΟΓΡΑΦΙΑ - ΙΣΤΟΣΕΛΙΔΕΣ	44

ΚΕΦΑΛΑΙΟ 2 - Εισαγωγή

Το πρώτο κεφάλαιο αποτελεί μία γενική εισαγωγή στην παρούσα διπλωματική εργασία και περιλαμβάνει: την περιγραφή του αντικειμένου της εργασίας, μία αναφορά στην αρχιτεκτονική mvc ,περιγραφή της υλοποίησης με την χρήση του framework asp.net mvc 5 και τέλος μία περίληψη των θεμάτων που θα αναπτυχθούν.

Στη εργασία που ακολουθεί θα ασχοληθούμε με το μαθηματικό εκπαιδευτικό λογισμικό. Αναζητήσαμε προγράμματα που αναφέρονται σε αυτό, θα προσπαθήσουμε να τα περιγράψουμε και να τα κατηγοριοποιήσουμε. Φυσικά ήταν αδύνατον να τα προσθέσουμε όλα τα προγράμματα σ αυτήν την εργασία πρώτον γιατί ο αριθμός τους είναι πολύ μεγάλος και δεύτερον γιατί πολλά από αυτά είναι παρόμοια και έχουν την ίδια χρήση, έτσι λοιπόν ξεχωρίσαμε κάποια από αυτά τα οποία θα παρουσιαστούν στην συνέχεια. Πριν αυτού όμως ας δούμε τι είναι το εκπαιδευτικό λογισμικό και γιατί πρέπει να το χρησιμοποιήσουμε. Με τον όρο "εκπαιδευτικό λογισμικό" αναφερόμαστε στο σύνολο των εφαρμογών για ηλεκτρονικούς υπολογιστές ή τον παγκόσμιο ιστό που εξυπηρετούν εκπαιδευτικούς σκοπούς. Κάποια λογισμικά εξυπηρετούν τους εκπαιδευτικούς διευκολύνοντας τους στο έργο τους. Άλλες φορές, ενισχύουν την προσπάθεια του μαθητή, παρέχοντάς τους εργαλεία εξάσκησης, πηγές γνώσεις, κ.ά. Επιπλέον, υπάρχει εκπαιδευτικό λογισμικό που χρησιμοποιείται στο συντονισμό του εκπαιδευτικού έργου, τη συγκέντρωση και οργάνωση του εκπαιδευτικού υλικού, την κατάρτιση και τήρηση του προγράμματος εκπαίδευσης και, γενικότερα, τη διοίκηση και λειτουργία ενός εκπαιδευτικού οργανισμού. Εκτός από το λογισμικό που τίθεται στην υπηρεσία του εκπαιδευτικού και του συστήματος εκπαίδευσης, υπάρχει και εκπαιδευτικό λογισμικό προσαρμοσμένο στις ιδιαιτερότητες του εκπαιδευόμενου. Έτσι, αρκετά διαδεδομένα είναι τα συστήματα εξ αποστάσεως εκπαίδευσης, ενώ έχουν ξεκινήσει να λειτουργούν και εξελιγμένα συστήματα αυτοεκπαίδευσης, είτε από απόσταση, είτε με την βοήθεια ενός εκπαιδευτικού ή εκπαιδευτικού οργανισμού. Εκπαιδευτικό λογισμικό μπορεί να θεωρηθεί κάθε οργανωμένη πηγή γνώσης, όπως ηλεκτρονικές βιβλιοθήκες, εγκυκλοπαίδειες, ψηφιακές συλλογές οπτικοακουστικού υλικού, κλπ. Τέλος, υπάρχουν αρκετά ηλεκτρονικά παιχνίδια με καθαρά εκπαιδευτικό χαρακτήρα. Με το εκπαιδευτικό λογισμικό επιδιώκεται να δημιουργηθεί ένα μαθησιακό περιβάλλον με σκοπό η διδασκαλία να γίνει πιο δημιουργική, ενεργητική και διερευνητική, αξιοποιώντας τις δυνατότητες που προσφέρουν τα εκπαιδευτικά προγράμματα κάνοντας το μάθημα πιο ελκυστικό και προκλητικό. Δηλαδή το εκπαιδευτικό λογισμικό είναι ένα μέσο για την επίτευξη των στόχων που θέτει κάθε μάθημα και για την ποιοτική βελτίωση της διδασκαλίας. Σήμερα, υπάρχουν πολλά εκπαιδευτικά λογισμικά για την διδασκαλία πολλών μαθημάτων (μαθηματικά, φυσική, χημεία κ.ά.) κάποια αναφέρονται σε μια συγκεκριμένη θεματική ενότητα ενώ κάποια άλλα μπορούν να χρησιμοποιηθούν σε περισσότερα από ένα μαθήματα. Τα προγράμματα αυτά εξυπηρετούν ανάγκες ποικίλων επιπέδων, από διδασκαλία σε μικρά παιδιά έως διδασκαλία σε πανεπιστημιακό επίπεδο. Δυστυχώς, ελάχιστα από αυτά χρησιμοποιούνται στη χώρα μας, και πολύ λίγα είναι προσαρμοσμένα στην ελληνική γλώσσα και πραγματικότητα. Ειδικότερα, το εκπαιδευτικό λογισμικό αναμένεται να συμβάλει

- στη φιλικότερη, ελκυστικότερη, πλουσιότερη και πολύπλευρη παρουσίαση της ύλης,
- στη βιωματική προσέγγιση της γνώσης,
- στην πρόκληση του ενδιαφέροντος του μαθητή μέσα από την δημιουργία
- στην καλύτερη κατανόηση της ύλης μέσα από τις δραστηριότητες τον πειραματισμό και την διερεύνηση,
- στην συμπύκνωση και παρουσίαση της ύλης είτε οπτικά είτε ακουστικά.
- στην εξοικονόμηση χρόνου και κόπου του μαθητή και του εκπαιδευτικού,
- στην ανάπτυξη συνεργατικότητας μέσα στην τάξη πέρα από την ατομική δουλειά κάθε μαθητή.

Το εκπαιδευτικό λογισμικό συχνά αποτελεί μέρος του συνολικού διδακτικού υλικού μιας εκπαιδευτικής διαδικασίας και πρέπει να είναι σχεδιασμένο και κατασκευασμένο έτσι ώστε να διευκολύνει τη μάθηση και να εκπληρώνει συγκεκριμένους μαθησιακούς στόχους. Είναι ιδιαίτερα σημαντικό να μπορεί ο εκπαιδευτής-δάσκαλος αφενός μεν να συνεισφέρει στη διαδικασία σχεδίασης και ανάπτυξης του εκπαιδευτικού λογισμικού, αφετέρου δε να μπορεί να αξιολογήσει σωστά το εκπαιδευτικό λογισμικό που θα χρησιμοποιήσει. Με δεδομένη τη σημασία της συνεισφοράς του καλού εκπαιδευτικού λογισμικού στην εκπαιδευτική διαδικασία, η αξιολόγησή του καθίσταται ιδιαίτερα σημαντική. Το εκπαιδευτικό λογισμικό θα πρέπει να αξιολογείται από την εκπαιδευτική-μαθησιακή αλλά και από τεχνολογική άποψη. Τα λογισμικά που θα ασχοληθούμε είναι ελεύθερα λογισμικά/λογισμικά ανοικτού κωδικά(ΕΛ/ΛΑΚ) είτε εμπορικά λογισμικά. Δίνουμε έμφαση στα ελεύθερα λογισμικά, γι αυτό το λόγο πρέπει να δούμε τι είναι το ελεύθερο λογισμικό. Το ελεύθερο λογισμικό όπως ορίζεται από το Ίδρυμα Ελευθέρου Λογισμικού (Free Software Foundation)[από Βικιπαιδεία], είναι λογισμικό που μπορεί να χρησιμοποιηθεί, αντιγραφεί, μελετηθεί, τροποποιηθεί και αναδιανεμηθεί χωρίς περιορισμό. Η ελευθερία από τέτοιους περιορισμούς είναι βασικό στοιχείο στην ιδέα του ελευθέρου λογισμικού. Το αντίθετο του ελευθέρου λογισμικού είναι το ιδιόκτητο λογισμικό, και όχι το λογισμικό που πωλείται για κέρδος, όπως το εμπορικό λογισμικό. Το ελεύθερο λογισμικό ορισμένες φορές αναφέρεται και σαν ανοιχτό λογισμικό ή λογισμικό ανοιχτού κώδικα (Free Open Source Software - FOSS). Εν γένει, σύμφωνα με την ισχύουσα νομοθεσία περί πνευματικής ιδιοκτησίας, η ελεύθερη αντιγραφή, διανομή και τροποποίηση του λογισμικού δεν επιτρέπεται. Για το λόγο αυτό, οι εκδόσεις ελευθέρου λογισμικού κάνουν χρήση ειδικής άδειας (free software licence) σύμφωνα με την οποία, παραχωρείται το δικαίωμα αντιγραφής, τροποποίησης και αναδιανομής του λογισμικού, στους χρήστες. Επομένως ένα ελεύθερο λογισμικό μπορεί να χρησιμοποιηθεί από πολλούς, να τροποποιηθεί ή και να ελεγχθεί για την καλύτερη χρήση του, βασικό ρόλο παίζει και το κόστος για την χρήση η την απόκτηση του που φυσικά είναι πολύ μικρότερο από αντίστοιχα εμπορικά λογισμικά. Ένα κομμάτι που επηρεάζει αρνητικά τα ελεύθερα λογισμικά είναι ότι δεν έχει ελεγχόμενη ανάπτυξη και πρέπει να περάσουν πολλές εκδόσεις για την καλύτερευση και σταθεροποίηση του λογισμικού.

ΚΕΦΑΛΑΙΟ 3 - Ανασκόπηση πεδίου

3.1 ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΛΟΓΙΣΜΙΚΩΝ

Η κατηγοριοποίηση των εκπαιδευτικών λογισμικών με απασχόλησε πολύ και με έφερε αντιμέτωπη στο ερώτημα με ποια κριτήρια και ποιες μεθόδους γίνεται η κατηγοριοποίηση; Είχε δίκιο ο Taylor(1980) ο οποίος θεωρούσε τον υπολογιστή καθηγητή, εργαλείο ή διδασκόμενο; Ή μήπως η στρατηγική του Means(1994) ήταν πιο κατάλληλη περιγράφοντας τέσσερις διαφορετικές κατηγορίες (διδασκαλία, εξερεύνηση, εργαλείο, επικοινωνία) για τα προγράμματα με βάση την χρήση τους ; Πολλά από τα είδη εκπαιδευτικού λογισμικού που έχουν αναπτυχθεί μέχρι σήμερα θέτουν τον υπολογιστή στη θέση του καθηγητή-στηριζόμενα κυρίως στη συμπεριφοριστική αντίληψη (βλ παρακάτω) και άλλα αξιοποιούν τη δυνατότητα για εξερεύνηση και επικοινωνία για να υλοποιήσουν μια περισσότερο εποικοδομητική διδακτική προσέγγιση. Εμείς θα τα κατηγοριοποιήσουμε με βάση την χρήση τους στη μαθησιακή διαδικασία, το βαθμό αλληλεπίδρασης και την παιδαγωγική προσέγγιση. Υπάρχει η πιθανότητα ένα λογισμικό να υπάγεται σε περισσότερες από μια κατηγορίες. Κατηγορίες λογισμικού:

1. Προγράμματα εξάσκησης και πρακτικής. (drill and practice). Στηρίζονται στις αρχές της προγραμματισμένης διδασκαλίας και έχουν απλή γραμμική μορφή καλώντας τον εκπαιδευόμενο να απαντήσει σε μια σειρά ερωτήσεων. Τα προγράμματα αυτά στηρίζονται αποκλειστικά στη διατύπωση της σωστής απάντησης και δεν εκμεταλλεύονται το λάθος του μαθητή ώστε να τον οδηγήσουν σταδιακά στην σωστή απάντηση. Τα συστήματα εξάσκησης και πρακτικής στοχεύουν στην παροχή άσκησης ώστε να αναπτυχθούν και να βελτιωθούν γνώσεις και δεξιότητες. Σε αντίθεση με τα συστήματα καθοδήγησης που προσφέρουν έναν ολοκληρωμένο κύκλο διδασκαλίας (στη σύγχρονη μάλιστα μορφή τους κάνουν χρήση πολλαπλών μορφών πληροφορίας, όπως κείμενα, ήχοι, βίντεο, εικόνες και κινούμενες εικόνες), τα συστήματα εξάσκησης και πρακτικής στοχεύουν ένα διαφορετικό κοινό αφού αφορούν μαθητές ή χρήστες που είναι ήδη εξοικειωμένοι στον έναν ή στον άλλο βαθμό με το αντικείμενο διδασκαλίας. Με άλλα λόγια, δεν στοχεύουν στην παροχή νέας πληροφορίας αλλά στον έλεγχο των αποκτηθέντων εκτός συστήματος γνώσεων.

2. Εκπαιδευτικό παιχνίδι . Υπάρχουν δυο είδη (α) τα παιχνίδια περιπέτειας όπου οι μαθητές μαθαίνουν παίζοντας. και (β) τα παιχνίδια υπολογισμών που προωθούν την υπολογιστική ικανότητα (π.χ. Προγράμματα τα οποία ελέγχουν την ορθότητα των πράξεων). Τα εκπαιδευτικά ηλεκτρονικά παιχνίδια είναι τα παιχνίδια τα οποία ενθαρρύνουν την ανάπτυξη της λογικής και την απόκτηση δεξιοτήτων και γνώσης με έναν ευχάριστο τρόπο . Το υπόβαθρό τους σχετίζεται με κομμάτια γνώσης τα οποία οι χρήστες πρέπει να εφαρμόσουν με σκοπό να επιτύχουν τους στόχους που τους προτείνονται. Από τις πρώτες έρευνες που έγιναν για την χρήση των παιχνιδιών στην εκπαίδευση (Gordon, 1970) αποδείχθηκε ότι αποτελούν μία πηγή κινήτρου για τους χρήστες να δοκιμάσουν τις γνώσεις τους, να τις αναπτύξουν εφαρμόζοντάς τις καθώς και να μάθουν πράγματα που δεν γνωρίζουν ενώ ταυτόχρονα διασκεδάζουν. Συγκεκριμένα, η χρήση των πολυμέσων,

οι ελκυστικές ιστορίες που παρουσιάζουν πραγματικούς ή φανταστικούς στόχους και πράκτορες (agents) που συνοδεύουν το χρήστη κατά τη διάρκεια του παιχνιδιού (δίνοντάς τους κίνητρο να συνεχίσουν το παιχνίδι και εφοδιάζοντας τους με ανατροφοδότηση) αυξάνουν την μαθησιακή επίδοση. Ένα εκπαιδευτικό παιχνίδι πρέπει να καθοδηγήσει, να προωθήσει τη διανοητικότητα, ή τη συναισθηματική ή/και φυσική ανάπτυξη. Ένα εκπαιδευτικό παιχνίδι μπορεί να διδάξει ένα παιδί για ένα ιδιαίτερο θέμα ή μπορεί να βοηθήσει ένα παιδί να αναπτύξει μια ιδιαίτερη ικανότητα. Η βασική διαφορά είναι η εκμάθηση και η ανάπτυξη του παιδιού που συνδέονται με την αλληλεπίδραση με το παιχνίδι.

3. Λογισμικό εκμάθησης (tutorial). Τα προγράμματα αυτά i) προσομοιώνουν έναν τρόπο εκπαίδευσης με την μορφή ερωτήσεων αυξανόμενης δυσκολίας ii) παρουσιάζουν τις ασκήσεις iii) εξηγούν την εργασία που πρέπει να εκτελεστεί iv) ελέγχουν τα αποτελέσματα v) μετρούν την απόδοση του χρήστη και vi) του δίνουν συμβουλές. Παρέχουν την δυνατότητα παρουσίασης στοιχείων θεωρίας στους μαθητές τα οποία ενισχύονται με τη χρήση διαφορετικών μέσων όπως ήχο, κείμενο και υπερκείμενο (συνδεδεμένο με διαφορετικά αρχεία), εικόνες και βίντεο. Στοχεύουν στην πληρέστερη κατανόηση του αντικείμενου που η θεωρία περιγράφει και στην αύξηση του ενδιαφέροντος των μαθητών, κάνοντας το μάθημα ελκυστικότερο.

4. Προγράμματα προσομοίωσης (simulations) Αναπαριστούν σχηματικά μια πραγματική κατάσταση ή ένα φαινόμενο. Διακρίνονται στις ακόλουθες κατηγορίες: i. Προσομοίωση καταστάσεων και επίλυση προβλημάτων. Τοποθετούν τον χρήστη σε μια προβληματική κατάσταση την οποία πρέπει να επίλυση παίρνοντας υπόψιν διάφορους περιορισμούς και συνθήκες ii. Προσομοίωση περιπτώσεων. Ο χρήστης τοποθετείται σε μια κατάσταση όπου μπορεί να παρέμβει ενώ η σκέψη και ο συλλογισμός του καθοδηγούνται από τον υπολογιστή με βάση κάποια αποτελέσματα στα οποία καταλήγει μετά από την ανακάλυψη διάφορων κανόνων, νόμων κ.λπ. iii. Προσομοίωση με χρήση κινούμενων εικόνων ή σχεδίων iv. Προγράμματα εικονικής πραγματικότητας (virtual reality programs). Εξειδικευμένη και προωθημένη μορφή προσομοίωσης και αναπαράσταση πραγματικών καταστάσεων (με χρήση γυαλιά κάσκας κλπ που φορά ο χρήστης αισθάνεται την εικονική πραγματικότητα ως πραγματική).

5. Αλληλεπιδραστικές προσομοίωσης στις φυσικές επιστήμες. Έπεται των πειραματικών διαδικασιών, πρώτα εξερεύνηση του πραγματικού κόσμου και μετά του εικονικού. Δεν αντικαθιστούν την δουλειά στο εργαστήριο. Με τις αλληλεπιδραστικές προσομοίωσης οι μαθητές έχουν την δυνατότητα, να σχεδιάζουν και να πραγματοποιούν τα δικά τους πειράματα, να εξερευνούν και να οπτικοποιούν τις συνέπειες των συλλογισμών τους, να διαπιστώνουν τις αντιφάσεις τους και να αποκτούν γνώση των λανθασμένων τους ιδεών και αντιλήψεων και τέλος να αποκτούν πολύτιμη εμπειρία.

6. Ανοικτά περιβάλλοντα διερεύνησης και μελέτης. Αυτά τα προγράμματα ζητούν από τους μαθητές να επιλύσουν και να διερευνήσουν προβλήματα στηριζόμενοι σε γνώσεις που αποκτήθηκαν νωρίτερα. Αυτά τα προγράμματα μπορούν να αποτελούν έναυσμα για την διερευνητική μάθηση στην τάξη. Δίνουν την δυνατότητα να εφαρμοστούν οι κλασσικές

στρατηγικές επίλυσης προβλημάτων και βοηθούν το μαθητή να δημιουργήσει και να αναπτύξει περισσότερο τη δική του στρατηγική επίλυσης.

7. Προγράμματα δημιουργίας και έκφρασης. Υπάρχουν τα ακόλουθα είδη: (α) προγράμματα για την παράγωγή κινούμενων σχεδίων (β) προγράμματα που βοηθούν στην καλλιτεχνική δημιουργία (γ) προγράμματα για την δημιουργία άλλων προγραμμάτων και (δ) προγράμματα δημιουργίας παιχνιδιών. Τα προγράμματα αυτά υποστηρίζουν την δημιουργικότητα των μαθητών η οποία μπορεί να σχετίζεται με όλες τις μορφές λογού και με όλες τις καλλιτεχνικές μορφές. 8. Προγράμματα μοντελοποίησης. Τυποποιούν οπτικά σύμβολα και συμπεριφορές με στόχο την αφαιρετική περιγραφή της λειτουργίας και της δομής ενός υπολογιστικού συστήματος. Μια μεγάλη κατηγορία εκπαιδευτικού λογισμικού που εμπερικλείει στις λειτουργίες του δραστηριότητες χειρισμού εικονικών και συμβολικών παραστάσεων που αναπαριστούν αντικείμενα, έννοιες, ιδιότητες ή πράξεις πάνω στον πραγματικό κόσμο καθώς και τη δυνατότητα σύνδεσής τους, επιτρέποντας την έκφραση της δομής και των αλληλεξαρτήσεων τους, είναι το λογισμικό μοντελοποίησης. Δεδομένου ότι τα μοντέλα είναι αναπαραστάσεις της δομής ενός συστήματος, χρειάζονται εργαλεία μοντελοποίησης τα οποία να κατασκευάζουν αυτές τις αναπαραστάσεις. Η ανάπτυξη υπολογιστικών μοντέλων παρέχει τη δυνατότητα χειρισμού τους (και όχι χειρισμού των ίδιων των αντικειμένων), και επιτρέπει τη δυνατότητα έκφρασης (δραστηριότητες μοντελοποίησης, με δημιουργία νέων μοντέλων) και διερεύνησης (δραστηριότητες διερεύνησης έτοιμων μοντέλων μέσω της προσομοίωσής τους) συλλογισμών. Στο πλαίσιο αυτό, τα λογισμικά μοντελοποίησης οφείλουν να διαθέτουν: α) Εργαλεία για οικοδόμηση και ανάλυση μοντέλων (έκφραση ενός μοντέλου): οι υπάρχουσες μορφές αναπαράστασης (λεκτικές, εικονικές, μαθηματικές, διαγραμματικές, κλπ.) υλοποιούνται μέσω του υπολογιστικού περιβάλλοντος. Ένα σημαντικό στοιχείο με πολλαπλές δυνατότητες είναι οι πιθανές μορφές αναπαράστασης οι οποίες δεν μπορούν να υλοποιηθούν χωρίς τη χρήση υπολογιστών. β) Εργαλεία ελέγχου εγκυρότητας του μοντέλου (διερεύνηση ενός μοντέλου): τα εργαλεία αυτά χρησιμοποιούνται για τη σύγκριση των προβλέψεων δεδομένων μοντέλων με τα εμπειρικά δεδομένα. 9. Προγράμματα άμεσου χειρισμού. Ο μαθητής μπορεί να χειρίζεται άμεσα ένα ψηφιακό αντικείμενο δοκιμάζοντας εύκολα τις ιδέες του και έχει άμεση ανάδραση των αλλαγών και των επιδράσεων σε διαφορετικές αναπαραστάσεις. Οι χρήστες έχουν ενεργητικό ρόλο .

10. Ευφυή συστήματα διδασκαλίας (Intelligent Tutoring Systems - ITS) . Τα ITS είναι πολύπλοκα προγράμματα που χρησιμοποιούνται από τους εκπαιδευόμενους σαν ένας ακούραστος δάσκαλος που προσαρμόζεται στις γνωστικές ιδιαιτερότητες και στην εξατομικευμένη πρόοδο κάθε μαθητή. Τέτοια συστήματα βασίζονται σε ένα μεγάλο ποσό γνώσης από τον εκπαιδευτικό χώρο και πολλά από αυτά χρησιμοποιούν παιδαγωγικές μεθόδους . Οι ιδιαιτερότητες των μαθητών και η πρόοδος τους αποθηκεύονται σε αυτό που αποκαλούμε «μοντέλο μαθητή». Αυτό επιτυγχάνεται χρησιμοποιώντας κυρίως τεχνικές Τεχνητής Νοημοσύνης για την αναπαράσταση των παιδαγωγικών αποφάσεων καθώς της γνώσης και τις πληροφορίες που αφορούν κάθε μαθητή . Η βασική αρχιτεκτονική ενός Ευφυούς Συστήματος Διδασκαλίας αποτελείται από τα ακόλουθα συστατικά μέρη: (α) το επίπεδο που περιέχει το εκπαιδευτικό υλικό, (β) το μοντέλο μαθητή που καταγράφει πληροφορίες που αφορούν τον χρήστη, (γ) το παιδαγωγικό μοντέλο που

εμπεριέχει γνώση σχετικά με τις διάφορες παιδαγωγικές αποφάσεις, (δ) τη διασύνδεση χρήστη.

3.2 ΕΙΔΗ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΛΟΓΙΣΜΙΚΟΥ

Ο Means (1994) περιγράφει τέσσερις κατηγορίες δυνατές χρήσεις του υπολογιστή ως προς τη χρήση του για εκπαιδευτικούς σκοπούς: Ο υπολογιστής χρησιμοποιείται για διδασκαλία, για εξερεύνηση, ως εργαλείο και για επικοινωνία. Πολλά από τα είδη εκπαιδευτικού λογισμικού που έχουν αναπτυχθεί μέχρι σήμερα θέτουν τον υπολογιστή στη θέση του καθηγητή ενώ άλλα αξιοποιούν τη δυνατότητα για εξερεύνηση και επικοινωνία, υλοποιώντας έτσι μια περισσότερο εποικοδομητική διδακτική προσέγγιση. Κατά τους Paterson και Strickland (Paterson, Strickland, 1986) το εκπαιδευτικό λογισμικό μπορεί να ταξινομηθεί με κριτήριο τη χρήση του στη μαθησιακή διαδικασία ως εξής :

- Λογισμικό εξάσκησης (Drill & Practice).
- Λογισμικό Παρουσίασης (Tutorial).
- Εκπαιδευτικό Παιχνίδι (Educational game).
- Προσομοίωση (Simulation).
- Επίλυση προβλήματος (Problem solving).
- Περιβάλλοντα Εικονικής Πραγματικότητας (Virtual Reality)

Ένα άλλο κριτήριο ταξινόμησης των εκπαιδευτικών λογισμικών είναι η χρήση των τεχνολογικών μέσων στην κατασκευή τους. Αυτά είναι τα πολυμέσα, υπερμέσα (multimedia) και υπερκείμενα, αλλά πρέπει να σημειωθεί ότι αυτά ενυπάρχουν στις προηγούμενες κατηγορίες, (π.χ ένα λογισμικό εξάσκησης ή ένα λογισμικό παρουσίασης μπορεί να είναι multimedia). Μια άλλη ταξινόμηση εφαρμόζει ως κριτήριο το βαθμό αλληλεπίδρασης μεταξύ λογισμικού και χρήστη διακρίνοντας δύο βασικές κατηγορίες εκπαιδευτικών λογισμικών τα ανοικτά και κλειστά περιβάλλοντα.

Τέλος είναι απαραίτητη η αναφορά στην κατηγοριοποίηση εκπαιδευτικού λογισμικού με κριτήριο την παιδαγωγική προσέγγιση που αυτό υιοθετεί. Σύμφωνα με αυτό το κριτήριο και αναλόγως της παιδαγωγικής προσέγγισης του, αποδίδεται σε ένα εκπαιδευτικό λογισμικό ο χαρακτηρισμός του διερευνητικού εκπαιδευτικού λογισμικού.

Επίσης λογισμικά γενικής χρήσης όπως το Word, το Excel και το Internet θεωρούνται από πολλούς εκπαιδευτικά εργαλεία επειδή με αυτά μπορούν να σχεδιαστούν και να υλοποιηθούν εκπαιδευτικές δραστηριότητες για διάφορα γνωστικά αντικείμενα. Στη συνέχεια, περιγράφονται τα χαρακτηριστικά των παραπάνω ειδών εκπαιδευτικού λογισμικού μαζί με τα ισχυρά και αδύνατα σημεία τους.

Λογισμικό εξάσκησης. Τα προγράμματα αυτά δίνουν τη δυνατότητα στους μαθητές να εξασκηθούν σε ύλη την οποία ήδη έχουν διδαχθεί. Είναι αξιοποιήσιμα από τους καθηγητές που θέλουν να εξασκήσουν τους μαθητές τους σε ένα συγκεκριμένο θέμα αλλά και να έχουν τη δυνατότητα άμεσου ελέγχου των επιδόσεων των μαθητών τους. Συμπεριλαμβάνουν ένα σύνολο ερωτήσεων - ασκήσεων στις οποίες ο μαθητής καλείται να απαντήσει και να αξιολογηθεί ανάλογα με την επίδοσή του. Αν και θεωρούνται

κατάλληλα για την επανάληψη, δεν εμφανίζονται τα τελευταία χρόνια ως αυτόνομο λογισμικό αλλά ενσωματώνονται σε άλλου τύπου λογισμικά.

Λογισμικό παρουσίασης. Αυτού του είδους το λογισμικό μπορεί να παρουσιάζει την ήδη διδαγμένη ύλη ή και την ύλη που δεν έχει ακόμη διδαχθεί. Ένα καλό πρόγραμμα παρουσίασης διαθέτει οθόνες βοήθειας (help screen) που παρέχουν περισσότερες πληροφορίες εξηγήσεις και παραδείγματα. Η λειτουργία αυτών των προγραμμάτων είναι παρόμοια με αυτή που κάνει ένας καθηγητής ή ένα σχολικό βιβλίο όταν παρουσιάζει νέες έννοιες ή δίνει πληροφορίες στους μαθητές. Ο υπολογιστής παρουσιάζει τις καινούριες έννοιες με χρήση κειμένου, παραδειγμάτων, animation, video, περιγραφής, ερωτήσεων και προβλημάτων. Ο κύκλος πληροφορία-ερώτηση- ανάδραση- επαναλαμβάνεται καθ' όλη τη διάρκεια της εφαρμογής. Ένα «ιδανικό» πρόγραμμα παρουσίασης παρέχει τη δυνατότητα παρουσίασης του υλικού ανάλογα με τις ανάγκες του κάθε χρήστη - μαθητή.

Διδακτικά και Μορφωτικά παιχνίδια. Βοηθούν στην απόκτηση και ανάπτυξη δεξιοτήτων σε περιβάλλον παιχνιδιού. Το παιχνίδι χρησιμοποιείται ως κίνητρο για το μαθητή ώστε αυτός να αποκτήσει συγκεκριμένες δεξιότητες και γνώσεις καθώς περιηγείται στο λογισμικό και ολοκληρώνει τις διαδικασίες του παιχνιδιού. Είναι μάλλον το πιο δύσκολο είδος λογισμικού για αξιολόγηση δεδομένου ότι είναι αμφίβολο αν οι ικανότητες που προωθούνται από το παιχνίδι είναι δυνατόν να μεταφέρουν και έννοιες. Είναι σημαντικό να πούμε πως ένα καλό παιχνίδι πρέπει να έχει μαθησιακούς στόχους συμβατούς με το πρόγραμμα σπουδών. Τα εκπαιδευτικά παιχνίδια δίνουν στους μαθητές τη δυνατότητα να διδαχθούν σε ένα πλαίσιο που αυξάνει την κινητοποίηση και τον ενθουσιασμό αλλά και την προσοχή στη μαθησιακή διαδικασία

Προσομοίωση. Η προσομοίωση δίνει τη δυνατότητα υλοποίησης καταστάσεων που δεν θα ήταν δυνατόν να υλοποιηθούν με άλλο τρόπο. Με την προσομοίωση δίνεται η ευκαιρία να δοκιμαστεί η ικανότητα και τα αντανακλαστικά των χρηστών σε πραγματικές συνθήκες. Στην τάξη η προσομοίωση δίνει τη δυνατότητα να εξασκηθεί ο μαθητής σε καταστάσεις που σε πραγματικό περιβάλλον θα ήταν ριψοκίνδυνο να υλοποιηθούν (π.χ ένα επικίνδυνο πείραμα χημείας κ.λ.π.), ή ακόμα σε καταστάσεις που η υλοποίηση τους είναι πολυδάπανη, χρονοβόρος κλπ. Η προσομοίωση δημιουργεί μια αναπαράσταση ή ένα μοντέλο ενός πραγματικού συστήματος ή φαινομένου στην οθόνη κάτω από ρεαλιστικές συνθήκες. Η αναπαράσταση αυτής της διαδικασίας πρέπει να δίνεται «ζωντανά», να εμπλέκει

αρκετά το χρήστη ώστε η «εμπειρία» να έχει νόημα, να έχει μια ποικιλία στόχων, να λύνει προβλήματα. Μπορεί να αποτελέσει ιδανικό εργαλείο για την ενσωμάτωση διαφορετικών επιστημών στην ίδια επιστημονική ενότητα, κυρίως για Φυσική, Μαθηματικά, Κοινωνιολογία και Γλώσσα. Τα λογισμικά προσομοίωσης κατά κανόνα στηρίζονται σε μια σειρά αλγορίθμων και οι μαθητές έχουν τη δυνατότητα να αλλάζουν τις τιμές ορισμένων μεταβλητών και να παρατηρούν τα αποτελέσματα της πράξης τους.

Λογισμικό επίλυσης προβλήματος. Αυτά τα προγράμματα ζητούν από τους μαθητές να επιλύσουν προβλήματα στηριζόμενοι σε γνώσεις που αποκτήθηκαν νωρίτερα. Στα

περισσότερα προγράμματα προσομοίωσης και στα παιχνίδια υπάρχουν χαρακτηριστικά problem solving. Μπορούν να αποτελέσουν σπουδαία εναύσματα για τη διερευνητική μάθηση στην τάξη. Δίνουν τη δυνατότητα να εφαρμοστούν οι κλασικές στρατηγικές επίλυσης προβλημάτων. Επίσης βοηθούν το μαθητή να δημιουργήσει και να αναπτύξει περισσότερο τη δική του στρατηγική επίλυσης (Paterson & Strickland 1986). Προσφέρουν ένα πλαίσιο μέσα στο οποίο οι μαθητές μπορούν να αποκτήσουν και να βελτιώσουν τις δεξιότητες επίλυσης προβλήματος. Αυτό το πλαίσιο τις περισσότερες φορές προσομοιώνει ένα πραγματικό φαινόμενο. Τα προγράμματα προσομοίωσης θα πρέπει να ζητούν από το χρήστη να εφαρμόσει αποδεκτές αρχές ή κανόνες για να καταλήξει σε συμπεράσματα και λύσεις. Θα πρέπει επίσης να δίνουν στο χρήστη τη δυνατότητα να παραμετροποιεί το πρόβλημα και να περιλαμβάνουν μια εξήγηση ή μια γραφική αναπαράσταση της τελικής κατάστασης από τις απαντήσεις που δόθηκαν κατά την προσπάθεια επίλυσης. Τέλος η χρήση τους θα πρέπει να αναπτύσσει στο χρήστη μια εκτίμηση και κατανόηση των αλγοριθμικών μεθόδων, να αποθαρρύνουν τις διαδικασίες δοκιμής - λάθους και αντίστοιχα να ενθαρρύνουν αποφάσεις και λύσεις που απορρέουν από πνευματική διαδικασία.

Περιβάλλοντα Εικονικής Πραγματικότητας. Οι τεχνολογίες εικονικής πραγματικότητας χρησιμοποιούνται προς το παρόν κυρίως σε ερευνητικό επίπεδο και λιγότερο στην εκπαιδευτική διαδικασία δεδομένου ότι έχουν πολύ πρόσφατα εμφανιστεί και απαιτούν συνήθως εξειδικευμένα τεχνολογική υποδομή και εξοπλισμό. Ως εικονική πραγματικότητα ορίζεται η αλληλεπίδραση σε τρισδιάστατο χώρο που χρησιμοποιείται για προσομοίωση πραγματικών ή μη καταστάσεων.

Η επιστημονική και τεχνολογική τεκμηρίωση των εφαρμογών της εικονικής πραγματικότητας στη διδασκαλία δεν μπορεί να θεωρηθεί ακόμη πλήρως εμπειριστατωμένη. Ωστόσο χαρακτηριστικά της εικονικής πραγματικότητας όπως η ισχυρή αλληλεπίδραση, η άμεση ανταπόκριση του συστήματος στις ενέργειες του χρήστη και η ελευθερία κινήσεων στους εικονικούς κόσμους μπορούν να επηρεάσουν θετικά τη μαθησιακή διαδικασία.

3.3 ΕΦΑΡΜΟΓΕΣ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΛΟΓΙΣΜΙΚΟΥ

ΦΥΣΙΚΗ – ΠΡΟΣΟΜΟΙΩΣΕΙΣ ΠΕΙΡΑΜΑΤΩΝ



Περιγραφή

Το λογισμικό Φυσική – Προσομοιώσεις πειραμάτων είναι ένα ολοκληρωμένο μαθησιακό υπολογιστικό περιβάλλον, του οποίου η χρήση στοχεύει στο να βοηθήσει τη διδασκαλία της φυσικής κυρίως στο λύκειο. Η επιλογή των θεματικών ενοτήτων και ο σχεδιασμός των εκπαιδευτικών περιβαλλόντων που περιλαμβάνει στηρίχθηκαν σε παιδαγωγικές μεθόδους διδασκαλίας (διερευνητική, ανακαλυπτική) οι οποίες κρίθηκαν κατάλληλες από τους επιστημονικούς υπεύθυνους για την κατανόηση φυσικών φαινομένων και εννοιών.

Τύπος εκπαιδευτικού λογισμικού

Ανοικτό μαθησιακό περιβάλλον

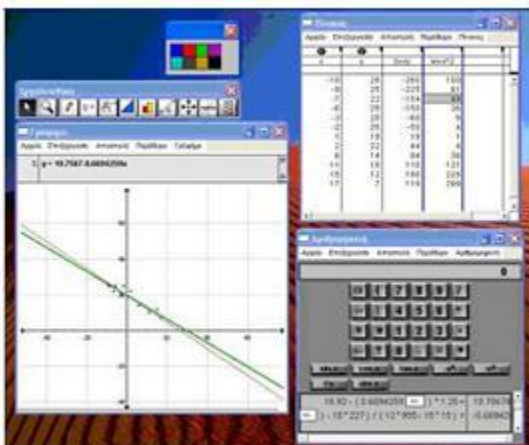
Θεματική κατάταξη

Φυσική αγωγή

Εταιρεία ανάπτυξης

Edusoft

Function Probe



Περιγραφή

Πολυ-εποπτικό εργαλείο για τη σύγχρονη άλγεβρα, την τριγωνομετρία και την ανάλυση, που επιτρέπει τη διερεύνηση των συναρτήσεων και την μαθηματική μοντελοποίηση. Το εκπαιδευτικό λογισμικό διερευνητικού χαρακτήρα Function Probe είναι ένα κατεχοχόν «ανοιχτό» περιβάλλον μάθησης. Το Function Probe είναι προορισμένο να χρησιμοποιείται σε μαθήματα άλγεβρας, τριγωνομετρίας και ανάλυσης (πριν τον διαφορικό και τον ολοκληρωτικό λογισμό) σε τάξεις του Γυμνασίου και του Λυκείου. Αποτελεί ένα ευέλικτο και δυναμικό εργαλείο, σχεδιασμένο έτσι ώστε να είναι εξίσου εύκολο στην εκμάθηση και τη χρήση. Οι μαθητές έχουν την δυνατότητα να συμμετέχουν ενεργά στη χρήση του προγράμματος κατά την διάρκεια της προσπάθειας που καταβάλλουν στην επίλυση προβλημάτων. Οι καθηγητές έχουν την δυνατότητα να χρησιμοποιούν το Function Probe για να επιδεικνύουν τεχνικές επίλυσης προβλημάτων και για να καθοδηγούν συζητήσεις μέσα στην τάξη. Το Function Probe δεν είναι αυστηρά συνδεδεμένο με συγκεκριμένο διδακτικό υλικό. Αντίθετα, είναι σχεδιασμένο έτσι ώστε να είναι συμβατό με ενέργειες και αναπαραστάσεις που οι μαθητές δημιουργούν και χρησιμοποιούν σε μια ποικιλία προβλημάτων, που μπορεί να συναντήσουν μελετώντας μαθηματικά. Το Function Probe είναι, επίσης, ένα ιδανικό εργαλείο για την επεξεργασία και την μοντελοποίηση δεδομένων σε μαθήματα που βασίζονται στην άλγεβρα και την τριγωνομετρία. Πρόκειται για ένα πολυ-εποπτικό πακέτο που περιλαμβάνει τρία ξεχωριστά εργαλεία: το Γράφημα, τον Πίνακα δεδομένων και την Αριθμομηχανή.

Τύπος εκπαιδευτικού λογισμικού

Εργαλείο εκπαιδευτικού λογισμικού

Θεματική κατάταξη

Μαθηματικά

Εταιρεία ανάπτυξης

Cornel Research Foundation Quest Math & Science Multimedia, Inc

eCity – Εικονική Πόλη



Περιγραφή

Ολοκληρωμένο εκπαιδευτικό παιχνίδι συνεργατικής μάθησης που υποστηρίζει τη δημιουργία προσομοιώσεων στο πλαίσιο μίας εικονικής πόλης. Περιλαμβάνει σύνθετα προβλήματα που μπορούν να εντοπιστούν στο περιβάλλον μίας πόλης (διανομή ενέργειας, ανανεώσιμες πηγές ενέργειας, προστασία από πλημμύρες, ρύπανση, δίκτυο ΜΜΕ, κ.α.) Ακολουθεί την προβληματοκεντρική μάθηση (PBL), μία πρακτική ενεργητικής μάθησης κατά την οποία οι μαθητές διερευνούν τον τρόπο επίλυσης ενός προβλήματος, συχνά συνδεδεμένου με καταστάσεις της πραγματικής ζωής. Συμβάλλει στην ανάπτυξη γνώσεων σε μαθηματικά και επιστήμες για χρήση στο πλαίσιο εργασίας ενός ατόμου ως επιστήμονας μηχανικός. Καλεί τους μαθητές να λύσουν σύνθετα προβλήματα που απαιτούν την εφαρμογή γνώσεων από διαφορετικά πεδία επιστημών και σχετίζονται με την ασφάλεια, την ποιότητα ζωής, τις επικοινωνίες, την παραγωγή ενέργειας, τη μόλυνση του περιβάλλοντος και άλλα σχετικά θέματα.

Τύπος εκπαιδευτικού λογισμικού

Ανοικτό μαθησιακό περιβάλλον

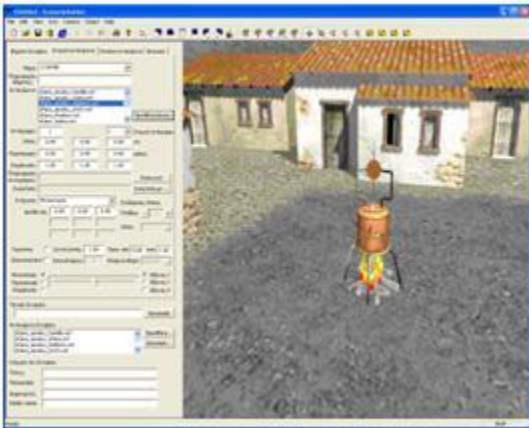
Θεματική κατάταξη

Περιβαλλοντική Εκπαίδευση / Εκπαίδευση για την Αειφόρο Ανάπτυξη

Εταιρεία ανάπτυξης

Multilateral Projects

Άξονες



Περιγραφή

Το Ολοκληρωμένο Προϊόν Εκπαιδευτικού Λογισμικού «ΑΞΟΝΕΣ» αποτελεί ένα περιβάλλον μάθησης που αξιοποιεί τις σύγχρονες εφαρμογές των Τεχνολογιών Πληροφορικής και Επικοινωνίας (κυρίως το Διαδίκτυο και τα αλληλεπιδραστικά πολυμέσα) για να υποστηρίξει εκπαιδευτικές δραστηριότητες ενεργούς διερεύνησης και οικοδόμησης των γνώσεων σε συνθήκες κοινωνικής αλληλεπίδρασης και συνεργασίας στα γνωστικά αντικείμενα της Τεχνολογίας Α' και Γ' Γυμνασίου, της Βιολογίας Α' και Γ' Γυμνασίου και της Ιστορίας Β' και Γ' Γυμνασίου. Το Ολοκληρωμένο Προϊόν Εκπαιδευτικού

Λογισμικού «ΑΞΟΝΕΣ» αποτελείται από τα τρία παρακάτω υποσυστήματα: Το ΑΞΟΝΕΣ 3Δ Εργαλείο Συγγραφής: είναι λογισμικό δημιουργίας τρισδιάστατων εικονικών αναπαραστάσεων το οποίο επιτρέπει στο χρήστη να δημιουργήσει ή να μετατρέψει εκπαιδευτικές δραστηριότητες εντός των εικονικών κόσμων για διάφορα γνωστικά αντικείμενα. Το ΑΞΟΝΕΣ 3Δ Εφαρμογή Εκτέλεσης είναι το περιβάλλον εκτέλεσης των τρισδιάστατων εικονικών αναπαραστάσεων και των εμπριεχόμενων εκπαιδευτικών δραστηριοτήτων που δημιούργησε ο καθηγητής με το ΑΞΟΝΕΣ 3Δ Εργαλείο Συγγραφής, από τους μαθητές. Η εφαρμογή εκτέλεσης είναι ενσωματωμένη και προσβάσιμη μέσα από το Διαδικτυακό Περιβάλλον Ολοκλήρωσης και ενοποιείται με τις εκπαιδευτικές λειτουργίες του ολοκληρωμένου περιβάλλοντος. Το ΑΞΟΝΕΣ Αυτοτελές Διαδικτυακό Περιβάλλον Ολοκλήρωσης είναι μια διαδικτυακή πλατφόρμα διάθεσης εκπαιδευτικού περιεχομένου και εκπαιδευτικών σεναρίων (συμπεριλαμβανομένων και των τρισδιάστατων εικονικών αναπαραστάσεων και δραστηριοτήτων εντός αυτών) για τα προσφερόμενα.

Τύπος εκπαιδευτικού λογισμικού

Ανοικτό μαθησιακό περιβάλλον

Θεματική κατάταξη

Βιολογία

Εταιρεία ανάπτυξης

SYSTEMA TECHNOLOGIES S.A

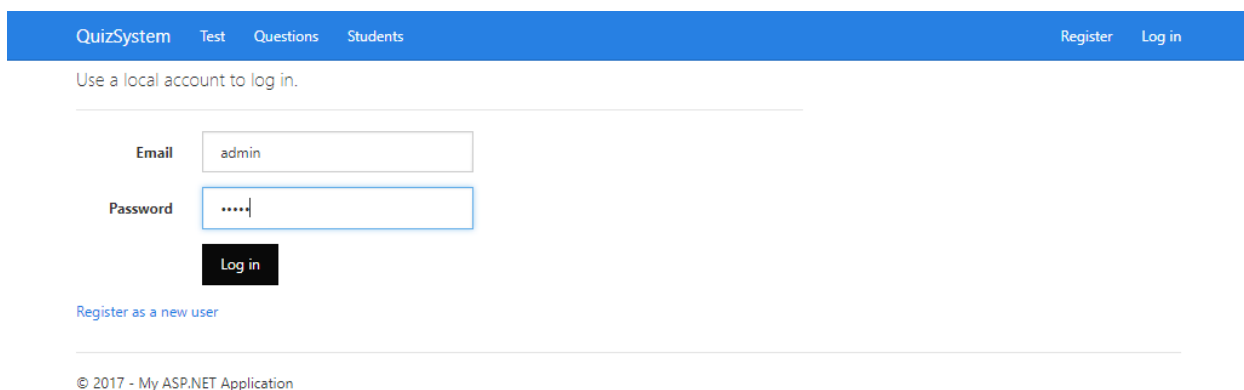
ΚΕΦΑΛΑΙΟ 4 - Παρουσίαση και χρήση εφαρμογής

4.1 Εγχειρίδιο χρήστη – περιβάλλον καθηγητή

Όπως έχουμε αναφέρει ήδη, το συγκεκριμένο project αφορά την ανάπτυξη μιας διαδικτυακής εκπαιδευτικής εφαρμογής. Στην συγκεκριμένη εφαρμογή ο εκάστοτε χρήστης μπορεί να συνδεθεί ως μαθητής η καθηγητής ανάλογα με την ιδιότητα του. Παρακάτω θα αναφέρουμε αναλυτικά την διαδικασία σύνδεσης του ρόλου “καθηγητή” καθώς και τις βασικές λειτουργίες που μπορεί να εκτελέσει.

4.1.1 Αρχική οθόνη

Στην αρχική οθόνη της εφαρμογής ο χρήστης καλείται να πληκτρολογήσει το email και τον κωδικό του ώστε να μπορεί αν συνδεθεί στην εφαρμογή (εικόνα 1). Αν ο χρήστης δεν είναι δηλωμένος στο σύστημα μπορεί να δημιουργήσει έναν νέο λογαριασμό από την επιλογή “register as new user”, αφού πληκτρολογήσει το όνομα, το επίθετο, το κωδικό και το email του (εικόνα 2).



QuizSystem Test Questions Students Register Log in

Use a local account to log in.

Email admin

Password ****

Log in

[Register as a new user](#)

© 2017 - My ASP.NET Application

εικόνα 1- Αρχική οθόνη

QuizSystem Test Questions Students Register Log in

Account

Password

FirstName

LastName

Email

Create

© 2017 - My ASP.NET Application

εικόνα 2 – οθόνη εγγραφής χρήστη

4.1.2 Διαχειριστική οθόνη καθηγητή

Σε αυτή την οθόνη ο χρήστης με ρόλο “καθηγητής” μπορεί να δει βασικές πληροφορίες που αφορούν τους μαθητές του. Αναλυτικότερα μπορεί να δει το σύνολο των χρηστών που έχουν εγγραφεί στην εφαρμογή, το σύνολο των μαθητών που έχουν ολοκληρώσει κάποιο διαγώνισμα (Τεστ) αλλά και το μέσο όρο της βαθμολογίας του. Τέλος στο κάτω μέρος της οθόνης προβάλεται μια λίστα με τους μαθητές, το ακριβές ποσοστό επιτυχίας (score) αλλά και το κεφάλαιο που έχουν επιλέξει ότι τους αρέσει περισσότερο βάση των ερωτήσεων που έχουν απαντήσει.

QuizSystem Test Questions Students Hello admin admin! Log Off

Quiz System

Quiz System is a service for students where they can pass test and get results online, all scores saved to database.

Users

6 students have already registered.

[Register »](#)

Questions

3 different chapters, 31 questions in total. Login and try yourself.

[Login »](#)

Scores

3 tests already completed. Average score is 11,11 %

[Pass Test »](#)

Quiz results

Student Name	Score	Chapter liked most
kyriakos karageo	8,333333333333333	Chapter1
student student	25	Chapter3
student student	0	Chapter3

4.1.3 Οθόνη διαχείρισης ερωτήσεων (test)

Σε αυτή την οθόνη εμφανίζονται τα κεφάλαια ύλης, ο αριθμός των ερωτήσεων που περιέχουν ανα κατηγορία καθώς και αριθμός των likes που έχουν κάνει οι μαθητές .

QuizSystem Test Questions Students Hello admin admin! Log Off

Chapters

Chapter Name	Gap Questions	Variants Questions	Likes
Chapter1	5	18	1 Details
Chapter2	4	0	0 Details
Chapter3	4	0	2 Details

© 2017 - My ASP.NET Application

Επιπλέον στην παραπάνω οθόνη υπάρχει η επιλογή Details (λεπτομέρειες), με την οποία μπορούμε να προσθέσουμε ερωτήσεις στο συγκεκριμένο κεφάλαιο , να επεξεργαστούμε τις υπάρχουσες ή και να διαγράψουμε.

Question text	Answer	
question_gap_____	yellow	Edit Delete
question2_gap____	green	Edit Delete
question2_gap_____	asda	Edit Delete
question2_gap_____	question2	Edit Delete
questiongap_____	yellow	Edit Delete

4.1.4 Δημιουργία νέας ερώτησης συμπλήρωσης κενών

Με την επιλογή create new (δημιουργία ερώτησης) μπορούμε να εισάγουμε μια νέα ερώτηση επιλέγοντας το κείμενο και την σωστή απάντηση.

QuizSystem	Test	Questions	Students	Hello admin admin!	Log Off
EditGapQuestion					
QuestionGap					
Text	<input type="text" value="question_gap_____"/>				
Answer	<input type="text" value="yellow"/>				
ChapterID	Chapter1				
	<input type="button" value="Save"/>				
Back to List					
© 2017 - My ASP.NET Application					

4.1.5 Δημιουργία ερωτήσεων πολλαπλής επιλογής

Σε αυτή την οθόνη εμφανίζεται η λίστα των ερωτήσεων πολλαπλής επιλογής , μέσω της οποίας μπορούμε να δημιουργήσουμε νέα, να διαγράψουμε τις υπάρχουσες ή και να τις επεξεργαστούμε.

QuizSystem	Test	Questions	Students	Hello admin admin!	Log Off
hgfdls			@	Edit Delete	
question 1?		A a B a C b	A a	Edit Delete	
question_2?		A a B a C b	A a	Edit Delete	
question 3?		A a B a C b	A a	Edit Delete	
question 4?		A a B b C a	A a	Edit Delete	
question 6?		A a B b C a	A a	Edit Delete	
question 7?		A a B a C b	A a	Edit Delete	
question 8?		A a B a C b	A a	Edit Delete	
question 9?		A a B b C a	A a	Edit Delete	

4.1.6 Επεξεργασία ερωτήσεων πολλαπλής επιλογής.

Στην συγκεκριμένη οθόνη ο χρήστης μπορεί να δημιουργήσει μια νέα ερώτηση πολλαπλής επιλογής, επιλέγοντας το κεφάλαιο που θέλει, το κείμενο της ερώτησης, τις πιθανές απαντήσεις αλλά και την σωστή επιλογή.

QuizSystem Test Questions Students Hello admin admin! Log Off

Edit Variant Question

Select Chapter
Chapter1 ▼

Write question
question 1?

Write answer
a Add Variant

Write another possible answers
a Remove
b Remove

Submit

[Back to List](#)

© 2017 - My ASP.NET Application

4.1.7 Οθόνη μαθητών

Σε αυτή την οθόνη ο χρήστης μπορεί να δει μια λίστα με τους μαθητές που έχουν εγγραφεί στο σύστημα, να δει περισσότερες λεπτομέρειες, να επεξεργαστεί αλλά και να διαγράψει από την λίστα.

Index

Not Approved

AccountType	FirstName	LastName	Email	Password	
Student	kyriakos	karageo	gmail	pass	Edit Details Delete
Teacher	admin	admin	admin	admin	Edit Details Delete
Student	student	student	student@gmail.com	student	Edit Details Delete
Student	dd	dsfe	me	me	Edit Details Delete
Student	kyriakos	kyriakos	kyriakos@gmail.com	123456	Edit Details Delete
Student	kyriakos	kyriakos	kyriakos@gmail.com	kyriakos	Edit Details Delete
Student	kyriakos	kyriakos	kyriakos@gmail.com	123456	Edit Details Delete

© 2017 - My ASP.NET Application

4.1.8 Οθόνη έγκρισης εγγραφής μαθητών

Σε αυτή την οθόνη ο χρήστης μπορεί να δει τους μαθητές που έχουν κάνει αίτηση έγκρισης λογαριασμού και έχει την δυνατότητα να εγκρίνει το αίτημα ή και αν το διαγράψει.

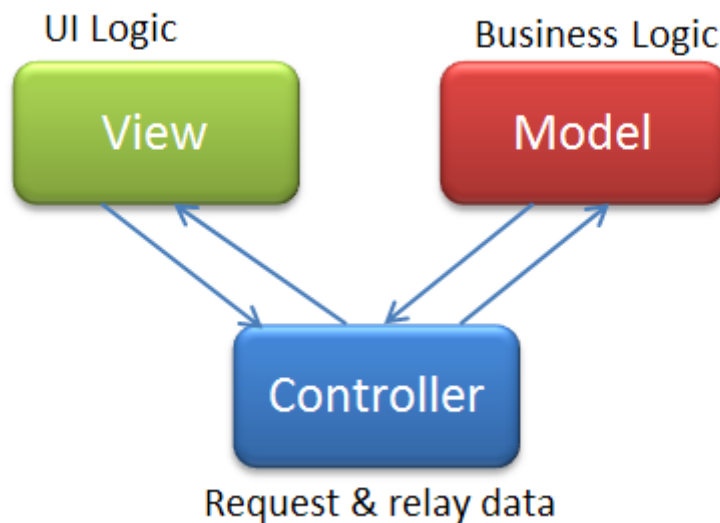
For Approve

FirstName	LastName	Email	Password	
kyriakos	kyriakos	kyriakos@gmail.com	kyriakos	Approve Delete
kyriakos	kyriakos	kyriakos@gmail.com	123456	Approve Delete

© 2017 - My ASP.NET Application

4.2 Αρχιτεκτονική της εφαρμογής

Η παρούσα εργασία είναι ένα ASP.NET MVC 5 project , όπου το MVC 5 είναι ένα template που προσφέρει το Visual Studio, όσον αφορά την δημιουργία διαδικτυακών εφαρμογών. Το συγκεκριμένο template προσφέρει τόσο την οργάνωση των αρχείων και των φακέλων που πρέπει να αποτελείται το project, όσο και την λογική με την οποία αυτά πρέπει να αλληλεπιδρούν με τον χρήστη. Στο MVC template, όπως αυτό υλοποιήθηκε από τη Microsoft και εμφανίζεται στην εικόνα 1, η οργάνωση και τα ονόματα των φακέλων και των αρχείων έχουν ιδιαίτερη σημασία. Σε αυτό το σημείο πρέπει να αναφερθεί ότι, στην πραγματικότητα το μοντέλο MVC (Model View Controller) της Microsoft είναι βασισμένο στο MVVC (Model ViewModel View Controller).



- **Model:** Είναι οι κλάσεις που αναπαριστούν τις οντότητες που διαχειρίζεται η εφαρμογή. Για παράδειγμα, στην παρούσα εφαρμογή υπάρχει η κλάση Clip που έχει όλες τις ιδιότητες και ενέργειες που θέλουμε να έχει ένα Video Clip, καθώς και το πώς αυτή θα αποθηκεύεται ή και θα εξάγεται από την βάση δεδομένων.
- **ViewModel:** Είναι οι κλάσεις που μοντελοποιούν το πώς θα αλληλεπιδρούν με το View. Για παράδειγμα, στην κλάση Clip του μοντέλου υπάρχει ένα πεδίο Fingerprint που κρατάει την πληροφορία που θα βοηθήσει στην αναζήτηση του βίντεο, αλλά στον χρήστη δεν πρέπει να εμφανίζεται αυτή η πληροφορία. Έτσι, στο ViewModel δεν εμφανίζεται ως πεδίο. Το ViewModel βασίζεται στις κλάσεις του Model και είναι τις περισσότερες φορές υποσύνολο όσον αφορά τα πεδία της αντίστοιχης κλάσης του Model.

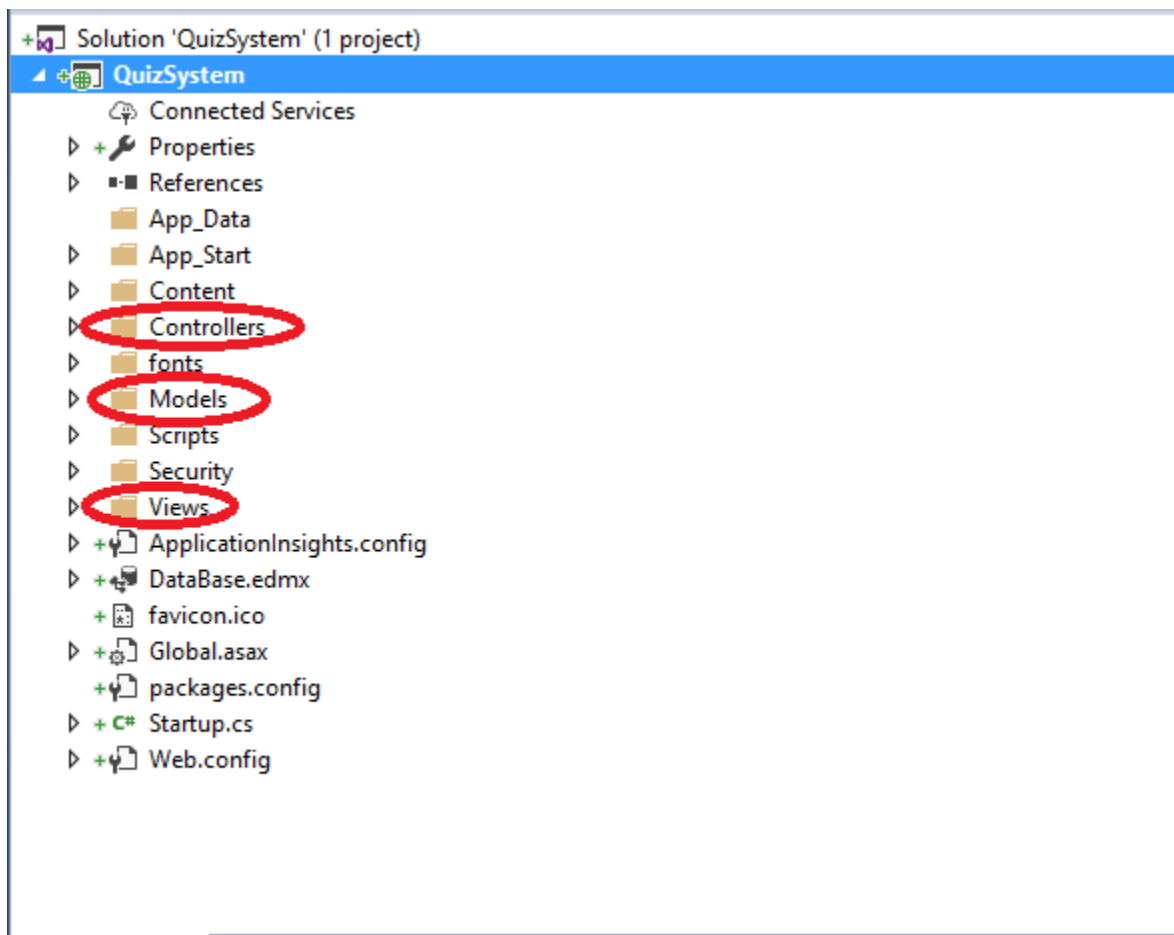
- View: Είναι τα αρχεία που διαμορφώνουν το τι θα εμφανίζεται στον χρήστη βάσει του ViewModel. Συνήθως, όπως και σε αυτή την εφαρμογή, είναι οι html (cshtml) σελίδες.
- Controller: Είναι οι κλάσεις που περιέχουν τις μεθόδους που εκτελούνται έπειτα από ενέργειες του χρήστη ή της εφαρμογής. Ο Controller αποτελείται από καθαρό κώδικα (code behind), C# στην περίπτωση αυτή, που εκτελεί την λογική που πρέπει να αποδοθεί στις ενέργειες του χρήστη.

Το μοντέλο ανάπτυξης MVVC μπορεί να χρησιμοποιηθεί σε πληθώρα εφαρμογών (διαδικτυακές εφαρμογές, εφαρμογές desktop, εφαρμογές για κινητές συσκευές κτλ) και προσφέρει πολλά πλεονεκτήματα:

- Διαχωρισμός της εμφάνισης από τις οντότητες. Έτσι, αν χρειαστεί κάποια στιγμή αλλαγή της εμφάνισης, δεν επηρεάζεται κάποιο άλλο μέρος της εφαρμογής. Επιπλέον μπορεί να αλλάξει η οντότητα ή/και ο τρόπος που αυτή αποθηκεύεται στην βάση δεδομένων, χωρίς να υπάρχει ανησυχία για την εμφάνιση της στον χρήστη.
 - Διαχωρισμός της λογικής (ενεργειών) από την εμφάνιση ή/και των οντοτήτων. Και πάλι είναι δυνατόν, αν σε μια μεταγενέστερη έκδοση της εφαρμογής αποφασιστεί ότι θα πρέπει να γίνει κάτι επιπλέον, για παράδειγμα υπολογισμός στατιστικών στοιχείων και αποθήκευση τους στη βάση, να μπορεί να μεταβληθεί ο κώδικας, ώστε να εκτελέσει την επιπλέον ενέργεια χωρίς να χρειαστεί να μεταβληθεί κάποιο άλλο κομμάτι της εφαρμογής. Έτσι, τα πλεονεκτήματα του MVVC είναι:
1. Συντηρησιμότητα (Maintainability)
 2. Επεκτασιμότητα (Scalability). Για παράδειγμα, οι κλάσεις του Model μπορούν να χρησιμοποιηθούν και σε άλλη εφαρμογή.
 3. Προγραμματισμός βάσει δοκιμών (Test driven programming): Τα ξεχωριστά μέρη της εφαρμογής μπορούν να αναπτυχθούν και να δοκιμαστούν ξεχωριστά

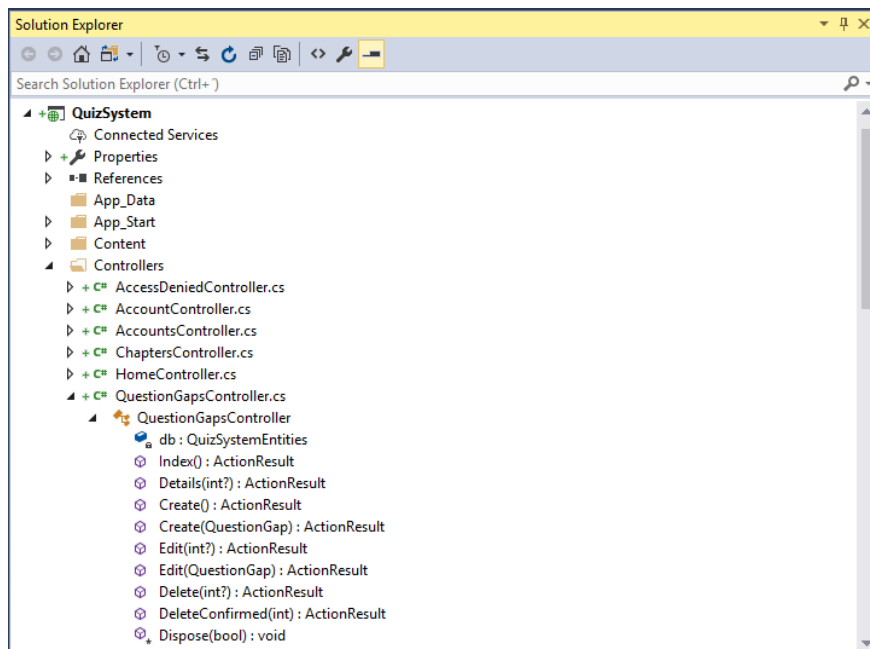
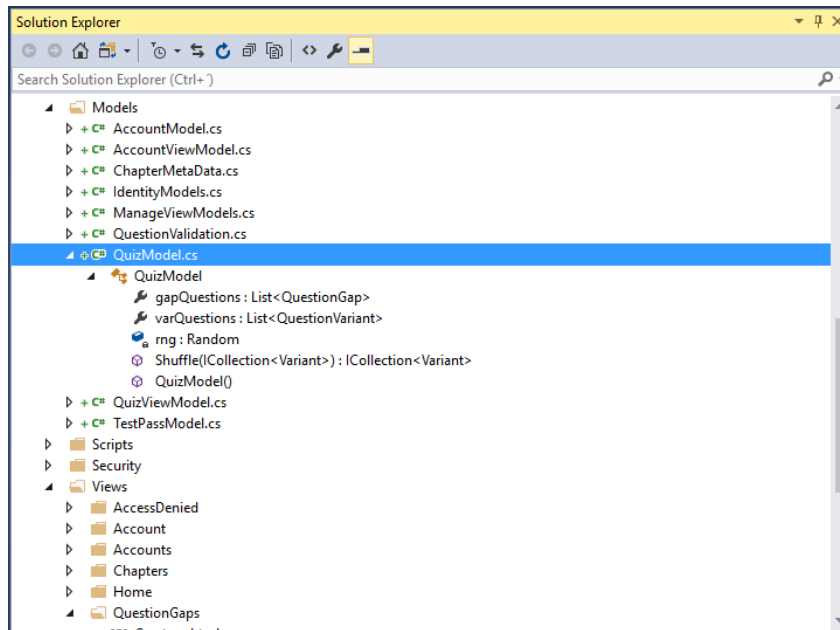
4.2.1 MVC - Αντιστοιχία με την Εφαρμογή

Στις εικόνες 2 και 3 παρουσιάζεται η οργάνωση των φακέλων και των αρχείων. Στη συνέχεια θα δοθεί ένα παράδειγμα χρήσης του μοντέλου MVC, για να εξεταστεί η σημασία που έχουν στο πρότυπο MVC. Όπως φαίνεται στην εικόνα 2, για κάθε στοιχείο του μοντέλου MVC υπάρχει και ο αντίστοιχος φάκελος (το Model και το ViewModel βρίσκονται στον φάκελο Models).

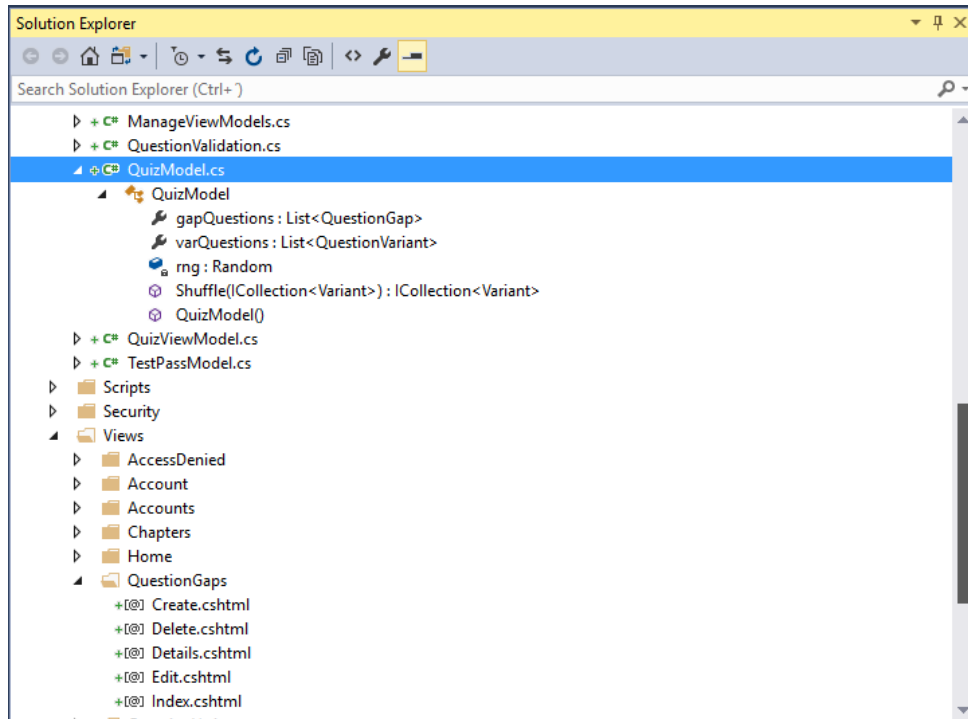


Στην εικόνα 3 παρουσιάζονται τα περιεχόμενα των φακέλων. Με μπλε χρώμα έχουν επισημανθεί τα αρχεία και οι μέθοδοι που θα χρησιμοποιηθούν στο παράδειγμα. Τα ονόματα των φακέλων και των αρχείων έχουν ιδιαίτερη σημασία στην υλοποίηση του MVC από την Microsoft, για τον λόγο αυτό θα μαρκάρονται σύμφωνα με την αλληλεπίδραση που έχουν. Συγκεκριμένα:

1. Ο φάκελος Controller περιέχει την κλάση QuestionGapsController.
2. Η κλάση QuestionGapsController περιέχει δύο υπερφορτωμένες μεθόδους για την ενέργεια Create.
3. Ο φάκελος Model περιέχει την κλάση QuizModel.



4. Ο φάκελος View περιέχει τον φάκελο QuestionGaps.Στον οποίο βρίσκεται το αρχείο Create.cshtml.



Πριν το παράδειγμα, θα πρέπει να εξεταστεί ακόμη ένα αρχείο που βρίσκεται στον φάκελο App_Start και λέγεται RouteConfig.cs. Το συγκεκριμένο αρχείο περιέχει την ομώνυμη κλάση με την στατική μέθοδο RegisterRoutes.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;
namespace VideoSearch
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id =
                    UrlParameter.Optional }
            );
        }
    }
}
```

}}

Στο προηγούμενο κομμάτι κώδικα γίνεται register η μορφή που θα έχουν τα url της εφαρμογής και το πώς αυτά θα καλούν τα αντίστοιχα μέρη της εφαρμογής. Έτσι, τα url της εφαρμογής έχουν το εξής format: `https://Quiz/ {controller}/ {action}/ {id}` όπου: {controller}: το όνομα του Controller {action}: το όνομα της μεθόδου του Controller {id}: κάποιο προαιρετικό id. Το όνομα του Controller ορίζεται να είναι το όνομα του αρχείου χωρίς την λέξη Controller. Για παράδειγμα, το όνομα του αρχείου είναι `QuestionGapsController`, αλλά στο url θα υπάρχει μόνο η πρώτη λέξη, δηλαδή `Clips`. Επίσης ορίζεται ποιος είναι ο προεπιλεγμένος Controller και ποια η προεπιλεγμένη ενέργεια αυτού κατά την εκκίνηση της εφαρμογής, δηλαδή όταν στο url υπάρχει μόνο το `https://Quiz/` θα εκτελείται η μέθοδος `Index` του `HomeController`. Τώρα είναι η κατάλληλη στιγμή να αναφερθούν οι δύο υπερφορτωμένες μέθοδοι με το όνομα `Create` της κλάσης `QuestionGapsController`. Οι μέθοδοι των Controllers αντιστοιχούν στα http verbs GET και POST, με κάθε μέθοδο να επιστρέφει ένα View στο τέλος των ενεργειών της. Σε ποιο verb αντιστοιχεί κάθε μέθοδος ορίζεται από τον προγραμματιστή με τα properties, γνωστά και ως annotations, που δίνουν επιπλέον ιδιότητες στις μεθόδους. Έστω ότι ο χρήστης επέλεξε να κάνει εισαγωγή ενός βίντεο επιλέγοντας το κατάλληλο link. Τότε στην γραμμή διευθύνσεων θα εμφανιστεί το url

Αυτό σημαίνει ότι θα πρέπει, σε απόκριση του χρήστη, να εκτελεστεί ο κώδικας του `QuestionGapsController` και συγκεκριμένα η μέθοδος `Create`. Σε αυτή την περίπτωση, επειδή δεν αποστέλλεται κάτι στο server αλλά γίνεται αίτηση περιεχομένου από αυτόν, θα εκτελεστεί η `Create` που αντιστοιχεί στο verb GET. Με την εκτέλεση της μεθόδου γίνεται η όποια εργασία ορίστηκε να γίνεται και στο τέλος η μέθοδος επιστρέφει την κλάση του `.Net View`. Λόγω της χρήσης του `QuestionGapsController`, η κλάση `View` θα ψάξει στον φάκελο `View` και θα αναζητήσει τον υποφάκελο `QuestionGaps`. Εκεί θα πρέπει να βρει ένα `cshtml` αρχείο με όνομα ίδιο με αυτό της μεθόδου και θα το εμφανίσει στον χρήστη. Στην συνέχεια ο χρήστης μπορεί να συμπληρώσει τα πεδία του view και να εκτελέσει την 'Υποβολή', δηλαδή να στείλει τα δεδομένα στο server. Έτσι θα εκτελεστεί η μέθοδος `Create` του `QuestionGapsController` που έχει οριστεί για το verb POST. Θα εκτελέσει τις ενέργειες, ενδεχομένως να αποθηκεύσει τα δεδομένα στην βάση και με την σειρά της θα επιστρέψει την κλάση `View`, που και πάλι θα την αναζητήσει στον φάκελο `View` και στον υποφάκελο `Clips` με το όνομα `Create.cshtml`.

4.3 Λεπτομέρειες υλοποίησης

Σε αυτό το υποκεφάλαιο θα παρουσιάσουμε ορισμένες τεχνικές λεπτομέρειες που χρειάστηκε να ρυθμιστούν έτσι ώστε να μπορεί να είναι η εφαρμογή λειτουργική.

4.3.1 Εργαλεία υλοποίησης

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκαν:

- Το περιβάλλον εργασίας `Visual Studio 2015 Express` (IDE: `Intergraded`)

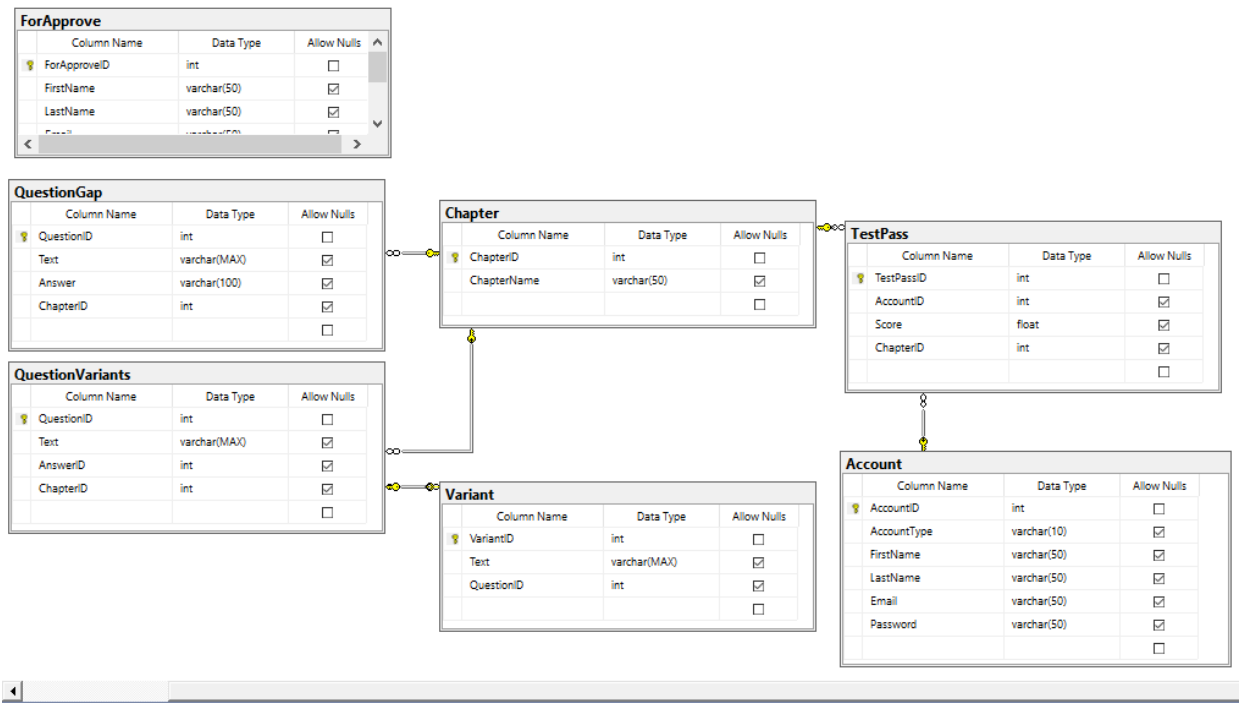
- Development Environment).
- Το .Net framework, έκδοση 4.5.
- Η γλώσσα προγραμματισμού C#, που είναι μέρος του .Net framework.
- Η βάση δεδομένων Microsoft SQL 2016 Express.
- Η τεχνολογία Microsoft Entity Framework 6 της Microsoft για ORM (Object Relational Mapping).
- Το JQuery framework, έκδοση 2.1.3, για client side actions. Είναι βασισμένο σε javascript και θεωρείται στις μέρες μας ως το πιο σταθερό framework για ενέργειες που πρέπει να γίνουν από την πλευρά του browser.
- Το Bootstrap, έκδοση 3.3.1, το γνωστό framework για την διαμόρφωση της εμφάνισης των σελίδων.
- Όλα τα παραπάνω συνδυάστηκαν και χρησιμοποιήθηκαν σε ένα ASP.NET MVC 5 project.

Η εφαρμογή αποτελείται από συνολικά 26 html σελίδες που είναι επιφορτισμένες για την εμφάνιση των δυνατοτήτων της εφαρμογής. Για την ακρίβεια οι σελίδες είναι cshtml, δηλαδή με το πρόθεμα cs (C Sharp) πριν το html, διότι σε αυτές τις σελίδες εκτός από καθαρό html περιεχόμενο μπορούμε να έχουμε και κώδικα C#. Για να μπορεί ο server να ξεχωρίσει των κώδικα C# από το html περιεχόμενο, σε αυτά τα αρχεία ο κώδικας C# ξεκινάει με το σύμβολο @. Η συγκεκριμένη σύνταξη ονομάζεται Razor syntax.

4.4 Βάση δεδομένων

Νομοτελειακά, κάθε εφαρμογή η οποία έχει δυναμικό περιεχόμενο, δηλαδή περιεχόμενο που μεταβάλλεται με τον καιρό, όπως είναι και η παρούσα εφαρμογή, μιας και οι χρήστες της μπορούν ανά πάσα στιγμή να ανεβάζουν δεδομένα και να μεταβάλλουν το περιεχόμενο της, πρέπει να έχει μια βάση δεδομένων. Για τη δημιουργία της βάσης δεδομένων της εφαρμογής μας, χρησιμοποιήθηκε ο Microsoft SQL Server 2017.

Η βάση είναι οργανωμένη σε επτά πίνακες, τέσσερις από τους οποίους σχετίζονται με την διαχείριση των διαγωνισμάτων (tests) και οι υπόλοιποι με την διαχείριση χρηστών. Αυτοί δεν είναι ανεξάρτητοι, αντιθέτως συνδέονται με σχέσεις που εκφράζονται στα σχεσιακά μοντέλα που θα περιγραφούν στη συνέχεια.



4.4.1 Πίνακες βάσης δεδομένων

Πίνακας QuestionGap

Ένας από τους κύριους πίνακες της εφαρμογής είναι ο QuestionGap. Κάθε γραμμή αυτού του πίνακα αντιπροσωπεύει την οντότητα μίας ερώτησης. Τα πεδία που περιέχει είναι QuestionId, Text, Answer, ChapterId.

- QuestionId: INT, δηλώνει το Id του κάθε ερώτησης. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- Text: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- Answer : VARCHAR(100), αποθηκεύει το κείμενο της απάντησης.
- ChapterId: INT, , δηλώνει το Id του κάθε κεφαλαίου. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται ένα νέο κεφάλαιο.

Πίνακας QuestionVariants

Ένας από τους κύριους πίνακες της εφαρμογής είναι ο QuestionVariants. Κάθε γραμμή αυτού του πίνακα αντιπροσωπεύει την οντότητα ερώτησης πολλαπλής επιλογής. Τα πεδία που περιέχει είναι QuestionId, Text, Answer, ChapterId.

- QuestionId: INT, δηλώνει το Id του κάθε ερώτησης. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- Text: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- AnswerId : INT(100), δηλώνει το id της ερώτησης.
- ChapterId: INT, δηλώνει το Id του κάθε κεφαλαίου.

Πίνακας Variant

Ένας άλλος πίνακας τη εφαρμογής είναι ο Variant. Κάθε γραμμή αυτού του πίνακα αντιπροσωπεύει την οντότητα ερώτησης πολλαπλής επιλογής. Τα πεδία που περιέχει είναι VariantId, Text, QuestionId.

- VariantId: INT, δηλώνει το Id του κάθε ερώτησης πολλαπλής επιλογής. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- Text: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- QuestionId : INT, δηλώνει το id της ερώτησης.

Πίνακας Chapter

Ο πίνακας τη εφαρμογής Chapter. Κάθε γραμμή αυτού του πίνακα αντιπροσωπεύει την οντότητα κεφάλαιο. Τα πεδία που περιέχει είναι ChapterId, ChapterName.

- ChapterId: INT, δηλώνει το Id του κάθε κεφαλαίου. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται ένα νέο κεφάλαιο.
- ChapterName: NVARCHAR(MAX), αποθηκεύει το κείμενο με την ονομασία του κεφαλαίου.

Πίνακας TestPass

Ο πίνακας τη εφαρμογής είναι ο TestPass αντιπροσωπεύει την οντότητα ερώτησης πολλαπλής επιλογής. Τα πεδία που περιέχει είναι TestPassId, AccountId, Score,ChapterId.

- TestPassId: INT, δηλώνει το Id του διαγωνίσματος (τεστ) Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- AccountId: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- Score: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- ChapterId

Πίνακας Account

Ο πίνακας τη εφαρμογής Account αντιπροσωπεύει την οντότητα λογαριασμός χρήστη. Τα πεδία που περιέχει είναι τα :

- AccountId: INT, δηλώνει το Id του κάθε ερώτησης. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- AccountType: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- FirstName: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- LastName: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- Email: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- Password: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.

Πίνακας ForApprove

Ένας άλλος πίνακας τη εφαρμογής είναι ο ForApprove. Κάθε γραμμή αυτού του πίνακα περιέχει ένα 'flag' που αντιπροσωπεύει την έγκριση δημιουργίας λογαριασμού ενός χρήστη. Τα πεδία που περιέχει είναι VariantId, Text, QuestionId.

- VariantId: INT, δηλώνει το Id του κάθε ερώτησης. Επίσης, έχει οριστεί να αυξάνεται από το σύστημα κατά μία μονάδα κάθε φορά που εισάγεται μία νέα ερώτηση.
- Text: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.
- QuestionId: NVARCHAR(MAX), αποθηκεύει το κείμενο της ερώτησης.

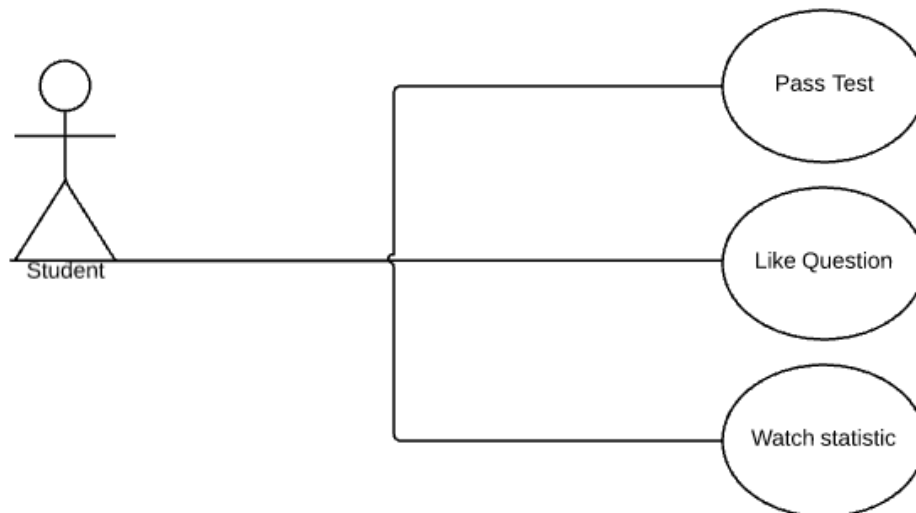
4.5 Διαγράμματα UML

Τα κυριότερα διαγράμματα της UML είναι τα εξής:

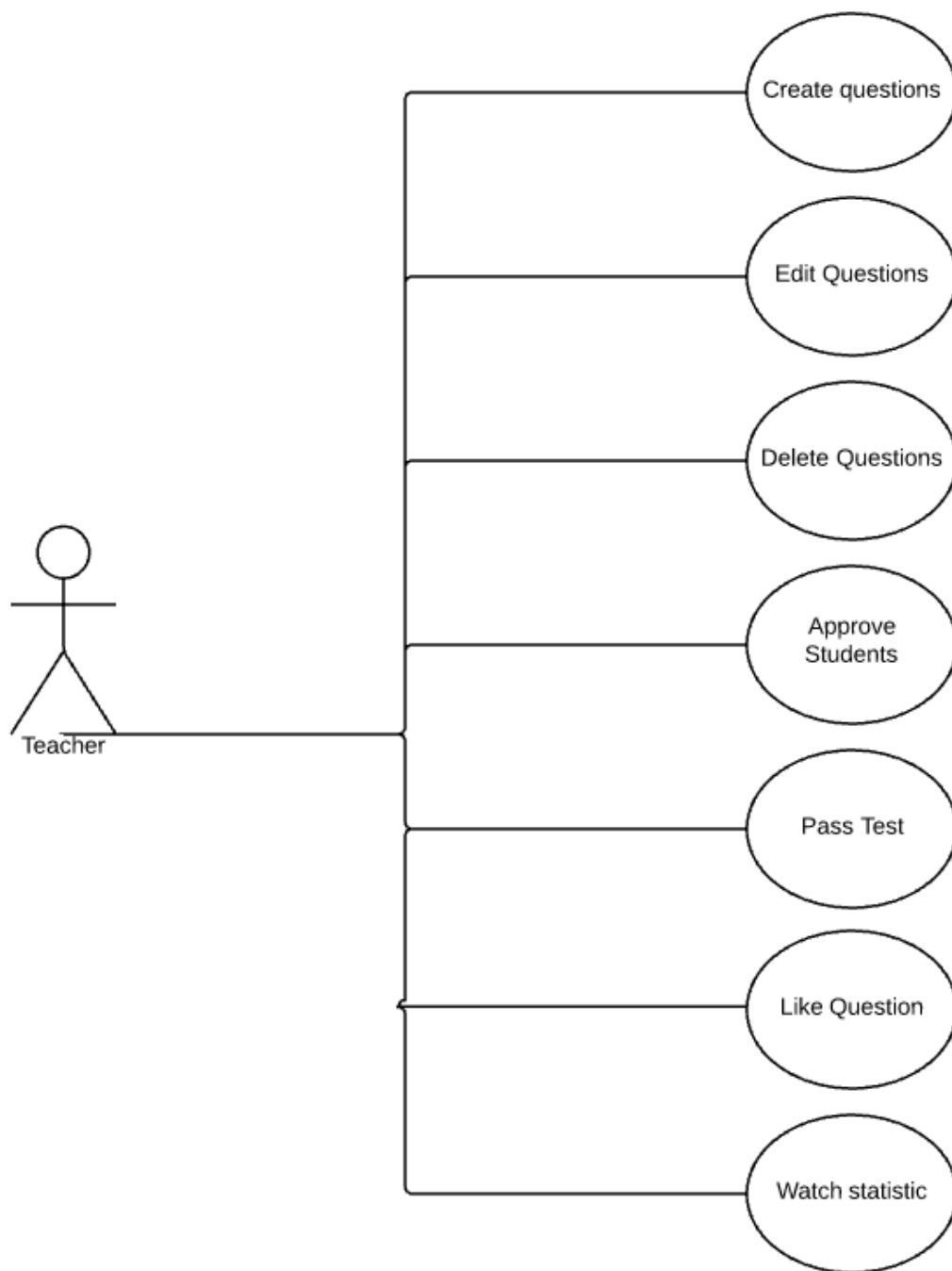
- Διάγραμμα περιπτώσεων χρήσης (Use case diagram)
- Διάγραμμα κλάσεων (Class diagram)
- Διάγραμμα δραστηριότητας (Activity diagram)
- Διάγραμμα συστατικών (Component diagram)

Διάγραμμα περιπτώσεων χρήσης

Διάγραμμα περιπτώσεων χρήσης Το διάγραμμα περιπτώσεων χρήσης στη UML χρησιμοποιείται για την μοντελοποίηση της λειτουργικότητας ενός συστήματος, όπως αυτή γίνεται αντιληπτή από τον εξωτερικό χρήστη. Τα διαγράμματα αυτά διαμερίζουν τη λειτουργικότητα του συστήματος σε συναλλαγές που έχουν νόημα για τους χρήστες του συστήματος ή αλλιώς χειριστές (actors). Τα επιμέρους τμήματα της λειτουργικότητας ονομάζονται περιπτώσεις χρήσης (use cases). Το σύνολο των περιπτώσεων χρήσης συνιστούν τη συμπεριφορά του συστήματος. Τα βασικά διαγραμματικά στοιχεία του διαγράμματος περιπτώσεων χρήσης είναι το σύστημα, ο χειριστής, η περίπτωση χρήσης και οι σχέσεις μεταξύ τους. Τα στοιχεία αυτά φαίνονται παρακάτω στο Σχήμα



Διάγραμμα περιπτώσεων -μαθητής



Διάγραμμα περιπτώσεων - καθηγητής

Η αξία του διαγράμματος περιπτώσεων χρήσης είναι ιδιαίτερα σημαντική, διότι καθορίζει τις λειτουργικές απαιτήσεις, οι οποίες θα αποτελέσουν σημείο αναφοράς καθ' όλη τη

διάρκεια ανάπτυξης του συστήματος. Ο σημαντικότερος ρόλος του συγκεκριμένου διαγράμματος είναι ότι αποτελεί ένα μέσο επικοινωνίας μεταξύ πελατών και σχεδιαστών, όσον αφορά στη λειτουργικότητα του συστήματος. Η απλότητα των συμβολισμών το καθιστά ιδανικό για αυτό το σκοπό, παρέχοντας τη δυνατότητα εύκολης αντίληψης του συνόλου των λειτουργιών καθώς και εύκολης τροποποίησής τους. Ο χειριστής αντιπροσωπεύει μια εξωτερική οντότητα, άνθρωπο ή σύστημα, η οποία αλληλεπιδρά με το σύστημα.

Ο χειριστής αναπαριστά ένα ρόλο, όχι έναν μεμονωμένο χρήστη του συστήματος, μιας και ο ίδιος χρήστης μπορεί να αλληλεπιδρά με το σύστημα με πολλαπλούς ρόλους. Οι χειριστές είναι κλάσεις με το στερεότυπο «actor», όπου το όνομα της κλάσης γενικά αναπαριστά το ρόλο του χειριστή. Το σύμβολο του χειριστή φαίνεται στο Σχήμα 2.2. Στο διάγραμμα περιπτώσεων χρήσης χρησιμοποιείται μόνο η σχέση γενίκευσης ανάμεσα σε χειριστές, προκειμένου να περιγραφεί η κοινή συμπεριφορά ανάμεσα τους, την οποία και κληρονομούν από μια πρόγονο κλάση χειριστή.

Ο τυπικός ορισμός μιας περίπτωσης χρήσης είναι μια ακολουθία ενεργειών που πραγματοποιείται από το σύστημα για την παραγωγή μετρήσιμων αποτελεσμάτων που έχουν νόημα για τον χρήστη.

Η περίπτωση χρήσης ορίζει ένα συγκεκριμένο τρόπο Actor UseCase association generalization <> <> 27 χρησιμοποίησης του συστήματος, προσδιορίζοντας την αλληλεπίδραση ανάμεσα σε έναν ή περισσότερους χειριστές και το σύστημα. Το στιγμιότυπο μιας περίπτωσης χρήσης ονομάζεται σενάριο (scenario), και αναπαριστά ένα συγκεκριμένο μονοπάτι εκτέλεσης (execution path) μέσα στο σύστημα. Ο συμβολισμός φαίνεται στο Σχήμα 2.2. Η περίπτωση χρήσης έχει τα ακόλουθα χαρακτηριστικά:

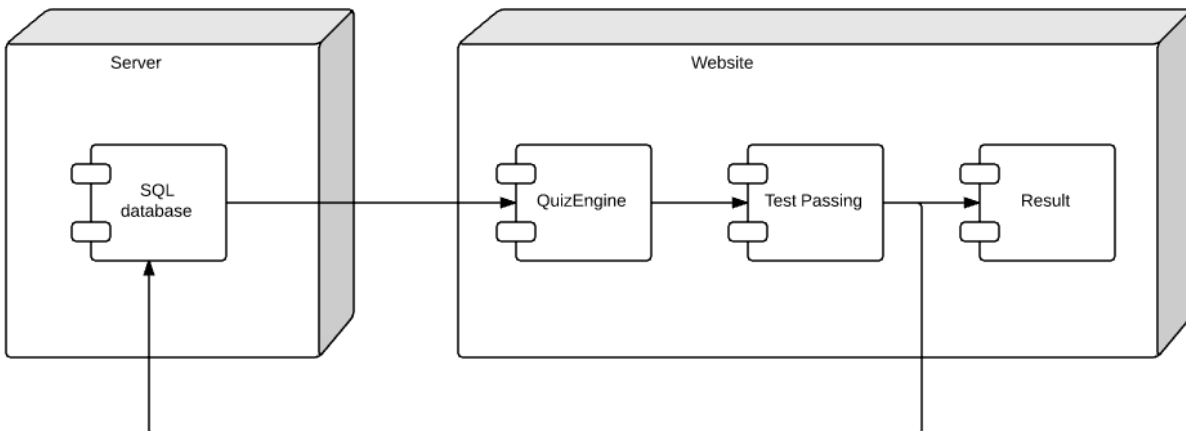
1. Ξεκινάει πάντα από ένα χειριστή.
2. Πρέπει να επιστρέφει κάποιου είδους απτή πληροφορία στο χρήστη.
3. Μια περίπτωση χρήσης είναι πλήρης, με την έννοια ότι αποτελεί μια πλήρη περιγραφή. Μια περίπτωση χρήσης δε θεωρείται ότι έχει ολοκληρωθεί μέχρις ότου η τελική πληροφορία παραχθεί, ακόμη κι αν απαιτούνται γι' αυτό πολλαπλές αλληλεπιδράσεις μεταξύ των αντικειμένων. Ένα σύνηθες λάθος είναι η διαίρεση μιάς περίπτωσης χρήσης σε μικρότερες, οι οποίες παράγουν ενδιάμεσα αποτελέσματα.

Ανάμεσα στις περιπτώσεις χρήσης υπάρχουν τρία είδη σχέσεων: η επέκταση (extends), η συμπερίληψη (uses ή includes) και η ομαδοποίηση (grouping). Οι σχέσεις αυτές φαίνονται στο Σχήμα 2.2. Η σχέση της επέκτασης είναι μια σχέση γενίκευσης που χρησιμοποιείται στην περίπτωση όπου μια περίπτωση χρήσης συμπεριλαμβάνει ένα τμήμα, όχι απαραίτητα ολόκληρο, την συμπεριφορά της περίπτωσης χρήσης που επεκτείνει. Τέτοιου είδους περιπτώσεις χρήσης χρησιμοποιούνται στο χειρισμό εξαιρέσεων. Η σχέση της συμπερίληψης είναι και αυτή μια σχέση γενίκευσης που χρησιμοποιείται στην περίπτωση όπου μια περίπτωση χρήσης συμπεριλαμβάνει την πλήρη λειτουργικότητα μιας άλλης. Όταν ένα σύνολο περιπτώσεων χρήσης παρουσιάζουν σε κάποια τμήματα κοινή συμπεριφορά, η σχέση αυτή χρησιμοποιείται για τη μοντελοποίηση αυτής της κοινής συμπεριφοράς σε μια περίπτωση χρήσης που

χρησιμοποιείται από τις υπόλοιπες. Τέλος με τη σχέση της ομαδοποίησης, περιπτώσεις χρήσης, οι οποίες διαθέτουν παρόμοια συμπεριφορά ή σχετίζονται με κάποιο τρόπο μεταξύ τους, οργανώνονται σε πακέτα.

Διάγραμμα συστατικών

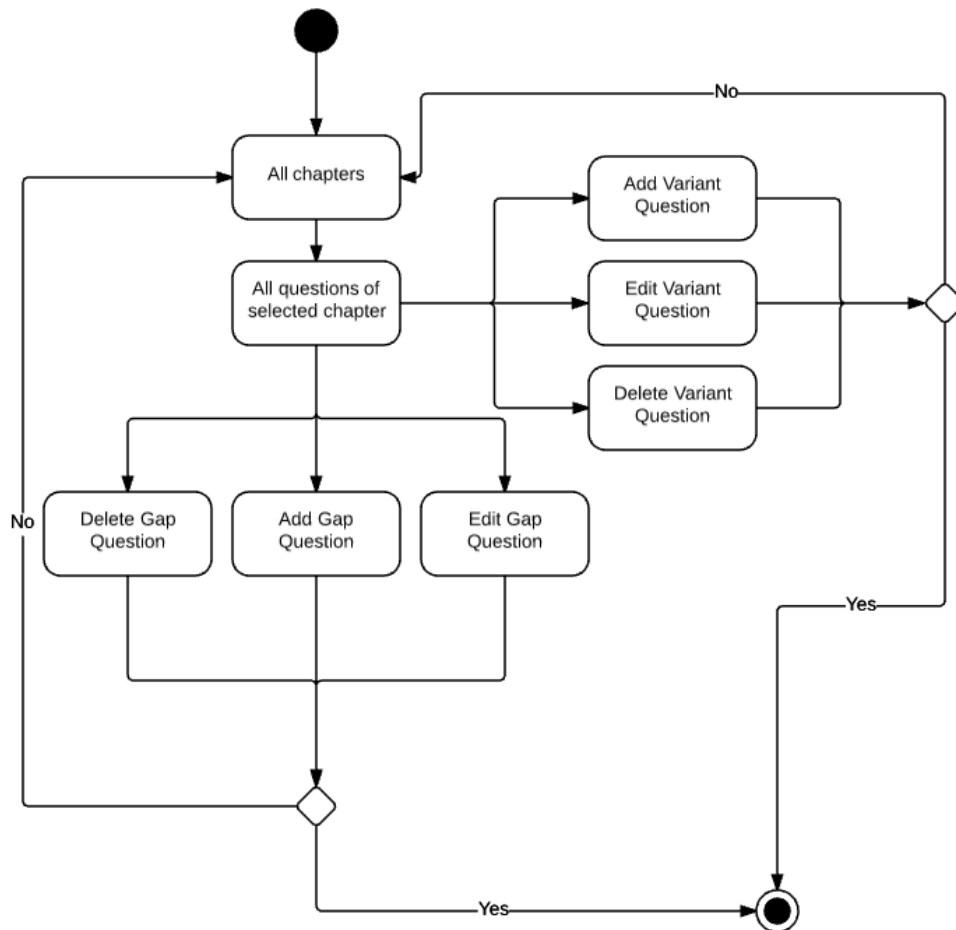
Ένα συστατικό (component) είναι μια φυσική μονάδα υλοποίησης κώδικα με σαφώς προσδιορισμένες διασυνδέσεις, η οποία αποτελεί επαναχρησιμοποιήσιμο τμήμα του συστήματος. Σε ένα αντικειμενοστρεφές σύστημα ένα συστατικό ενσωματώνει την υλοποίηση μίας ή περισσότερων κλάσεων. Καλά σχεδιασμένα συστατικά δε θα πρέπει να εξαρτώνται άμεσα από άλλα συστατικά αλλά μόνο από διασυνδέσεις. Οι εξαρτήσεις έχουν επίδραση στη συντήρηση ενός συστήματος λογισμικού. Αν κάποιο συστατικό A εξαρτάται από κάποιο άλλο συστατικό B, οποιαδήποτε αλλαγή στο B μπορεί να επηρεάσει το A. Υπό την ίδια έννοια, οι εξαρτήσεις καθορίζουν την ευκολία επαναχρησιμοποίησης ενός συστατικού. Στην περίπτωση όπου ένα συστατικό στο σύστημα μπορεί να



αντικατασταθεί από κάποιο άλλο, που υποστηρίζει τις ίδιες διασυνδέσεις, δεν επιφέρονται αλλαγές στο υπόλοιπο σύστημα. Ένα διάγραμμα συστατικών απεικονίζει το δίκτυο των εξαρτήσεων μεταξύ των συστατικών του συστήματος. Μια εξάρτηση μεταξύ δύο συστατικών υποδηλώνει ότι για την ορθή λειτουργία του ενός συστατικού απαιτείται η ύπαρξη ενός άλλου. Το συστατικό συμβολίζεται ως ένα ορθογώνιο ενώ οι εξαρτήσεις συμβολίζονται ως διακεκομμένες ακμές με κατεύθυνση από το εξαρτώμενο συστατικό προς αυτό που παρέχει τις λειτουργίες. Στα διαγράμματα συστατικών υπάρχει η δυνατότητα απεικόνισης λογικών τμημάτων ενός συστήματος με τη χρήση των πακέτων. Ένα πακέτο περιλαμβάνει ένα σύνολο από συστατικά τα οποία έχουν λειτουργική συνάφεια. Ο συμβολισμός ενός πακέτου στη UML επιτυγχάνεται με τη χρήση ενός φακέλου

Διάγραμμα δραστηριότητας

Ένας γράφος δραστηριότητας είναι μια ειδική μορφή μηχανής καταστάσεων που έχει ως στόχο τη μοντελοποίηση των υπολογισμών και της ροής της εργασίας. Οι καταστάσεις του γράφου δραστηριότητας αναπαριστούν τις καταστάσεις εκτέλεσης ενός υπολογισμού, όχι τις καταστάσεις των αντικειμένων που συμμετέχουν. Υπό κανονικές συνθήκες, ένας γράφος δραστηριότητας προϋποθέτει ότι οι υπολογισμοί πραγματοποιούνται χωρίς εξωτερικές γεγονοδηγούμενες διακοπές, αλλιώς είναι

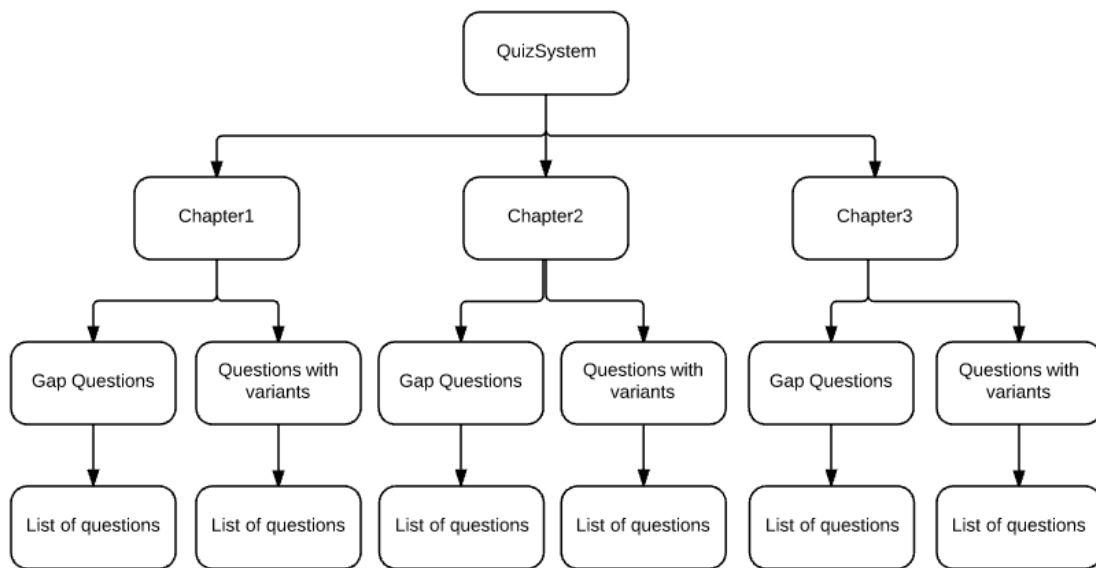


Διαγράμματα αντικειμένων

Τα διαγράμματα αντικειμένων περιγράφουν ένα σύνολο αντικειμένων καθώς και την σχέσεις τους μια δεδομένη χρονική στιγμή .

Ένα διάγραμμα αντικειμένων συνήθως περιέχει Αντικείμενα (objects) και Συνδέσμους (Links) .

Τα διαγράμματα αντικειμένων χρησιμοποιούνται για να καταγράψουν στατικές δομές αντικειμένων , είναι ένα στιγμιότυπο του διαγράμματος κλάσεων .Η στατική όψη ενός διαγράμματος συνεργασίας



ΚΕΦΑΛΑΙΟ 5 - Συμπεράσματα

Αποτελεί σήμερα κοινή διαπίστωση ότι η εποχή μας χαρακτηρίζεται από τη ραγδαία εξέλιξη των επιστημών και της τεχνολογίας και ότι η επιστημονική και η τεχνολογική πρόοδος επιδρούν καταλυτικά στην οικονομική και κοινωνική ζωή, στις εργασιακές σχέσεις, στην εκπαίδευση και στον πολιτισμό. Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας, λόγω του εγκάρσιου χαρακτήρα τους, διαπερνούν σταδιακά το σύνολο του κοινωνικού ιστού και καθιστούν καταλυτική την παρουσία τους σε όλο το φάσμα των ανθρώπινων δραστηριοτήτων. Η εισαγωγή στο σχολείο των νέων τεχνολογιών της πληροφορίας και της επικοινωνίας, είναι πλέον αναγκαιότητα. Στόχος πρέπει να είναι αφενός η εξοικείωση των μαθητών και των εκπαιδευτικών με τη χρήση τους και αφετέρου

η αξιοποίησή τους στη διαδικασία της μάθησης. Οι εφαρμογές των νέων τεχνολογιών της πληροφορίας και της επικοινωνίας και κυρίως η ενοποίηση τριών - μέχρι πρόσφατα διακριτών - τεχνολογικών κλάδων, της πληροφορικής, των τηλεπικοινωνιών και των οπτικοακουστικών μέσων, επιδρούν καταλυτικά σε όλους σχεδόν τους τομείς της ανθρώπινης δραστηριότητας. Επιφέρουν σημαντικές αλλά και ραγδαίες αλλαγές στο χώρο της εργασίας, στην οικονομία, στον πολιτισμό και στην κοινωνία ολόκληρη. Αγγίζοντας το νέο αιώνα οι σημερινοί μαθητές καλούνται να ζήσουν στην «Κοινωνία της Πληροφορίας» που τη χαρακτηρίζει ένας εμφανής δυναμισμός και ευρύτατη χρήση των νέων τεχνολογιών.

Η ανάπτυξη των υπολογιστών δεν θα μπορούσε να αφήσει ανεπηρέαστα τα μέσα και τις μεθόδους άσκησης του εκπαιδευτικού έργου. Ο εκπαιδευτικός χώρος μπορεί να προσφέρει στην ηλεκτρονική τεχνολογία ένα ευρύ πεδίο εφαρμογών με πολλαπλά οφέλη για όλους όσους συμμετέχουν στην εκπαιδευτική διαδικασία. Πρώτα, η χρήση ηλεκτρονικών προγραμμάτων εκπαιδευτικού χαρακτήρα μπορεί να προσδώσει νέο περιεχόμενο, υψηλό ενδιαφέρον και μεγαλύτερη αποτελεσματικότητα στην διδακτική πράξη. Ο παραδοσιακός τρόπος διδασκαλίας, που ήθελε το δάσκαλο και το διδακτικό βιβλίο μοναδικές πηγές γνώσης ενισχύεται - χωρίς να καταργείται - με τη χρήση ενός μέσου που πολλαπλασιαστικά μπορεί να λειτουργήσει για την επίτευξη του προσδοκώμενου μορφωτικού αποτελέσματος.

Με τη βοήθεια του υπολογιστή η απόκτηση της γνώσης απαλλάσσεται από την τυποποίηση και την παθητικότητα που ταλαιπωρεί τη σχολική μετάδοσή της και συνδυάζεται με το παιχνίδι και την αυτενέργεια. Τα εκπαιδευτικά προγράμματα περιέχουν ένα θησαυρό γνώσεων, η κατάκτηση των οποίων προϋποθέτει την κινητοποίηση της φαντασίας, της επινοητικότητας, της κριτικής σκέψης και της εξερευνητικής διάθεσης, που στη μέχρι τώρα σχολική πράξη έμεναν συνήθως ανενεργές. Οι επικρατούσες απόψεις σχετικά με την αναγκαιότητα της Πληροφορικής στην εκπαίδευση προκάλεσαν σύγχυση και παράλληλα, ποικίλα σχόλια και αντιδράσεις των εκπαιδευτικών. Οι αντιδράσεις εντοπίζονται σχετικά με τα πλεονεκτήματα και τα μειονεκτήματα της ένταξης της Πληροφορικής στην εκπαίδευση.

Τα πλεονεκτήματα σχετίζονται κυρίως με την καλύτερη κατανόηση της ύλης από τους μαθητές και την δυνατότητα εξατομίκευσης της διδασκαλίας. Καθώς ο υπολογιστής, μπορεί να δώσει τη δυνατότητα στο μαθητή να αναπτύξει επιστημονικό τρόπο σκέψης. Όσον αφορά στα μειονεκτήματα της ένταξης και χρήσης των νέων Τεχνολογιών της Πληροφορικής στην εκπαίδευση, αυτά συνοψίζονται στην άποψη ότι είναι πιθανό να προκαλέσουν ομοιομορφία στη διδασκαλία και την αξιολόγηση, πράγμα το οποίο μπορεί να δυσκολέψει τη παροχή σύνθετης γνώσης προς τους μαθητές. Επιπλέον, οι νέες τεχνολογίες μπορούν να συνεισφέρουν στη βελτίωση και τον επαναπροσανατολισμό της διαδικασίας της μάθησης σε μια κατεύθυνση όπου η μάθηση θα γίνει ενεργητική και οι μαθητές:

- ◆ θα πειραματίζονται
- ◆ θα αναζητούν, θα ανακαλύπτουν και θα χαίρονται τη γνώση

- ◆ θα μαθαίνουν να συνεργάζονται, να είναι μεθοδικοί, να παίρνουν πρωτοβουλίες, να θέτουν στόχους, να επιχειρηματολογούν και να σκέφτονται και να εκφράζονται ελεύθερα
- ◆ θα καλλιεργούν τις κλίσεις και τα ταλέντα τους
- ◆ θα αγαπούν τη μάθηση. Η εισαγωγή των υπολογιστών στην εκπαίδευση έχει επιφέρει σημαντικές θετικές και αρνητικές επιπτώσεις. Οι θετικές επιπτώσεις είναι:
- ◆ Η μάθηση μπορεί να αντιμετωπίζεται από τα παιδιά σαν μια διαδικασία ευχάριστη και ενδιαφέρουσα.
- ◆ Οι υπολογιστές μειώνουν την προσπάθεια για αποστήθιση και μπορούν να συμβάλλουν στην οργάνωση του γνωστικού υλικού.
- ◆ Ο προγραμματισμός ενός υπολογιστή προϋποθέτει ευρύτητα στη γνώση, ευελιξία στη σκέψη και ικανότητα στη πρόβλεψη σφαλμάτων. Λειτουργεί δηλαδή προς την κατεύθυνση της πνευματικής καλλιέργειας.
- ◆ Ο μαθητής είναι σε θέση να σκέφτεται και να γράφει ελεύθερα, να αναπτύσσει τις ιδέες του, έχοντας επίγνωση ότι

5.1 Μελλοντικές βελτιώσεις

Εκτός από τα πλεονεκτήματα της εφαρμογής, καλό θα ήταν να αναφέρουμε και ορισμένα μειονεκτήματα που παρουσιάζει τονίζοντας παράλληλα τις αλλαγές που θα μπορούσαν να γίνουν μελλοντικά.

Η πρώτη πιθανή βελτίωση που πρέπει να εφαρμοσθεί σε μία νέα έκδοση της εφαρμογής που δημιουργήσαμε, είναι η ύπαρξη της δυνατότητας αλλαγής γλώσσας. Εκτός από ελληνικά, θα ήταν καλό να υπήρχε υποστήριξη της εφαρμογής τουλάχιστον στην αγγλική γλώσσα. Αυτή θα ήταν μία σημαντική βελτίωση γιατί θα επέτρεπε την χρήση της εφαρμογής από άτομα που δεν μιλούν ελληνικά.

Η επόμενη πιθανή βελτίωση αφορά την ενσωμάτωση ζωντανής επικοινωνίας μεταξύ καθηγητή μαθητή μέσω παραθύρου διαλόγου (chat). Κάτι που θα βελτίωνε σημαντικά την επικοινωνία τους και θα βοηθούσε στην ταχύτερη επίλυση των δυσκολιών που αντιμετωπίζουν οι μαθητές.

Επιπροσθέτως μια άλλη σημαντική προσθήκη θα ήταν η δημιουργία αποθηκευτικού χώρου όπου μαθητές και καθηγητές θα μπορούσαν να ανεβάζουν τις σημειώσεις του και να οργανώνουν κατάλληλα το υλικό τους.

ΒΙΒΛΙΟΓΡΑΦΙΑ - ΙΣΤΟΣΕΛΙΔΕΣ

2. Pro ASP.NET MVC 5 Platform– Freeman, Adam
3. Programming ASP.NET MVC 5– http://www.c-sharpcorner.com/ebooks/programming-aspdot-net-mvc_54. Design Patterns MVC Pattern - Tutorialspoint:https://www.tutorialspoint.com/design_pattern/mvc_pattern.htm– Εισαγωγή στην Ανάπτυξη Εφαρμογών Android (Σημειώσεις σεμιναρίου)
4. Student GURU c# tutorial: <http://studentguru.gr/w/tutorials/59.2-c6>. Tutorials for SQL Server: <https://docs.microsoft.com/en-us/sql/sql-server/tutorials-for-sql-server-2016>
5. Wikipedia – UML Diagrams: https://en.wikipedia.org/wiki/Activity_diagram
6. Model–view–controller - Wikipedia: <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
7. ASP.NET MVC – Wikipedia https://en.wikipedia.org/wiki/ASP.NET_MVC
8. ASP.NET MVC 5 | Microsoft Docs: <https://docs.microsoft.com/en-us/aspnet/mvc/mvc5>