

Ανοικτό Πανεπιστήμιο Κύπρου
Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή
στα Πληροφοριακά και Επικοινωνιακά Συστήματα



**Σχεδιασμός και Υλοποίηση Εκπαιδευτικής Ιστοσελίδας με
Στοιχεία Εξατομίκευσης και Μοντελοποίησης Χρήστη
(Personalization and User Modeling)**

Γεωργία Μάρκου

**Επιβλέπων Καθηγητής
Δρ. Δημήτρης Κυριάκου**

Μάιος 2014

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Σχεδιασμός και Υλοποίηση Εκπαιδευτικής Ιστοσελίδας με Στοιχεία Εξατομίκευσης και Μοντελοποίησης Χρήστη (Personalization and User Modeling)

Γεωργία Μάρκου

Επιβλέπων Καθηγητής
Δρ. Δημήτρης Κυριάκου

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Μάιος 2014

Περίληψη

Στην παρούσα διατριβή παρουσιάζεται η σχεδίαση και υλοποίηση μιας εκπαιδευτικής ιστοσελίδας κάνοντας χρήση τεχνολογιών Εξατομίκευσης και Μοντελοποίησης Χρήστη. Η ιστοσελίδα αυτή δοθηκε προς χρήση σε μια ομάδα ανθρώπων/εκπαιδευόμενων, οι οποίοι με το πέρας της παρακολούθησης των μαθημάτων κλήθηκαν να απαντήσουν σε ερωτηματολόγιο. Από τη συλλογή των απαντήσεων έχουν προκύψει συμπεράσματα που οδήγησαν στην αξιολόγηση του ιστότοπου και στη σύγκρισή του με άλλες μεθόδους διδασκαλίας, όπως είναι οι στατικές εκπαιδευτικές ιστοσελίδες.

Summary

In this master thesis, the design and implementation of an educational website will be presented, using Personalization and User Modeling. This website had been visited by a group of people and after finishing their studying they had to answer some questions about it. In the end, all answers were gathered and processed in order for a conclusion to be made about the evaluation of the website and its comparison to other tutoring methods, such as static educational websites.

Ευχαριστίες

Ευχαριστώ θερμά τον καθηγητή μου, κ. Δημήτρη Κυριάκου, για την βοήθεια και την καθοδήγησή του καθώς και για την ευκαιρία που μου έδωσε να ασχοληθώ με το ενδιαφέρον αυτό θέμα. Επίσης, ευχαριστώ το σύζυγό μου, Ισίδωρο, για τη στήριξη του όλο το διάστημα εκπόνησης της διατριβής μου.

Περιεχόμενα

1	Εισαγωγή	1
1.2	Στόχος	1
1.3	Μεθοδολογία	2
1.4	Ερευνητικά ερωτήματα	3
2	Βιβλιογραφική ανασκόπηση	4
2.1	Εξατομίκευση (Personalization)	5
2.1.1	Ιστορική αναδρομή	6
2.2	Προσαρμοζόμενα Εκπαιδευτικά Υπερμέσα (ΑΕΗ)	8
2.3	Μοντελοποίηση Χρήστη (User Modelling)	9
2.3.1	Ιστορική αναδρομή	10
2.3.2	Δια βίου μοντελοποίηση Χρήστη (Lifelong User Modelling)	12
3	Προσαρμοζόμενα Υπερμέσα AHA & GALE	14
3.1	Αρχιτεκτονική Προσαρμοζόμενων Υπερμέσων (AHA)	15
3.1.1	Αρχιτεκτονική και λειτουργία AHA	16
3.2	GRAPPLE Project	22
3.2.1	Το σύστημα GALE	24
3.3	Σύγκριση AHA – GALE	27
3.3.1	Επιλογή του συστήματος	31
4	Υλοποίηση προσαρμοστικότητας εκπαιδευτικής εφαρμογής	33
4.1	Σχεδίαση μαθήματος	35
4.2	Σχεδίαση προσαρμοστικότητας περιεχομένου	39
4.2.1	Κύρια έννοια Introducing Java και υποέννοιες	40
4.2.2	Κύρια έννοια Control Structures και υποέννοιες	44
4.2.3	Κύρια έννοια Loops και υποέννοιες	47
4.2.4	Κύρια έννοια Methods και υποέννοιες	51
4.2.5	Κύρια έννοια Arrays και υποέννοιες	55
4.2.6	Ετικέτες	58

4.3	Σχεδίαση προσαρμοστικότητας πλοήγησης	63
4.3.1	Ανακεφαλαίωση	67
4.4	Έρευνα αξιολόγησης	68
5	Συμπεράσματα - συζήτηση.....	73
5.1	Ανάλυση αποτελεσμάτων έρευνας αξιολόγησης.....	73
6	Επίλογος	84
7	Βιβλιογραφία	86

Κεφάλαιο 1

Εισαγωγή

Καθώς οι άνθρωποι δεν έχουν όλοι τις ίδιες ανάγκες στον τρόπο εκμάθησης και παρακολούθησης εκπαιδευτικών προγραμμάτων, η ιδέα της χρήσης τεχνολογίας εξατομίκευσης και μοντελοποίησης χρήστη στις εκπαιδευτικές ιστοσελίδες δύναται να αποφέρει πολύ χρήσιμα αποτελέσματα στην εκπαίδευση. Ο χρήστης μπορεί να λαμβάνει κάθε φορά τις βέλτιστες πληροφορίες που αντιστοιχούν στο προφίλ του στην κατάλληλη στιγμή. Έτσι δεν θα χρειάζεται να παρακολουθήσει μια σειρά από παρουσιάσεις οι οποίες μπορεί να δυσχεραίνουν ή να παρατείνουν την εκπαίδευσή του, εξ αιτίας είτε της ακαταλληλότητας του χρόνου εμφάνισης του εκπαιδευτικού υλικού είτε της ακαταλληλότητας του περιεχομένου. Λόγω του πρώιμου σταδίου στο οποίο βρίσκεται η χρήση της καινούργιας αυτής τεχνολογίας, της εξατομίκευσης και μοντελοποίησης χρήστη, στους εκπαιδευτικούς ιστοχώρους, μια έρευνα για την αξιολόγηση ενός τέτοιου ιστότοπου, και τη σύγκρισή του με άλλους εκπαιδευτικούς ιστότοπους, για την εξαγωγή συμπερασμάτων για την χρήση και το βαθμό προσφοράς της τεχνολογίας αυτής στην ηλεκτρονική εκπαίδευση, καθίσταται αναγκαία.

1.1 Στόχος

Σκοπός της τρέχουσας διατριβής είναι η κατασκευή ενός εκπαιδευτικού ιστότοπου για την εκμάθηση της γλώσσας Java, ο οποίος θα καθοδηγεί ξεχωριστά κάθε χρήστη ανάλογα με το προφίλ του και την συμπεριφορά του κατά την πλοήγηση. Το προφίλ κάθε χρήστη θα είναι προσαρμόσιμο. Το ίδιο θα είναι και το περιεχόμενο κάθε ιστοσελίδας, το οποίο θα προσαρμόζεται δυναμικά για κάθε χρήστη σύμφωνα με τις εκάστοτε ανάγκες του. Η επιτυχής δημιουργία, συντήρηση και εκμετάλλευση των προφίλ των χρηστών αποτελεί βασικό στόχο καθώς επηρεάζει άμεσα τα αποτελέσματα της εκπαιδευτικής καθοδήγησης και συνεπώς ολόκληρης της έρευνας. Το συγκεκριμένο εκπαιδευτικό υλικό απευθύνεται στους μεταπτυχιακούς φοιτητές του τμήματος Πληροφοριακών και Επικοινωνιακών Συστημάτων του Ευρωπαϊκού Πανεπιστημίου Κύπρου και συγκεκριμένα σε όσους παρακολουθούν τη θεματική ενότητα ΠΕΣ511 – Προηγμένες τεχνολογίες ιστού και προγραμματισμός. Οι χρήστες θα αξιολογήσουν τον ιστότοπο ώστε να γίνει σύγκριση και να εξαχθούν συμπεράσματα για τους εκπαιδευτικούς ιστότοπους με χρήση τεχνολογιών εξατομίκευσης και μοντελοποίησης χρήστη σε σχέση με τις συμβατικούς εκπαιδευτικούς ιστότοπους.

1.2 Μεθοδολογία

Για τη μελέτη του συγκεκριμένου θέματος πραγματοποιήθηκε μικτή έρευνα. Δηλαδή, συνδυασμό ποιοτικής και ποσοτικής έρευνας. Στο πρώτο στάδιο, αυτό της ποιοτικής έρευνας, θα γίνει μια προσπάθεια για μια ερμηνευτική προσέγγιση κατασκευασμένη από φιλοσοφικές υποθέσεις που θα βασίζονται σε προσωπική εμπειρία της χρήσης εξατομίκευσης και μοντελοποίησης χρήστη σε εκπαιδευτικούς ιστότοπους. Σε δεύτερο χρόνο, θα προβούμε σε ποσοτική έρευνα μέσω διαδικτύου με τον σχεδιασμό ενός ερωτηματολογίου για την συλλογή και ανάλυση ποσοτικών δεδομένων από τις απαντήσεις των ίδιων των χρηστών. Το ερωτηματολόγιο, μετά το πέρας της παρακολούθησης ολόκληρου του εκπαιδευτικού υλικού, θα βρίσκεται στο διαδίκτυο και οι χρήστες θα κληθούν να το συμπληρώσουν. Τη δυνατότητα χρήσης του εκπαιδευτικού ιστότοπου θα έχουν όλοι οι φοιτητές της θεματικής ενότητας ΠΕΣ511 του Ευρωπαϊκού Πανεπιστημίου Κύπρου, από τους οποίους θα αντλήσουμε και τα δεδομένα της έρευνας. Τέλος, τα δεδομένα της ποσοτικής έρευνας θα αναλυθούν με Επαγωγική Στατιστική, η οποία περιλαμβάνει τα παρακάτω στάδια:

- Διατύπωση υποθέσεων (βασικά ερευνητικά ερωτήματα),
- Διαμόρφωση κριτηρίων για τη λήψη μιας απόφασης,

- Συλλογή δεδομένων από δείγματα και
- Αξιολόγηση της μηδενικής απόφασης (επιβεβαίωση ή απόρριψη υποθέσεων).

1.3 Ερευνητικά ερωτήματα

1. Η τεχνολογία εξατομίκευσης και μοντελοποίησης χρήστη προσφέρει καλύτερα μαθησιακά αποτελέσματα σε σχέση με παραδοσιακές μεθόδους διδασκαλίας (στατικές ιστοσελίδες, στατικές παρουσιάσεις PowerPoint);
2. Η χρήση τεχνικών εξατομίκευσης και μοντελοποίησης του χρήστη για την παρουσίαση εκπαιδευτικού υλικού καλύπτει επαρκώς τις εκπαιδευτικές ανάγκες των χρηστών;

Κεφάλαιο 2

Βιβλιογραφική ανασκόπηση

Οι όροι Εξατομίκευση (Personalization) και Μοντελοποίηση Χρήστη (User Modelling) έχουν κάνει την εμφάνισή τους πολλά χρόνια πριν και η έρευνα για την εφαρμογή τους στα εκπαιδευτικά συστήματα συνεχίζεται μέχρι και σήμερα. Η Μοντελοποίηση Χρήστη είναι η «καρδιά» των εκπαιδευτικών υπηρεσιών εξατομίκευσης, τα οποία προσφέρουν προσαρμοζόμενο περιεχόμενο μέσω παραλλαγών μεμονωμένων κομματιών και προσαρμοζόμενη απεικόνιση υπερσυνδέσμων [13]. Η έρευνα στο πεδίο της Μοντελοποίησης Χρήστη εξετάζει πώς δημιουργούνται οι υποθέσεις, για τις γνώσεις και τις προτιμήσεις του χρήστη, πώς αναπαρίστανται και πώς μπορούν να εκμεταλλευτούν από το σύστημα κατά τη διάρκεια της αλληλεπίδρασης με τον χρήστη [21].

Τα Έξυπνα Συστήματα Διδασκαλίας (Intelligent Tutoring Systems) είναι οι πρώτες εκπαιδευτικές εφαρμογές που εμφάνισαν στοιχεία Εξατομίκευσης και Μοντελοποίησης Χρήστη [28].

Πολύ σύντομα έγινε ξεκάθαρο ότι τα υπερκείμενα ή τα υπερμέσα παρέχουν την καλύτερη επιλογή για την οργάνωση εκπαιδευτικού υλικού που παραδίδεται σε απευθείας σύνδεση(online). Ο συνδυασμός ενός ITS με εκπαιδευτικό υλικό οργανωμένο ως υπερμέσα, ήταν η φυσική αφετηρία της έρευνας των **Προσαρμοζόμενων Εκπαιδευτικών Υπερμέσων (Adaptive Educational Hypermedia)**[2].

Υπάρχουν πολλοί ορισμοί και τεχνικές για Προσαρμοζόμενα Υπερμέσα. Μια καλή επισκόπηση παρουσιάζεται από τον Brusilovsky [4]. Αυτή η επισκόπηση χαρακτηρίζει 29 διαφορετικά Συστήματα Προσαρμοζόμενων Υπερμέσων ανάλογα κατηγοριοποιημένα με βάση τον τύπο της εφαρμογής και την τεχνική προσαρμογής που χρησιμοποιούν [11].

Τα τελευταία χρόνια ο όρος Δια Βίου Μοντελοποίηση Χρήστη ήρθε στο προσκήνιο και έχει δημιουργήσει έντονο προβληματισμό. Η δημιουργία τεχνολογίας για εξατομικευμένη δια βίου μάθηση αναγνωρίστηκε ως Πρόβλημα Μεγάλης Πρόκλησης από μεγάλα ερευνητικά κέντρα ανά τον κόσμο [18]. Ο όρος Δια Βίου Μοντελοποίηση Χρήστη χρησιμοποιήθηκε για πρώτη φορά από την Judy Kay και αναφέρεται στη σύλληψη μακρόχρονων πληροφοριών από τους χρήστες(Judy Kay 2012).

2.1 Εξατομίκευση (Personalization)

Εξατομίκευση είναι η διαλογή ή το φιλτράρισμα πληροφοριών για ένα άτομο χρησιμοποιώντας πληροφορίες για το άτομο αυτό [22]. Οι τεχνικές Εξατομίκευσης εξυπηρετούν ώστε το σύστημα να έχει τη δυνατότητα να μάθει το χρήστη, να θυμάται το χρήστη και να προσαρμόζει τα δεδομένα του χρήστη που διατηρεί στη μνήμη ανάλογα με τις αλλαγές στις ανάγκες του [1].

Τα χαρακτηριστικά χρηστών που χρησιμοποιούν οι τεχνικές Εξατομίκευσης είναι :

- Μορφωτικό επίπεδο
- Γνώσεις
- Ξένες γλώσσες
- Ενδιαφέροντα
- Προτιμήσεις

Με βάση τις παραπάνω πληροφορίες γίνεται η προσαρμογή του περιεχομένου που παρουσιάζεται στον χρήστη.

2.1.1 Ιστορική αναδρομή

Η πρώτη αναφορά της έννοιας της Εξατομίκευσης, στο εκπαιδευτικό πλαίσιο, έγινε από τον Victor Garcia Hoz το 1970 [31]. Στις αρχές τις δεκαετίας του '90 έγιναν οι πρώτες αναφορές για την υπηρεσία της Εξατομίκευσης στις ιστοσελίδες εμπορικού περιεχομένου(π.χ. Amazon, eBay, κ.α.). Στη συνέχεια οι τεχνικές εξατομίκευσης επεκτάθηκαν και σε άλλες εφαρμογές όπως οι μηχανές αναζήτησης και οι ψηφιακές βιβλιοθήκες. Η έρευνα για Εξατομίκευση στα εκπαιδευτικά συστήματα οδήγησε στην ανάπτυξη των Έξυπνων Συστημάτων Διδασκαλίας. Ήταν το 1982, όταν οι Sleeman & Brown επανεξέτασαν την τρέχουσα επιστημονική κατάσταση της τεχνολογίας CAI (Computer Aided Instruction) – Καθοδήγηση Βοηθούμενη από Υπολογιστή, και χρησιμοποίησαν για πρώτη φορά τον όρο ITS (Intelligent Tutoring Systems) Έξυπνα Συστήματα Διδασκαλίας για να περιγράψουν τα εξελισσόμενα αυτά συστήματα και για να διαχωριστούν από τα CAI [29].

Τα Έξυπνα Συστήματα Διδασκαλίας(ITS) είναι εκπαιδευτικές εφαρμογές οι οποίες περιλαμβάνουν ένα συνδυασμό από ενσωματωμένα συστήματα που παρακολουθούν την απόδοση του εκπαιδευόμενου εξατομικεύοντας την καθοδήγηση βάσει της προσαρμογής στο μαθησιακό στυλ και το γνωστικό επίπεδο του εκπαιδευόμενου, με τις κατάλληλες διδακτικές τακτικές [25].

Ο στόχος των ITS είναι η χρήση της γνώσης για ένα πεδίο και το μαθητή, καθώς επίσης και προκαθορισμένων στρατηγικών διδασκαλίας για την υποστήριξη ευέλικτης και εξατομικευμένης εκμάθησης και διδασκαλίας (Brusilovsky, 1999). Υπάρχουν τέσσερις βασικές τεχνολογίες ITS (Brusilovzky, 1999):

- Διαδοχή διδακτέας ύλης (curriculum sequencing): στο μαθητή παρέχεται η πιο κατάλληλα σχεδιασμένη αλληλουχία μερών γνώσης για εκμάθηση μέσα από μια σειρά εργασιών εκμάθησης.
- Έξυπνη ανάλυση λύσεων μαθητή (intelligent analysis of student's solution): ένας αναλυτής επεξεργάζεται τις τελικές απαντήσεις του μαθητή και του επιστρέφει μια επεξήγηση σφαλμάτων.

- Διαδραστική υποστήριξη επίλυσης προβλήματος (interactive problem solving support): στο μαθητή παρέχεται μια έξυπνη βοήθεια σε κάθε βήμα επίλυσης προβλήματος.
- Υποστήριξη επίλυσης προβλήματος βασιζόμενη σε παράδειγμα (example-based problem solving support): παρέχεται εκτεταμένη βοήθεια με τη μορφή παραδειγμάτων που βασίζονται σε προηγούμενα λυμένα προβλήματα.

Η **προσαρμογή** (ή εξατομίκευση) βασίζεται σε ένα μοντέλο χρήστη που αναπαριστά απόψεις σχετικές με τις προτιμήσεις του (Brusilovsky, 1996), τις γνώσεις του και τα ενδιαφέροντά του. Το σύστημα συγκεντρώνει πληροφορίες για τον χρήστη παρατηρώντας την χρήση που κάνει πάνω στην εφαρμογή και γενικότερα την συμπεριφορά του κατά την πλοήγηση.

Η **προσαρμοστικότητα** σε ένα εκπαιδευτικό σύστημα μέσω υπολογιστή είναι πλέον απαιτούμενη εξ αιτίας του ότι κάθε άνθρωπος κατέχει συγκεκριμένες ικανότητες, ταλέντα και όρια. Η εκπαιδευτική προσέγγιση του ένα μέγεθος για όλους ξεχνά ότι κάθε άτομο είναι διαφορετικό και έχει διαφορετικές ανάγκες. Σε απάντηση αυτής της απαίτησης, το ερευνητικό ενδιαφέρον για τα εξατομικευμένα εκπαιδευτικά συστήματα υπερμέσων έχει εξελιχθεί. Μία αναδυόμενη προσέγγιση είναι τα Προσαρμοζόμενα Εκπαιδευτικά Συστήματα Υπερμέσων τα οποία κληρονομούν και συνδυάζουν τα δυο παρακάτω συστήματα [1]:

- Έξυπνα Συστήματα Διδασκαλίας (Intelligent Tutoring Systems)
- Προσαρμοζόμενα Συστήματα Υπερμέσων (Adaptive Hypermedia Systems)

Τα **Προσαρμοζόμενα Συστήματα Υπερμέσων** (Adaptive Hypermedia Systems -AHS) είναι όλα τα συστήματα υπερμέσων και υπερκειμένου που αντανακλούν τα χαρακτηριστικά ενός χρήστη σε ένα μοντέλο τα οποία μπορούν να θεωρηθούν σαν μια εγγραφή για ένα χρήστη και που εφαρμόζει αυτό το μοντέλο να προσαρμόζει διάφορα κομμάτια του συστήματος στο χρήστη (Brusilovsky, 1996a). Τα AHS εφαρμόζουν διάφορες μορφές διαρκούς μοντέλου χρήστη για να εξατομικεύσουν το περιεχόμενο και τους υπερσυνδέσμους των σελίδων υπερμέσων για κάθε χρήστη. Παρά την παράκαμψη της έξυπνης μονάδας, η οποία ήταν το κύριο συστατικό ενός ITS, στη δομή των AHS, ήταν δυνατό να γεφυρωθεί το κενό μεταξύ των εκπαιδευτικών συστημάτων καθοδηγούμενων από υπολογιστή (ITS) και των εκπαιδευτικών συστημάτων καθοδηγούμενων από μαθητή (υπερμέσα) εκμεταλλευόμενου του μοντέλου χρήστη για να παρέχει αποτελεσματική προσαρμοστικότητα περιεχομένου (Brusilovsky, 1995). Ο στόχος των AHS

είναι να χρησιμοποιεί τα μοντέλα των χρηστών του μέσα από την μεταξύ τους αλληλεπίδραση με σκοπό την προσαρμογή του περιεχομένου υπερμέσων και των υπερσυνδέσμων στις ανάγκες τους [23].

Συνήθης κατάταξη χρηστών στα Προσαρμοζόμενα Συστήματα

- Αρχάριος (Novice)
- Μέσος (Intermediate)
- Εξειδικευμένος (Expert)

2.2 Προσαρμοζόμενα Εκπαιδευτικά Υπερμέσα

Στα Προσαρμοζόμενα Εκπαιδευτικά Υπερμέσα το εκπαιδευτικό υλικό προκύπτει από την εξατομικευμένη αλληλεπίδραση συστήματος και χρήστη στα πλαίσια συγκεκριμένων παιδαγωγικών στόχων, της γνώσης του συστήματος σχετικά με το γνωστικό αντικείμενο, καθώς και τα χαρακτηριστικά του εκπαιδευόμενου [17].

Οι βασικοί παράγοντες ενός προσαρμοζόμενου συστήματος είναι το γνωστικό και το μαθησιακό στυλ. Το γνωστικό στυλ αναπαριστά μια συλλογή ιδιοτήτων που συμβαίνουν σε πολλούς εκπαιδευόμενους και επιτρέπουν στο σύστημα να βγάλει συμπεράσματα βασιζόμενο σε μικρό αριθμό παρατηρήσεων, ενώ το μαθησιακό στυλ έχει μια ευρεία έννοια που περιλαμβάνει το γνωστικό μαζί με το συναισθηματικό και ψυχολογικό στυλ [15]. Τα Προσαρμοζόμενα Εκπαιδευτικά Υπερμέσα αποτελούνται συνήθως από τρία μέρη:

1. Μοντέλο πεδίου(domain model) για την παρουσίαση του συστήματος γνώσης του πεδίου.
2. Μοντέλο χρήστη (user model) που καταγράφει τα προσωπικά χαρακτηριστικά του εκπαιδευόμενου, όπως ηλικία, γλώσσα, μαθησιακό στυλ κλπ. καθώς και το μορφωτικό του επίπεδο.
3. Μηχανισμός προσαρμογής (adaptation module) που προσαρμόζει την παρουσίαση του περιεχομένου και την πλοήγηση του χρήστη όπως έχουν καταγραφεί στο μοντέλο του χρήστη.

Παρακάτω φαίνονται ορισμένα παραδείγματα εφαρμογών Προσαρμοζόμενων Εκπαιδευτικών Υπερμέσων (Adaptive Educational Hypermedia):

- Interbook (Brusilovsky, <http://www2.sis.pitt.edu/~peterb/InterBook.html>)
- AHA (Paul De Bra, <http://aha.win.tue.nl/>)
- GRAPPLE Project (<http://grapple.win.tue.nl/summary/>)
- i-Weaver (Wolf: Individualised Web-based learning)
- ELM-ART (Brusilovsky, Schwarz, & Weber, 1996)
- InterBook@AHA! (Ramp, De Bra, and Brusilovsky)
- NetCoach(<http://www.netcoach.eu.com/>)

2.3 Μοντελοποίηση Χρήστη (User Modeling)

Εξ ορισμού ένα Μοντέλο Χρήστη (User Model) είναι η αναπαράσταση της γνώσης και των προτιμήσεων, τις οποίες το σύστημα υποθέτει ότι ο χρήστης κατέχει [6]. Η αρχική εφαρμογή της Μοντελοποίησης Χρήστη γινόταν μέσα από την εφαρμογή και δεν υπήρχε σαφής διαχωρισμός ώστε να διαχειρίζεται ως ξεχωριστή συνιστώσα [20]. Τα μοντέλα χρήστη κατηγοριοποιούνται στις παρακάτω κατηγορίες [24] :

- Μοντέλο που βασίζεται σε στερεότυπα (stereotype model). Αναπαριστά μια ομάδα ανθρώπων που συγκεντρώνουν κοινά χαρακτηριστικά.
- Μοντέλο επικάλυψης (overlay model). Το μοντέλο του εκπαιδευόμενου αποτελεί υπομονάδα του μοντέλου πεδίου. Το πεδίο αποσυντίθεται σε ένα σύνολο στοιχείων και η επικάλυψη είναι το σύνολο της κτήσης των στοιχείων αυτών.
- Διαφορικό μοντέλο(differential model). Είναι βασικά μια επικάλυψη της αναμενόμενης γνώσης, η οποία με τη σειρά της είναι μια επικάλυψη της εξειδικευμένης γνώσης.

- Μοντέλο διατάραξης (perturbation model). Αναπαριστά τους εκπαιδευόμενους ως ένα υποσύνολο της γνώσης των εξειδικευμένων επιπρόσθετα στην χαμηλού επιπέδου γνώση τους.

2.3.1 Ιστορική αναδρομή

Η μελέτη του αντικειμένου της Μοντελοποίησης Χρήστη ξεκινά στα τέλη της δεκαετίας του '70 και μέχρι τα μέσα της δεκαετίας του '80 τα θέματα δημιουργίας, αναπαράστασης και εκμετάλλευσης υποθέσεων διερευνώνται διαμέσου των παρακάτω τεσσάρων πεδίων [7]:

- Συστήματα φυσικής γλώσσας στον τομέα της έρευνας τεχνητής νοημοσύνης (natural-language systems)
- Έξυπνα συστήματα διδασκαλίας (intelligent tutoring systems)
- Αλληλεπίδραση ανθρώπου υπολογιστή (human-computer interaction)
- Ανάκτηση πληροφορίας (information retrieval)

Αργότερα οι μεθοδολογίες εξελίχθηκαν σε μεγάλο βαθμό και εκτός από τις παραπάνω συνεισφορές στη Μοντελοποίηση Χρήστη έρχονται να συνεισφέρουν και άλλα πεδία όπως :

- Έξυπνοι πράκτορες (Intelligent Agents),
- Σημασιολογικός Ιστός (Semantic Web),
- Γνωσιακή Επιστήμη (Cognitive Science),
- Οπτικοποίηση (Visualization), κ.α.

Η έρευνα για συστήματα μοντελοποίησης χρήστη ξεκίνησε από την ανάπτυξη των Συστημάτων Κελύφους και έφτασε μέχρι την υλοποίηση των Διακομιστών Μοντελοποίησης Χρήστη.

- Συστήματα Κελύφους Μοντελοποίησης Χρήστη (User Model Shell Systems) – Χρησιμοποιούσαν στατικά μοντέλα Χρήστη όπου οι πληροφορίες συγκεντρώνονται μια φορά στην αρχή και δεν ενημερώνονται στη συνέχεια.

- Διακομιστές Μοντελοποίησης Χρήστη (User Modeling Servers)- Πρόκειται για δυναμικά μοντέλα χρήστη των οποίων τα προφίλ ανανεώνονται στη διάρκεια του χρόνου.

Τα **συστήματα κελύφους (User Modeling Shell Systems)** αφορούν στατικά μοντέλα χρήστη και διερευνώνται μέχρι το 1995 ως ενσωματωμένα στις εφαρμογές. Στον πυρήνα κάθε Συστήματος Μοντελοποίησης Χρήστη υπάρχει ένας μηχανισμός για την αναπαράσταση και την αιτιολόγηση των υποθέσεων που έχουν γίνει για τον χρήστη. Γι' αυτόν το λόγο, κάθε τέτοιο σύστημα κελύφους θα πρέπει να προσφέρει ικανότητες αναπαράστασης και αιτιολόγησης που θα καλύπτουν τις ανάγκες όσο το δυνατόν περισσότερων συστημάτων [26]. Η ανάγκη επαναχρησιμοποίησης οδήγησε στην αναζήτηση Συστημάτων Κελύφους ανεξάρτητα της εφαρμογής. Η πρώτη αναφορά των Συστημάτων κελύφους για τον διαχωρισμό της Μοντελοποίησης Χρήστη από τους υπόλοιπους μηχανισμούς μιας εφαρμογής έγινε από τον Kobsa το 2001. Η πρώτη αναφορά των Συστημάτων κελύφους για τον διαχωρισμό της Μοντελοποίησης Χρήστη από τους υπόλοιπους μηχανισμούς μιας εφαρμογής έγινε από τον Kobsa το 2001. Μερικά γνωστά συστήματα που αναπτύχθηκαν είναι:

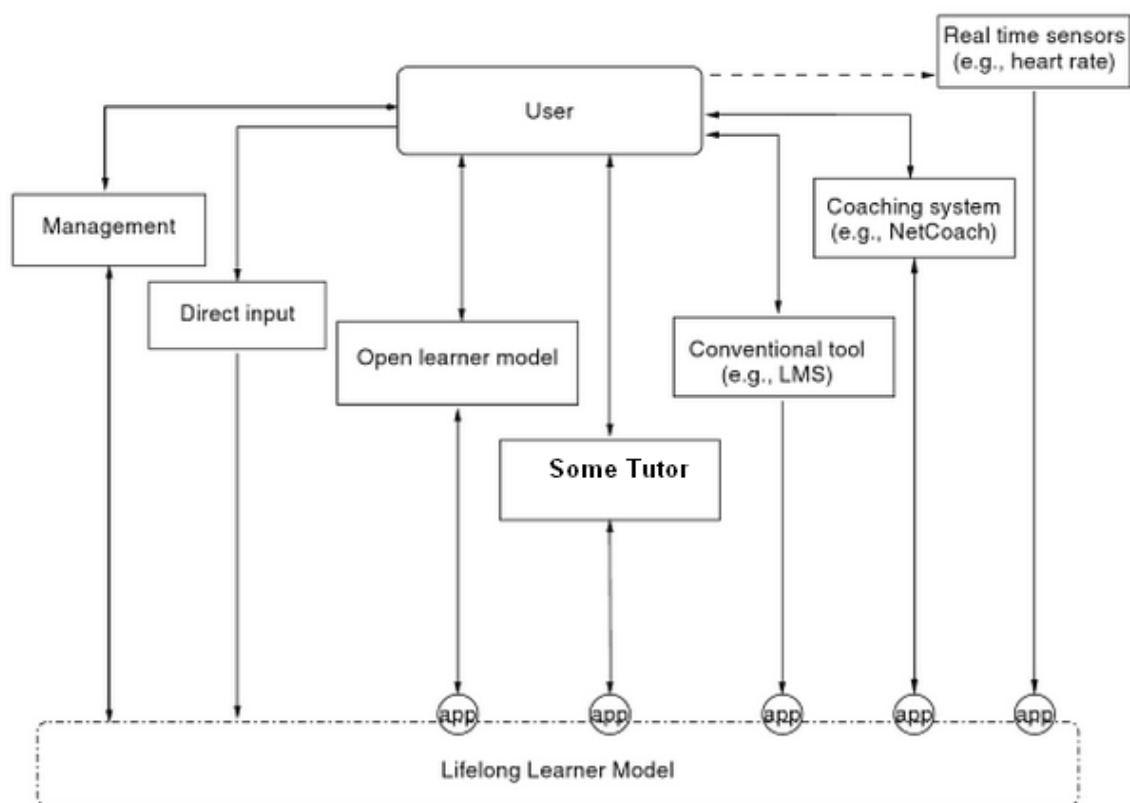
- UMT,
- BGP-MS,
- UM toolkit,
- Tagus

Από το 2000 και μετά ξεκινούν και αναπτύσσονται οι **Διακομιστές Μοντελοποίησης Χρήστη (User Modeling Servers)**. Οι Διακομιστές Μοντελοποίησης Χρήστη αφορούν δυναμικά Μοντέλα Χρήστη, δηλαδή μοντέλα τα οποία ανανεώνονται και προσαρμόζονται στις αλλαγές των χαρακτηριστικών του χρήστη. Η αρχική εφαρμογή αυτών των διακομιστών αφορούσε εμπορική χρήση. Στη συνέχεια επεκτάθηκε και σε άλλες περιοχές όπως τα εκπαιδευτικά συστήματα. Οι διακομιστές Μοντελοποίησης Χρήστη βασίζονται στην αρχιτεκτονική Πελάτη-Διακομιστή (Client-Server) διατηρώντας τις πληροφορίες για τα προφίλ των χρηστών στην μνήμη του διακομιστή. Έτσι κάθε εφαρμογή μπορεί ανά πάσα στιγμή ή ανά τακτά διαστήματα να έχει πρόσβαση σε αυτές τις πληροφορίες είτε για ανάκτηση είτε για ενημέρωσή τους [16]. Μερικά παραδείγματα συστημάτων Διακομιστών είναι :

- Group Lens
- ATG Adaptive Scenario Engine
- enQuiry™ Identity Service
- Personis
- CUMULATE

2.3.2 Δια βίου Μοντελοποίηση Χρήστη (Lifelong User Modeling)

Η διάχυτη και απανταχού χρήση του υπολογιστή συλλέγει και διατηρεί πολύ μεγάλο αριθμό προσωπικών πληροφοριών. Η τεράστια ποσότητα αυτών των πληροφοριών προσφέρει εξαιρετικά προνόμια. Ένα από αυτά είναι και η Δια βίου μάθηση [18]. Ο όρος χρησιμοποιήθηκε για πρώτη φορά από την Judy Kay και παρουσιάστηκε σαν μια προσπάθεια να μοντελοποιήσει τις καθημερινές δια-βίου αλληλεπιδράσεις των χρηστών με διάφορες υπηρεσίες του παγκόσμιου ιστού προσφέροντας παράλληλα στους χρήστες την δυνατότητα να εξετάζει με λεπτομέρεια και να ελέγχει όλη την διαδικασία εξατομίκευσης [19]. Αναφέρεται στη σύλληψη μακρόχρονων πληροφοριών από τους χρήστες (Judy Kay, 2012).



ΕΙΚΟΝΑ 2.1 – Μοντέλο δια-βίου εκπαιδευόμενου

Υπάρχουν πολλά θέματα τα οποία πρέπει να εξεταστούν και πολλά ερωτήματα που χρήζουν απαντήσεων πριν από την εφαρμογή ενός τέτοιου συστήματος. Τέτοια ερωτήματα αφορούν την απόδοση, τα πρότυπα που θα χρησιμοποιηθούν, την ασφάλεια των δεδομένων κ.α.

Κεφάλαιο 3

Προσαρμοζόμενα Υπερμέσα ΑΗΑ & GALE

Όλες σχεδόν οι δημοσιεύσεις για τα Προσαρμοζόμενα Υπερμέσα, που δημοσιεύτηκαν πριν από το 1996, περιγράφουν κλασικό προ-Ιστού υπερκείμενο ή υπερμέσα. Αντιθέτως, η πλειονότητα των δημοσιεύσεων που δημοσιεύτηκαν από το 1996 και μετά αφορούν Προσαρμοζόμενα Υπερμέσα που βασίζονται στον Ιστό [03]. Κάποια πρώιμα συστήματα Προσαρμοζόμενων Υπερμέσων που αναπτύχθηκαν για εκπαίδευση βασίζομενη στον ιστό, όπως είναι το ELM-ART [05], το Interbook (Brusilovsky, Schwarz, & Weber, 1996a) και το 2L670 (DeBra, 1996) παρείχαν την απόδειξη της ύπαρξης τέτοιων συστημάτων και επηρέασαν ορισμένα συστήματα που αναπτύχθηκαν αργότερα [03].

Η μετέπειτα δουλειά για τα δεύτερης γενιάς Προσαρμοζόμενα Εκπαιδευτικά Υπερμέσα έγινε μεταξύ του 1996 και του 2002. Υπήρξαν τρεις κατηγορίες ερευνητικών ομάδων. Μία ομάδα είχε ως κίνητρο την παραγωγή συστημάτων για διδακτική χρήση. Μια δεύτερη την ανάπτυξη νέων τεχνικών για τα συστήματα αυτά. Και μία τρίτη ομάδα την ανάπτυξη συντακτικών εργαλείων [3].

Παρά την σημαντική αυτή έρευνα, τα Προσαρμοζόμενα Υπερμέσα δεν κατάφεραν να επηρεάσουν αρκετά την εκπαίδευση βασιζόμενη στον ιστό. Μόνο πολύ λίγα από αυτά τα συστήματα χρησιμοποιούνται για την διδακτική πραγματικών μαθημάτων. Αντ' αυτών η πλειονότητα των μαθημάτων μέσω ιστού βασίζονται στα λεγόμενα LMS (Learning Management Systems- Συστήματα Διαχείρισης Μάθησης). Τα LMS παρέχουν λύσεις τόσο για τον εκπαιδευτή όσο και για τον εκπαιδευόμενο [3].

Ένας αριθμός πρωτότυπων ερευνών στα προσαρμοζόμενα συστήματα μάθησης εμπλουτισμένης με τεχνολογία (TEL – Technology Enhanced Learning) που δημιουργήθηκαν, χρησιμοποιήθηκαν κυρίως για λόγους επίδειξης ή για προσαρμοζόμενη παράδοση εκπαιδευτικού υλικού στα ινστιτούτα των συντακτών τους. Μια αξιοσημείωτη εξαίρεση αποτελεί το σύστημα AHA! (De Bra, 2006), το οποίο είναι μια γενικού σκοπού πλατφόρμα ανοιχτού κώδικα προσαρμοζόμενων υπερμέσων, και χρησιμοποιήθηκε από πολλούς ερευνητές και εκπαιδευτές ανά τον κόσμο. Ωστόσο, η σύνδεση της προσαρμοζόμενης τεχνολογίας σε εφαρμογές μάθησης παραμένει αρκετά περιορισμένη. Ο σκοπός του GRAPPLE project ήταν να το αλλάξει αυτό, εισάγοντας την προσαρμοζόμενη μάθηση εμπλουτισμένη με τεχνολογία (adaptive TEL) στο προσκήνιο. Αυτό επιτυγχάνεται με την ολοκλήρωση ενός προσαρμοζόμενου περιβάλλοντος TEL(πλέον χρησιμοποιείται το ALE –Adaptive Learning Environment- ως συντομογραφία) με τα Συστήματα Διαχείρισης Μάθησης (LMS-Learning Management Systems) χρησιμοποιώντας προσέγγιση υπηρεσιοκεντρικού πλαισίου [08].

3.1 Το σύστημα AHA

Το AHA αναπτύχθηκε αρχικά για να υποστηρίξει ένα διαδικτυακό μάθημα με κάποια καθοδήγηση για το χρήστη μέσα από υπό όρους επεξηγήσεις και απόκρυψη υπερσυνδέσμων. Μετά από ορισμένες αρχικές πειραματικές εκδόσεις η πρώτη επίσημη έκδοση, η έκδοση 1.0, δημοσιεύτηκε το 2000 . Σε σύγκριση με άλλα προσαρμοζόμενα συστήματα, όπως είναι το Interbook και πολλά άλλα, το AHA πλεονεκτεί ως προς την απλότητα. Από την αρχική έκδοση

έχει βέβαια εξελιχθεί στις εκδόσεις 2.0 και 3.0, οι οποίες διατηρούν ωστόσο την βασική απλότητα [09]. Ιδρυτής του συστήματος AHA είναι ο Paul De Bra.

3.1.1 Αρχιτεκτονική και λειτουργία AHA

Στο σύστημα AHA, το μάθημα βασίζεται σε έννοιες (concepts) οι οποίες έχουν χαρακτηριστικά(attributes). Οι τιμές των χαρακτηριστικών αυτών αλλάζουν όταν ο χρήστης πραγματοποιήσει κάποιο συμβάν. Έτσι ανανεώνεται και το Μοντέλο Χρήστη. Στο AHA, η γνώση για μία έννοια παράγεται είτε από τη μελέτη μιας μόνο σελίδας είτε από την εφαρμογή κάποιου τεστ. Αυτό συνεπάγεται ότι οι έννοιες έχουν εξαχθεί με δίκαιο τρόπο. Αν ο χρήστης πρέπει να διαβάσει πέντε σελίδες για να επιτύχει κάποιο συγκεκριμένο σκοπό, κάθε μια από τις πέντε σελίδες πρέπει να σχετίζεται με διαφορετική έννοια, και οι πέντε έννοιες μαζί να καθορίζουν τον επιθυμητό στόχο [12].

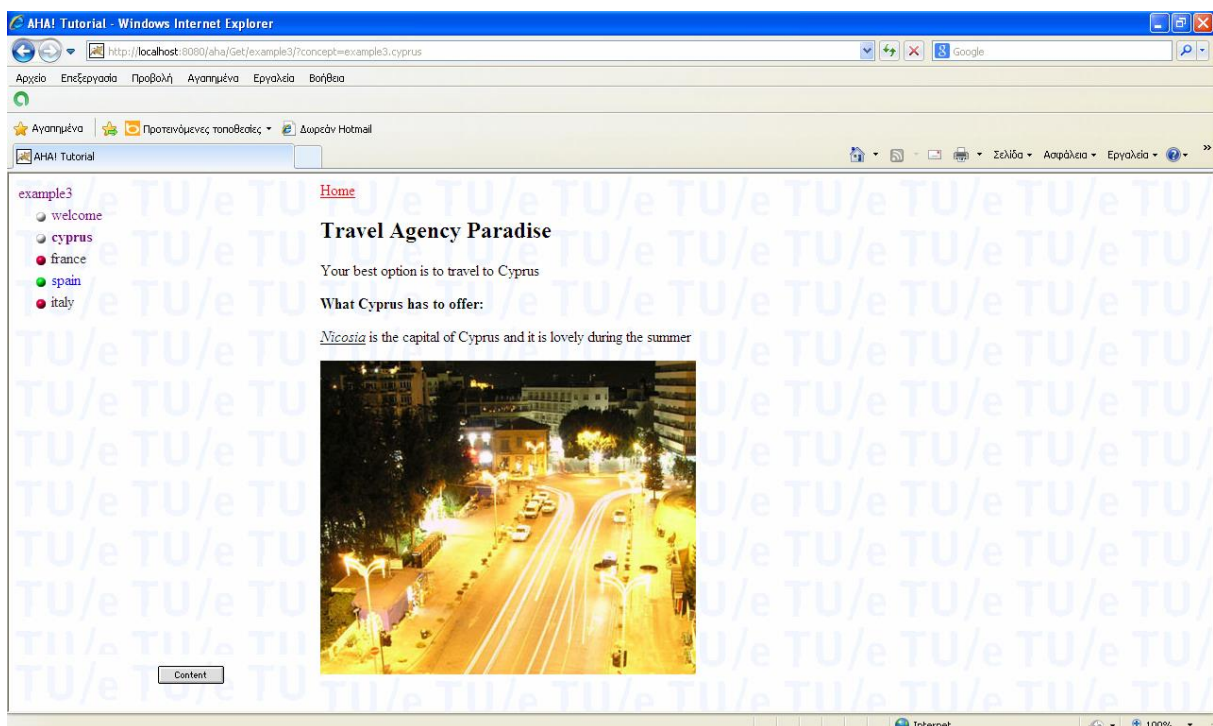
Εκτός από το σύνολο των γνωστών εννοιών, το λογισμικό AHA διατηρεί επίσης ένα αρχείο καταγραφής για κάθε χρήστη. Κάθε φορά που ο χρήστης έχει πρόσβαση σε μια σελίδα καταγράφονται δυο εισαγωγές εγγραφής, μία για την εκκίνηση και μία για την υποτιθέμενη λήξη της περιόδου που χρειάστηκε ο χρήστης για την ανάγνωση της σελίδας. (Με αυτό τον τρόπο καταγράφεται η διάρκεια ανάγνωσης για κάθε σελίδα). Για κάθε τεστ, καταγράφεται και η βαθμολογία. Το αρχείο καταγραφής είναι μέρος του Μοντέλου Χρήστη και μπορεί να χρησιμοποιηθεί για την ειδική σήμανση υπερσυνδέσμων στην σελίδα που έχουν προσπελαστεί και διαβαστεί από το χρήστη ώστε να διαφοροποιούνται από τους υπερσυνδέσμους που δεν έχουν διαβαστεί. Οι περισσότεροι WWW-browsers αλλάζουν το χρώμα των υπερσυνδέσμων με αυτό τον τρόπο, αλλά αυτή η συμπεριφορά εμπλέκεται με την προσαρμοζόμενη αναπαράσταση υπερσυνδέσμων. Επίσης οι χρήστες μπορούν να έχουν τη δυνατότητα να καθαρίσουν το ιστορικό του φυλλομετρητή και το σχετικό χρωματισμό των υπερσυνδέσμων και με αυτό τον τρόπο να απενεργοποιήσουν την καθοδήγηση που προσπαθεί να παρέχει η εφαρμογή προσαρμοζόμενου υπερκειμένου [12].

Το AHA είναι ένα εργαλείο για την δημιουργία εφαρμογών Προσαρμοζόμενων Υπερμέσων, αλλά δεν είναι ένα μαγικό ραβδί που μετατρέπει οποιοδήποτε υπερκείμενο ή υπεραρχείο σε ευκόλως χρησιμοποιήσιμη εφαρμογή Προσαρμοζόμενων Υπερμέσων [11]. Μερικά πολύ σημαντικά χαρακτηριστικά του συστήματος είναι ότι:

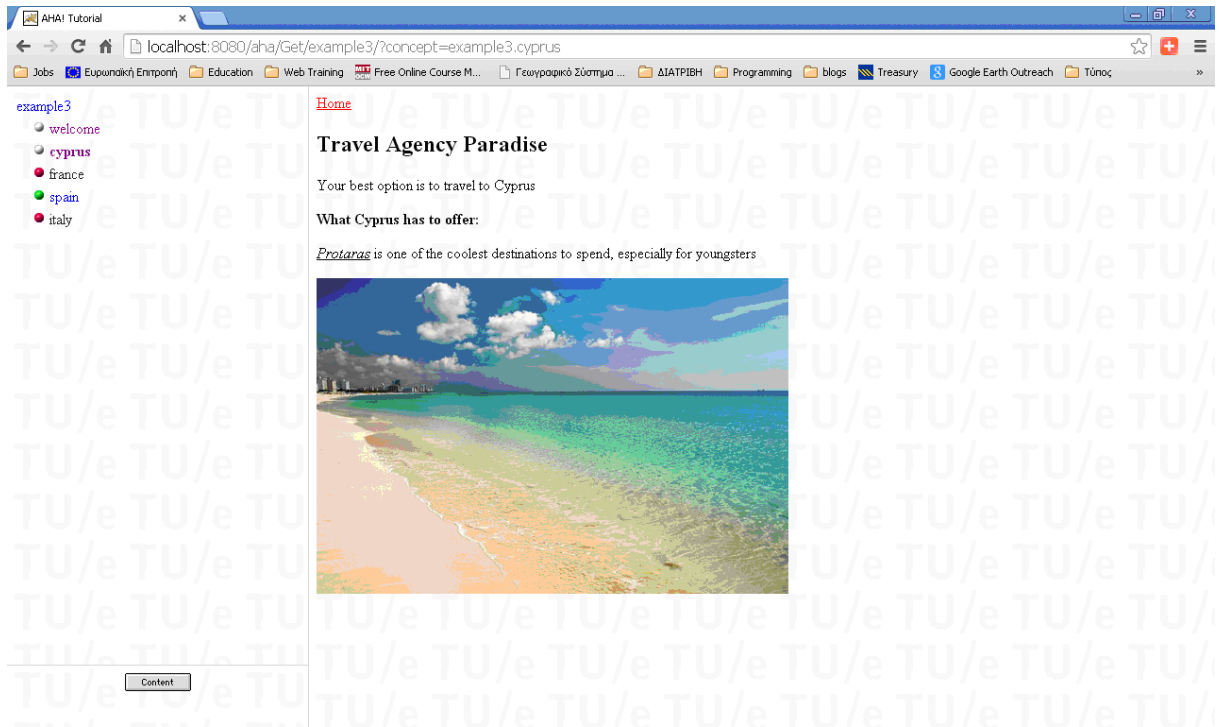
- παρέχει ανοιχτή αρχιτεκτονική, που σημαίνει ότι άλλα συστήματα μπορούν να ανακτήσουν και να ανανεώσουν ένα Μοντέλο Χρήστη [12] και,
- ότι διατηρεί ένα απλό Μοντέλο Χρήστη αποτελούμενο από ένα σύνολο μεταβλητών boolean. Στην αυθεντική εφαρμογή αυτές οι μεταβλητές χρησιμοποιούνται για την αναπαράσταση της γνώσης του χρήστη διαμέσου των ‘*εννοιών-concepts*’.

Οι τεχνικές προσαρμοστικότητας που χρησιμοποιεί το AHA είναι οι παρακάτω:

- **Προσαρμοζόμενη παρουσίαση - Adaptive presentation:** αυτές οι τεχνικές αποτελούνται και από την επιλογή των διαφόρων media ανάλογα με τις προτιμήσεις του χρήστη και από την προσαρμογή του περιεχομένου του κειμένου ή των πολυμέσων που βασίζεται σε ένα Μοντέλο Χρήστη. Ένα παράδειγμα προσαρμοζόμενης παρουσίασης είναι η εμφάνιση ή η απόκρυψη επιπλέον πληροφοριών ή και εικόνων, όπως επίσης και η μορφοποίηση όπως φαίνεται παρακάτω στις εικόνες 2 και 3 μετά από κάποιο συμβάν (π.χ. μετά από την 3^η επίσκεψη).

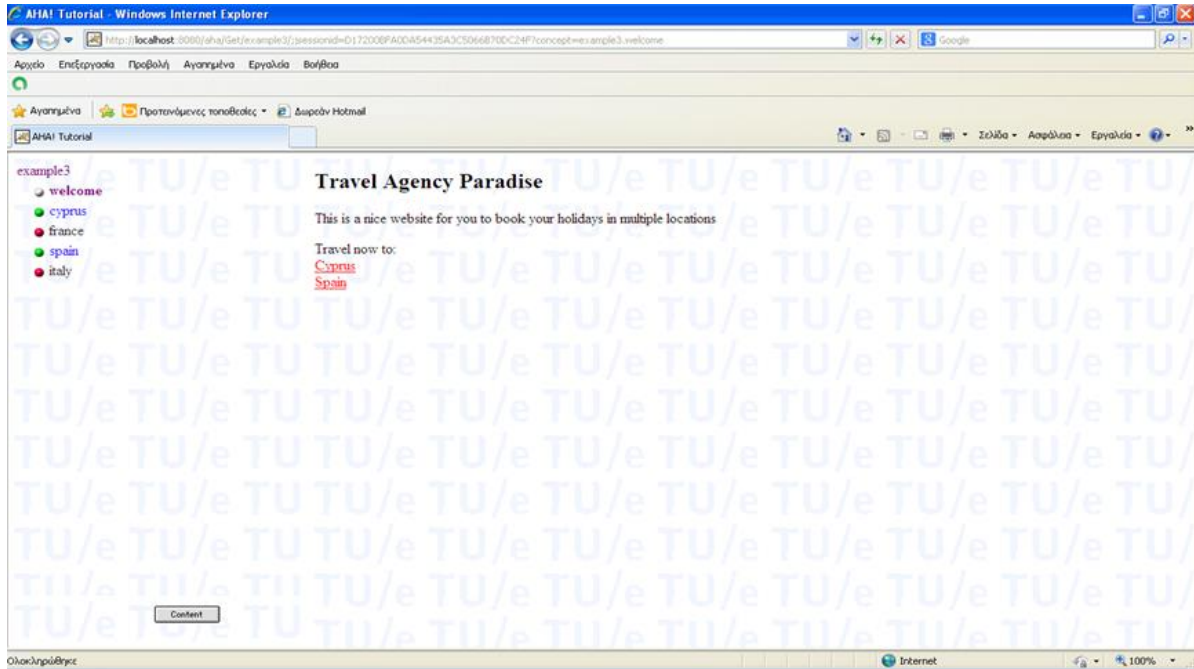


ΕΙΚΟΝΑ 3.1 – Προσαρμοζόμενη παρουσίαση στο AHA (το περιεχομενο της σελίδας cyprus κατά τις τρεις πρώτες επισκέψεις).

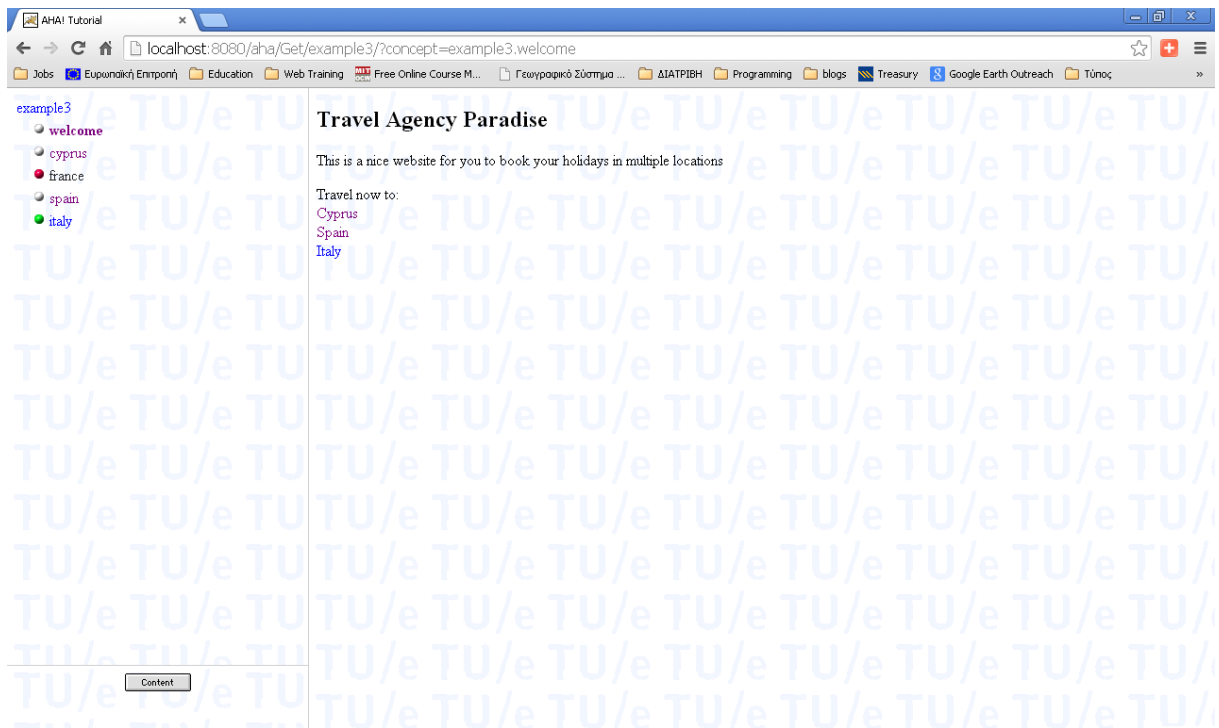


ΕΙΚΟΝΑ 3.2 – Προσαρμοζόμενη παρουσίαση στο ΑΗΑ (το περιεχόμενο της σελίδας cyprus μετά τις τρεις πρώτες επισκέψεις)

- **Υποστήριξη προσαρμοζόμενης πλοήγησης - Adaptive navigation support:** αυτές οι τεχνικές αλλάζουν τη δομή των υπερσυνδέσεων μεταξύ των σελίδων που αποτελούν το υπεραρχείο (hyperdocument). Ένα παράδειγμα προσαρμοζόμενης πλοήγησης είναι η αλλαγή στα χρώματα των υπερσυνδέσεων. Το χρώμα προσαρμόζεται όταν οι τοποθεσίες είναι κατάλληλες για να τις επισκεφθεί ο χρήστης, και όταν τις έχει ήδη επισκεφθεί ο χρήστης ή όχι, όπως φαίνεται στις εικόνες 4 και 5.



ΕΙΚΟΝΑ 3.3 – Προσαρμοζόμενη πλοήγηση στο AHA (ο χρήστης έχει επισκεφθεί μόνο τη σελίδα welcome. Οι σελίδες cyprus και spain είναι κατάλληλες για επίσκεψη και οι σελίδες france και italy είναι ακατάλληλες)



ΕΙΚΟΝΑ 3.4 - Προσαρμοζόμενη πλοήγηση στο ΑΗΑ (ο χρήστης έχει επισκεφθεί μόνο τη σελίδα welcome. Οι σελίδες cyprus και spain είναι κατάλληλες για επίσκεψη και οι σελίδες france και italy είναι ακατάλληλες)

Κάθε χρήστης είναι γνωστός στο σύστημα μέσω μιας μοναδικής ταυτότητας που διατηρείται μέσω των cookies. Ένας διακομιστής ΑΗΑ! μπορεί να έχει πολλές εφαρμογές και ο χρήστης χρησιμοποιώντας την μοναδική ταυτότητά του μπορεί να μεταβαίνει από τη μια στην άλλη αρκεί να χρησιμοποιεί τον ίδιο φυλλομετρητή (browser). Το ΑΗΑ καταγράφει την πρόσβαση του χρήστη σε κάθε σελίδα. Ο συντάκτης, μέσα από ένα σύνολο κανόνων προσαρμογής (adaptation rules), καθορίζει τον τρόπο ενημέρωσης του μοντέλου χρήστη, το οποίο αποτελείται από τις έννοιες (concepts) με τα χαρακτηριστικά τους (attributes). Κάποια άλλα σύνολα κανόνων, επίσης ορισμένα από τον συντάκτη καθορίζουν την εμφάνιση των υπερσυνδέσμων στο μενού πλοήγησης και τα χρώματά τους δείχνοντας κάθε φορά την τρέχουσα κατάσταση κάθε έννοιας.

Οι σελίδες κειμένου στο περιεχόμενο κάθε έννοιας περιέχουν κανόνες για την καταλληλότητα των προτάσεων. Οπότε, υπάρχει περίπτωση το περιεχόμενο μιας σελίδας να είναι διαφορετικό κάθε φορά. Οι σελίδες μπορεί να περιέχουν, επίσης, κομμάτια κειμένου των οποίων η εμφάνιση μπορεί να προσαρμοστεί (μεγαλύτερη γραμματοσειρά, κλπ). Κάτι τέτοιο δεν θα είναι ορατό στο χρήστη αν χρησιμοποιεί τα κουμπιά Πίσω ή Μπροστά του browser, τα οποία οδηγούν στην σελίδα όπως ακριβώς την είδε ο χρήστης τελευταία φορά. Το ΑΗΑ μπορεί να περιέχει φόρμες από όπου ο χρήστης μπορεί να αλλάξει τις ρυθμίσεις του μοντέλου χρήστη που χρησιμοποιείται

για την περίπτωση του, φόρμες για να αλλάξει τα χρώματα που χρησιμοποιούνται για την εμφάνιση των εννοιών στο μενού πλοήγησης, ή και φόρμες ερωτήσεων πολλαπλής επιλογής. Κάνοντας log off διαγράφεται η συνεδρία ο χρήστης οδηγείται στην αρχική σελίδα.

Οι συντάκτες(authors) είναι ειδικοί χρήστες οι οποίοι δημιουργούνται από τον διαχειριστή και έχουν πρόσβαση στο περιβάλλον εργασίας συντάκτη(Author Workplace) και στα διάφορα συντακτικά εργαλεία. Σε κάθε περίπτωση θα πρέπει να δημιουργηθεί μια βασική ιεραρχική δομή της οποίας η βάση θα έχει πάντα ένα όνομα σχετικό την σημασία της μάθησης, καθώς επίσης και μια φόρμα πρόσβασης η οποία θα υποστηρίζει και νέες εγγραφές. Η δημιουργία των εφαρμογών AHA περιλαμβάνει την συγγραφή σελίδων, την δημιουργία της δομής των εννοιών, τον ορισμό των κανόνων προσαρμογής, και την τελική εμφάνιση του περιεχομένου.

Το περιεχόμενο των σελίδων είναι γραμμένο σε XHTML, XHTML+AHA format ή XHTML Standard και η επεξεργασία γίνεται μέσω του XML parser ώστε να μπορεί ο συντάκτης να εντοπίσει κάποιο σφάλμα και να το διορθώσει. Οι κεφαλίδες και τα υποσελίδα χρησιμοποιούνται όπως τα αντικείμενα μέσω ετικετών. Οι ειδικές ετικέτες που επιτρέπεται να χρησιμοποιηθούν είναι η <variable> και η <handler> για τα χαρακτηριστικά του μοντέλου χρήστη και την πρόσβαση σε ειδικές σελίδες(logout,κλπ). Στο περιεχόμενο μπορούν, ακόμη, να χρησιμοποιηθούν ή να αποκρύπτονται προτάσεις υπό όρους. Αυτό προϋποθέτει η σελίδα να είναι γραμμένη σε XHTML+AHA format. Μπορούν επίσης να χρησιμοποιηθούν αντικείμενα υπό όρους ώστε να μπορούν να προστεθούν αρχεία XML. Τέλος υπάρχει δυνατότητα έμφασης ή και απόκρυψης κειμένου υπό όρους.

Η δομή των εννοιών δημιουργείται με τη βοήθεια του graph author tool, του concept editor, ή με απευθείας επεξεργασία των αρχείων XML.

- Το **graph author tool** είναι το κύριο συντακτικό εργαλείο για την δομή των εννοιών. Παρέχει τη δυνατότητα δημιουργίας ιεραρχικών εννοιών και των μεταξύ τους σχέσεων.
- Με το **concept editor**, που είναι λίγο πιο περίπλοκο, ορίζονται οι έννοιες και κανόνες προσαρμογής.

Τα δυο αυτά εργαλεία επεξηγούνται εκτενέστερα στο υποκεφάλαιο 3.3.

Η δομή των εννοιών είναι πάντα ίδια και περιλαμβάνει τα παρακάτω στοιχεία: *name, description, resource, title, hierarchy, attributes, adaptation rules*.

Οι κανόνες προσαρμογής χρησιμοποιούνται για την ενημέρωση των εννοιών του μοντέλου χρήστη (user model). Κάθε κανόνας σχετίζεται με ένα χαρακτηριστικό (attribute) μιας έννοιας και ενεργοποιείται κάθε φορά που η τιμή του χαρακτηριστικού αλλάζει.

3.2 Το GRAPPLE Project

Το GRAPPLE είναι ένα project, το οποίο διήρκησε 3 χρόνια, από το 2008 έως το 2011, για την δημιουργία ενός γενικού περιβάλλοντος προσαρμοζόμενης μάθησης (Generic Adaptive Learning Environment) και την ολοκλήρωσή του με διάφορα Συστήματα Διαχείρισης Μάθησης (LMS) για την ενεργοποίηση της προσαρμοζόμενης δια-βίου μάθησης για εκπαιδευόμενους ανώτερης εκπαίδευσης ή για βιομηχανίες. Το GRAPPLE προσφέρει μια νέα δομή, η οποία συνδυάζει δύο αντικείμενα. [10]:

- Τα Περιβάλλοντα Προσαρμοζόμενης Εκμάθησης (Adaptive Learning Environments), τα οποία προσφέρουν υποστήριξη στη διαδικασία εκμάθησης μέσω της προσαρμοζόμενης καθοδήγησης
- Και το εξατομικευμένο περιεχόμενο εκπαιδευτικού υλικού

Η δομή του GRAPPLE αποτελείται από (<http://grapple.win.tue.nl/summary/index.html>):

- LMS (Learning Managements System) - Σύστημα Διαχείρισης Εκμάθησης για την διαχείριση της διαδικασίας εκμάθησης, την πρόοδο και την υποστήριξη
- GALE (GRAPPLE Adaptive Learning Environment) για την προσαρμοζόμενη παράδοση του εκπαιδευτικού υλικού

- GUMF (GRAPPLE User Modeling Framework), ένα γενικό Framework Μοντελοποίησης Χρήστη που συλλέγει πληροφορίες για κάθε εκπαιδευόμενο από τη χρήση των συστημάτων LMS και GALE που κάνει ο ίδιος και τα διαφορετικά παραδείγματα που χρησιμοποιεί.
- Μια διαδικασία σύνδεσης χρήστη(sign-on) η οποία εξασφαλίζει ότι όλα τα στοιχεία του GRAPPLE αναγνωρίζουν τον εκπαιδευόμενο ως μια μοναδική ταυτότητα.
- GEB (Generic Event Bus), ένα γενικό Event Bus το οποίο παρέχει δυνατότητες ασύγχρονης ανταλλαγής πληροφορίας μεταξύ όλων των στοιχείων GRAPPLE.

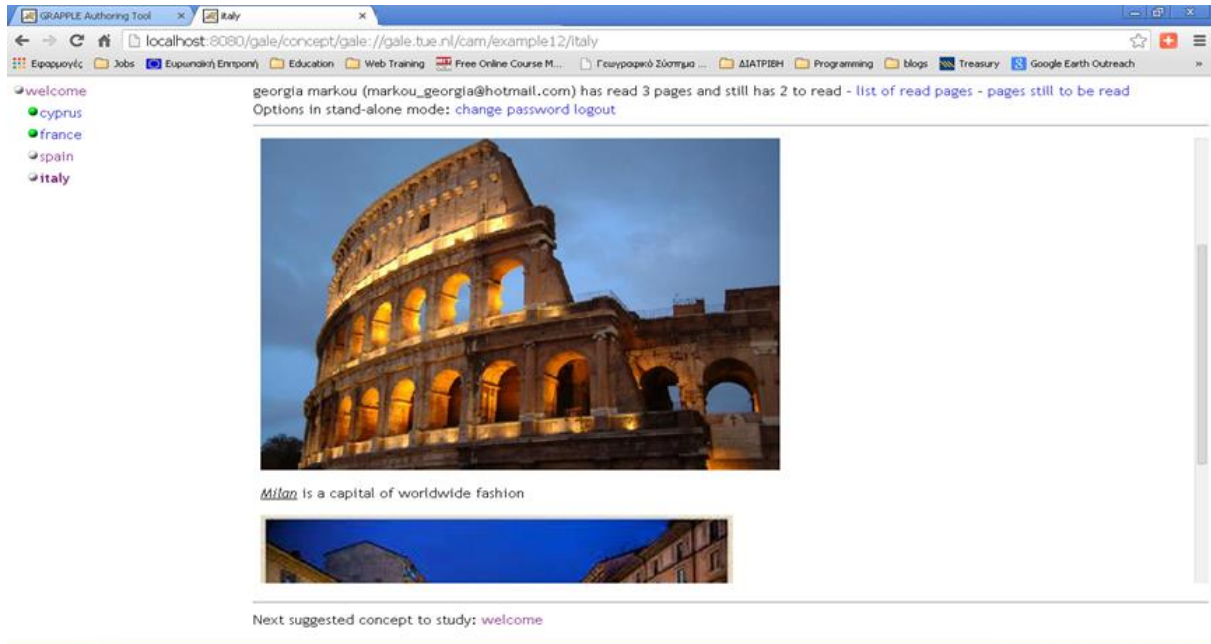
3.2.1 Το σύστημα GALE

Το σύστημα GALE είναι μια ευέλικτη, εκτενής μηχανή προσαρμογής που παρέχει και εξυπηρετεί τα προσαρμοζόμενα μαθήματα που έχουν συνταχθεί. Τα κύρια χαρακτηριστικά του είναι η υπηρεσία Μοντελοποίησης Χρήστη, η υπηρεσία μοντελοποίησης πεδίου/προσαρμογής και ο μηχανισμός προσαρμογής. Αυτά τα τρία συστατικά επικοινωνούν μεταξύ τους μέσω ενός διαύλου συμβάντων(event bus) το οποίο είναι επίσης προσβάσιμο και από άλλα συστατικά του GRAPPLE [30].

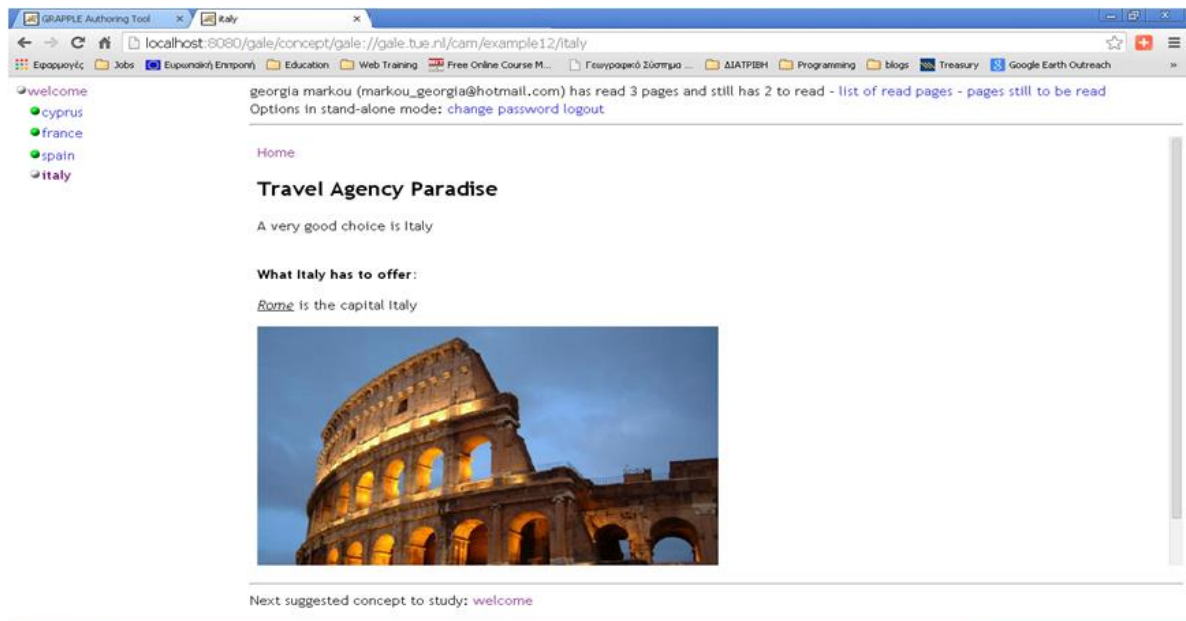
Οι πληροφορίες του χρήστη βρίσκονται κυρίως σε δυο σημεία μέσα στο GRAPPLE: Το GALE κρατά τις πληροφορίες που χρειάζονται για την εφαρμογή της προσαρμοστικότητας και το LMS αποθηκεύει τις προσωπικές πληροφορίες, την εγγραφή στο μάθημα, τη βαθμολογία και άλλες παρόμοιες πληροφορίες. Ωστόσο, όσο οι εκπαιδευόμενοι αυξάνουν τις εμπειρίες εκμάθησης, αυτό μπορεί να έχει ως αποτέλεσμα στο να έχουν την πληροφορία για τους εαυτούς τους και την μαθησιακή πρόοδό τους αποθηκευμένη σε πολλές τοποθεσίες [23].

Το σύστημα GALE, όπως και το σύστημα AHA, παρέχει :

- Προσαρμοζόμενη παρουσίαση (adaptive presentation). Όπως και στο ΑΗΑ προσαρμοζόμενη παρουσίαση είναι η τροποποίηση του περιεχομένου των σελίδων όταν ισχύει κάποια συνθήκη (π.χ. η γνώση για μια έννοια είναι τουλάχιστον 100).



ΕΙΚΟΝΑ 3.5 – προσαρμοζόμενη παρουσίαση στο GALE [όταν ο χρήστης έχει επισκεφτεί την Ισπανία(η γνώση για την έννοια είναι 100) στο περιεχομενο εμφανίζονται και πληροφορίες για τη Μίλαν

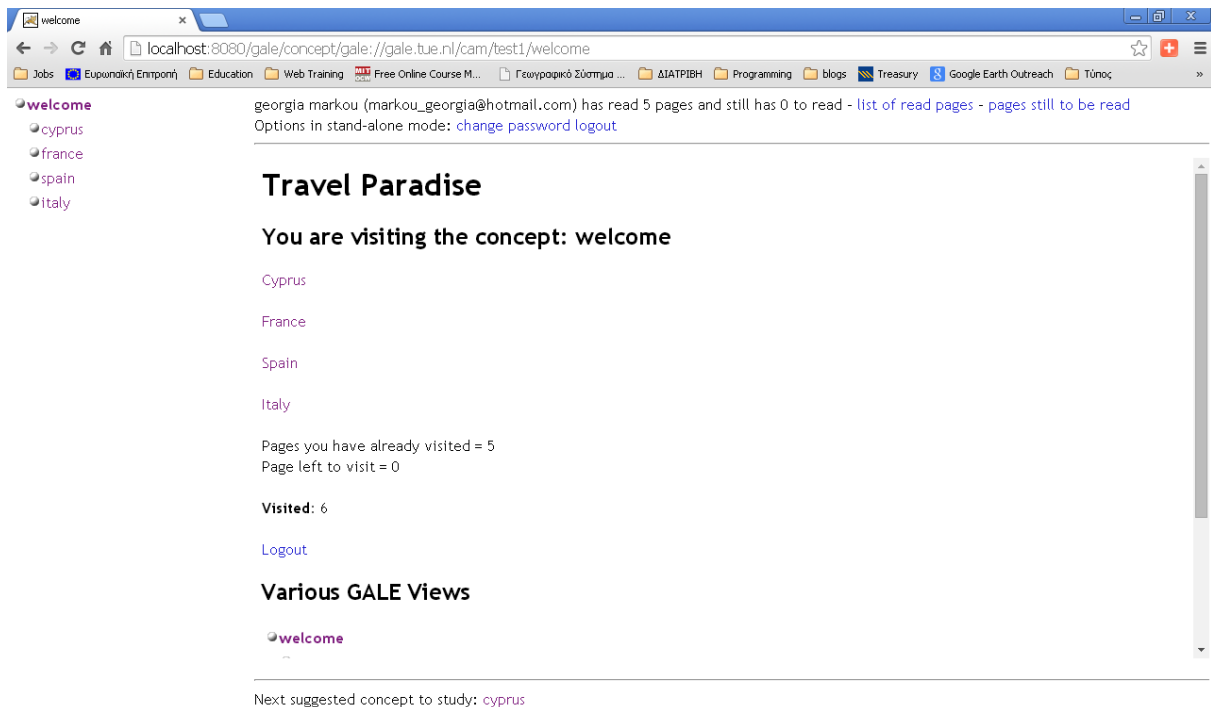


ΕΙΚΟΝΑ 3.6 - Προσαρμοζόμενη παρουσίαση στο GALE [όταν ο χρήστης δεν έχει επισκεφτεί την Ισπανία(η γνώση για την έννοια είναι 0) στο περιεχομενο εμφανίζονται μόνο οι πληροφορίες για τη Ρώμη]

- Προσαρμοζόμενη πλοήγηση (adaptive navigation). Κι εδώ όπως και στο ΑΗΑ προσαρμοζόμενη πλοήγηση είναι η τροποποίηση στο χρώμα γραμματοσειράς των υπερσυνδέσμων. Σύμφωνα με το χρώμα γραμματοσειράς ο χρήστης γνωρίζει ανα πάσα στιγμή ποιες σελίδες είναι κατάλληλες και ποιες σελίδες έχει ήδη επισκεφθεί όπως φαίνεται στις εικόνες 8 και 9.



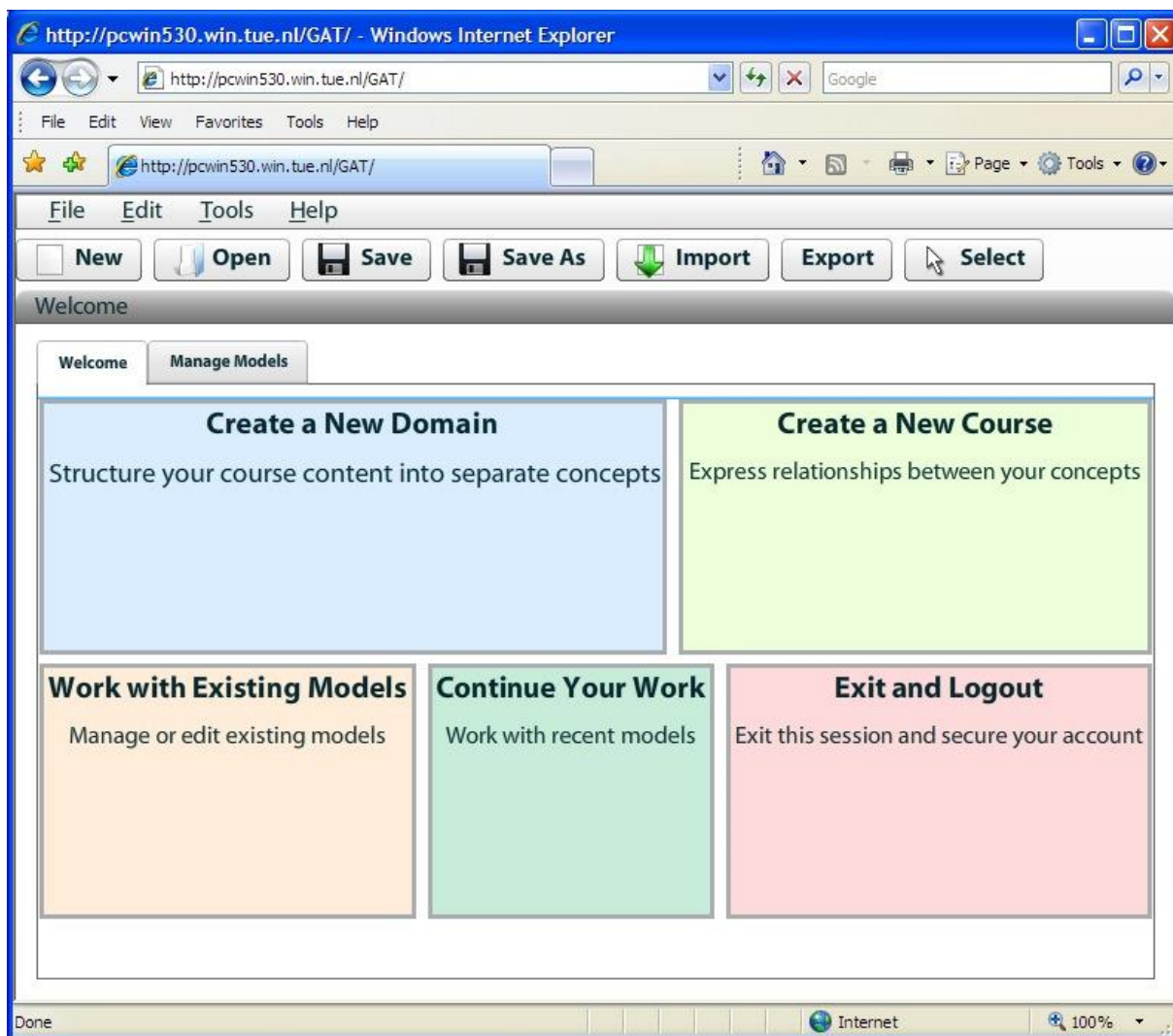
ΕΙΚΟΝΑ 3.7 – Προσαρμοζόμενη πλοήγηση στο GALE (Ο χρήστης έχει επισκεφτεί μόνο τη σελίδα welcome. Οι σελίδες cyprus, france και spain είναι κατάλληλες για επίσκεψη και εμφανίζονται στο περιεχομενο δεξιά. Η σελίδα italy είναι ακατάλληλη)



ΕΙΚΟΝΑ 3.8 - Προσαρμοζόμενη πλοήγηση στο GALE (Ο χρήστης έχει επισκεφτεί όλες τις σελίδες. Έτσι το χρώμα γραμματοσειράς των υπερσυνδέσμων είναι μωβ τόσο στο μενού αριστερά όσο και στο περιεχόμενο δεξιά)

Το GALE είναι σχεδιασμένο όσο το δυνατόν πιο ευέλικτο και επεκτάσιμο για να δώσει στους συντάκτες την ελευθερία να επιλέξουν μέσα από ένα μεγάλο εύρος μεθόδων και τεχνικών. Τα μαθήματα (courses) αποτελούνται από έννοιες και συσχετίσεις που μαζί δομούν το μοντέλο πεδίου (domain model). Για κάθε έννοια το GALE παρακολουθεί την αλληλεπίδραση του χρήστη με την έννοια. Αυτό επιτυγχάνεται κρατώντας πληροφορίες στις ιδιότητες (attributes) του μοντέλου χρήστη (User Model). Με κάθε έννοια το GALE συσχετίζει την συμπεριφορά ή τους κανόνες προσαρμογής (adaptation rules). Το GALE διαθέτει το DM (domain model) editor, το CAM (Conceptual Adaptation Model) editor και το PRT (pedagogical relationship type) Tool για τη σύνταξη του μοντέλου πεδίου, των κανόνων προσαρμογής και την δημιουργία νέων κανόνων αντίστοιχα. Και τα δυο είναι μέρη του GAT (GRAPPLE Authoring Tool). Όπως φαίνεται στην εικόνα 19 επιλέγοντας Create a New Domain ξεκινά το DM editor και επιλέγοντας Create a New Course ξεκινά το CAM editor. Με το CAM editor σχεδιάζονται οι παιδαγωγικές σχέσεις (pedagogical relationships) μεταξύ των εννοιών και από εκεί ένας compiler (μεταγλωττιστής) παράγει αυτόματα τους κανόνες προσαρμογής. Η σύνταξη ξεχωριστών κανόνων προσαρμογής δυσχεραίνει τους συντάκτες. Το GRAPPLE διαθέτει πλούσιο σύνολο από προκαθορισμένους κανόνες ώστε να χρησιμοποιήσει κανείς το PRT Tool μόνο σε εξειδικευμένες περιπτώσεις. Για κάθε έννοια του μοντέλου πεδίου πρέπει να κατασκευαστεί αντίστοιχα μια

ιστοσελίδα σε μορφή xhtml με το εκπαιδευτικό υλικό. Ή εναλλακτικά, μπορεί να υπάρχουν πολλές διαθέσιμες σελίδες για την ίδια έννοια και το GALE να αποφασίζει μέσω των ιδιοτήτων (attributes) του μοντέλου χρήστη ποια θα χρησιμοποιήσει κάθε φορά. Η XHTML δίνει τη δυνατότητα για χρήση των gale tags ώστε στο περιεχόμενο να παρουσιάζεται υπό όρους κείμενο.



ΕΙΚΟΝΑ 3.9 - GRAPPLE Authoring Tool

3.3 Σύγκριση AHA – GALE

Η επιλογή του καταλληλότερου συστήματος για την εκπαιδευτική εφαρμογή πραγματοποιήθηκε μετά από εκπαίδευση και εξάσκηση σε παραδείγματα και στα δυο συστήματα, AHA και GALE. Αναλυτικότερα, η εκπαίδευση ξεκίνησε από το σύστημα AHA και το

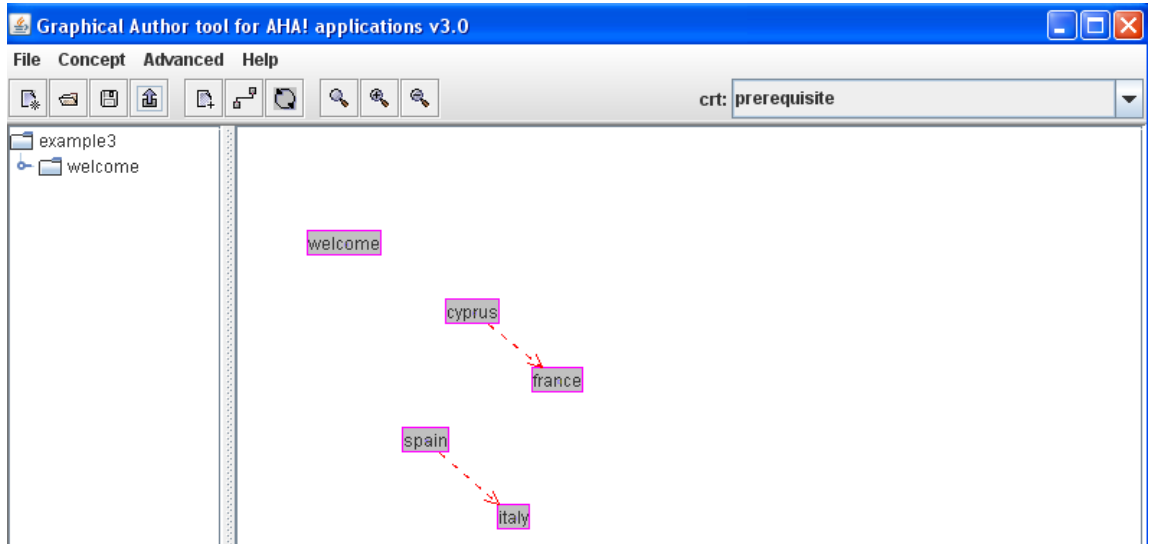
παράδειγμα που αναφέρει το ίδιο το tutorial του AHA στην επίσημη ιστοσελίδα του <http://aha.win.tue.nl:18080/aha/tutorial/>. Στη συνέχεια, και με την επιστημονική υποστήριξη του επιβλέποντος καθηγητή κ. Κυριάκου, υλοποιήθηκε ένα πολυπλοκότερο παράδειγμα και στα δύο συστήματα. Το παράδειγμα περιλάμβανε τη δημιουργία ιστοσελίδας ενός τουριστικού πρακτορείου και ορισμένων προορισμών με προσαρμοζόμενο περιεχόμενο.

Το βασικό μενού περιλαμβάνει τις κύριες σελίδες welcome, Cyprus, France, Spain, Italy. Το πράσινο χρώμα κουκίδας (μπλε χρώμα γραμματοσειράς υπερσυνδέσμου) σημαίνει ότι η σελίδα είναι κατάλληλη για επίσκεψη. Το κόκκινο χρώμα κουκίδας(μαύρο χρώμα γραμματοσειράς υπερσυνδέσμου) ότι η σελίδα είναι ακατάλληλη για επίσκεψη(η γνώση των προηγούμενων εννοιών δεν επαρκεί για την ανάγνωση αυτής της έννοιας). Και, τέλος, το άσπρο χρώμα κουκίδας(μωβ χρώμα γραμματοσειράς υπερσυνδέσμου) ότι τη σελίδα την επισκέφθηκε ο χρήστης όταν αυτή ήταν κατάλληλη.

Στο παράδειγμα, η γνώση(knowledge) για κάθε έννοια αυξάνεται κατά 10 μονάδες κάθε φορά που ο χρήστης επισκέπτεται την κάθε σελίδα και η επισκεψιμότητα(visited) κατά 1 μονάδα. Όταν ο χρήστης επισκεφθεί 5 φορές τη σελίδα Cyprus, τότε το ενδιαφέρον του(interest) θα ορίζεται στις 100 μονάδες. Το ίδιο ισχύει και για τη σελίδα Spain. Η σελίδα France θα φαίνεται κατάλληλη(suitable) όταν η γνώση της σελίδας Cyprus είναι τουλάχιστον 30 μονάδες. Η σελίδα Italy θα φαίνεται κατάλληλη όταν η επισκεψιμότητα της σελίδας Spain είναι τουλάχιστον 2. Στην πρώτη σελίδα, τη σελίδα welcome, ο χρήστης θα μπορεί να βλέπει τους υπερσυνδέσμους μόνο των σελίδων που είναι κατάλληλες. Όταν το ενδιαφέρον της σελίδας Cyprus είναι μικρότερο από 100 τότε θα εμφανίζεται μόνο το περιεχόμενο για την Λευκωσία. Όταν είναι μεγαλύτερο ή ίσο με 100 τότε εμφανίζεται στη σελίδα και το περιεχόμενο για τον Πρωταρά. Τις πρώτες 3 φορές που ο χρήστης θα επισκεφθεί τη σελίδα France, θα βλέπει μόνο το περιεχόμενο για το Παρίσι. Από την τέταρτη φορά θα βλέπει και πληροφορίες για τη Λυών.

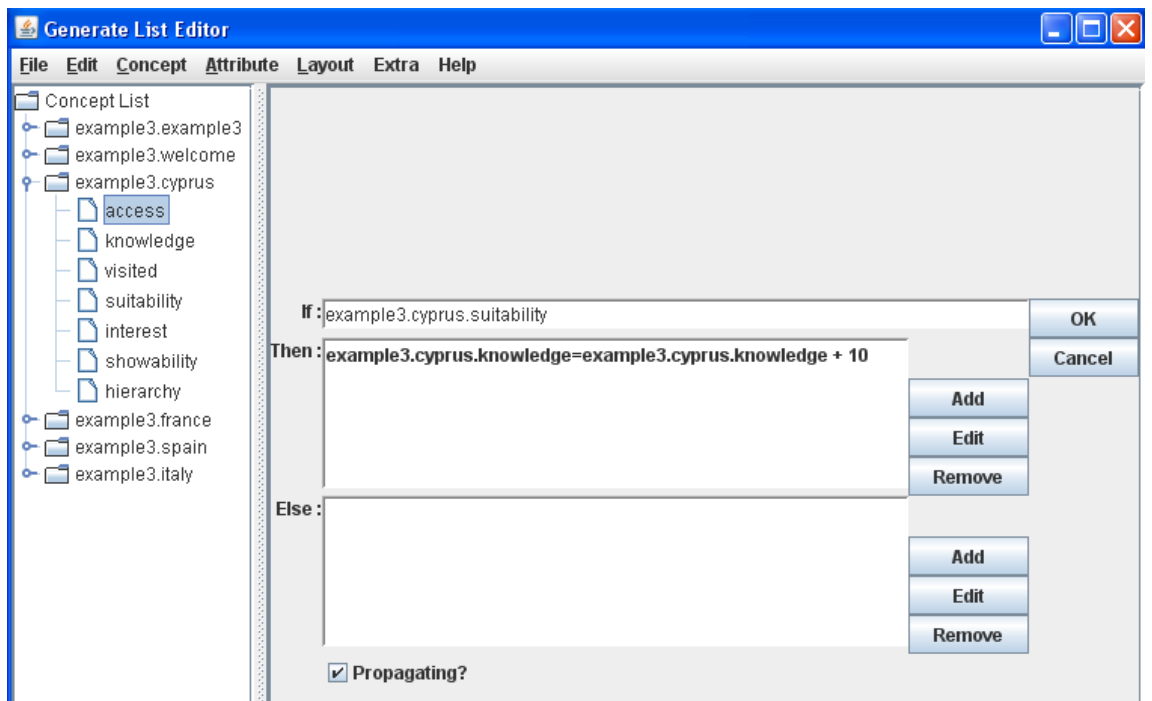
Για την εφαρμογή του παραδείγματος στο AHA! ακολουθήθηκαν τα παρακάτω βήματα:

1. Αρχικά, σχεδιάστηκε η δομή των σελίδων (ονόματα σελίδων και εννοιών, ιεραρχία, σχέσεις) και μετά δημιουργήθηκαν οι έννοιες και οι μεταξύ τους σχέσεις με το graph author tool.



ΕΙΚΟΝΑ 3.10 – AHA GRAPH AUTHOR TOOL

2. Στη συνέχεια ορίστηκαν τα attributes και τα adaptation rules (κανόνες προσαρμογής) με το concept editor.



ΕΙΚΟΝΑ 3.11 – AHA Concept editor

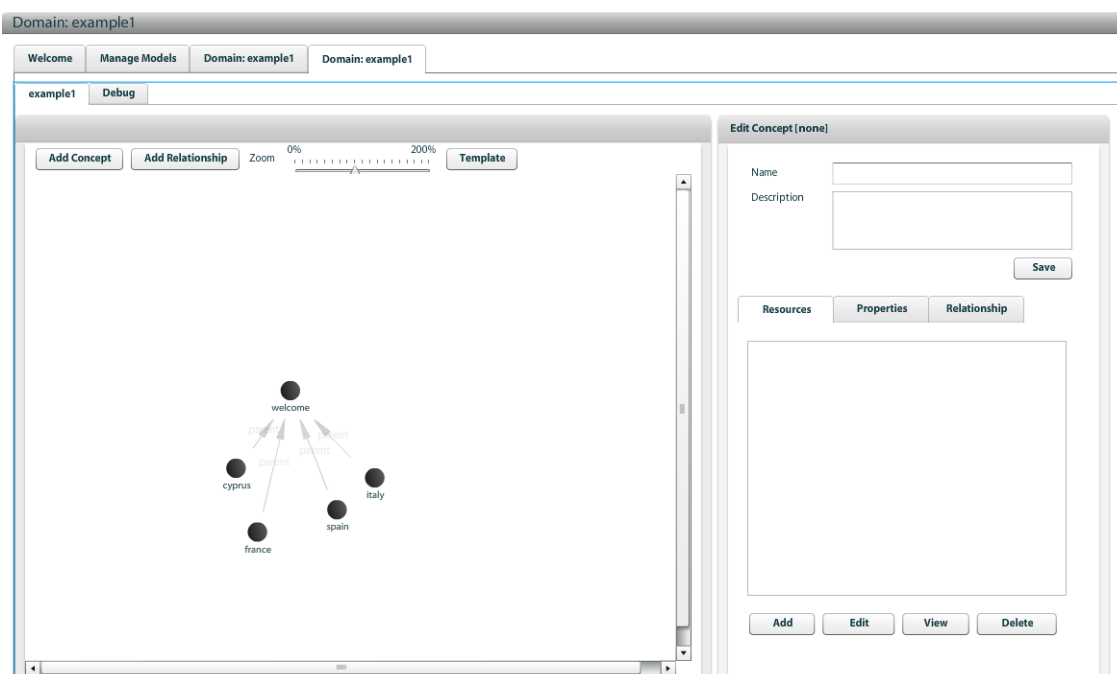
3. Στο επόμενο βήμα δημιουργήθηκαν οι σελίδες xhtml ή html και αποθηκεύτηκαν σε ένα νέο φάκελο (πχ example3) μέσα στον φάκελο aha. Προσοχή στον ίδιο φάκελο θα πρέπει να υπάρχουν τόσο τα δυο αρχεία LayoutConfig.xml και ConceptTypeConfig.xml, που

αφορούν την εμφάνιση του μενού και των περιεχομένων, όσο και το αρχείο index.html ως αρχική σελίδα που θα εμφανίζεται στο χρήστη, και το αρχείο Registration.html ως η σελίδα εισόδου πριν μπει στον προσαρμοζόμενο περιβάλλον του παραδείγματος. Οπότε αυτά αντιγράφηκαν από το φάκελο tutorial.

4. Τέλος, ελέγχεται η λειτουργία από την αρχική σελίδα index η οποία θα πρέπει να εμφανιστεί πρώτη, στη συνέχεια η σελίδα registration και μετά η πρώτη σελίδα του παραδείγματος, η σελίδα welcome.

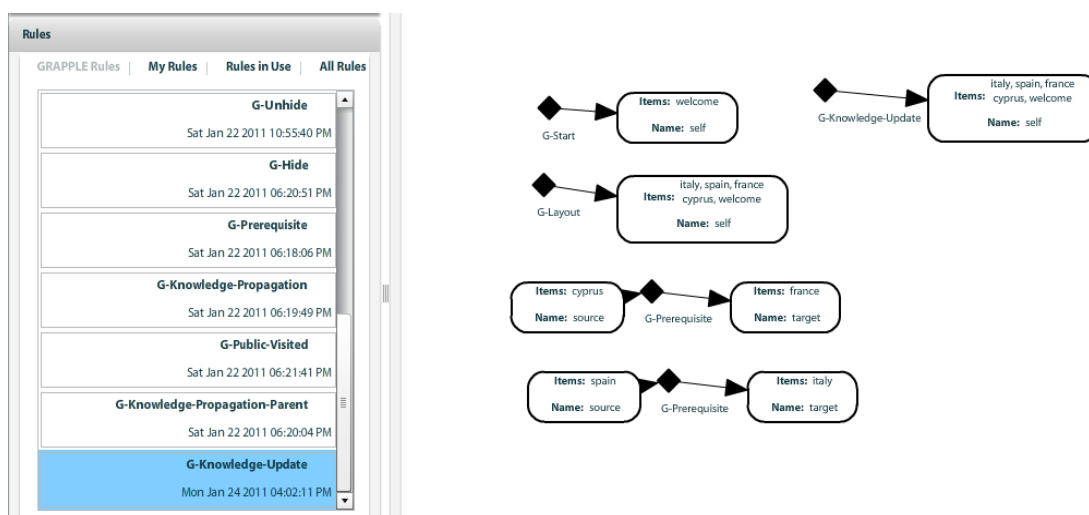
Για την εφαρμογή του παραδείγματος στο GALE ακολουθήθηκαν τα παρακάτω βήματα:

1. Σχεδιάστηκε το domain model με το GAT (Gale Authoring Tool). Δηλαδή οι έννοιες (concepts) και η δομή τους, κύριες σελίδες και υποσελίδες. Σε κάθε έννοια προστέθηκαν οι σελίδες xhtml ως resources, και οι ιδιότητές (properties) τους. Οι σελίδες xhtml που χρησιμοποιήθηκαν είναι οι ίδιες που χρησιμοποιήθηκαν και με το σύστημα ΑΗΑ.



ΕΙΚΟΝΑ 3.12 –Δημιουργία Domain model στο Graph Author Tool

2. Στη συνέχεια και αφού το domain model είναι έτοιμο, δημιουργείται το μάθημα(course) όπου ορίζονται οι κανόνες προσαρμοστικότητας (adaptation rules).



ΕΙΚΟΝΑ 3.13 – Δημιουργία Course και κανόνων προσαρμογής στο Gat

3. Τέλος γίνεται έλεγχος ορθής λειτουργίας από την αρχική σελίδα index η οποία θα πρέπει να εμφανιστεί πρώτη, στη συνέχεια η σελίδα registration και μετά η πρώτη σελίδα του παραδείγματος, η σελίδα welcome.

3.3.1 Επιλογή του συστήματος

Ένα σημαντικό πλεονέκτημα του συστήματος GALE είναι η απλότητά στη σχεδίαση. Με το GAT (Gale Authoring Tool) γίνεται η σχεδίαση τόσο του domain model όσο και των κανόνων προσαρμογής(adaptation rules).

Στο AHA η σχεδίαση των κανόνων προσαρμογής πρέπει να ολοκληρώνεται σε ένα χρόνο συνεχόμενα και όσο είναι σε λειτουργία το Concept Editor. Σε περίπτωση που κλείσει το Concept Editor χάνεται ο κώδικας των κανόνων και την επόμενη φορά θα πρέπει να ξαναδημιουργηθεί από την αρχή. Αυτό το σημαντικό μειονέκτημα καθιστά το σύστημα δύσχρηστο, ιδίως σε εφαρμογές που έχουν πολλούς κανόνες προσαρμογής.

Το GALE είναι μια ακριβής επανασχεδίαση του συστήματος AHA! [14]. Στο AHA! μπορεί να εφαρμοστεί προσαρμοστικότητα στις σελίδες που ήταν αποθηκευμένες σε μετακινούμενους web servers. Η προσαρμοστικότητα, ωστόσο, έπρεπε να καθορίζεται στην τοποθεσία AHA! η

οποία εφάρμοσε την προσαρμοστικότητα. Στο GALE μπορούν να κατανεμηθούν όχι μόνο οι σελίδες-πηγές αλλά και ο καθορισμός της προσαρμοστικότητας. Αυτό είναι ένα σημαντικό βήμα καθώς σημαίνει ότι κάποιος μπορεί να εγκαταστήσει ένα μηχανισμό προσαρμοστικότητας «τοπικά» και ο μηχανισμός αυτός να κάνει χρήση προσαρμοστικότητας που έχει οριστεί σε κάποια άλλη τοποθεσία και να τον εφαρμόσει σε σελίδες-πηγές που βρίσκονται σε μια τρίτη τοποθεσία. Αυτό ονομάζεται Προσαρμοστικότητα ανοιχτού μοντέλου [27].

Για τους λόγους που αναφέρθηκαν παραπάνω το σύστημα GALE ήταν αυτό που επιλέχθηκε για την κατασκευή της εκπαιδευτικής εφαρμογής.

Κεφάλαιο 4

Υλοποίηση

Προσαρμοστικότητας

εκπαιδευτικής εφαρμογής

Για την ανάπτυξη της εκπαιδευτικής εφαρμογής χρησιμοποιήθηκε εκπαιδευτικό υλικό σύμφωνα με τη διδακτέα ύλη του μαθήματος ΠΕΣ511 – “Προηγμένες τεχνολογίες ιστού και προγραμματισμός” στο οποίο διδάσκονται βασικές αρχές της γλώσσας Java. Το εκπαιδευτικό υλικό οργανώθηκε σε κεφάλαια, τα οποία αντιστοιχούν σε κύριες έννοιες και υποκεφάλαια που αντιστοιχούν σε υποέννοιες. Το περίγραμμα του μαθήματος διαμορφώθηκε όπως φαίνεται στον παρακάτω πίνακα.

Κεφάλαιο	Υποκεφάλαια
Introducing Java	Variables
	Data
Control Structures	If, if/else nested control structures
	Switch
	Logical operators
Loops	For

	While
	Do/while
Methods	Modifiers
	Parameters
	Void method
	Scope & overloading
Arrays	Creating and Processing
	Arrays as parameters
	Enhanced for

ΠΙΝΑΚΑΣ 4.1 – Περίγραμμα μαθήματος

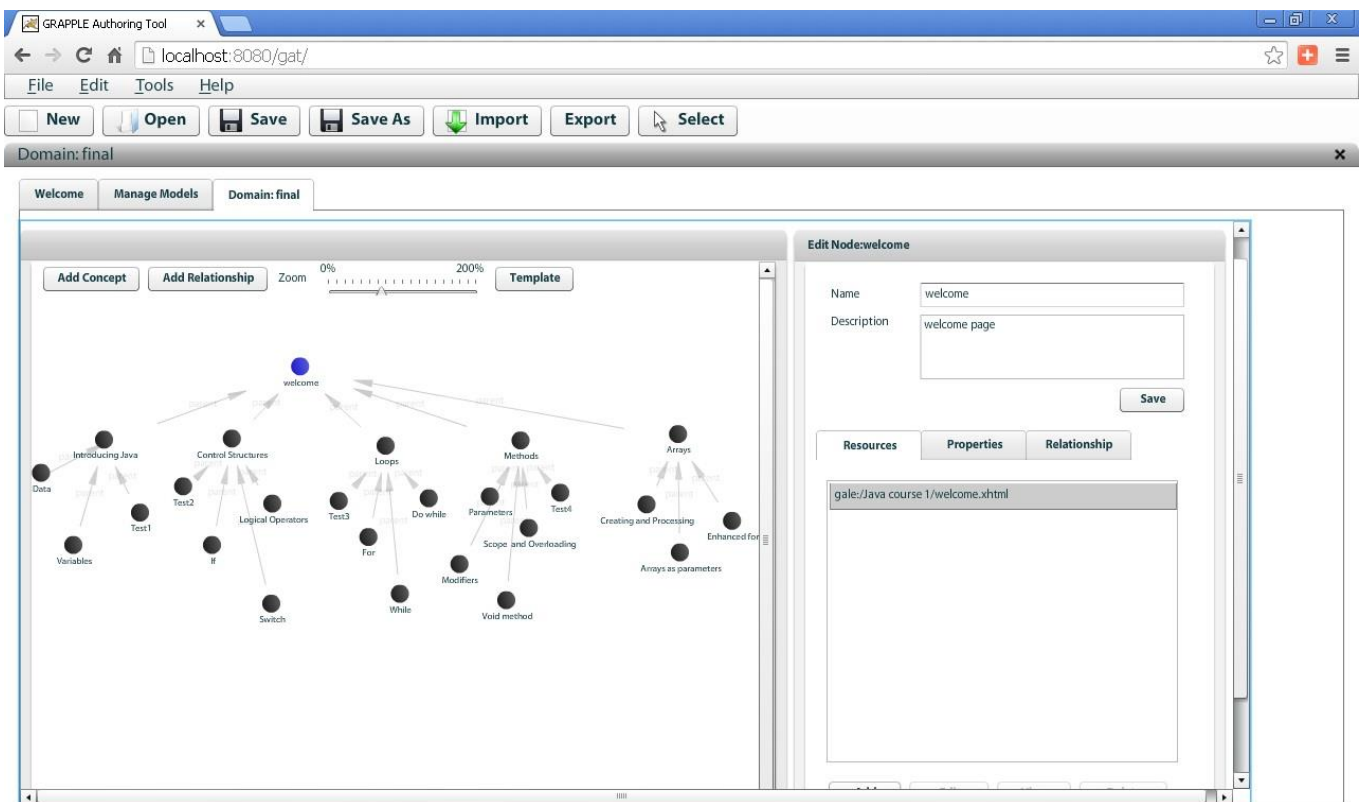
Αφού ολοκληρώθηκε η οργάνωση του κεφαλαίων, δημιουργήθηκε για καθένα από τα κεφάλαια και τα υποκεφάλαια μία ιστοσελίδα σε μορφή xhtml με το αντίστοιχο περιεχόμενο. Το περιεχόμενο των σελίδων πρέπει να είναι σε XHTML για να μπορούν να αποκρύπτονται προτάσεις υπό όρους. Το μοντέλο πεδίου (domain model) και οι κανόνες προσαρμογής (adaptation rules) που δημιουργούνται λίγο αργότερα μέσα από το Graph Author Tool αποθηκεύονται σε μορφή XML. Η επεξεργασία γίνεται μέσω του XML parser ώστε να μπορεί ο συντάκτης να εντοπίσει κάποιο σφάλμα και να το διορθώσει. Οι κεφαλίδες και τα υποσελίδα χρησιμοποιούνται μέσω ετικετών <tags>, όπως στην απλή HTML. Οι ειδικές ετικέτες που επιτρέπεται να χρησιμοποιηθούν είναι η <variable> και η <handler> για τα χαρακτηριστικά του μοντέλου χρήστη και την πρόσβαση σε ειδικές σελίδες (logout, κλπ).

Πριν από όλες τις σελίδες που αναφέρονται παραπάνω κατασκευάστηκε μια «πρώτη» σελίδα, η σελίδα welcome. Η σελίδα welcome είναι η αρχική σελίδα του μαθήματος και περιλαμβάνει μια εισαγωγή σχετικά με το τι πρόκειται να ακολουθήσει. Εξηγεί εν συντομία την τεχνολογία που χρησιμοποιείται καθώς και τον τρόπο λειτουργίας της. Επίσης, είναι η σελίδα αναφοράς προόδου για τον χρήστη καθώς μπορεί να την επισκέπτεται οποιαδήποτε στιγμή για να παρακολουθεί την πρόοδο που έχει σημειώσει σε κάθε έννοια του μαθήματος. Η εμφάνιση όλων των σελίδων καθορίζεται από δυο αρχεία XML:

- Το αρχείο **LayoutConfig.xml**, το οποίο αφορά την συνολική εμφάνιση του περιεχομένου και την συνολική μορφοποίηση,
- Και το αρχείο **ConceptTypeConfig.xml**, το οποίο αφορά την παρουσίαση των εννοιών και των υπερσυνδέσμων των εννοιών.

4.1 Σχεδίαση μαθήματος

Στη συνέχεια κατασκευάστηκε το μοντέλο πεδίου (Domain Model) για να περιγράψει τη δομή της εκπαιδευτικής εφαρμογής, με βάση τα κεφάλαια και την ιεραρχία που αναφέρθηκαν πιο πάνω. Για κάθε έννοια ορίστηκαν οι πηγές(Resources), οι ιδιότητες (Properties) και οι μεταξύ τους σχέσεις (Relationship).



ΕΙΚΟΝΑ 4.1 – ΜΟΝΤΕΛΟ ΠΕΔΙΟΥ(DOMAIN MODEL)

Κάθε έννοια συνδέθηκε με την αντίστοιχη ιστοσελίδα της μορφής xhtml μέσω του ορισμού των πηγών (Resources) όπως φαίνεται δεξιά στην εικόνα 16.



ΕΙΚΟΝΑ 4.2 - Resources (πηγές)

Οι ιδιότητες (properties), για όλες τις έννοιες εκτός από τις έννοιες welcome και Enhanced for, για λόγους που αναφέρονται παρακάτω, είναι οι εξής:

title= [ο τίτλος της σελίδας]

type=page [ο τύπος της έννοιας ορίζεται σε σελίδα]

order=[αριθμός π.χ. 1,2,3,κλπ. που δείχνει τη σειρά εμφάνισης της έννοιας στο βασικό μενού]

next=[ο τίτλος της επόμενης, σε σειρά εμφάνισης, έννοιας]

Η έννοια welcome είναι η πρώτη σε σειρά έννοια και για το λόγο αυτό περιλαμβάνει μία ακόμη ιδιότητα την

start=true [ορίζει ότι είναι η πρώτη σελίδα εμφάνισης],

ενώ η έννοια Enhanced for είναι η τελευταία σε σειρά έννοια και συνεπώς δεν περιλαμβάνει την ιδιότητα **next**.

Edit Node: Introducing Java

Name:

Description:

title	Introducing Java
type	page
order	1
next	Data

ΕΙΚΟΝΑ 4.3 - Properties (ιδιότητες)

Και τέλος, οι σχέσεις ορίστηκαν όπως φαίνεται στον παρακάτω πίνακα.

Έννοια πηγή	Είδος σχέσης	Έννοια προορισμός
welcome	is-parent-of	Introducing Java Control Structures Loops Methods Arrays
Introducing Java	is-parent-of	Data Variables Test1
Control Structures	is-parent-of	Test2 If Switch Logical Operators
Loops	is-parent-of	Test3 For

		While Do..while
Methods	is-parent-of	Parameters Modifiers Void method Scope and Overloading Test4
Arrays	is-parent-of	Creating and Processing Arrays as parameters Enhanced for

ΠΙΝΑΚΑΣ 4.2 - Σχέσεις

Εκτός από τις ήδη υπάρχουσες σχέσεις που προσφέρει το GAT υπάρχει δυνατότητα να δημιουργηθούν και νέες από το χρήστη όπως η σχέση parent που φαίνεται στην εικόνα 18.

Edit Node: Control Structures

Name: Control Structures

Description:

Save

Resources | Properties | Relationship

Control Structures	parent	welcome
if	parent	Control Structures
Switch	parent	Control Structures
Logical Operators	parent	Control Structures
Test2	parent	Control Structures

ΕΙΚΟΝΑ 4.4 – Relationships (σχέσεις)

Σκοπεύοντας στην παρουσίαση όσο το δυνατόν περισσότερων δυνατοτήτων του συστήματος, σχεδιάστηκαν διαφορετικές τεχνικές προσαρμοστικότητας για κάθε κεφάλαιο. Μερικές από τις τεχνικές που χρησιμοποιήθηκαν για την υλοποίηση της προσαρμοστικότητας είναι οι παρακάτω:

Knowledge – Είναι η ιδιότητα(attribute) της γνώσης για την εκάστοτε έννοια και αλλάζει κάθε φορά που ο χρήστης επισκέπτεται κάποια την αντίστοιχη σελίδα. Είναι ακέραιος αριθμός, του οποίου η τιμή παραμένει αποθηκευμένη και μετά την εκτέλεση του προγράμματος.

Access – Είναι η ιδιότητα της πρόσβασης σε μια σελίδα και ενεργοποιείται κάθε φορά που ο χρήστης επισκέπτεται τη σελίδα αυτή. Είναι τύπου Boolean έχει προσωρινή ισχύ για όσο εκτελείται το πρόγραμμα.

Visited – Είναι η ιδιότητα της επισκεψιμότητας και αριθμεί τις επισκέψεις του χρήστη στην εκάστοτε σελίδα. Είναι ακέραιος αριθμός, και η τιμή του παραμένει αποθηκευμένη και μετά την εκτέλεση του προγράμματος

Interest – Είναι η ιδιότητα του ενδιαφέροντος για μια σελίδα και απεικονίζει το βαθμό ενδιαφέροντος το χρήστη για την συγκεκριμένη σελίδα. Είναι ακέραιος αριθμός.

Multiple choice test – Είναι ερωτήσεις πολλαπλής επιλογής που μπορούν είτε να ενσωματωθούν σε υπάρχουσες σελίδες είτε να υπάρξουν ως αυτόνομες σελίδες τεστ, και έχουν τη μορφή αυτοαξιολόγησης.

4.2 Σχεδίαση προσαρμοστικότητας περιεχομένου

Στο υποκεφάλαιο αυτό ορίζονται οι προδιαγραφές προσαρμοστικότητας του συστήματος. Πρώτα και κύρια ορίστηκε ως κοινή προδιαγραφή η δυνατότητα του χρήστη να παρακολουθεί οποιαδήποτε έννοια με οποιαδήποτε σειρά χωρίς να υπάρχουν προαπαιτούμενες έννοιες. Στη συνέχεια, για κάθε υποέννοια προστέθηκαν ερωτήσεις πολλαπλής επιλογής τύπου τεστ είτε σε

ξεχωριστή σελίδα είτε ενσωματωμένες στο εκπαιδευτικό υλικό. Οι ερωτήσεις πολλαπλής επιλογής εκτός από το ρόλο της προσαρμοστικότητας, διαθέτουν και ρόλο αξιολογητή. Στο πάνω μέρος κάθε υποέννοιας-σελίδας εμφανίζεται η τρέχουσα τιμή της ιδιότητας knowledge. Παρακάτω αναλύεται η τεχνική που χρησιμοποιήθηκε για κάθε έννοια.

4.2.1 Κύρια έννοια Introducing Java και υποέννοιες

Η σελίδα Introducing Java έχει τρεις υποσελίδες-υποέννοιες, την Data (εικ. 20), την Variables (εικ. 21) και την Test1 (εικ. 22). Κάθε επίσκεψη του χρήστη στις υποσελίδες Data και Variables προσθέτει στην ιδιότητα knowledge των αντίστοιχων υποεννοιών 50 μονάδες. Η σελίδα Test1 περιλαμβάνει ερωτήσεις πολλαπλής επιλογής (multiple choice) για την αξιολόγηση της γνώσης των υποεννοιών. Η σελίδα Test1 εμφανίζεται μόνο όταν ο χρήστης έχει επισκεφθεί-μελετήσει τις υποέννοιες Data και Variable. Η σελίδα Variables εμφανίζεται μόνο όταν ο της επισκεφθεί την σελίδα Data. Η ορθή συμπλήρωση του τεστ θα δώσει άλλες 50 μονάδες στη ιδιότητα knowledge των δυο υποεννοιών (εικ. 23). Αν ο χρήστης απαντήσει λανθασμένα και επιτύχει σκορ μικρότερο από 100 (εικ. 24) τότε μηδενίζεται η ιδιότητα knowledge και των δύο υποεννοιών και προτείνεται στο χρήστη η επανάληψη ανάγνωσής των. Ο χρήστης μπορεί να απαντήσει το τεστ μόνο δυο φορές. Στην τρίτη προσπάθεια εμφανίζεται ανάλογο μήνυμα στην οθόνη (εικ. 25).

Introducing Java

localhost:8080/gale/concept/gale://gale.tue.nl/cam/final7/Introducing_Java

georgia markou (markou_georgia@hotmail.com) has read 7 pages and still has 18 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Introducing Java

Java is a high-level programming language originally developed by Sun Microsystems and released in 1995. Java runs on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX.

When we consider a Java program it can be defined as a collection of objects that communicate via invoking each other's methods. Let us now briefly look into what do class, object, methods and instance variables mean.

Object - Objects have states and behaviors. Example: A dog has states - color, name, breed as well as behaviors -wagging, barking, eating. An object is an instance of a class.

Class - A class can be defined as a template that describes the behaviors/states that object of its type support.

Methods - A method is basically a behavior. A class can contain many methods. It is in methods where the logics are written, data is manipulated and all the actions are executed.

Instance Variables - Each object has its unique set of instance variables. An object's state is created by the values assigned to these instance variables.

First program in Java

Here's a simple code that would print the words Hello World.

```
public class MyFirstJavaProgram {
    public static void main(String []args) {
        System.out.println("Hello World");
    }
}
```

You need to read Data in order for the Variables page to be revealed. Test1 page will be shown to you right after you finish reading both Data and Variables, and then you will be able to take the test. Be careful with your answers because you can only perform the test twice.

Next suggested concept to study: [Data](#)

ΕΙΚΟΝΑ 4.5 –Η σελίδα Introducing Java

Data

localhost:8080/gale/concept/gale://gale.tue.nl/cam/final7/Data

georgia markou (markou_georgia@hotmail.com) has read 8 pages and still has 17 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Data

Your Knowledge about Data is: 50.0

Data types are distinguished in two separate types. Primitive data types and reference data types. Reference data types are Arrays, Classes and Interfaces. We will not expand to reference data types at this point of tutorial.

The distinction for the primitive data types carries on as shown in the next image.

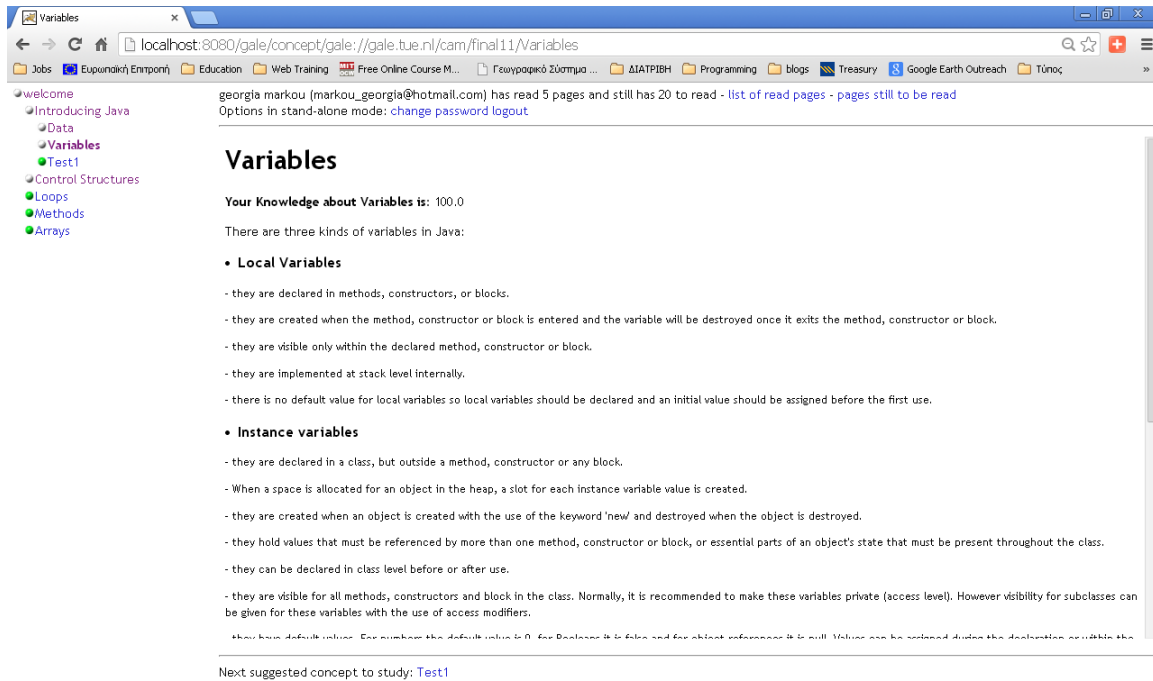
```

graph TD
    P[Primitive Data Type] --> N[Numbers]
    P --> C[Characters]
    P --> L[Logical]
    N --> I[Integers]
    N --> R[Real Numbers]
    I --> byte
    I --> short
    I --> int
    I --> long
    R --> float
    R --> double
    C --> char
    L --> boolean
    
```

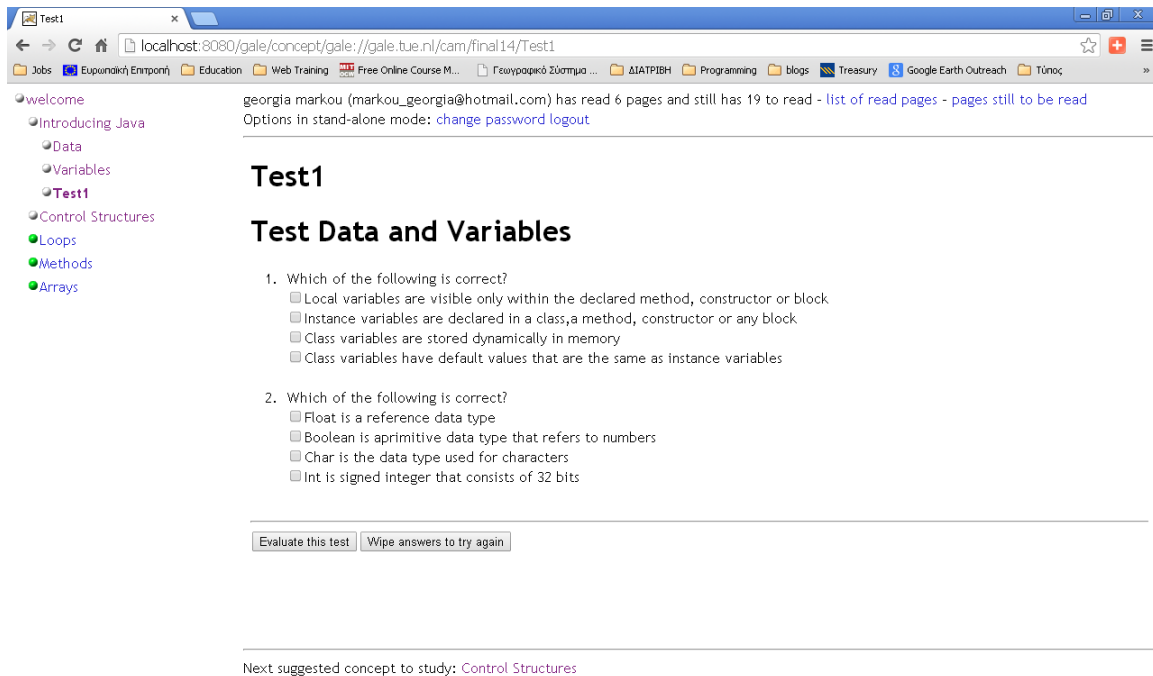
Name	Data	Range	Default Value	Size
byte	signed integer	[-128, 127]	0	8 bits
short	signed integer	[-32768, 32767]	0	16 bits
int	signed integer	[-2147483648, 2147483647]	0	32 bits
long	signed integer	[-9223372036854775808, 9223372036854775807]	0	64 bits
float	floating-point	MIN: ±1.4E-45 MAX: ±3.4028235E+38	0.0	32 bits
double	floating-point	MIN: ±4.9E-324 MAX: ±1.7976931348623157E+308	0.0	64 bits
char	Unicode	['\u0000', '\uFFFF']	'\u0000'	16 bits
boolean	logical value	{false, true}	false	≈ 1 bit

Next suggested concept to study: [Variables](#)

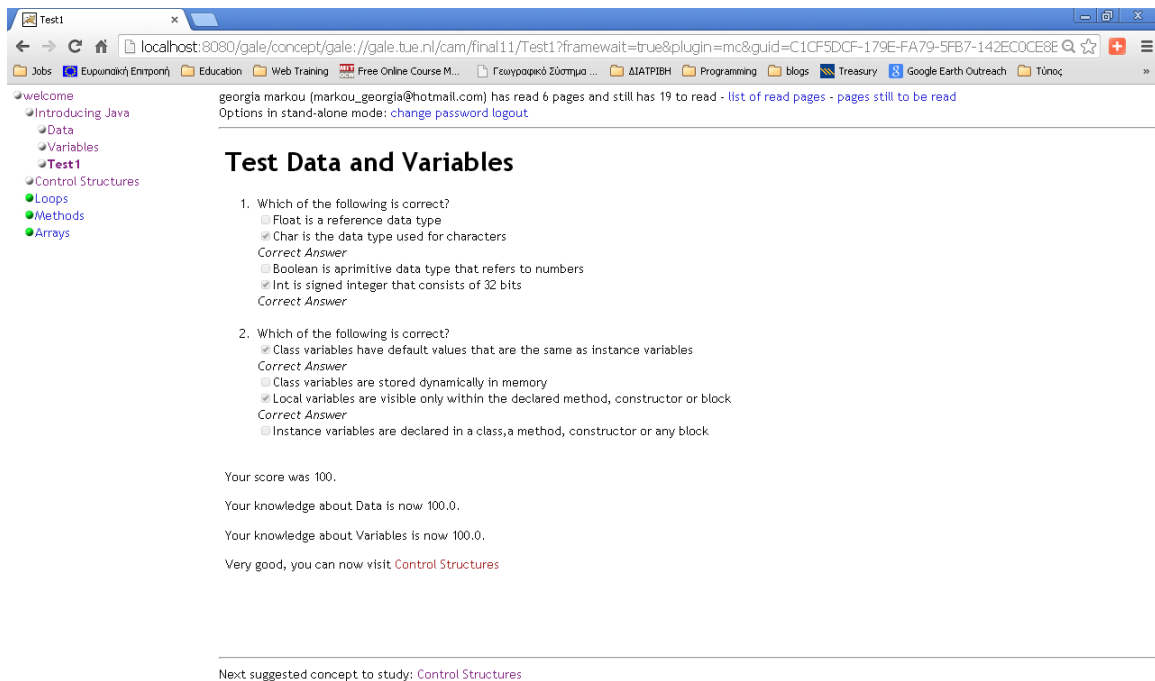
ΕΙΚΟΝΑ 4.6 - Η σελίδα data



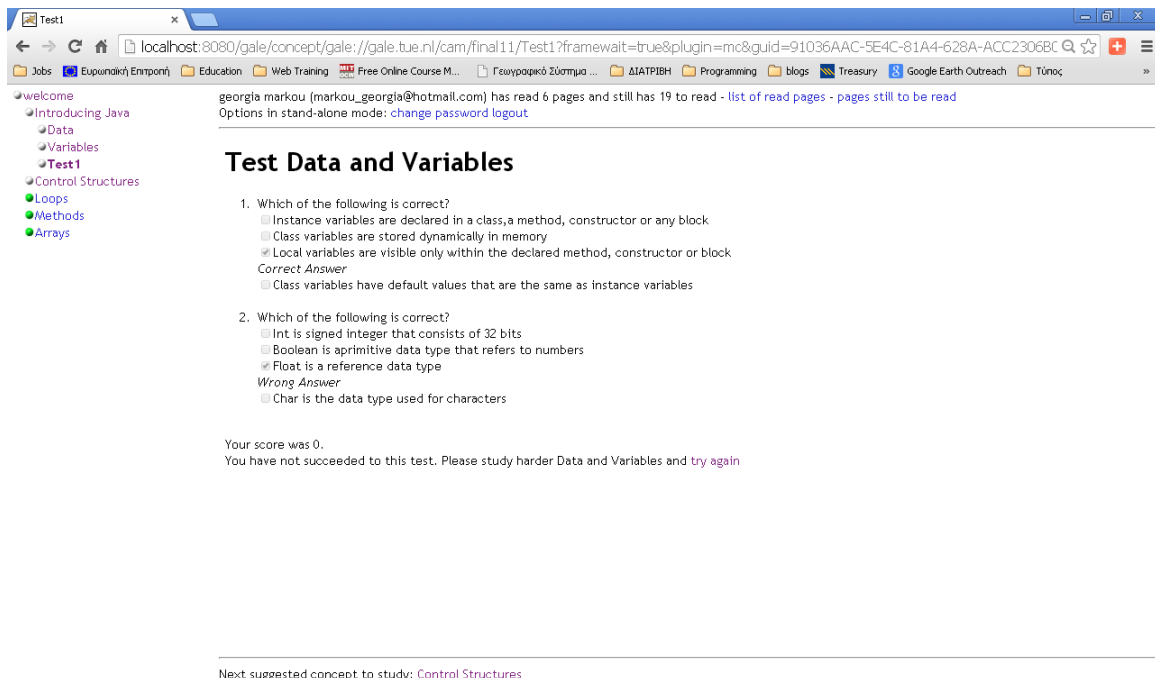
EΙΚΟΝΑ 4.72 - Η σελίδα Variables



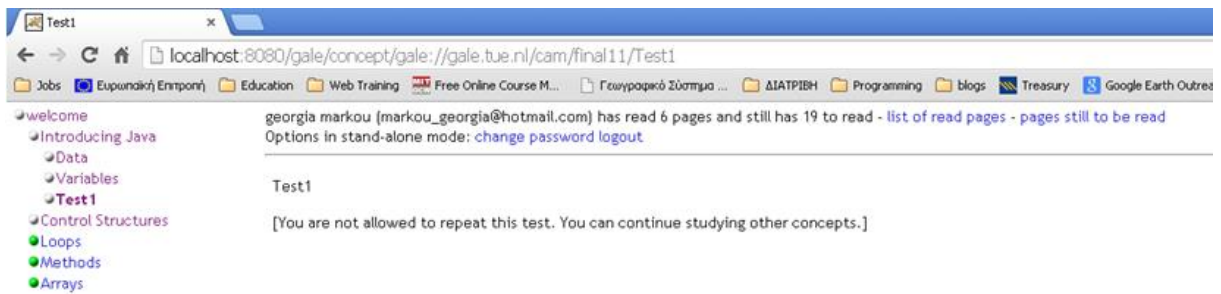
EΙΚΟΝΑ 4.8 - To Test1



ΕΙΚΟΝΑ 4.9 - Το Test1 μετά από επιτυχή ολοκλήρωση



ΕΙΚΟΝΑ 4.10 - Το Test1 μετά από ανεπιτυχή ολοκλήρωση



ΕΙΚΟΝΑ4.11 - Το Test1 την 3η προσπάθεια συμπλήρωσης

4.2.2 Κύρια έννοια Control Structures και υποέννοιες

Κι εδώ προστέθηκε το Test2 ως νέα υποσελίδα (εικ. 26). Ο χρήστης πρέπει να συμπληρώσει το τεστ για να εμφανιστούν οι τρεις υποσελίδες-υποέννοιες If (εικ. 29), Switch (εικ. 30), Logical Operators (εικ. 31), οι οποίες είναι αρχικά κρυφές. Μετά την συμπλήρωσή του, το σκορ, που έχει πετύχει ο χρήστης, ανατίθεται στην ιδιότητα knowledge κάθε μιας από τις τρεις υποέννοιες και ο χρήστης καθοδηγείται να αγνοήσει ή να μελετήσει τις τρεις υποέννοιες αν επιτύχει (εικ. 27) ή αποτύχει (εικ. 28) στο τεστ αντίστοιχα. Οι ερωτήσεις είναι δύο και συνεπώς υπάρχουν τρεις περιπτώσεις σκορ. Δυο πετυχημένες απαντήσεις δίνουν το 100%, μία πετυχημένη απάντηση δίνει το 50% και καμία πετυχημένη απάντηση δίνει 0%. Η επίσκεψη των υποσελίδων If, Switch, Logical Operators δίνει 50 επιπλέον μονάδες στη γνώση κάθε υποέννοιας. Ο χρήστης μπορεί να επαναλάβει το τεστ όσες φορές θέλει.

georgia markou (markou_georgia@hotmail.com) has read 6 pages and still has 19 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Control Structures

Control structures are blocks of code that dictate the flow of control. A simple example of a control structure is, if a then b else c. a control structure is a container for a series of function calls, instructions and statements. There are only three necessary control structures needed to write programs: sequence, selection, and iteration. Sequence refers to the line-by-line execution as used in your programs so far. The program enters the sequence, does each step, and exits the sequence.

Selection is the control structure that allows choice among different directions. Java provides different levels of selection:

1. One-way selection with an if structure
2. Two-way selection with an if-else structure
3. Multiple selection with a switch structure

Iteration refers to looping. Loops are discussed in the next chapter.

Selection

if structure

if/else structure

switch structure

Next suggested concept to study: [Test2](#)

ΕΙΚΟΝΑ 4.12 –Η σελίδα Control Structures

georgia markou (markou_georgia@hotmail.com) has read 7 pages and still has 18 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

```

System.out.println("You are but a wee child!");
}
else if (age < 19)
{
System.out.println("You are no longer a child, but a budding teenager.");
}
else
{
if (age < 65)
{
System.out.println("You are an adult!");
}
else
{
System.out.println("You are now a senior, enjoy the good life friends!");
}
}
System.out.println("Also, since you are over the age of 19, you deserve a drink!");
}

```

False
 True
That is correct. Well done!

Your score was 100.
If you have not already done it, refresh your browser now.

Your knowledge about If statement is now 100.0.

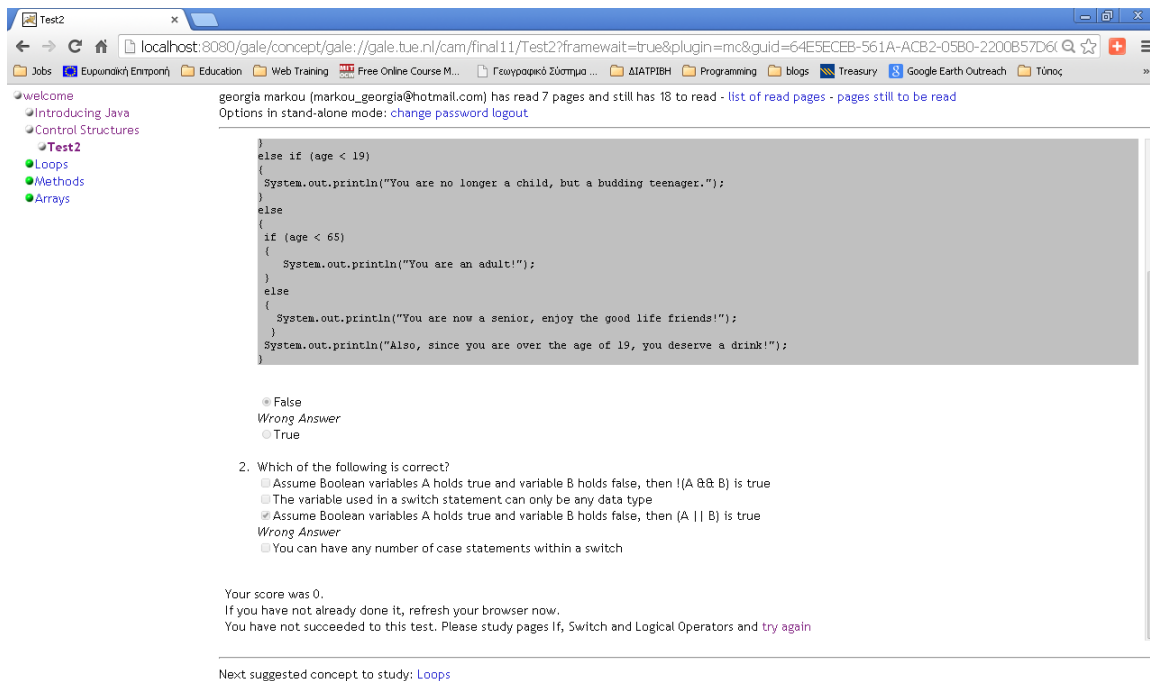
Your knowledge about Switch statement is now 100.0.

Your knowledge about Logical Operators is now 100.0.

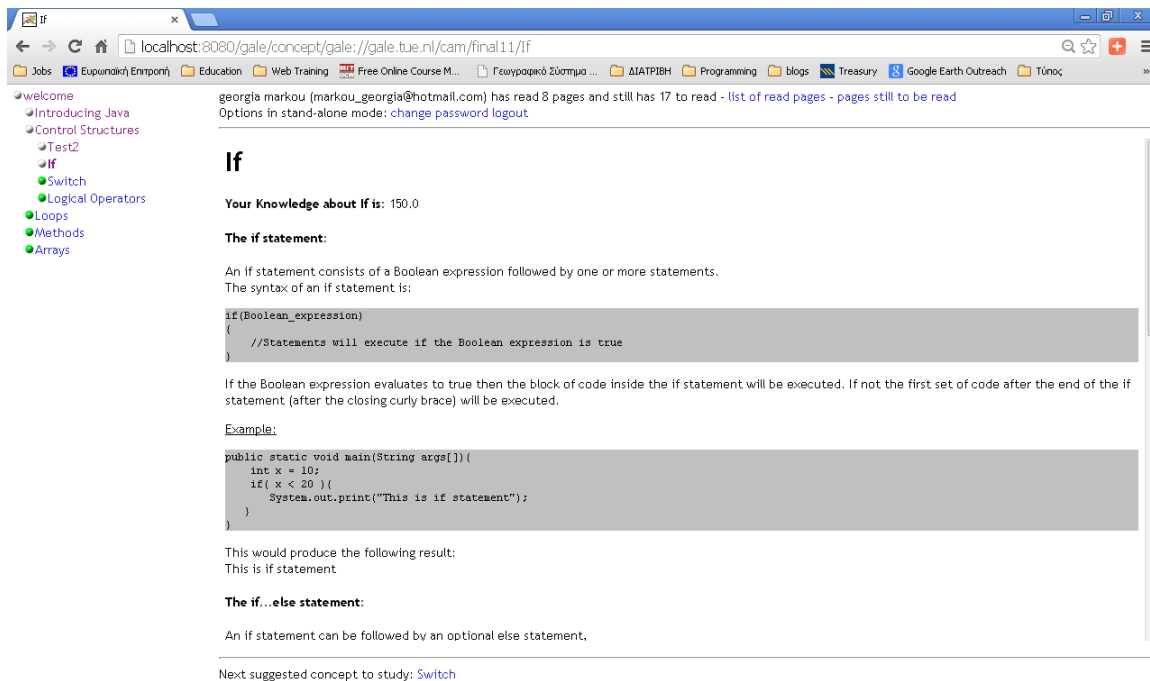
Very good, you can ignore pages If, Switch and Logical Operators and visit [Loops](#)

Next suggested concept to study: [If](#)

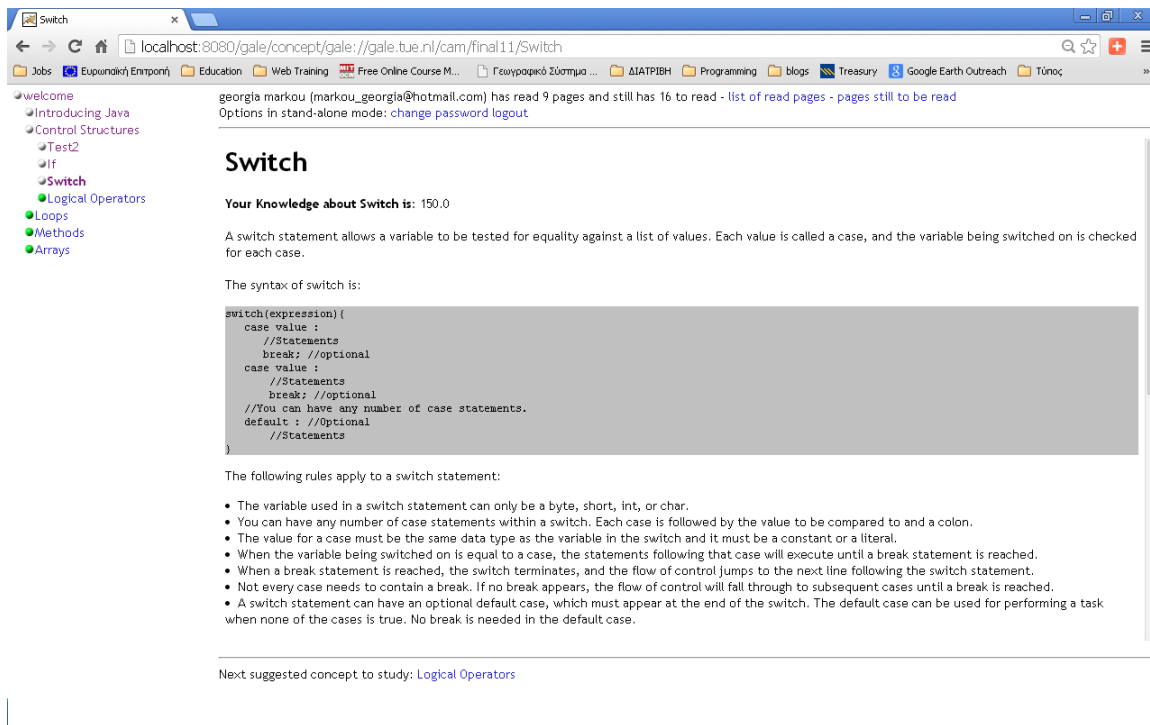
ΕΙΚΟΝΑ 4.13 - Το Test2 μετά την επιτυχή ολοκλήρωσή του



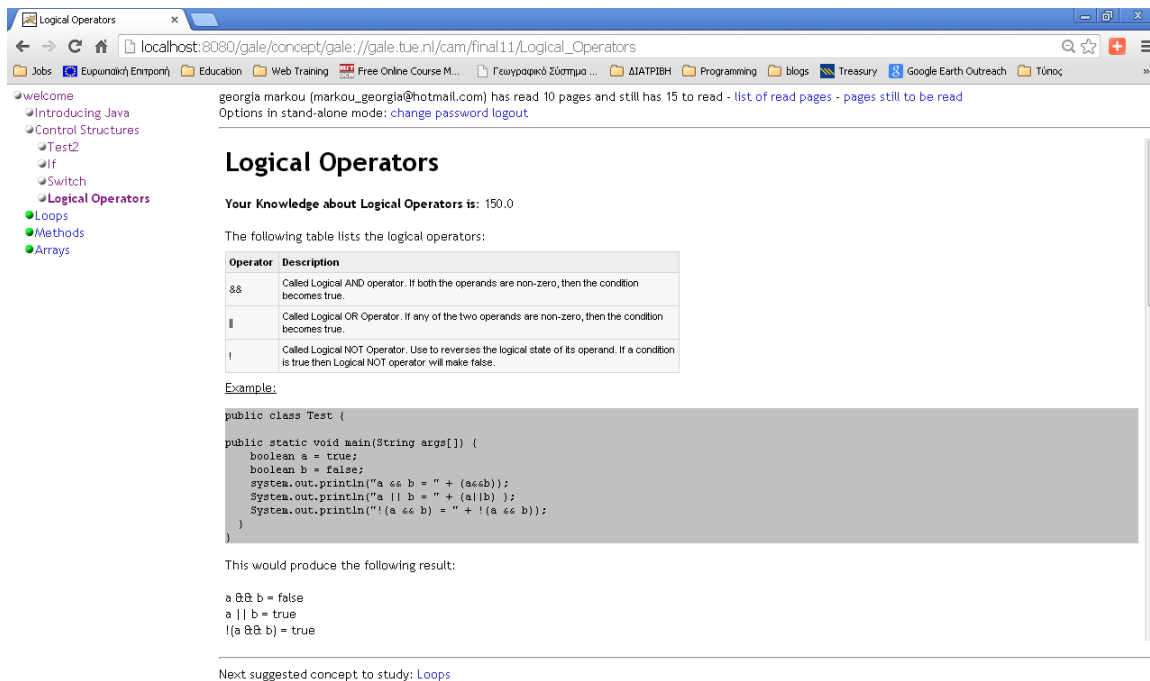
ΕΙΚΟΝΑ 4.14 - Το Test2 μετά την ανεπιτυχή ολοκλήρωσή του



ΕΙΚΟΝΑ 4.15 - Η σελίδα If



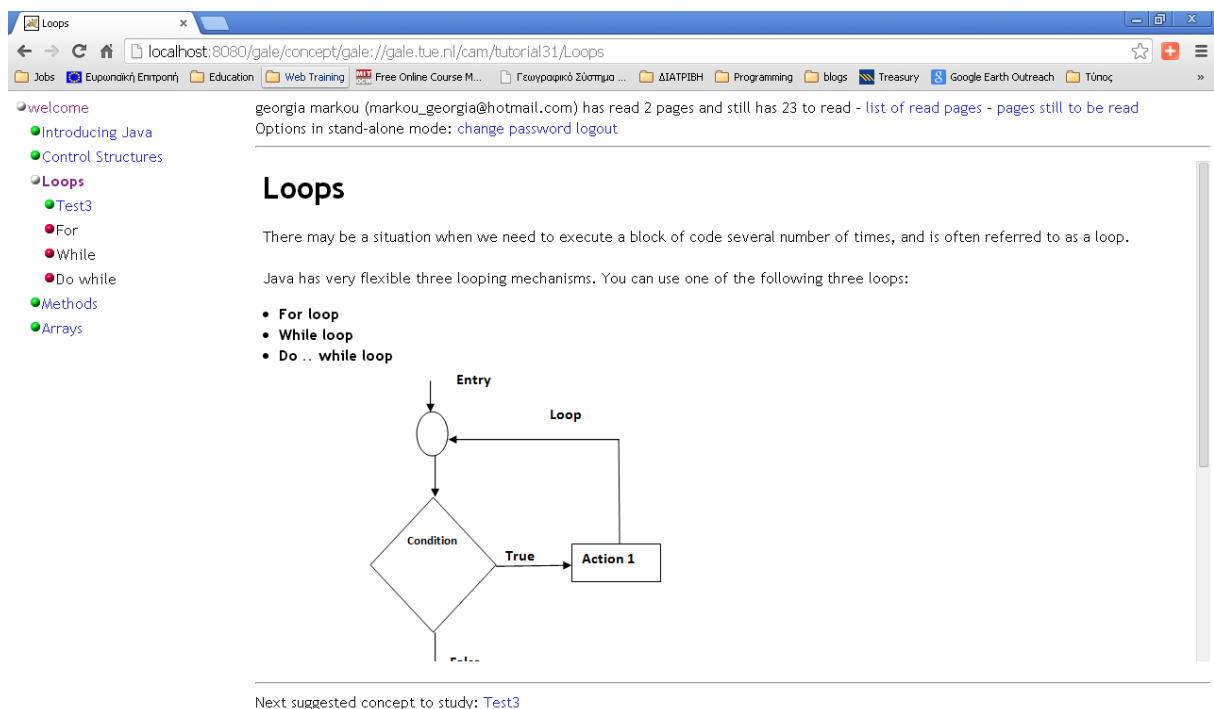
ΕΙΚΟΝΑ 4.16 - Η σελίδα Switch



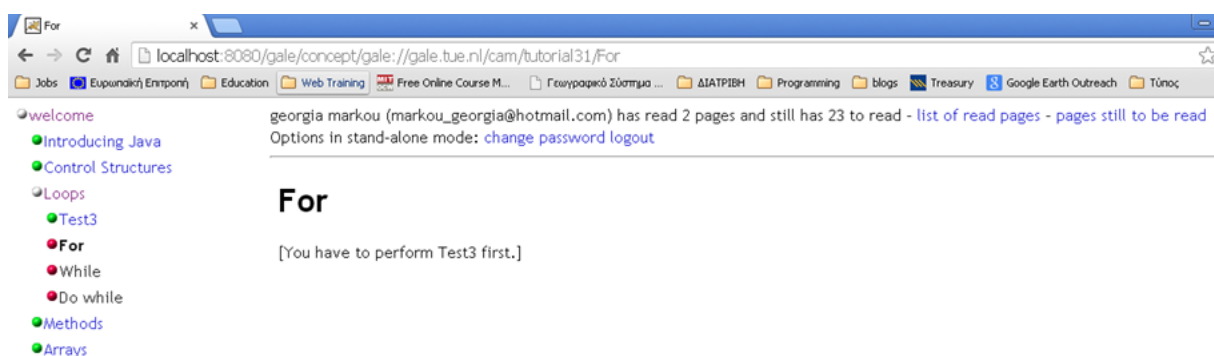
ΕΙΚΟΝΑ 4.17 - Η σελίδα Logical Operators

4.2.3 Κύρια έννοια Loops και υποέννοιες

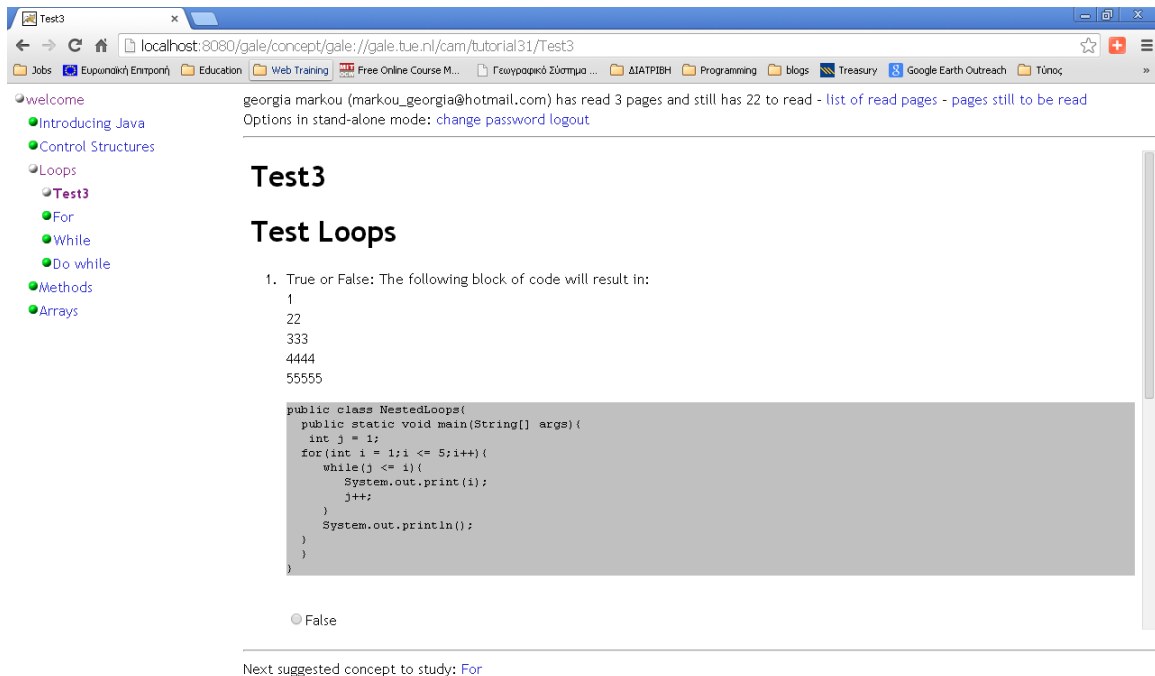
Ο χρήστης πρέπει και πάλι να συμπληρώσει πρώτα το Test3 (εικ. 34), η οποία είναι η πρώτη κατά σειρά εμφάνιση υποσελίδα. Μόνο αν συμπληρωθεί το τεστ μπορεί ο χρήστης να μελετήσει τις υποσελίδες- υποέννοιες For, While, Do while (εικ. 33). Αν το τεστ συμπληρωθεί σωστά τότε οι υποσελίδες δεν εμφανίζουν παραδείγματα επεξήγησης των υποεννοιών (εικ. 36). Διαφορετικά οι σελίδες περιλαμβάνουν και παραδείγματα επεξήγησης (εικ. 35). Ο χρήστης μπορεί να συμπληρώσει μόνο δυο φορές το τεστ. Σε επόμενη προσπάθεια εμφανίζεται ανάλογο μήνυμα στην οθόνη (εικ. 38). Η ορθή συμπλήρωση του τεστ δίνει στο χρήστη 100 μονάδες γνώσης, ενώ κάθε επίσκεψη προσθέτει 50 μονάδες.



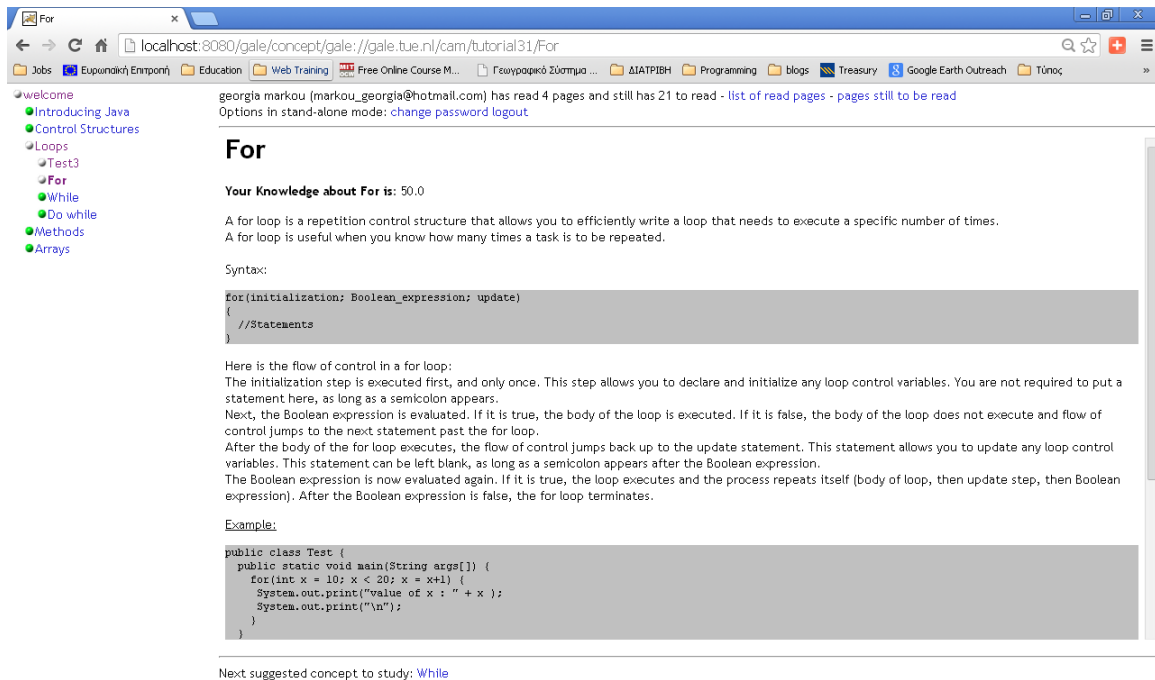
ΕΙΚΟΝΑ 4.18 – Η σελίδα Loops



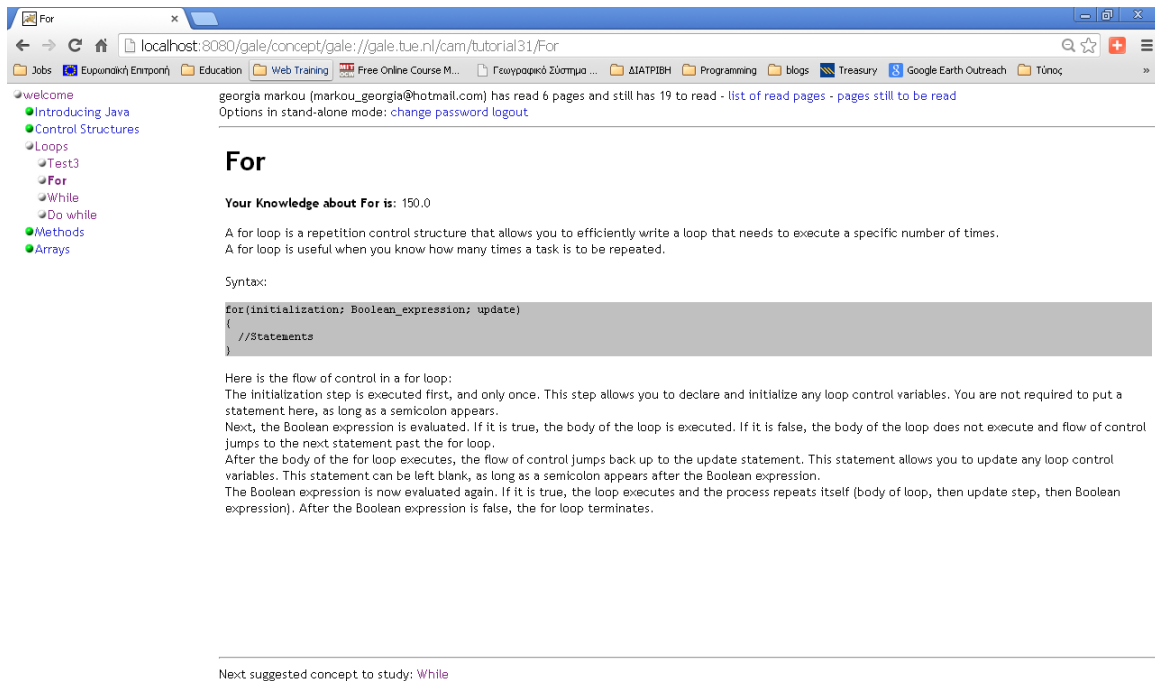
ΕΙΚΟΝΑ 4.193 - Η σελίδα For πριν την λήψη του Test3



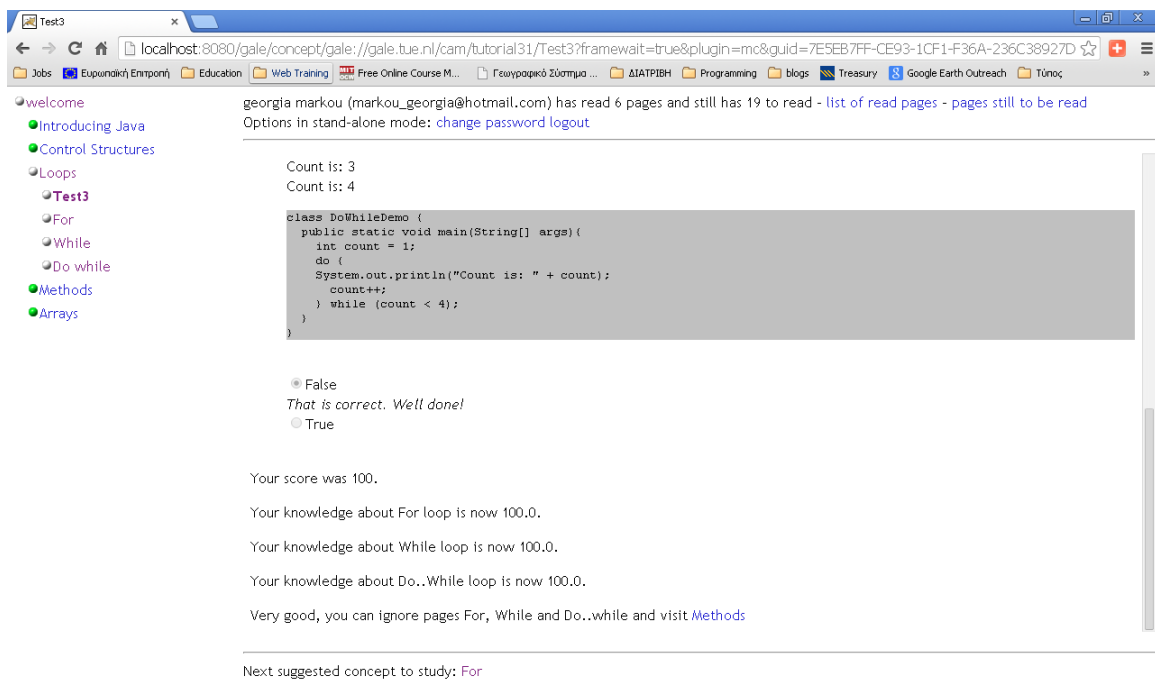
EΙΚΟΝΑ 4.20 - To Test3



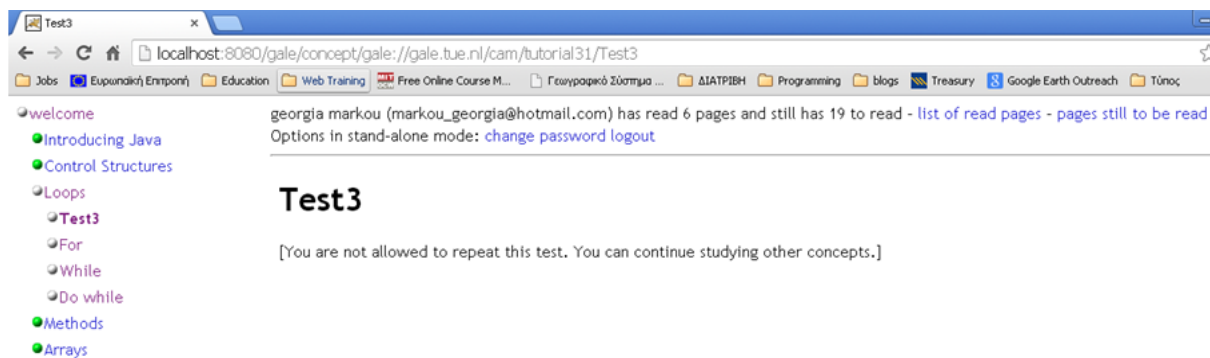
Εικόνα 4.21 - Η σελίδα For μετά την ανεπιτυχή ολοκλήρωση του τεστ



ΕΙΚΟΝΑ 4.22 - Η σελίδα For μετά την επιτυχή ολοκλήρωση του τεστ



ΕΙΚΟΝΑ 4.23 - Το Test3 μετά την επιτυχή ολοκλήρωση



ΕΙΚΟΝΑ 4.24 - Το Test3 μετά από δυο επαναλήψεις

4.2.4 Κύρια έννοια Methods και υποέννοιες

Για την κύρια έννοια Methods (εικ. 39) ο χρήστης πρέπει να μελετήσει τις υποέννοιες Modifiers (εικ. 40), Parameters (εικ. 41), Void method (εικ. 42), Scope and Overloading (εικ. 43) και μετά να συμπληρώσει το Test4 (εικ. 44). Για να μπορέσει να κατανοήσει την έννοια Scope and Overloading θα πρέπει πρώτα να μελετήσει την έννοια Parameters. Μόνο όταν αναγνώσει όλες τις υποσελίδες θα εμφανιστούν οι ερωτήσεις του τεστ. Διαφορετικά εμφανίζεται ανάλογο μήνυμα στην οθόνη (εικ. 45). Ο χρήστης, στη συνέχεια, μπορεί να επαναλάβει το τεστ όσες φορές θέλει. Όταν ο χρήστης έχει ολοκληρώσει το τεστ, ανάλογα με το αποτέλεσμα, προσαρμόζεται η μορφοποίηση των υποσελίδων Modifiers (εικ. 46), Parameters, Void method, Scope and Overloading σε επόμενη επίσκεψη. Δηλαδή, αν ο χρήστης απαντήσει λανθασμένα τότε η σελίδα εμφανίζεται με διαφορετικό χρώμα φόντου. Μπορεί επίσης, να υπάρχουν προτάσεις με έντονη γραφή, υπογράμμιση ή επισήμανση στο κείμενο. Η επίσκεψη του χρήστη στις υποσελίδες Modifiers, Parameters, Void method, Scope and Overloading προσθέτει 50 μονάδες γνώσης σε κάθε έννοια που επισκέπτεται και η επιτυχής ολοκλήρωση του τεστ μπορεί να δώσει απευθείας τις 100 για κάθε έννοια.

georgia markou (markou_georgia@hotmail.com) has read 2 pages and still has 23 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Methods

A Java method is a collection of statements that are grouped together to perform an operation. When you call the `System.out.println` method, for example, the system actually executes several statements in order to display a message on the console. Now you will learn how to create your own methods with or without return values, invoke a method with or without parameters, overload methods using the same names, and apply method abstraction in the program design.

Creating a Method:

In general, a method has the following syntax:

```
modifier returnType methodName(list of parameters) {
    // Method body;
}
```

A method definition consists of a method header and a method body. Here are all the parts of a method:

- Modifiers
- Return Type
- Method Name
- Parameters
- Method Body

Define a method
modifier return value type method name formal parameters

Next suggested concept to study: [Modifiers](#)

EIKONA 4.25 – Methods

georgia markou (markou_georgia@hotmail.com) has read 9 pages and still has 16 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Modifiers

Your Knowledge about Modifiers is: 50.0

Access Control Modifiers:

Java provides a number of access modifiers to set access levels for classes, variables, methods and constructors. The four access levels are:

- Visible to the package, the default. No modifiers are needed.
- Visible to the class only (private).
- Visible to the world (public).
- Visible to the package and all subclasses (protected).

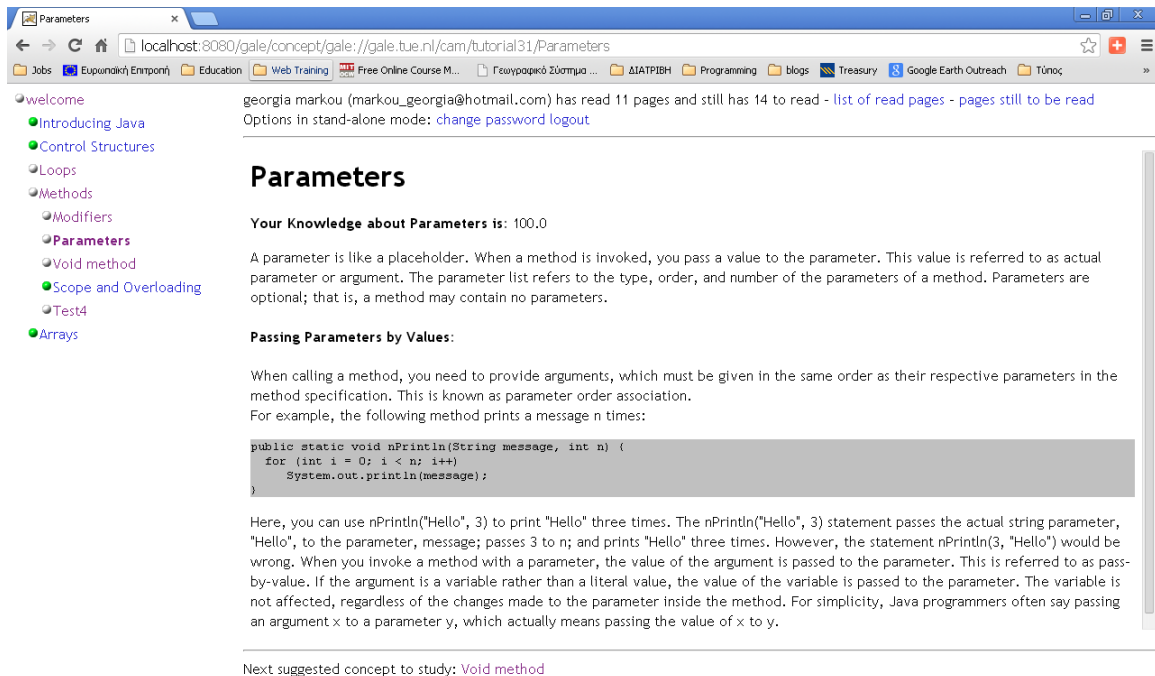
Non Access Modifiers:

Java provides a number of non-access modifiers to achieve many other functionality.

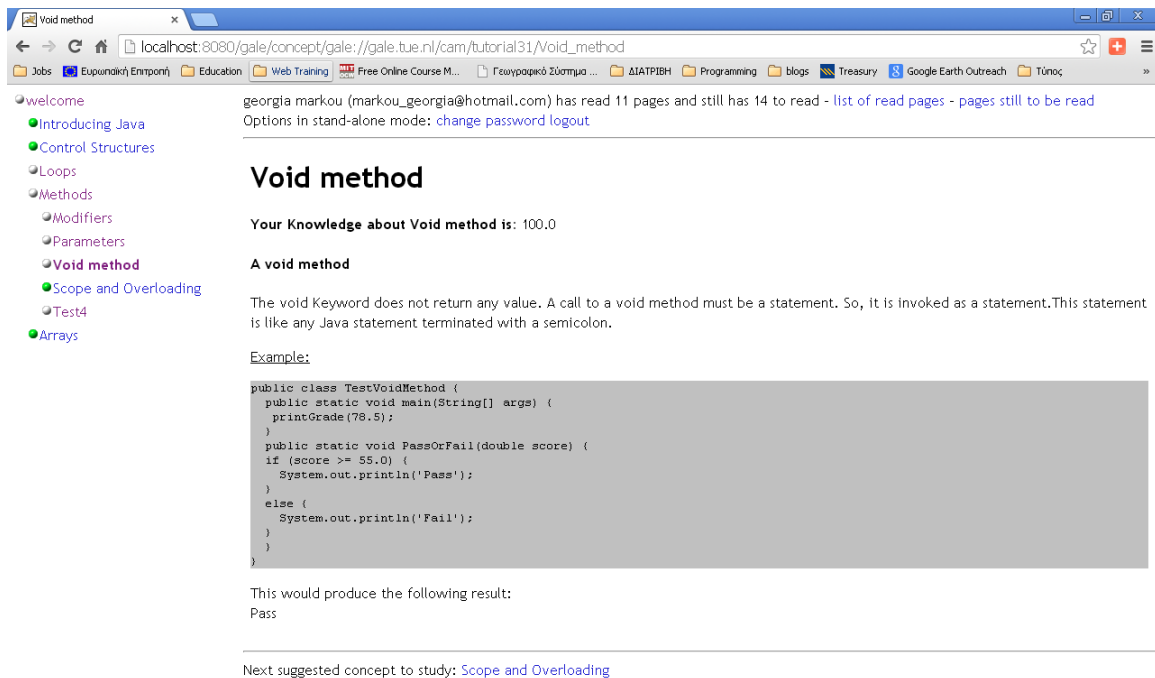
- The static modifier for creating class methods and variables.
- The final modifier for finalizing the implementations of classes, methods, and variables .
- The abstract modifier for creating abstract classes and methods .
- The synchronized and volatile modifiers, which are used for threads .

Next suggested concept to study: [Parameters](#)

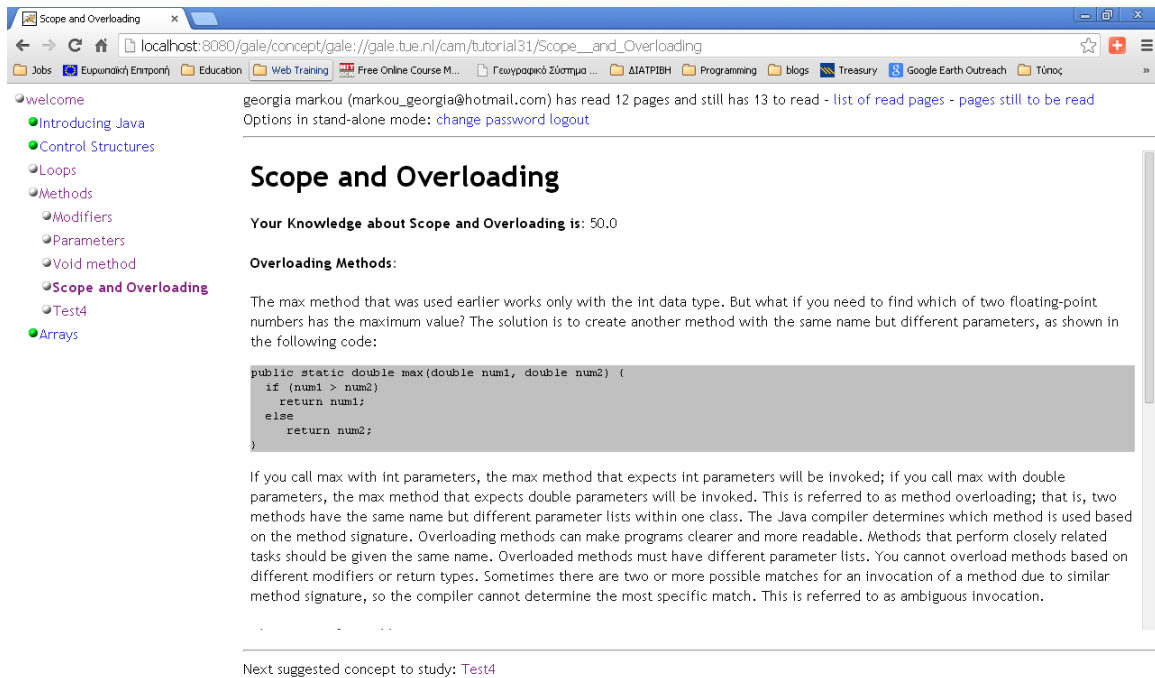
EIKONA 4.26 -Η σελίδα Modifiers



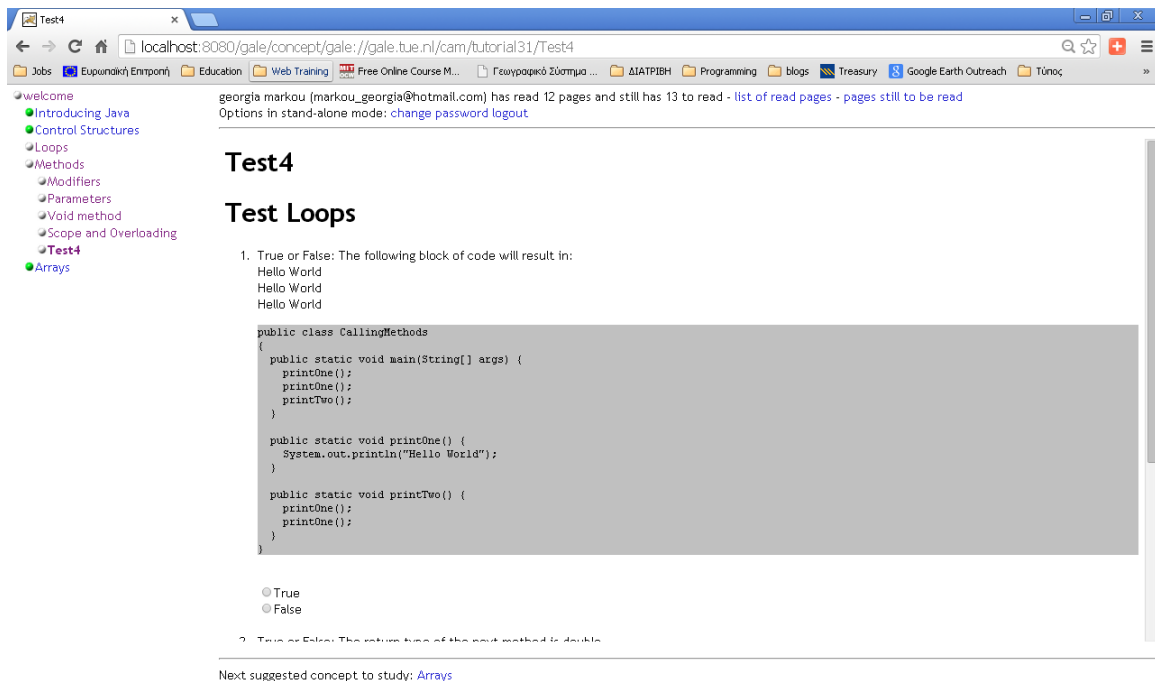
ΕΙΚΟΝΑ 4.27 - Η σελίδα Parameters



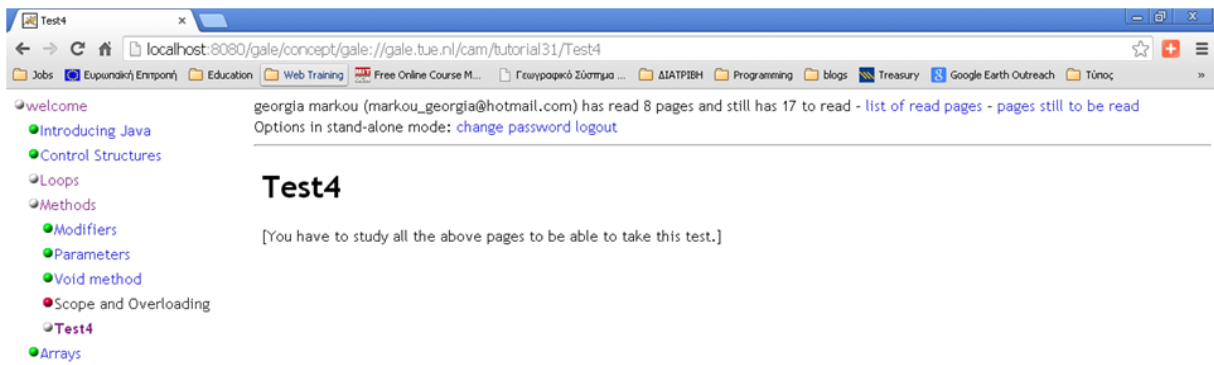
ΕΙΚΟΝΑ 4.28 - Η σελίδα Void Method



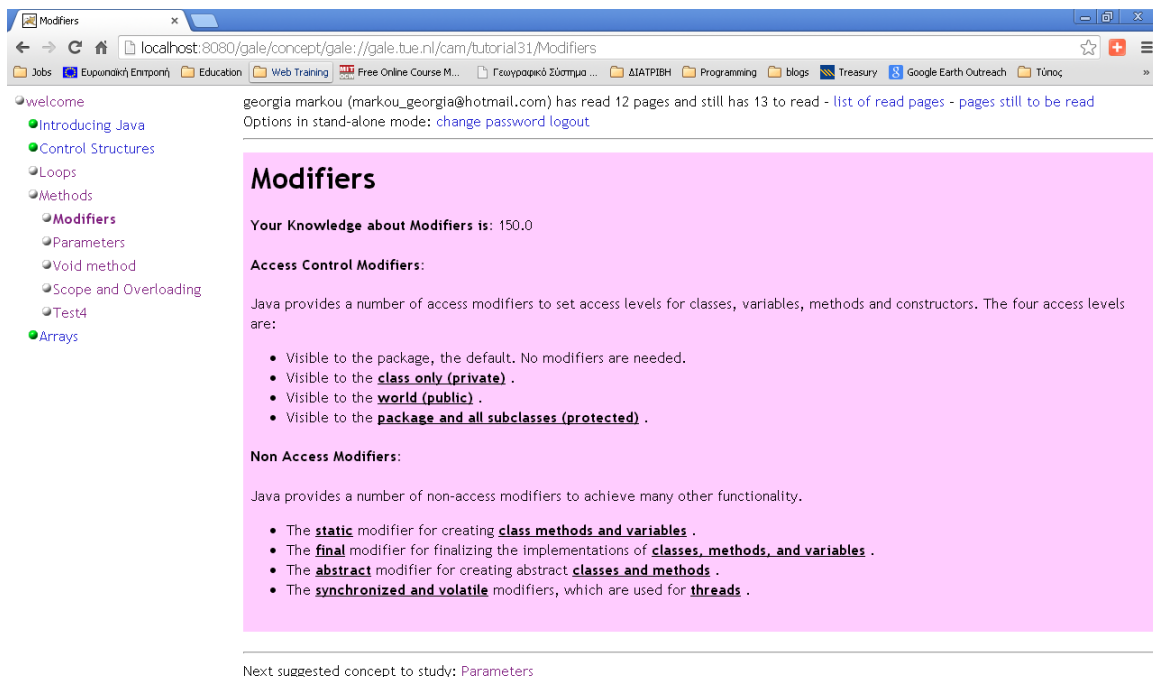
EΙΚΟΝΑ 4.29 - Η σελίδα Scope and Overloading



EΙΚΟΝΑ 4.30 - To Test4



ΕΙΚΟΝΑ 4.31 - Το Test4 πριν την ανάγνωση των υποσελίδων Modifiers, Parameters, Void method, Scope and Overloading



ΕΙΚΟΝΑ 4.32 - Η σελίδα Modifiers μετά την ανεπιτυχη ολοκλήρωση του τεστ

4.2.5 Κύρια έννοια Arrays και υποέννοιες

Η έννοια Arrays (εικ. 47) περιλαμβάνει τρεις υποσελίδες-έννοιες. Την Creating and Processing (εικ. 48), την Arrays as parameters (εικ. 49), και την Enhanced for (εικ. 50). Όταν ο χρήστης επισκεφθεί τις υποσελίδες, στο κάτω μέρος καθεμιάς θα βρει ένα μίνι τεστ αξιολόγησης της συγκεκριμένης υποέννοιας (εικ. 51). Το τεστ αυτό αλλάζει σε κάθε επόμενη επίσκεψη εφόσον ο χρήστης συμπληρώσει το τρέχον τεστ. Οι μονάδες γνώσης ορίζονται σε 50 με την επίσκεψη του χρήστη και φτάνουν τις 100 όταν απαντήσει σωστά στις ερωτήσεις. Επίσης, όταν ο χρήστης

απαντήσει σωστά τότε στις επόμενες επισκέψεις, στη θέση του τεστ θα εμφανίζεται ανάλογο μήνυμα.

georgia markou (markou_georgia@hotmail.com) has read 2 pages and still has 23 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Arrays

Java provides a data structure, the array, which stores a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type. Instead of declaring individual variables, such as number0, number1, ..., and number99, you declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables.

This tutorial introduces how to declare array variables, create arrays, and process arrays using indexed variables.

Declaring Array Variables:
To use an array in a program, you must declare a variable to reference the array, and you must specify the type of array the variable can reference. Here is the syntax for declaring an array variable:

```
dataType[] arrayRefVar; // preferred way.
```

or

```
dataType arrayRefVar[]; // works but not preferred way.
```


Note: The style dataType[] arrayRefVar is preferred. The style dataType arrayRefVar[] comes from the C/C++ language and was adopted in Java to accommodate C/C++ programmers.

Example:
The following code snippets are examples of this syntax:

```
double[] myList; // preferred way.
```

or

```
double myList[]; // works but not preferred way.
```

 Study concepts Creating and Processing, Arrays as parameters, Enhanced for and at the bottom of every page you will find a mini test to evaluate your knowledge about the specific concept. This test will be appearing until you answer correctly.

Next suggested concept to study: [Creating and Processing](#)

EIKONA 4.33 – Arrays

georgia markou (markou_georgia@hotmail.com) has read 3 pages and still has 22 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Creating and Processing

Your Knowledge about Creating and Processing is: 50.0

Creating Arrays:

You can create an array by using the new operator with the following syntax:

```
arrayRefVar = new dataType[arraySize];
```

The above statement does two things:
It creates an array using new dataType[arraySize];
It assigns the reference of the newly created array to the variable arrayRefVar.

Declaring an array variable, creating an array, and assigning the reference of the array to the variable can be combined in one statement, as shown below:

```
dataType[] arrayRefVar = new dataType[arraySize];
```

Alternatively you can create arrays as follows:

```
dataType[] arrayRefVar = {value0, value1, ..., valuek};
```

The array elements are accessed through the index. Array indices are 0-based; that is, they start from 0 to arrayRefVar.length-1.

Example:
Following statement declares an array variable, myList, creates an array of 10 elements of double type and assigns its reference to myList:

```
double[] myList = new double[10];
```

Following picture represents array myList. Here, myList holds ten double values and the indices are from 0 to 9.

```
myList [reference] → [myList[0] | 5.6 | ... ]
```

Next suggested concept to study: [Arrays as parameters](#)

EIKONA 4.34 - Η σελίδα Creating and Processing

georgia markou (markou_georgia@hotmail.com) has read 4 pages and still has 21 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Arrays as parameters

Your Knowledge about Arrays as parameters is: 50.0

Just as you can pass primitive type values to methods, you can also pass arrays to methods. For example, the following method displays the elements in an int array:

```
public static void printArray(int[] array) {
    for (int i = 0; i < array.length; i++) {
        System.out.print(array[i] + " ");
    }
}
```

You can invoke it by passing an array. For example, the following statement invokes the printArray method to display 3, 1, 2, 6, 4, and 2:

```
printArray(new int[]{3, 1, 2, 6, 4, 2});
```

Returning an Array from a Method:
A method may also return an array. For example, the method shown below returns an array that is the reversal of another array:

```
public static int[] reverse(int[] list) {
    int[] result = new int[list.length];
    for (int i = 0, j = result.length - 1; i < list.length; i++, j--) {
        result[j] = list[i];
    }
    return result;
}
```

Test Arrays as parameters

1. True or False: The following block of code will result in:
element 0=?

Next suggested concept to study: [Enhanced for](#)

ΕΙΚΟΝΑ 4.35 - Η σελίδα Arrays as parameters

georgia markou (markou_georgia@hotmail.com) has read 5 pages and still has 20 to read - [list of read pages](#) - [pages still to be read](#)
Options in stand-alone mode: [change password](#) [logout](#)

Enhanced for

Your Knowledge about Enhanced for is: 50.0

As of Java 5, the enhanced for loop was introduced. This is mainly used for Arrays.
Syntax:

```
for(declaration : expression)
{
    //Statements
}
```

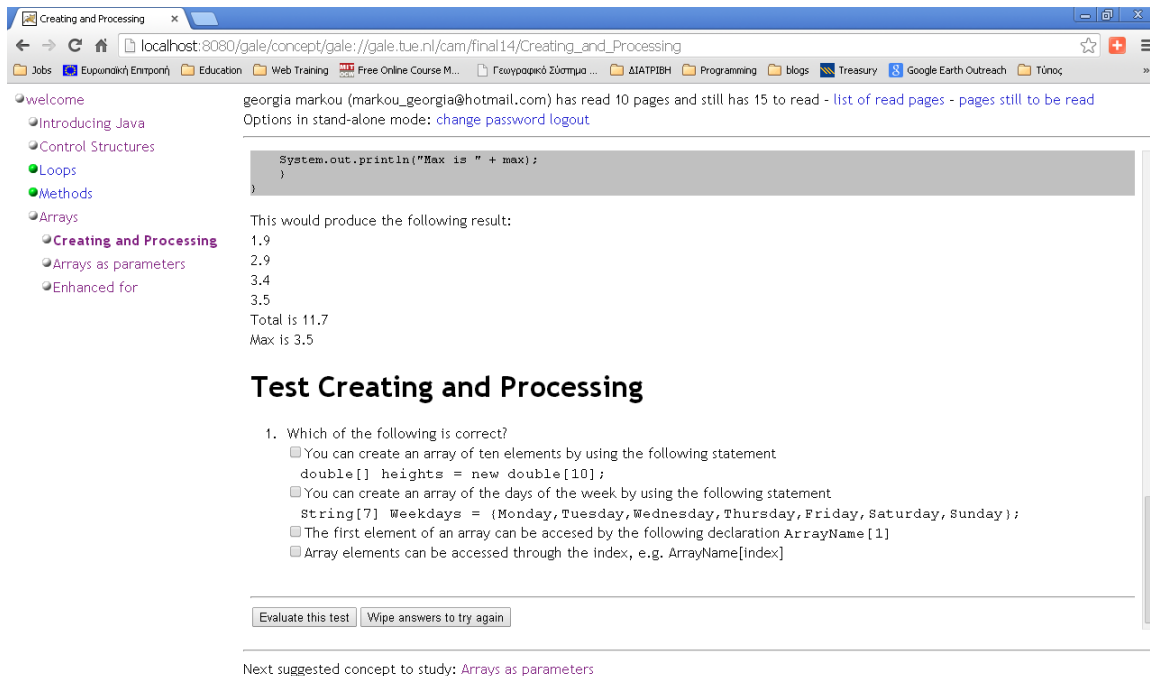
Declaration: The newly declared block variable, which is of a type compatible with the elements of the array you are accessing. The variable will be available within the for block and its value would be the same as the current array element.
Expression: This evaluates to the array you need to loop through. The expression can be an array variable or method call that returns an array.

Example:

```
public class Test {
    public static void main(String args[]){
        int [] numbers = {10, 20, 30, 40, 50};
        for(int x : numbers ){
            System.out.print( x );
            System.out.print(",");
        }
        System.out.print("\n");
        String [] names = {"James", "Larry", "Tom", "Lacy"};
        for( String name : names ) {
            System.out.print( name );
            System.out.print(",");
        }
    }
}
```

Next suggested concept to study: [welcome](#)

ΕΙΚΟΝΑ 4.36 - Η σελίδα Enhanced for



ΕΙΚΟΝΑ 4.37 - Το επόμενο test της σελίδας Creating and Processing μετά την συμπλήρωση του 1ου

4.2.6 Ετικέτες GALE

Για το προσαρμοζόμενο κείμενο του περιεχόμενου το GALE διαθέτει κάποια Modules καθένα από τα οποία αντιστοιχίζεται σε ένα tag. Οπότε όταν ο επεξεργαστής XML βλέπει ένα tag το μεταφράζει σε module και εφαρμόζει την προσαρμοστικότητα. Τα modules που χρησιμοποιεί το GALE είναι :

- IfModule <if>- χρησιμοποιείται για την χρήση της δομής if
- ObjectModule<object> - χρησιμοποιείται για την υπο όρους συμπερίληψη αρχείων-πηγών
- AdaptLinkModule <a>- χρησιμοποιείται για την προσαρμοζόμενη χρήση των υπερσυνδέσμων
- VariableModule <variable>- χρησιμοποιείται για την εισαγωγή τιμής ιδιότητας του μοντέλου χρήστη

- AttrVariableModule <attr-variable>- χρησιμοποιείται για να εισάγει την τιμή σε ιδιότητα του μοντέλου χρήστη
- ForModule <for>- Χρησιμοποιείται για τη χρήση της δομής επανάληψης for
- CountModule <count>- χρησιμοποιείται για την καταμέτρηση των αντικειμένων
- MCModule <test>- χρησιμοποιείται για τα multiple choice tests
- PluginModule -<plugin> χρησιμοποιείται για υπερσυνδέσμους ειδικών λειτουργιών όπως το Logout
- ViewModule <view>- για την αναπαράσταση στοιχείων του μοντέλου πεδίου
- HTMLModule <html> - χρησιμοποιείται για την αντικατάσταση της ετικέτας html που βρίσκεται στο πάνω μέρος της σελίδας από κάποια άλλη δομή.

Η ετικέτα <test> χρησιμοποιείται για την κατασκευή multiple choice test. Στο παράδειγμα που παρουσιάζεται παρακάτω φαίνονται και κάποιες πρόσθετες ετικέτες που χρησιμοποιούνται ως μέρος του τεστ όπως είναι οι <question> και <answer>.

```
<gale:test
    title="Example Test"
    action="#{conceptone#knowledge,value.intValue()};"
    expr="{conceptone#knowledge} &lt; 100"
    ask="2"
    alt="[You are not allowed to repeat this test.]"
    verbose="true">

    <gale:question
        answers="2"
        right="1">
        True or False: The capital of Cyprus is Nicosia
```

```

        <gale:answer correct="true">True <gale:explain>That
is correct. Well done!</gale:explain></gale:answer>

        <gale:answer correct="false">False
<gale:explain>Wrong Answer</gale:explain> </gale:answer>

    </gale:question>

    <gale:question
        answers="4"
        right="2">

        Which of the following cities are located in
France?

        <gale:answer correct="true"> Lyon
<gale:explain>Correct Answer</gale:explain></gale:answer>

        <gale:answer correct="false"> Athens<gale:explain>
Wrong Answer</gale:explain></gale:answer>

        <gale:answer correct="true">
Paris<gale:explain>Correct Answer</gale:explain></gale:answer>

        <gale:answer
correct="false">Barcelona<gale:explain>Wrong
Answer</gale:explain></gale:answer>

    </gale:question>

    <gale:result>

    Your score was <gale:variable
expr="\${conceptone#knowledge}.intValue()" />. <br />

    <gale:if expr="\${conceptone#knowledge} == 100">

    <gale:then>

    <br />Very good, you can now visit <gale:a
href="concepttwo">Concepttwo</gale:a>

    </gale:then>

    <gale:else>

    You have not succeeded to this test. Please <gale:a
href="conceptone">try again </gale:a>

    </gale:else>

    </gale:if>

    </gale:result>

```

Η ετικέτα <test> έχει 6 παραμέτρους:

- Η παράμετρος title ορίζει τον τίτλο του τεστ και το παρουσιάζει με την μορφή του header 1 (h1).
- Η παράμετρος action περιγράφει τις εντολές που θα πρέπει να εκτελέσει το GALE όταν ο χρήστης ολοκληρώσει το τεστ. Το σκορ του τεστ είναι διαθέσιμο στην μεταβλητή value και μπορεί να χρησιμοποιηθεί. Στο παράδειγμα "{conceptone#knowledge,value.intValue()}" η τιμή της value θα αποθηκευτεί στην ιδιότητα knowledge της έννοιας conceptone. Έτσι αν ο χρήστης επιτύχει σκορ ίσο με 100 τότε το 100 θα ανατεθεί στην ιδιότητα knowledge.
- Η παράμετρος expr είναι μια έκφραση τύπου Boolean η οποία είναι αληθής όσο επιτρέπεται στο χρήστη να κάνει το τεστ. Στο παράδειγμα "{conceptone#knowledge} < 100" σημαίνει ότι το τεστ επιτρέπεται όταν η ιδιότητα knowledge της conceptone είναι μικρότερη από 100. Διαφορετικά ο χρήστης δεν μπορεί να κάνει το τεστ.
- Η τιμή της παραμέτρου ask δείχνει πόσες ερωτήσεις περιλαμβάνει το τεστ. Το τεστ μπορεί να περιέχει πολλές ερωτήσεις από τις οποίες μόνο ο αριθμός της ask θα εμφανιστεί και η επιλογή των ερωτήσεων γίνεται τυχαία και παρουσιάζονται με τυχαία σειρά.
- Η τιμή της παραμέτρου alt είναι το μήνυμα που εμφανίζεται στο χρήστη όταν δεν επιτρέπεται πλέον να κάνει το τεστ.
- Η σημαία verbose έχει τιμή Boolean και δείχνει αν θα δθούν διευκρινίσεις μετά τη λήξη του τεστ. Όταν ορίζεται ψευδής ο χρήστης λαμβάνει μόνο το σκορ. Όταν ορίζεται αληθής ο χρήστης μπορεί να δει σε ποιες ερωτήσεις απάντησε σωστά και σε ποιες λανθασμένα επεξηγώντας ταυτόχρονα τις απαντήσεις.

Η ετικέτα <question> έχει 2 παραμέτρους :

- Τον αριθμό όλων των απαντήσεων και
- τον αριθμό των σωστών απαντήσεων.

Η ερώτηση μπορεί να περιέχει πολλές απαντήσεις και πολλές ορθές απαντήσεις. Οι απαντήσεις επιλέγονται τυχαία και παρουσιάζονται με τυχαία σειρά. Τα Radio buttons εμφανίζονται όταν υπάρχει μόνο μία σωστή απάντηση και τα checkboxes όταν οι σωστές απαντήσεις είναι


```

</gale:then>
<gale:else>

&nbsp;
  </gale:else>
</gale:if>
</gale:else>
</gale:if>

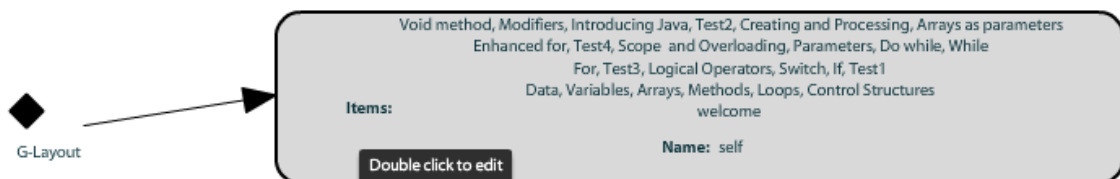
```

4.3 Σχεδίαση προσαρμοστικότητας πλοήγησης

Η υλοποίηση της προσαρμοστικότητας περιλαμβάνει τον ορισμό των κανόνων προσαρμογής (adaptation rules) και την τροποποίηση των πηγών (resources), δηλαδή των σελίδων xhtml για την κατά συνθήκη προσαρμογή του περιεχομένου.

Οι κανόνες προσαρμογής ορίζονται από τη σχεδίαση του Course Model στο Graph Author Tool. Εδώ καθορίζεται η παιδαγωγική δομή του μαθήματος γι' αυτό και οι κανόνες προσαρμογής αναφέρονται στο Graph Author Tool ως Pedagogical Relationship Type - PRT. Το αποτέλεσμα συχνά αναφέρεται ως Conceptual Adaptation Model ή CAM. Εξ ορισμού το GAT έχει ορισμένους δικούς του κανόνες από τους οποίους χρησιμοποιήθηκαν οι παρακάτω:

G-layout – Καθορίζει ποιες έννοιες θα χρησιμοποιούν την ειδική μορφή εμφάνισης του GALE, η οποία περιλαμβάνει ιεραρχική πλοήγηση, κεφαλίδα στο πάνω μέρος της σελίδας και υποσέλιδο στο κάτω μέρος της σελίδας. Διαθέτει μόνο έννοια-ες προορισμό. Ο κανόνας G-Layout εφαρμόστηκε σε όλες τις έννοιες (εικ. 23).



ΕΙΚΟΝΑ 4.39 – Ο κανόνας G-layout όπως απεικονίζεται στο course model του GAT

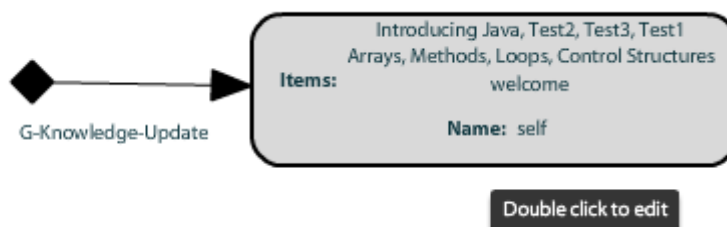
G-Prerequisite – Καθορίζει την καταλληλότητα μιας έννοιας. Ο κανόνας αυτός έχει μια ή περισσότερες έννοιες πηγή και μια ή περισσότερες έννοιες προορισμό. Αρχικά η έννοια προορισμός είναι ακατάλληλη για επίσκεψη και όταν η ιδιότητα knowledge της έννοιας πηγή φτάσει το επίπεδο “level”, που ορίζεται στον κώδικα του κανόνα και είναι εξ ορισμού ίσο με 80, τότε η έννοια προορισμό γίνεται κατάλληλη.

Ο κανόνας G-Prerequisite εφαρμόστηκε στις έννοιες Test3, For, While και Do while(εικ. 24).



ΕΙΚΟΝΑ 4.40 – Ο κανόνας G-Prerequisite όπως απεικονίζεται στο course model του GAT

G-Knowledge-Update – Διαθέτει μόνο έννοια-ες προορισμό Όταν η έννοια είναι κατάλληλη και την επισκέπτεται ο χρήστης τότε η ιδιότητα Knowledge γίνεται 100 αλλιώς γίνεται 35. Όταν η έννοια είναι κατάλληλη και την επισκέπτεται ο χρήστης τότε ο μετρητής των επισκέψεων αυξάνεται. Η γνώση (Knowledge) για μια έννοια δεν κερδίζεται απλά με την ανάγνωση της έννοιας αυτής αλλά χρειάζεται η ανάγνωση των υποεννοιών της. Δηλαδή, αν μια έννοια έχει δυο υποέννοιες η γνώση αποκτάται κατά το 1/3 από την κύρια έννοια και 1/3 από κάθε μια από τις δυο υποέννοιες. Ο κανόνας εφαρμόστηκε στις έννοιες welcome, Introducing Java, Control Structures, Loops, Methods, Arrays (εικ. 25).



ΕΙΚΟΝΑ 4.41 – Ο κανόνας G-Knowledge-Update όπως απεικονίζεται στο course model του GAT

G-Start – Προσδιορίζει ποια έννοια θα είναι η εναρκτήρια έννοια. Διαθέτει μόνο έννοια-ες προορισμό. Ο κανόνας G-Start εφαρμόστηκε στην έννοια welcome (εικ. 26).



ΕΙΚΟΝΑ 4.42 – Ο κανόνας G-Start όπως απεικονίζεται στο course model του GAT

G-Hide – Διαθέτει έννοια-ες πηγή και έννοια-ες προορισμό. Όταν ο χρήστης επισκέπτεται την έννοια πηγή η έννοια προορισμός γίνεται κρυφή. Ο κανόνας G-Hide δεν εφαρμόζεται αυτούσιος σε κάποια έννοια δίνεται όμως ο ορισμός του για την κατανόηση ενός νέου κανόνα που θα δημιουργηθεί παρακάτω και ο οποίος βασίζεται στον κώδικα προσαρμογής του.

G-Unhide – Διαθέτει έννοια-ες πηγή και έννοια-ες προορισμό. Όταν ο χρήστης επισκέπτεται την έννοια πηγή η έννοια προορισμός γίνεται ορατή. Ο κανόνας G-Unhide εφαρμόζεται μια φορά για το ζευγάρι εννοιών Data-Variables και μία φορά για το ζευγάρι Variables-Test1.



ΕΙΚΟΝΑ 4.43 - Ο κανόνας G-Unhide όπως απεικονίζεται στο course model του GAT

Εκτός από τους κανόνες προσαρμογής που προσφέρει το GALE υπάρχει η δυνατότητα για την δημιουργία νέων κανόνων από το χρήστη. Για την εκπαιδευτική εφαρμογή χρειάστηκε να δημιουργηθούν τρεις νέοι κανόνες όπως ορίζονται παρακάτω.

Ο κανόνας **G-Knowledge-Update-2** βασίστηκε στον ήδη υπάρχων κανόνα G-Knowledge-Update και η λειτουργία του είναι να ορίζει στο 50 την ιδιότητα της γνώσης (knowledge) μιας έννοιας όταν ο χρήστης επισκέπτεται την έννοια αυτή και η έννοια αυτή είναι κατάλληλη. Αν είναι ακατάλληλη και η γνώση για την έννοια αυτή είναι μικρότερη από 10 (level) και την επισκεφθεί ο χρήστης τότε η ιδιότητα knowledge ορίζεται στο 10. Ο κανόνας G-Knowledge-Update-2 εφαρμόζεται στις έννοιες Creating and Processing, Arrays as parameters, Enhanced for. Ο κώδικας προσαρμοστικότητας του κανόνα τροποποιείται ως εξής :


```

%self%

event + `

    if ($#suitability)

        ##visited, $#visited+1;

        ##own_knowledge, 50;

        if (!$#suitability && $#own_knowledge < %level%)

            ##own_knowledge, %level%; `

        #own_knowledge event + `

        ##knowledge,$#knowledge+changed.diff/($<-(parent).length+1);
` }

}

```

Ο κανόνας **G-Hide-At-First-Visit** βασίζεται στον κανόνα G-Hide και χρησιμοποιείται για την απόκρυψη της έννοιας προορισμός μόνο στην πρώτη επίσκεψη του χρήστη στην έννοια πηγή. Ο κώδικας του κανόνα τροποποιείται ως εξής:

```

%source%

    event + `

        if ($%source%#visited==1)

            #%target%#hierarchy, false;

` }

}

```

Ο κανόνας G-Hide-At-First-Visit εφαρμόζεται στις έννοιες welcome ως έννοια πηγή και Logical Operators, Switch, If, Test1 και Variables.

Ο κανόνας **G-Knowledge-Update 50** βασίζεται στον κανόνα G-Knowledge-Update και προσθέτει 50 μονάδες σε κάθε επίσκεψη του χρήστη στην αντίστοιχη έννοια όταν αυτή είναι κατάλληλη. Αν είναι ακατάλληλη και η γνώση για την έννοια αυτή είναι μικρότερη από 10 (level) και την επισκεφθεί ο χρήστης τότε η ιδιότητα knowledge ορίζεται στο 10. Ο κανόνας G-

Knowledge-Update 50 εφαρμόζεται στις έννοιες Void method, Test4, Variables, Data, If, Switch, Logical Operators, For, While, Do while, Parameters, Scope and Overloading, Modifiers.

```

%self%
    event + `
    if ($#suitability)
        ##visited, $#visited+1;
        ##own_knowledge, $#own_knowledge+50;
    if (!$#suitability && $#own_knowledge < %level%)
        ##own_knowledge, %level%; `
        #own_knowledge event + `
        ##knowledge,    $#knowledge+changed.diff/($&lt;- (parent) .length+1);
    `
}
}

```

Ο κανόνας **G-Prerequisite-100** βασίζεται στον κανόνα G-Prerequisite και η λειτουργία του είναι να ορίζει ως κατάλληλη την έννοια προορισμός όταν η γνώση της έννοιας πηγή είναι μεγαλύτερη από 40(level). Ο κανόνας G-Prerequisite-100 εφαρμόζεται στο ζεύγος εννοιών Parameters και Scope and Overloading.

```

%target%
#suitability & !`($%source%#knowledge>%level%) `

```

4.3.1 Ανακεφαλαίωση

Ανακεφαλαιώνοντας οι τεχνικές προσαρμοστικότητας που χρησιμοποιήθηκαν σε κάθε έννοια και αναφέρθηκαν αναλυτικά παραπάνω στο παρόν κεφάλαιο συνοψίζονται στον παρακάτω πίνακα.

Έννοια (Υποέννοιες)	Τεχνική προσαρμοστικότητας περιεχομένου	Τεχνική προσαρμοστικότητας πλοήγησης

Introducing Java (Data, Variables)	Knowledge Visited Multiple choice test	G-layout G-Knowledge-Update G-Knowledge-Update 50 G-Hide-At-First-Visit G-Unhide
Control Structures (If, Switch, Logical Operators)	Knowledge Visited Multiple choice test	G-layout G-Knowledge-Update G-Knowledge-Update 50 G-Hide-At-First-Visit
Loops (For, While, Do while)	Knowledge Multiple choice test	G-layout G-Prerequisite G-Knowledge-Update G-Knowledge-Update 50
Methods (Parameters, Modifiers, Void method, Scope and Overloading)	Knowledge Multiple choice test	G-layout G-Knowledge-Update G-Knowledge-Update 50 G-Prerequisite-100
Arrays (Creating and Processing, Arrays as parameters, Enhanced for)	Knowledge Multiple choice test	G-layout G-Knowledge-Update G-Knowledge-Update-2

ΠΙΝΑΚΑΣ 4.3 - Παρουσίαση τεχνικών προσαρμοστικότητας περιεχομένου και πλοήγησης που χρησιμοποιήθηκε σε κάθε έννοια

4.4 Έρευνα αξιολόγησης

Στη συνέχεια, η εκπαιδευτική εφαρμογή δόθηκε στον καθηγητή κ. Κυριάκου για εγκατάσταση στα εργαστήρια του Ευρωπαϊκού Πανεπιστημίου Κύπρου με σκοπό την αξιολόγηση του μαθήματος από πραγματική ομάδα σπουδαστών. Η έρευνα πραγματοποιήθηκε σε ένα σύνολο σπουδαστών που έχουν διδαχθεί το μάθημα Programming Principles I της Java. Συγκεκριμένα στην έρευνα αξιολόγησης έλαβαν μέρος 14 μαθητές από το Ευρωπαϊκό Πανεπιστήμιο Κύπρου, οι οποίοι κλήθηκαν να απαντήσουν σε ερωτηματολόγιο αμέσως μετά την παρακολούθηση της εκπαιδευτικής εφαρμογής. Οι απαντήσεις των συμμετεχόντων συγκεντρώθηκαν και

παρουσιάζονται στο επόμενο κεφάλαιο σε στατιστικά γραφήματα μαζί με την ανάλυση και τον σχολιασμό για κάθε μία από τις ερωτήσεις.

Το ερωτηματολόγιο περιλάμβανε τις παρακάτω ερωτήσεις :

1. Πιστεύετε ότι έχει αξία η εξατομίκευση του εκπαιδευτικού υλικού(αλλαγή περιεχομένου/εμφάνισης, εμφάνιση/απόκρυψη σελίδων) στην διαδικασία εκμάθησης;

Ναι

Όχι

Η πρώτη ερώτηση έχει σκοπό την εξαγωγή γενικευμένου συμπεράσματος από την ομάδα χρηστών και την προσωπική τους άποψη που βασίζεται στις γνώσεις τους και στην εμπειρία που έχουν για το αν έχει αποτέλεσμα η χρήση τεχνικών εξατομίκευσης στην εκπαιδευτική διαδικασία.

2. Βαθμολογήστε πόσο συντέλεσε η εξατομίκευση να επιτύχετε γρηγορότερα το στόχο σας (1= καθόλου, 5=πάρα πολύ).

1 2 3 4 5

Πιο ειδικά, η δεύτερη ερώτηση στοχεύει να αποτυπώσει αν λόγω της εξατομίκευσης οι χρήστες ολοκλήρωσαν το μάθημα γρηγορότερα και σε τι βαθμό συντέλεσε η εξατομίκευση σε αυτό. Η ταχύτερη εκμάθηση δίνει ως αποτέλεσμα την ευκολία στην κατανόηση του εκπαιδευτικού υλικού..

3. Ποια στρατηγική χρησιμοποιήθηκε στην 1η έννοια, Introducing Java, και τις υποσελίδες της;

Το τεστ ήταν στην αρχή

Το τεστ ήταν στο τέλος

Υπήρχε ένα τεστ σε κάθε σελίδα

Δεν υπήρχε τεστ

Οι ερωτήσεις 3-7 αποσκοπούν στο να ανιχνεύσουν αν οι τεχνικές εξατομίκευσης έχουν γίνει αντιληπτές από τους χρήστες.

4. Ποια στρατηγική χρησιμοποιήθηκε στην 2η έννοια, Control Structures, και τις υποσελίδες της;
- Το τεστ ήταν στην αρχή
 - Το τεστ ήταν στο τέλος
 - Υπήρχε ένα τεστ σε κάθε σελίδα
 - Δεν υπήρχε τεστ
5. Ποια στρατηγική χρησιμοποιήθηκε στην 3η έννοια, Loops, και τις υποσελίδες της;
- Το τεστ ήταν στην αρχή
 - Το τεστ ήταν στο τέλος
 - Υπήρχε ένα τεστ σε κάθε σελίδα
 - Δεν υπήρχε τεστ
6. Ποια στρατηγική χρησιμοποιήθηκε στην 4η έννοια, Methods, και τις υποσελίδες της;
- Το τεστ ήταν στην αρχή
 - Το τεστ ήταν στο τέλος
 - Υπήρχε ένα τεστ σε κάθε σελίδα
 - Δεν υπήρχε τεστ
7. Ποια στρατηγική χρησιμοποιήθηκε στην 5η έννοια, Arrays, και τις υποσελίδες της;
- Το τεστ ήταν στην αρχή
 - Το τεστ ήταν στο τέλος
 - Υπήρχε ένα τεστ σε κάθε σελίδα
 - Δεν υπήρχε τεστ
8. Σε ποιες έννοιες μπορεί ένας χρήστης να επαναλάβει το Test όσες φορές θέλει;
- Arrays
 - Introducing Java
 - Control Structures
 - Loops
 - Methods

Η λογική πίσω από την παραπάνω ερώτηση είναι αν η καθοδήγηση των χρηστών από το σύστημα, με βάση το μοντέλο χρήστη, έγινε κατανοητή.

9. Βαθμολογήστε πόσο θετικά ή αρνητικά λειτούργησε η προσαρμοστικότητα στο μενού πλοήγησης (αλλαγή στα χρώματα των κουκίδων και των γραμματοσειρών των links)(1=αρνητικά, 5=θετικά);
- 1 2 3 4 5

Η παραπάνω ερώτηση στοχεύει στην αξιολόγηση της προσαρμοστικότητας των υπερσυνδέσμων του μενού πλοήγησης (link adaptation).

10. Ήταν αντιληπτή η προσαρμοστικότητα του περιεχομένου του μαθήματος ανάλογα με την συμπεριφορά πλοήγησης του χρήστη;
- Ναι
Ναι σε κάποιες έννοιες
Όχι
Δεν κατάλαβα την ερώτηση

Στην παραπάνω ερώτηση διερευνάται αν γίνεται αντιληπτή από τους χρήστες η προσαρμοστικότητα περιεχομένου και τα υπό όρους κρυφά σημεία του κειμένου τα οποία υπάρχουν σε ορισμένες έννοιες και επανεμφανίζονται μόνο κάτω από συγκεκριμένες συνθήκες.

11. Σε ποια μαθήματα θα θέλατε το εκπαιδευτικό υλικό να παραδίδεται με τεχνικές εξατομίκευσης (Personalization);
- Μαθηματικά
Στατιστική
Ψυχολογία
Άλλες γλώσσες προγραμματισμού
Data structures and algorithms
Άλλη, διευκρινίστε ποιά.

Στην τελευταία ερώτηση οι συμμετέχοντες καλούνται να απαντήσουν αν

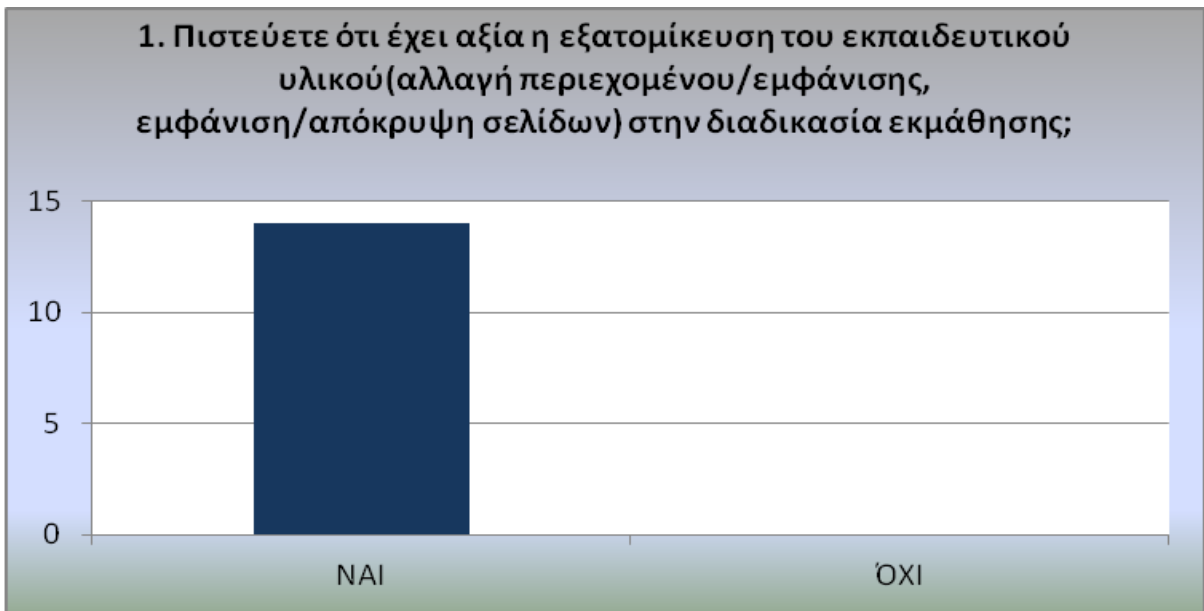
επιθυμούν και σε κάποια άλλα μαθήματα η ύλη να παραδίδεται με τεχνικές εξατομίκευσης. Η θετική απάντηση των χρηστών θα δείξει αν πραγματικά έμειναν ικανοποιημένοι με την εκπαιδευτική αυτή ιστοσελίδα και τις τεχνικές εξατομίκευσης.

Κεφάλαιο 5

Συμπεράσματα -Συζήτηση

Στην έρευνα συμμετείχαν 14 φοιτητές του Ευρωπαϊκού Πανεπιστημίου Κύπρου, οι οποίοι έχουν παρακολουθήσει το μάθημα Programming Principles I, που περιλαμβάνει υλικό από τις βασικές αρχές της γλώσσας Java.

Στην πρώτη ερώτηση οι φοιτητές κλήθηκαν να απαντήσουν από γενικής απόψεως αν θεωρούν ότι έχει αξία η χρήση τεχνολογιών εξατομίκευσης στην εκπαιδευτική διαδικασία, όπου το 100% των συμμετεχόντων απάντησαν θετικά (γράφημα 1). Το μεγάλο αυτό ποσοστό αντικατοπτρίζει την αιτία για τη μακρόχρονη και πολύκαρπη έρευνα στο πεδίο των εκπαιδευτικών συστημάτων που χρησιμοποιούν εξατομίκευση.



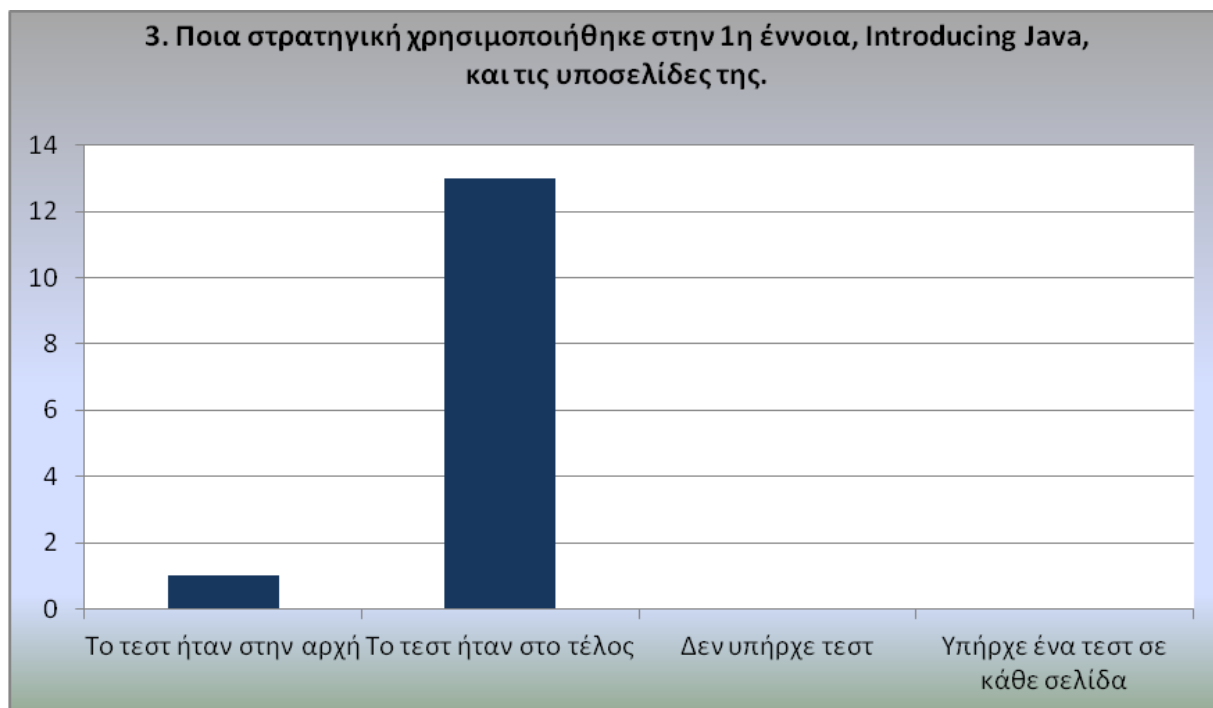
ΓΡΑΦΗΜΑ 5.1 – Απαντήσεις από την 1^η ερώτηση του ερωτηματολογίου

Στην 2^η ερώτηση ζητείται από το χρήστη να απαντήσει εξειδικευμένα για την εκπαιδευτική εφαρμογή στην οποία είχε την εμπειρία, αν η εξατομίκευση βοήθησε στην ταχύτερη εκμάθηση της εκπαιδευτικής ύλης. Από τις απαντήσεις και όπως φαίνεται παρακάτω στο γράφημα 2, δεν υπήρχε κανένας χρήστης ο οποίος να απάντησε αρνητικά. Το σύνολο των χρηστών απάντησαν ότι οι τεχνολογίες εξατομίκευσης που χρησιμοποιήθηκαν τους βοήθησαν τουλάχιστον αρκετά να ολοκληρώσουν γρηγορότερα με το περίπου 30% να ισχυρίζεται ότι τους βοήθησαν πάρα πολύ.



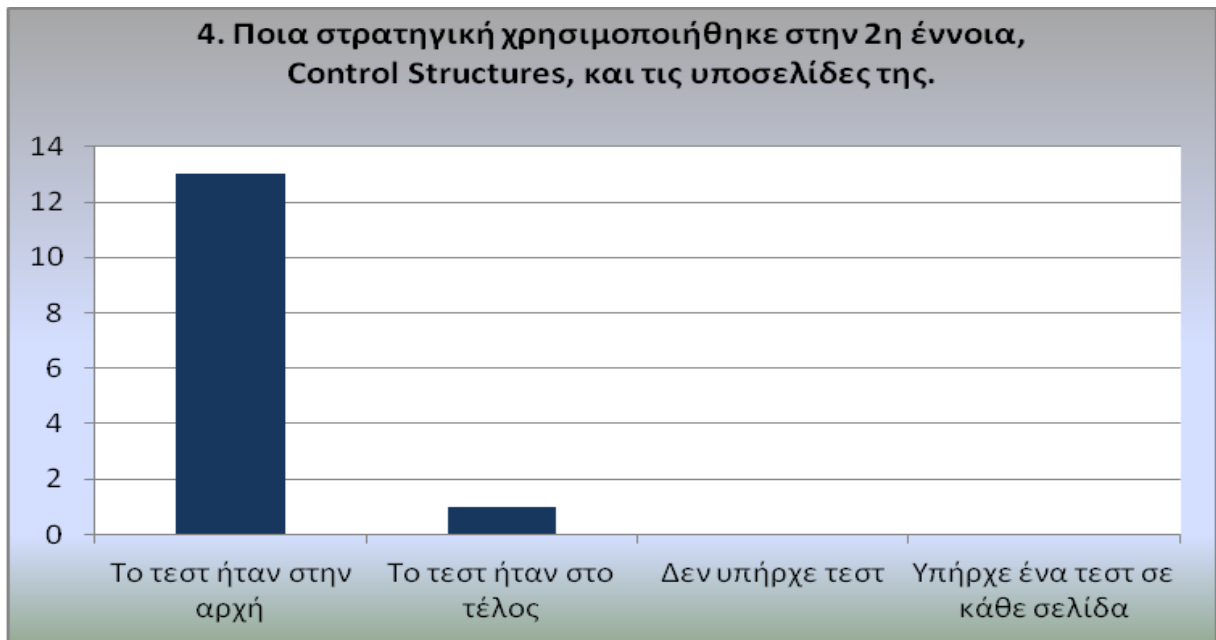
ΓΡΑΦΗΜΑ 5.2 – Απαντήσεις από την 2^η ερώτηση του ερωτηματολογίου

Οι επόμενες πέντε ερωτήσεις αφορούν τις έννοιες μία προς μία, και τις τεχνικές εξατομίκευσης και αν αυτές ήταν αντιληπτές από τους χρήστες. Σε κάθε ερώτηση υπάρχει μία σωστή απάντηση την οποία πρέπει να επιλέξουν οι συμμετέχοντες. Στην πρώτη έννοια, την Introducing Java, όλοι οι χρήστες πλην ενός αντιλήφθηκαν σωστά την ύπαρξη του τεστ στο τέλος της αντίστοιχης διδακτικής ύλης (γράφημα 3).



ΓΡΑΦΗΜΑ 5.3 – Απαντήσεις από την 3^η ερώτηση του ερωτηματολογίου

Ομοίως και στην δεύτερη έννοια, την Control Structures, πάνω από το 90% των χρηστών απάντησαν σωστά (γράφημα 4). Στην πρώτη έννοια το τεστ ήταν στην αρχή ενώ στην δεύτερη έννοια το τεστ ήταν στο τέλος. Εφ' όσον το ποσοστό του λάθους είναι ίδιο και στις δυο περιπτώσεις δεν μπορούμε να εξάγουμε συμπέρασμα για το αν η χρήση του τεστ στην αρχή είναι προτιμότερη έναντι της χρήσης του τεστ στο τέλος.



ΓΡΑΦΗΜΑ 5.4 – Απαντήσεις από την 4η ερώτηση του ερωτηματολογίου

Στις δύο επόμενες έννοιες, την Loops και την Methods, οι χρήστες απάντησαν σωστά στο σύνολο (γραφήματα 5 & 6). Αυτό μπορεί να συνέβη λόγω της σειράς μελέτης των εννοιών. Τις δυο τελευταίες έννοιες οι χρήστες επισκέφθηκαν πιθανότατα πιο πρόσφατα κι έτσι ανακάλεσαν ευκολότερα από την μνήμη τους την τεχνική εξατομίκευσης που χρησιμοποιήθηκε σε αυτές. Γι' αυτό το λόγο δεν υπήρχε καμία λανθασμένη απάντηση. Και εδώ όπως και στις δυο πρώτες έννοιες δεν μπορούμε να εξάγουμε συμπέρασμα για το αν η χρήση του τεστ στην αρχή είναι προτιμότερη έναντι της χρήσης του τεστ στο τέλος. Συνεπώς μπορεί να θεωρηθεί ότι η χρήση του τεστ στην αρχή είναι εξίσου ορατή με τη χρήση του τεστ στο τέλος.



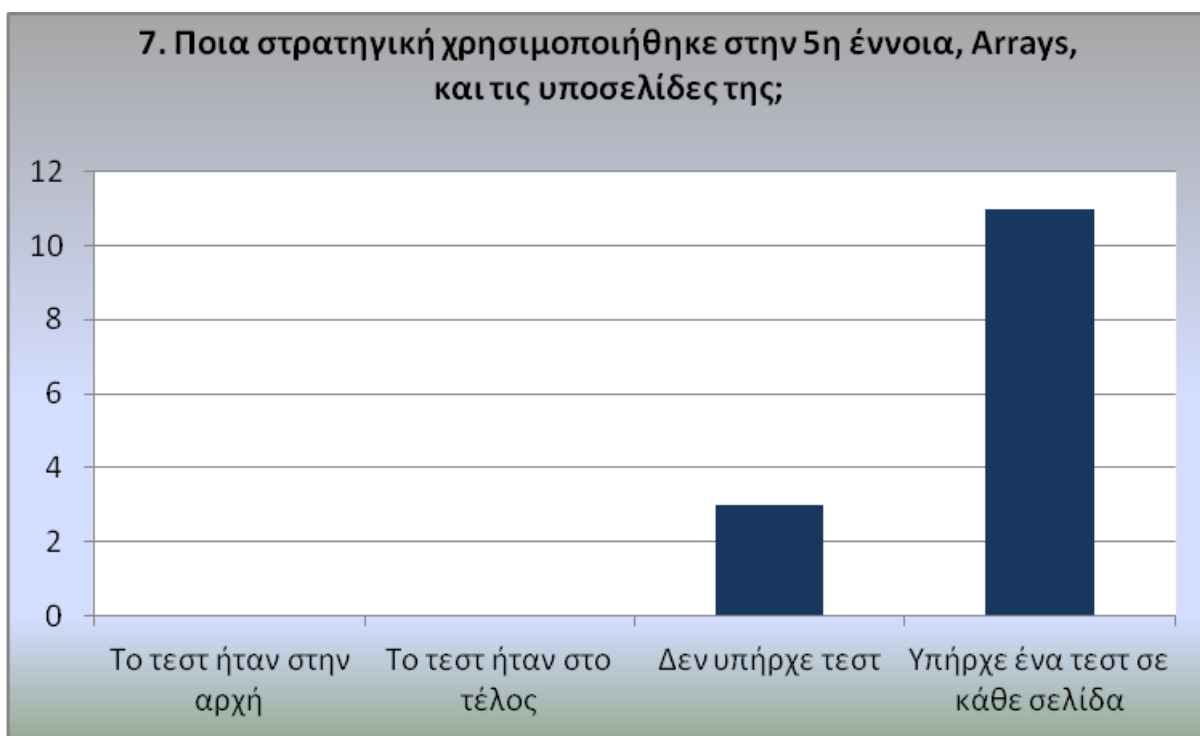
ΓΡΑΦΗΜΑ 5.5 – Απαντήσεις από την 5η ερώτηση του ερωτηματολογίου



ΓΡΑΦΗΜΑ 5.6 – Απαντήσεις από την 6η ερώτηση του ερωτηματολογίου

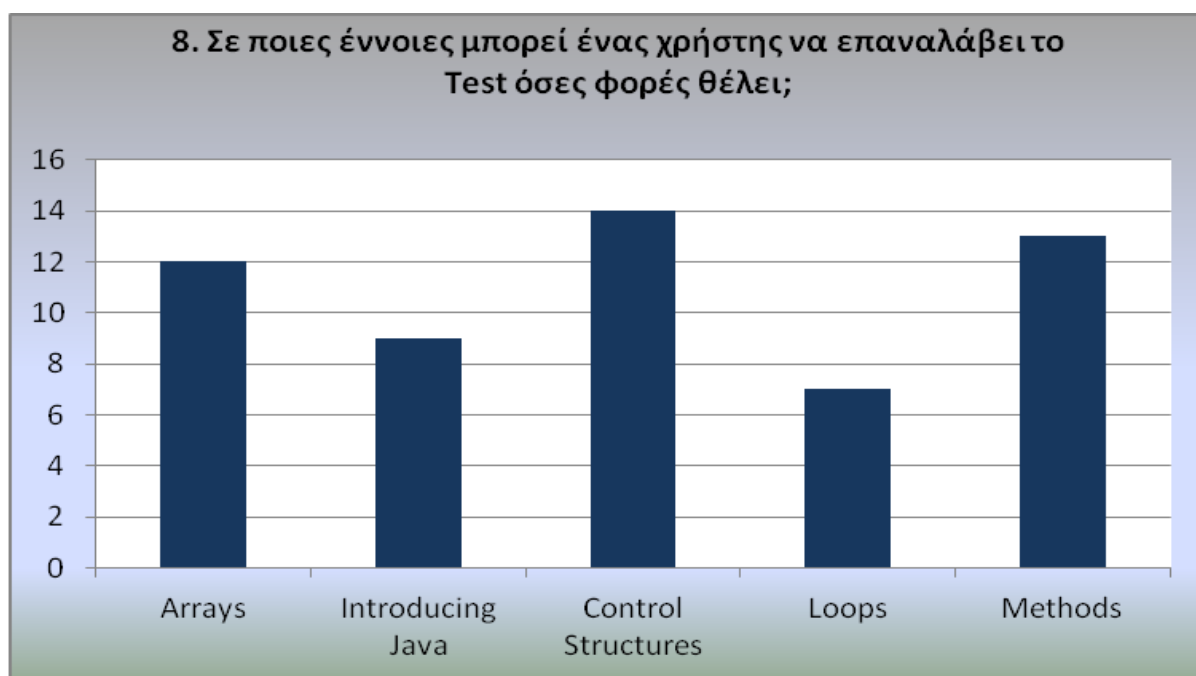
Σε αντίθεση με τις προηγούμενες τέσσερις έννοιες, η έννοια Arrays έχει περισσότερες από μία λανθασμένες απαντήσεις (γράφημα 7). Συγκεκριμένα, 3 χρήστες δεν εντόπισαν καθόλου το τεστ το οποίο υπήρχε στο κάτω μέρος κάθε υποσελίδας και αφορούσε την ύλη της συγκεκριμένης σελίδας. Δεδομένου του ποσοστού των λανθασμένων απαντήσεων η στρατηγική της υπάρξης

τεστ σε κάθε σελίδα δεν θεωρείται αρκετά ξεκάθαρη και πιθανόν να χρειάζεται κάποια επήληξη ή καθοδήληση.



ΓΡΑΦΗΜΑ 5.7 – Απαντήσεις από την 7η ερώτηση του ερωτηματολόγιου

Στην επόμενη ερώτηση οι χρήστες καλούνται να απαντήσουν σε ποιες έννοιες μπορούν να επαναλάβουν το τεστ πολλές φορές. Η σωστή απάντηση είναι στις έννοιες Control Structures, Methods και Arrays. Παρά το ότι μόνο τέσσερις απάντησαν σωστά, από τους υπόλοιπους 10 όλοι περιλάμβαναν την έννοια Control Structures στις απαντήσεις τους. Άρα συμπεραίνουμε ότι όλοι οι χρήστες επανέλαβαν το τεστ της συγκεκριμένης έννοιες παραπάνω από δυο φορές. Την έννοια Arrays επέλεξε επίσης το 80% των χρηστών που περιείχαν και λανθασμένες επιλογές στις απαντήσεις τους. Αυτό το 80% συνεπώς συμπλήρωσε το τεστ των υποσελίδων της έννοιας Arrays πολλές φορές. Ενώ το αντίστοιχο ποσοστό για την έννοια Methods είναι 90%. Από τις δυο έννοιες Introducing Java και Loops στις οποίες ο χρήστης μπορεί να επαναλάβει το τεστ μόνο δυο φορές, στη δεύτερη η συμπλήρωση του τεστ ως δύο φορές φαίνεται να είναι πιο ξεκάθαρη καθώς αποτελεί τις λιγότερες στις απαντήσεις των χρηστών. Επομένως η καθοδήγηση των χρηστών από το σύστημα είναι σχετικά αντιληπτή, με ένα μικρό περιθώριο βελτίωσης στις περιπτώσεις του περιορισμένου αριθμού επανάληψης των τεστ.



ΓΡΑΦΗΜΑ 5.8 – Απαντήσεις από την 8η ερώτηση του ερωτηματολογίου

Η προσαρμοστικότητα του μενού πλοήγησης είναι οι αλλαγές στο χρώμα γραμματοσειράς των υπερσυνδέσμων ανάλογα με την καταλληλότητα και το αν έχει αναγνωστεί μια σελίδα. Η επόμενη ερώτηση αξιολογεί την προσαρμοστικότητα στο μενού πλοήγησης και κατά πόσο λειτούργησε θετικά ή αρνητικά (γράφημα 9). Από τις απαντήσεις των χρηστών φαίνεται να επιδρά θετικά στο μεγαλύτερο ποσοστό των χρηστών. Είναι σημαντικό ότι δεν υπάρχει καμία

απάντηση για αρνητική επίδραση ενώ μόνο το 40% περίπου απάντησε ότι η ύπαρξη της προσαρμοστικότητας στο μενού πλοήγησης λειτούργησε ουδέτερα.



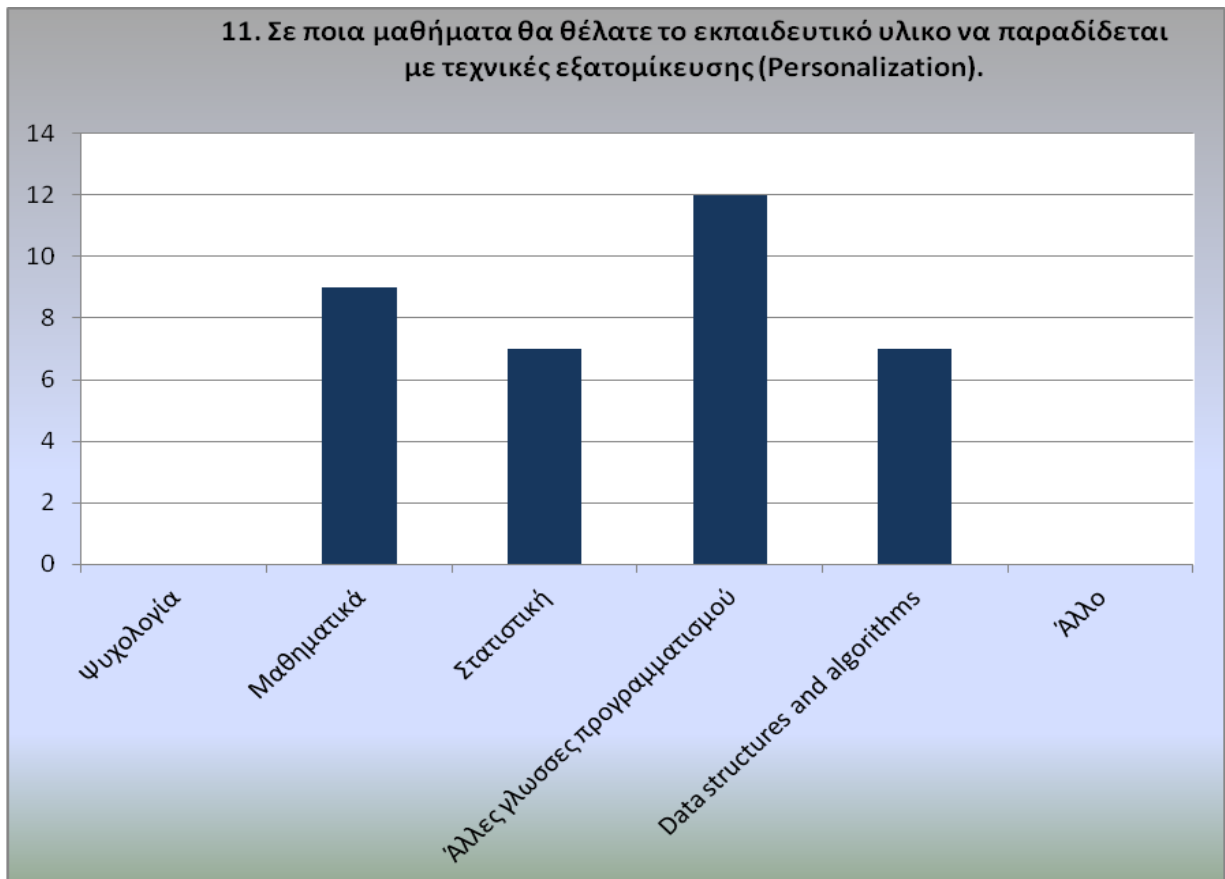
ΓΡΑΦΗΜΑ 5.9 – Απαντήσεις από την 9η ερώτηση του ερωτηματολογίου

Στο γράφημα 10 φαίνεται αν οι χρήστες αντιλήφθηκαν τις αλλαγές στο κείμενο και την μορφοποίηση του περιεχομένου που συμβαίνουν όταν ισχύουν κάποιες συνθήκες όπως π.χ. αποτυχία στο τεστ, επανάληψη ανάγνωσης σελίδας, κλπ. Οι αλλαγές αυτές συνιστούν την προσαρμοστικότητα του περιεχομένου(content adaptation)και έχουν οριστεί στις έννοιες Methods, Loops και Arrays. Το σύνολο των χρηστών έχει αντιληφθεί τις αλλαγές αυτές με το 65% να μπορεί να προσδιορίσει και ότι δεν υπήρχε προσαρμοστικότητα περιεχομένου σε όλες τις έννοιες του μαθήματος.



ΓΡΑΦΗΜΑ 5.10 – Απαντήσεις από την 10η ερώτηση του ερωτηματολογίου

Η τελευταία ερώτηση ζητά από τους χρήστες να επιλέξουν άλλα μαθήματα στα οποία θα ήθελαν το εκπαιδευτικό υλικό να παραδίδεται με τεχνικές εξατομίκευσης. Από το γράφημα 11 φαίνεται ότι οι χρήστες έχουν πολλές επιλογές, κάτι που σημαίνει ότι ήταν ικανοποιημένοι από την εμπειρία τους στο μάθημα της Java και γι' αυτό το λόγο επιλέγουν και άλλα μαθήματα, με κυρίαρχη επιλογή το μάθημα Άλλες γλώσσες προγραμματισμού, στα οποία θα ήθελαν να εφαρμοστεί το ίδιο. Γενικά οι επιλογές περιλαμβάνουν διάφορα μαθήματα που αφορούν τις θετικές επιστήμες. Και ενώ υπάρχει ως διαθέσιμη επιλογή το μάθημα της Ψυχολογίας, δεν έχει επιλεγεί από κανένα χρήστη. Για το μάθημα της Ψυχολογίας συμπεραίνουμε ότι οι συμμετέχοντες καλύπτονται από τα άλλα εκπαιδευτικά μέσα (στατικές ιστοσελίδες, παρουσιάσεις Powerpoint, κ.α.), ενώ για τα υπόλοιπα μαθήματα που υπάρχουν ως επιλογές και έχουν επιλεγεί από τους χρήστες τουλάχιστον μία φορά συμπεραίνουμε ότι η τεχνολογία εξατομίκευσης και μοντελοποίησης χρήστη προσφέρει καλύτερα μαθησιακά αποτελέσματα σε σχέση με παραδοσιακές μεθόδους διδασκαλίας.



ΓΡΑΦΗΜΑ 5.11 – Απαντήσεις από την 11η ερώτηση του ερωτηματολογίου

Το γενικό συμπέρασμα είναι ότι οι χρήστες αποκόμισαν θετικές εμπειρίες από την εκπαιδευτική εφαρμογή της Java με τεχνικές εξατομίκευσης και θεωρούν ότι η εξατομίκευση έχει θετική επίδραση στην εκπαιδευτική διαδικασία. Αυτό αποτυπώνεται τόσο στην τελευταία ερώτηση όπου από τις απαντήσεις (35 επιλογές) συμπεραίνουμε ότι κάθε χρήστης επέλεξε κατά μέσο όρο 2,5 μαθήματα επιπλέον στα οποία επιθυμεί το εκπαιδευτικό υλικό να παραδίδεται με τεχνικές εξατομίκευσης όσο και στην 2^η ερώτηση όπου το 100% απάντησε ότι η εξατομίκευση συντέλεσε τουλάχιστον αρκετά στην είτευξη του στόχου. Όσον αφορά συγκεκριμένα τις τεχνικές του συστήματος GALE, η άποψη των χρηστών ήταν κατά το μεγαλύτερο ποσοστό (80-90%), σε ορισμένες περιπτώσεις και κατά το σύνολο, ότι είναι αντιληπτές και κατανοητές.

Για την πραγματοποίηση της έρευνας αξιολόγησης υπήρξαν κάποιοι **περιορισμοί** οι οποίοι είναι σημαντικό να αναφερθούν. Δεδομένου ότι αυτό που αξιολογείται είναι οι τεχνικές προσαρμοστικότητας που χρησιμοποιήθηκαν και όχι το εκπαιδευτικό υλικό, η παράδοση της εκπαιδευτικής αυτής εφαρμογής σε εκπαιδευόμενους στους οποίους είναι άγνωστο το γνωστικό αντικείμενο της Java θεωρείται «πειραματισμός» και δεν είναι αποδεκτό για το επίπεδο αυτό.

Επειδή, λοιπόν ήταν δύσκολο να βρεθούν εκπαιδευόμενοι που θα έπρεπε τόσο να έχουν διδαχθεί τις βασικές έννοιες της Java όσο και να μπορούν να εγκαταστήσουν την εφαρμογή, προχωρήσαμε, με τη βοήθεια του επιβλέποντος καθηγητή στην εγκατάσταση της εφαρμογής στα εργαστήρια του Ευρωπαϊκού Πανεπιστημίου Κύπρου, όπου είναι εκπαιδευτής ο Κ. Κυριάκου, και στην ολοκλήρωση της έρευνας με μαθητές του μαθήματος Programming Principles I.

Κεφάλαιο 6

Επίλογος

Ολοκληρώνοντας την παρούσα μεταπτυχιακή διατριβή γίνεται μία σύντομη ανασκόπηση επισημαίνοντας ορισμένα σημεία, τα οποία για διάφορους λόγους κρίθηκαν σημαντικά για την πορεία της διατριβής.

Το πολύ μεγάλο εύρος του αντικειμένου της εξατομίκευσης (Personalization) και της μοντελοποίησης χρήστη (User Modeling) προκαλεί από μόνο του μια μεγάλη σύγχυση προσθέτοντας επιπλέον προβληματισμό όταν αυτές οι δυο έννοιες συνδυάζονται με τα εκπαιδευτικά υπερμέσα. Πότε ξεκίνησαν, πως εξελίχθηκαν και σε ποιο σημείο βρίσκονται τώρα τα εκπαιδευτικά υπερμέσα που κάνουν χρήση εξατομίκευσης και μοντελοποίησης χρήστη είναι κάποια από τα ερωτήματα που έχουν απαντηθεί στα δυο πρώτα κεφάλαια. Η εκπαίδευση στο σύστημα AHA πριν από την εκπαίδευση στο σύστημα GALE θεωρήθηκε αναγκαία εφ' όσον το σύστημα GALE βασίζεται στο AHA και αποτελεί τη συνέχειά του. Αυτή η αναγκαία σειρά προτεραιότητας θεωρήθηκε εκ των υστέρων πολύ χρήσιμη καθώς η γνώση για την χρήση τεχνικών εξατομίκευσης στην εκπαιδευτική διαδικασία αποκτήθηκε σταδιακά και παράλληλα με την αύξηση πολυπλοκότητας από το πρώτο στο δεύτερο και η μετέπειτα γνωριμία με το σύστημα GALE ήταν κατά πολύ ευκολότερη και πιο κατανοητή. Στη σχεδίαση του μαθήματος αρχικά θεωρήθηκε αναγκαία η δυνατότητα επιλογής ανάγνωσης εννοιών από τους ίδιους τους

χρήστες (να μπορεί δηλαδή ο χρήστη να διαβάσει μόνο την έννοια Methods και τις υποσελίδες της χωρίς να επισκεφθεί τις προηγούμενες) κι έτσι οι χρήστες δεν έχουν τη δυνατότητα να μεταβούν από ένα μέρος του μαθήματος σε ένα άλλο χωρίς να κάνουν χρήση του κυρίως μενού πλοήγησης που βρίσκεται στα αριστερά της σελίδας. Μπορούν να μετακινούνται μέσα στο μάθημα από σελίδα σε σελίδα μόνο επιλέγοντας τον αντίστοιχο υπερσύνδεσμο του αριστερού μενού που οδηγεί στην επιθυμητή σελίδα. Σε δεύτερο χρόνο και μετά από όλη την εμπειρία που αποκτήθηκε για την κατασκευή αυτού του εκπαιδευτικού ιστότοπου θα μπορούσε να αναθεωρηθεί η σχεδίαση ως προς την καθοδήγηση του χρήστη από το σύστημα. Για τον λόγο, λοιπόν, της καθοδηγούμενης διδασκαλίας και ίσως αυτό να ληφθεί ως συμβουλή για την επόμενη ερευνητική εργασία, θα μπορούσε το εκπαιδευτικό υλικό να παραδίδεται εξ ολοκλήρου με σειρά προτεραιότητας των εννοιών που ορίζεται από τον συντάκτη χωρίς ο χρήστης να έχει την δυνατότητα κατά την πρώτη ανάγνωση να επιλέξει οποιαδήποτε έννοια. Κάτι τέτοιο σημαίνει ότι δεν θα δίνεται η δυνατότητα στο χρήστη να επιλέξει να μελετήσει μόνο μία έννοια του μαθήματος η οποία μπορεί να βρίσκεται 3^η κατά σειρά και οι δυο πρώτες να είναι προαπαιτούμενες. Αυτό, όμως, μπορεί να θεωρηθεί ασήμαντο όταν η ομάδα εκπαιδευόμενων στην οποία στοχεύει ο συντάκτης είναι, για παράδειγμα, μία τάξη κάποιου τμήματος πανεπιστημίου που διδάσκεται για πρώτη φορά το συγκεκριμένο μάθημα.

Η συγκεκριμένη μεταπτυχιακή διατριβή, με την καθοδήγηση του επιβλέποντος καθηγητή δρ. Δημήτρη Κυριάκου, μπορεί να παρουσιαστεί σε συνέδριο για καινοτόμες εκπαιδευτικές τεχνολογίες, μιας και καλύπτει μία ανάγκη στην εκπαιδευτική κοινότητα που δεν καλύπτεται από υπάρχουσες εφαρμογές.

Μετά το πέρας της παρούσας κατασκευής του εκπαιδευτικού ιστότοπου για το μάθημα της Java, η έρευνα και η εφαρμογή νέων συνδυασμών τεχνικών εξατομίκευσης με το σύστημα GALE θα συνεχιστεί και σε άλλα μαθήματα, οδηγούμενη από τις επιλογές των χρηστών όπως αυτοί απάντησαν στην αντίστοιχη ερώτηση του ερωτηματολογίου για τα μαθήματα των θετικών επιστημών. Το συγκεκριμένο θέμα μπορεί να επεκταθεί ακόμα περισσότερο, αποτελώντας κίνητρο για μελλοντική έρευνα, διασυνδέοντας την εκπαιδευτική ιστοσελίδα με τα κοινωνικά δίκτυα (π.χ. Facebook) ώστε με την εγγραφή ενός νέου χρήστη το σύστημα να έχει έτοιμα στοιχεία για την δημιουργία του μοντέλου χρήστη από το προφίλ του στο συγκεκριμένο κοινωνικό δίκτυο.

Βιβλιογραφία

- [01] Apple W. P. Fok, Horach H. S. IP: Personalized Educational Hypermedia, 2008
- [02] Brusilovsky P., Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-based Education,, 2000
- [03] Brusilovsky, Adaptive Hypermedia for Education and Training, Adaptive Technologies for Training and Education, Paula J. Durlach, Alan M. Lesgold, 2012
- [04] Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. User Modeling and User-Adapted Interaction, 6, 1996
- [05] Brusilovsky, Schwarz, & Weber, World-Wide Intelligent Textbooks, 1996
- [06] Cheng, M., Gao, X., Gao, S., Xu, D.: Design and implementation of a braincomputer interface with high transfer rates, 2002
- [07] Cheng, M., Gao, X., Gao, S., Xu, D.: Design and implementation of a braincomputer interface with high transfer rates. IEEE Transactions on Biomedical Engineering 49, 1181–1186, 2002
- [08] De Bra P., Smits, D., Pechenizkiy, M., Vasilyeva, E., Adaptivity in GRAPPLE: Adaptation in Any Way You Like, 1997
- [09] De Bra Paul, Ad Aerts, Bart Berden, Barend de Lange, Brendan Rousseau, Tomi Santic, David Smits, Natalia Stash, AHA! The Adaptive Hypermedia Architecture, 2003
- [10] De Bra P., Smits, D., van der Sluijs, K., Cristea, I. A., Foss, J., Glahn, C., and Steiner, C.: GRAPPLE1: Learning Management Systems Meet Adaptive Learning Environments, 2013

- [11] De Bra, P., Calvi, L: AHA! An open Adaptive Hypermedia Architecture, 1998
- [12] De Bra, P., Calvi, L: AHA: a Generic Adaptive Hypermedia System, 1998
- [13] De Bra, P., Calvi, L.: AHA: A Generic Adaptive Hypermedia System. In: Proc. of the 2nd Workshop on Adaptive Hypertext and Hypermedia, Pittsburgh, pp. 5–12, 1998
- [14] De Bra, P., Smits, D., Stash, N.: The Design of AHA!, In: ACM Hypertext'06: Proceedings of the 17th ACM Conference on Hypertext and Hypermedia, pp. 133-134, 2006
- [15] Felder, R, "Learning and teaching styles in engineering education", Engr. Education, 1988
- [16] Fink, J., User Modeling Servers, Requirements, Design, and Evaluation, 2003
- [17] Grigoriadou, M. and Papanikolaou, K., Authoring Personalized Interactive Content, 2006
- [18] Kay J: Lifelong Learner Modeling for Lifelong Personalized Pervasive Learning, 2008
- [19] Kay, J., Lifelong user models, memory and learning. ResearchChannel.org, 2007
- [20] Kobsa, A., Generic User Modeling Systems, 2000
- [21] Kobsa, A.,: User Modelling and User Adapted Interaction in an Intelligent Tutoring System, Volume 1 issue 1, 1991
- [22] Koch et al: Communities and Personalization for Individuals Products, 2002

- [23] Kyriacou D., Enriching Lifelong User modeling in a Social E-networking and E-commerce Era, 2010
- [24] Nguyen Loc, Phung Do: Combination of Bayesian Network and Overlay model in User Modeling, 2009
- [25] Pipatsarun Phobun, Jiracha Vicheanpanya: Adaptive intelligent tutoring systems for e-learning systems, 2010
- [26] Pohl Wolfgang: Logic-Based Representation and Reasoning for User Modeling Shell Systems, 1998
- [27] Smits, D., DeBra, P., GALE: Generic Adaptation Language and Engine, 2012
- [28] Urban-Lurain, M, Intelligent Tutoring Systems: An Historic Review in the Context of the Development of Artificial Intelligence and Educational Psychology, στο <http://www.cse.msu.edu/rgroups/cse101/ITS/its.htm>. Τελευταία αναφορά 16-05-2014.
- [29] Urban-Lurain M.: Intelligent Tutoring Systems: An Historic Review in the Context of the Development of Artificial Intelligence and Educational Psychology
- [30] Van der Sluijs, K., Houben, G.J. A Generic Component for Exchanging User Models between Web-based Systems, 2006
- [31] Personalized learning στην τοποθεσία www.wikipedia.org. Τελευταία αναφορά 16-05-2014.