

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή στα Πληροφοριακά Συστήματα



Ανάλυση, Σχεδίαση και Υλοποίηση Προσωποποιημένων Πυλών
για Εκπαιδευτικά Ιδρύματα: Εφαρμογή ΤΕΙ Κρήτης

Γεώργιος Γεωργιάς

Επιβλέπουσα Καθηγήτρια
Αγγελική Κοκκινάκη

Ιανουάριος 2013

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

**Ανάλυση, Σχεδίαση και Υλοποίηση Προσωποποιημένων Πυλών
για Εκπαιδευτικά Ιδρύματα: Εφαρμογή ΤΕΙ Κρήτης**

Γεώργιος Γεωργιάς

**Επιβλέπουσα Καθηγήτρια
Αγγελική Κοκκινάκη**

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

Ιανουάριος 2013

Περίληψη

Η παρούσα μεταπτυχιακή διατριβή έχει ως σκοπό να αναλύσει, να σχεδιάσει και να υλοποιήσει μια διαδικτυακή πύλη (portal) που ενσωματώνει προσωποποιημένες διαδικτυακές υπηρεσίες εκπαιδευτικού περιβάλλοντος με πιστοποίηση. Το σύνολο των υποσυστημάτων που αναπτύχθηκαν βασίστηκαν σε λογισμικό ανοικτού κώδικα και σε τεχνολογίες και πρότυπα διαλειτουργικότητας (portlets, web services), ώστε να είναι διαθέσιμα και να μπορούν να λειτουργήσουν με το δυνατόν λιγότερους τεχνολογικούς περιορισμούς. Πιο συγκεκριμένα, στη διατριβή αυτή εκπονήθηκαν τα ακόλουθα:

- Ανάλυση απαιτήσεων/σύνταξη προδιαγραφών του συστήματος μετά από καταγραφή της παρούσας κατάστασης στο ΤΕΙ Κρήτης με προσδιορισμό κατηγοριών χρηστών και κατάλογου προσφερόμενων υπηρεσιών ανά κατηγορία χρηστών.
- Σχεδιασμός του συστήματος, και επιλογή των εργαλείων ανάπτυξης. Υποστηρίχθηκαν και εφαρμόστηκαν διεθνώς αναγνωρισμένοι κανόνες προτύπων και μελετήθηκε η πολυκαναλική διάθεση των προσφερόμενων υπηρεσιών σε φορητές συσκευές.
- Υλοποίηση της πύλης, με την δημιουργία του μηχανισμού πρόσβασης των χρηστών και των διαδικασιών ενσωμάτωσης-ενοποίησης των διαφορετικών εφαρμογών που θα χρησιμοποιηθούν, κυρίως ως προς τον τρόπο επικοινωνίας τους με το κοινό περιβάλλον πρόσβασης κι εκτέλεσης που θα προσφέρει το portal.
- Αξιολόγηση της πύλης από τους ίδιους τους χρήστες της και έλεγχος σε ποιό βαθμό είναι συμβατή με διεθνώς αναγνωρισμένα πρότυπα σχεδίασης ιστοχώρων.

Η δημιουργία μιας εξατομικευμένης πύλης απλοποιεί την αναζήτηση της επιθυμητής πληροφορίας. Η συγκέντρωση και ομαδοποίηση των υπηρεσιών ανάλογα το ρόλο του χρήστη, βοηθά στην αποτελεσματικότερη πρόσβαση και την άμεση ενημέρωση των χρηστών για διαθεσιμότητα νέων υπηρεσιών στο μέλλον.

Summary

This postgraduate thesis aims to analyze, design and implement a web portal in an educational institution that integrates password-protected personalized e-services.

Development of all the subsystems in this project, was based on open source software, technologies and interoperability standards (portlets, web services), that are available and can be operated with fewer possible technological limitations. In this project, we have conducted the following::

- Analyzed requirements and defined specifications of the portal, after writing down the current situation in TEI, and setting the classification of users, the list of the services offered respectively.
- Designed the system, and selected the development tools. We examined the compatibility against internationally recognized rules and standards and addressed the availability of developed e-services to mobile devices.
- Implementation of the portal, the creation of the mechanism for user access and the procedure of integration of the applications from different infrastructure that have been used.
- Evaluation of the usability of the portal by the users themselves and we will examine in what extent it is compatible with internationally recognized standards of design websites.

A personalized portal facilitates access to e-service for the users. The concentration and clustering of services depending on the user role, helps to effectively access and informing users directly about availability of new services in the future.

Ευχαριστίες

Για την υλοποίηση της μεταπτυχιακής διατριβής θα ήθελα να ευχαριστήσω αυτούς που συνέβαλαν με τον ένα ή άλλο τρόπο στην ολοκλήρωσή της:

- την Δρ. Αγγελική Κοκκινάκη, Αναπληρώτρια Καθηγήτρια στο Πανεπιστήμιο Λευκωσίας, η οποία ως επιβλέπουσα της διατριβής, ήταν πάντα διαθέσιμη όποτε τη χρειάστηκα και με τη πολύτιμη καθοδήγησή της ολοκληρώθηκε η εργασία.
- τον Δρ. Κώστα Βασιλάκη, Επιστημονικό Υπεύθυνο του Κέντρου Έλεγχου και Διαχείρισης Δικτύων (Κ.Ε.Ε.Δ.) του ΤΕΙ Κρήτης, οποίος πάντα πίστευε σε εμένα και μου πρότεινε το θέμα.
- τον κ. Μιχάλη Παναγιωτάκη, Γενικό Γραμματέα του ΤΕΙ Κρήτης και την κα Ειρήνη Κατεχάκη, προϊσταμένη του τμήματος Πληροφορικής της Διοίκησης, που στο διάστημα αυτό, επέτρεψαν μέρος της διπλωματικής εργασίας μου να υλοποιηθεί κατά τη διάρκεια της εργασίας μου ενώ, ταυτόχρονα περιόρισαν τις υπόλοιπες υποχρεώσεις μου στο τμήμα που έχω τη χαρά να ανήκω.
- το προσωπικό του Γρ. Διασύνδεσης του ΤΕΙ Κρήτης, του οποίου ήμουν υπεύθυνος μηχανογράφησης και η θέση εργασίας μου είναι στο χώρο τους, που υπόμεναν ίσως και περισσότερο από ότι θα έπρεπε την κυκλοθυμική συμπεριφορά μου.
- τον κ Ιωάννη Καπετανάκη, εξωτερικό συνεργάτη του Κ.Ε.Ε.Δ., για την πολύτιμη βοήθεια του σε θέματα ενσωμάτωσης του Κεντρικού καταλόγου (LDAP) του ΤΕΙ Κρήτης στο Κεντρικό Σύστημα Ταυτοποίησης της διαδικτυακής πύλης μας, αλλά και στην παροχή τεχνικής υποστήριξης σε θέματα έκδοσης πιστοποιητικού ασφαλείας για τον τομέα (domain) στο οποίο σε πιλοτική μορφή είναι εγκατεστημένη πύλη μας.

Περιεχόμενα

1.	Εισαγωγή.....	1
1.2	Πλαίσιο Αναφοράς.....	2
1.3	Κίνητρα.....	4
1.4	Σκοπός και Επιμέρους Στόχοι.....	4
1.5	Αναγκαιότητα και Σπουδαιότητα.....	5
1.6	Μεθοδολογική Προσέγγιση.....	6
1.7	Οφέλη.....	7
1.8	Περιορισμοί.....	8
1.9	Σύνοψη.....	9
2.	Βιβλιογραφική Επισκόπηση.....	10
2.1	Εισαγωγή.....	10
2.2	Portal (Διαδικτυακή Πύλη).....	11
2.2.1	Εισαγωγή	11
2.2.2	Η εξέλιξη των portals	12
2.2.3	Λειτουργικότητα	13
2.2.4	Εξατομίκευση πύλης	15
2.2.5	Διαχωρισμός Portal	16
2.2.6	Κέρδη από τη χρήση ενός portal	16
2.3	Portlets.....	17
2.3.1	Εισαγωγή	17
2.3.2	Λειτουργικότητα	18
2.3.3	Αρχιτεκτονική	19
2.3.4	Διαφορές και ομοιότητες με τα servlets	21
2.4	JSRs-Java Specification Requests.....	23
2.4.1	Web Services for Remote Portlets	24
2.5	Υποδομή πύλης.....	24
2.5.1	Το δοχείο του συστατικού στοιχείου πύλης (portlet container)	25
2.5.2	Ο διακομιστής πύλης (portal server)	25
2.6	LDAP.....	26
2.6.1	Ιστορική ανάδρομη	26
2.6.2	Πλεονεκτήματα	27

2.6.3	Προδιαγραφές	27
2.7	Shibboleth	31
2.7.1	Εισαγωγή	31
2.7.2	Έλεγχος πρόσβασης	32
2.8	Hypertext Transfer Protocol Secure (HTTPS)	33
2.8.1	Εισαγωγή	33
2.8.2	Ρύθμιση διακομιστή	35
2.8.3	SSL	36
2.9	Κρυπτογράφηση.....	38
2.10	Πιστοποιητικά	41
2.10.1	Εισαγωγή	41
2.10.2	Λειτουργικότητα	42
2.10.3	Υπογραφή πιστοποιητικού	42
2.11	Οδηγίες για την Προσβασιμότητα του Περιεχομένου του Ιστού.....	43
2.11.1	Εισαγωγή	43
2.11.2	Επίπεδα WCAG 2.0 Προσανατολισμού	44
2.11.3	Γενικές αρχές καθοδήγησης	45
2.11.4	Κριτήρια επιτυχίας	47
2.11.5	Τεχνικές Επάρκειας και Συμβουλευτικής	47
2.11.6	WCAG 2.0 έγγραφα	48
2.12	Μέθοδοι αξιολόγησης ευχρηστίας.....	49
2.12.1	Εισαγωγή	49
2.12.2	Μέθοδοι αξιολόγησης	50
2.12.3	Μέθοδος ελέγχου συμβατότητας με κανόνες και πρότυπα	51
2.12.4	Διερευνητικοί μέθοδοι	52
2.12.5	Συμπλήρωση ερωτηματολογίων	52
2.13	Σύνοψη	53
3.	Ανάλυση Απαιτήσεων	55
3.1	Εισαγωγή	55
3.2	Επισκόπηση ΤΕΙ Κρήτης.....	55
3.3	Τρέχουσα κατάσταση υποστήριξης χρηστών στο ΤΕΙ Κρήτης	56
3.4	Καταγραφή τμημάτων & υπηρεσιών.....	58
3.5	Κατάλογος ιστοσελίδων προς υλοποίηση	61

3.6	Κατηγορίες χρηστών	62
3.7	Περιγραφή λειτουργικότητας του συστήματος.....	62
3.7.1	Εισαγωγή	62
3.7.2	Προσωπική σελίδα χρήστη	63
3.8	Σύνοψη	64
4.	Μεθοδολογική Προσέγγιση & Εργαλεία Ανάπτυξης	66
4.1	Εισαγωγή	66
4.2	Μεθοδολογία ICONIX.....	67
4.2.1	Εισαγωγή	67
4.2.1	Use Case: Σύνδεση χρήστη με το λογαριασμό του	69
4.2.2	Use Case: Σύνδεση με το λογαριασμό, Robustness diagram	70
4.2.3	Use Case: Σύνδεση με το λογαριασμό, Sequence Diagram	71
4.3	uPortal	72
4.3.1	Εισαγωγή	72
4.3.2	Δυνατότητες	72
4.3.3	Έλεγχος ταυτότητας	73
4.4	Central Authentication System, CAS	73
4.5	uMobile	75
4.6	Spring.....	75
4.6.1	Εισαγωγή	76
4.6.2	Φιλοσοφία	77
4.7	Model-View-Controller (MVC)	78
4.8	HSQLDB.....	79
4.9	XSL	79
4.10	Fluid	80
4.11	Apache Maven.....	80
4.12	Apache Ant	81
4.13	Σύνοψη	81
5.	Υλοποίηση και Αντιμετώπιση Θεμάτων.....	83
5.1	Εισαγωγή	83
5.2	Επιλογή της πύλης.....	84
5.3	Κατηγορίες χρηστών	86

5.3.1	Εισαγωγή	86
5.3.2	Ταυτοποίηση ρόλων	87
5.3.3	Τεχνική υποστήριξη χρηστών πύλης	89
5.4	Κατηγορίες εφαρμογών/υπηρεσιών Πύλης	90
5.5	Ασφάλεια πύλης.....	91
5.5.1	Προστασία δεδομένων uPortal	92
5.5.2	Ασφάλεια συναλλαγών	92
5.6	Περιγραφή του μηχανισμού πιστοποίησης χρηστών.....	93
5.7	Πρόσβαση Διαδικτυακής Πύλης σε ΑΜΕΑ	94
5.8	Πολυκαναλική διάθεση των προσφερόμενων υπηρεσιών	95
5.9	Σύνοψη	98
6.	Σχεδίαση των Μικροεφαρμογών-Portlets	99
6.1	Εισαγωγή	99
6.2	Υπηρεσίες-εφαρμογές ανά χρήστη.....	99
6.3	Προδιαγραφές των portlets	103
6.4	Σχεδιασμός των portlets	104
6.6	Ενσωμάτωση διαδικτυακών υπηρεσιών του Ιδρύματος	107
6.7	Σύνοψη	108
7.	Αξιολόγηση της Πύλης	110
7.1	Εισαγωγή	110
7.2	Μέθοδος αξιολόγησης συμβατότητας.	111
7.2.1	Χρόνος φόρτωσης σελίδας	111
7.2.2	Συμβατότητα με κανόνες και πρότυπα	113
7.3	Διερευνητική μέθοδος συμπλήρωσης ερωτηματολογίων	113
7.4	Σύνοψη	117
8.	Σύνοψη	119
8.1	Εισαγωγή	119
8.2	Συμβολή.....	119
8.2	Μελλοντικές επεκτάσεις	122

Βιβλιογραφία.....	1244
-------------------	------

A.	Οδηγίες Πρόσβασης.....	A-1
A.1	Πρόσβαση στην εκπαιδευτική πύλη.....	A-1
A.2	Πρόσβαση στο υλικό της μεταπτυχιακής εργασίας	A-2
B.	Εγκατάσταση	B-1
B.1	Απαιτήσεις.....	B-1
B.1.2	Απαιτήσεις σε hardware:	B-1
B.1.3	Απαιτήσεις σε εγκατεστημένο λογισμικό	B-2
B.2	Εγκατάσταση και παραμετροποίηση προαπαιτούμενου λογισμικού	B-2
B.2.1	Java SE 1.6.21	B-2
B.2.2	Maven 3.0.3+	B-2
B.2.3	Ant 1.8.2+	B-3
B.2.4	Tomcat 6+	B-3
B.2.5	Μεταβλητές Περιβάλλοντος	B-5
B.2.6	Εγκατάσταση του uPortal	B-5
B.2.7	Επιλογή βάσης δεδομένων	B-6
B.3	Παραμετροποίηση αρχείων	B-6
B.3.1	Εγκατάσταση	B-7
B.4	Δημιουργία Διεπαφής.....	B-8
B.4.1	Γραφικό θέμα	B-8
B.4.2	Δημιουργία αρχείων μεταγλώττισης στα ελληνικά	B-9
B.5	Σύνδεση με καταλόγους δικτύου	B-10
B.6	Δημιουργία ρόλων	B-11
B.7	Υλοποίηση https-SSL	B-12
B.7.1	Μεταφορά πιστοποιητικού	B-13
B.7.2	Εισαγωγή στο tomcat:	B-13
B.8	Δημιουργία πιστοποιητικού	B-14
B.8.1	Δημιουργία νέου Αιτήματος Υπογραφής Πιστοποιητικού	B-15
Γ.	Επιστολές ανάθεσης έργου	Γ-1
Γ.1	Επιστολή αίτησης άδειας υλοποίησης έργου.....	Γ-1
Γ.2	Επιστολή έγκρισης ανάθεσης έργου.....	Γ-2

Κεφάλαιο 1

Εισαγωγή

Στα Ιδρύματα Τριτοβάθμιας Εκπαίδευσης (ΙΤΕ), όπου τυπικά συναντά κανείς δεκάδες τμήματα, εκατοντάδες καθηγητές και διοικητικό προσωπικό και χιλιάδες ενεργούς φοιτητές, υπάρχουν υπηρεσίες που διαμορφώνονται ανάλογα με την ιδιότητα εκείνου στον οποίο προσφέρονται και τη χρονική στιγμή κατά την οποία αιτούνται. Χαρακτηριστικό παράδειγμα ένας φοιτητής που θέλει να πληροφορείται με ειδήσεις που τον ενδιαφέρουν, θα πρέπει να διαβάζει καθημερινά τα νέα-ανακοινώσεις γενικότερου ενδιαφέροντος από τον κεντρικό ιστότοπο του ιδρύματος και τις πιο ειδικευμένες ανακοινώσεις από τις ιστοσελίδες της σχολής ή/και του τμήματος του. Επιπλέον, αν είναι μέλος μιας ή περισσότερων πολιτιστικών ομάδων (π.χ. την σκακιστική ή θεατρική λέσχη) θα πρέπει να επισκέπτεται και τους αντίστοιχους ιστοχώρους. Όταν ο φοιτητής είναι τελειόφοιτος (όπως αυτό ορίζεται κατά τους κανονισμούς του οικείου ΙΤΕ), τότε και μόνο τότε θα πρέπει να έχει πρόσβαση στον ιστότοπο του γραφείου που χειρίζεται τις πρακτικές ασκήσεις (internships). Αντιστοίχως, ο πτυχιούχος θα έχει πρόσβαση στις αγγελίες ευρέσεως εργασίας μετά την αποφοίτηση του. Τέτοιου είδους υπηρεσίες υποστηρίζονται πλέον από εφαρμογές Πληροφοριακών και Επικοινωνιακών Συστημάτων (ΤΠΕ) και συχνά είναι και διαδικτυακές.

Στο παραπάνω παράδειγμα είναι εμφανές, ότι για να ενημερώνεται ο φοιτητής σε ό,τι τον ενδιαφέρει, θα πρέπει να παρακολουθεί και να διαχειρίζεται ένα ικανό αριθμό διαδικτυακών εφαρμογών σε καθημερινή βάση. Ένας πιο αποδοτικός και λειτουργικός τρόπος θα ήταν να δίνεται η πρόσβαση σε αυτές τις υπηρεσίες μέσα από μια διαδικτυακή πύλη (portal), με εξατομικευμένο πληροφοριακό υλικό και υπηρεσίες που θα συγκεντρώνει ροές πληροφοριών από διάφορες διαδικτυακές εφαρμογές σύμφωνα με το προφίλ του χρήστη.

Η παρούσα διατριβή συζητά θέματα σχεδίασης και ανάπτυξης ενός πληροφοριακού συστήματος τύπου Portal και ως πεδίο εφαρμογής χρησιμοποιήθηκε το ΤΕΙ Κρήτης, ένα από τα μεγαλύτερα ανώτατα εκπαιδευτικά ιδρύματα της Ελλάδας,

Η δομή αυτού του κεφαλαίου είναι ως εξής. Στην ενότητα 1.2 περιγράφεται ο οργανισμός για τον οποίο σχεδιάστηκε η εφαρμογή. Τα κύρια και δευτερεύοντα κίνητρα για την εκπόνηση της συγκεκριμένης διατριβής του συντάκτη παρουσιάζονται στην ενότητα 1.3. Ο κύριος σκοπός και οι επιμέρους στόχοι της διατριβής αναλύονται στην ενότητα 1.4 ενώ η επόμενη ενότητα εξηγεί την ανάγκη που οδήγησε στην διατριβή αυτή. Η μεθοδολογική προσέγγιση που ακολουθήθηκε περιγράφεται με συντομία στην ενότητα 1.6 ενώ τα οφέλη της διατριβής συνοψίζονται στην ενότητα 1.7. Όπως και στα περισσότερα έργα, αντιμετωπίσαμε περιορισμούς που προέκυψαν από πίεση χρόνου και διοικητικές αγκυλώσεις. Οι περιορισμοί αυτοί εξηγούνται στην ενότητα 1.8. Τέλος, η ενότητα 1.9 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

1.2 Πλαίσιο Αναφοράς

Το Τ.Ε.Ι. Κρήτης περιλαμβάνει τέσσερεις Σχολές: 1) Σχολή Τεχνολογικών Εφαρμογών (Σ.Τ.ΕΦ), 2) Σχολή Διοίκησης και Οικονομίας (Σ.Δ.Ο.), 3) Σχολή Επαγγελματιών Υγείας & Πρόνοιας (Σ.Ε.Υ.Π.) και 4) Σχολή Τεχνολογίας Γεωπονίας (Σ.Τε.Γ.). Μέχρι πριν ένα χρόνο, υπήρχαν κάτω από το κεντρικό τομέα TEICRETE.GR διάφοροι άλλοι, όπως τομείς (κατάλογοι χρηστών) ανά σχολή (stef.teicrete.gr, sdo.teicrete.gr, steg.teicrete.gr, seyp.teicrete.gr), τομέας διοικητικών υπαλλήλων (admin.teicrete.gr), κάποιοι τομείς τμημάτων όπως epp.teicrete.gr & cs.teicrete.gr, ακόμα και γραφείων ή προγραμμάτων όπως το Γρ. Διασύνδεσης (career.teicrete.gr).

Ταυτόχρονα υπήρχαν διαδικτυακές υπηρεσίες, κάθε μια με τη δική της βάση χρηστών. Αυτό σημαίνει ότι ο χρήστης αυτής της υπηρεσίας συνδεόταν δίνοντας τα διαπιστευτήρια (username/password) που είχε ειδικά για αυτήν. Η πλειοψηφία των χρηστών κάθε φορά είχε

ανοίξει λογαριασμούς με διαφορετικά διαπιστευτήρια σε κάθε υπηρεσία με τη λογική της μεγαλύτερης ασφάλειας. Κάτι τέτοιο είχε το αυτονόητο πρόβλημα υποστήριξης: ο χρήστης προφανώς δεν μπορούσε να θυμάται τα διαπιστευτήρια αυτά ανά υπηρεσία οπότε καθημερινά το helpdesk του τμήματος πληροφορικής είχε λειτουργική επιβάρυνση με το να θυμίζει ή να επαναφέρει τα διαπιστευτήρια στους χρήστες που τα λησμονούσαν.

Πριν ένα χρόνο, τέθηκε ως στόχος η διασύνδεση των διαπιστευτηρίων όλων των συνδεδεμένων ιδρυμάτων (Shibboleth) και άρχισε η επίμονη και χρονοβόρα εργασία μεταφοράς των λογαριασμών των χρηστών σε ένα ενιαίο κεντρικό κατάλογο. Έτσι, έγινε ενοποίηση των τομέων των σχολών με το τομέα διοικητικών υπαλλήλων κάτω από ένα νέο τομέα `staff.teicrete.gr`. Κάποιες υπηρεσίες όπως το ηλ. ταχυδρομείο που είναι ίσως η πιο βασική ακολούθησαν, όπως ήταν λογικό, άμεσα την υποστήριξη του νέου τομέα.

Παραμένουν όμως ακόμα για διαφόρους λόγους κάποιοι ξεχωριστοί τομείς σε διάφορα τμήματα και γραφεία.

Το θέμα της ενοποίησης όλων των τομέων καθυστερεί υπερβολικά και ο λόγος είναι ότι στο ίδρυμα υπάρχουν δυο διαχειριστικές αρχές για τις υπηρεσίες πληροφορικής, το Κέντρο Ελεγχου & Διαχείρισης Δικτύων (ΚΕΕΔ) και το τμήμα πληροφορικής της διοίκησης (Τ.Π.Δ), το καθένα με διαφορετική φιλοσοφία, διαφορετικές υπηρεσίες και διαφορετικές τεχνολογίες καταλόγων δικτύου. Πιο συγκεκριμένα, το Τ.Π.Δ. ακολουθεί `microsoft-active directory`, ενώ το ΚΕΕΔ υιοθετεί `open source-linux & openldap`.

Η ενοποίηση των διαχειριστικών αρχών έχει αποφασιστεί να γίνει μελλοντικά αλλά αυτό δεν έπρεπε να εμποδίζει την διασύνδεση μας με το Shibboleth. Έτσι, αποφασίστηκε η χρήση ενός κοινού LDAP το οποίο θα ενημερώνεται καθημερινά από τους επιμέρους καταλόγους δικτύων και θα έχει κεντρικά τη βάση όλων των χρηστών του ιδρύματος, ώστε να μπορέσουμε να συνδεθούμε άμεσα με το Shibboleth.

Όμως, η διαδικασία ενοποιήσεων τομέων εγείρει κάποια προβλήματα. Η ενοποίηση τομέων προϋπόθετε διευθέτηση των λογαριασμών χρηστών, ώστε να υπάρχει ένας μοναδικός λογαριασμός για κάθε χρήστη. Αυτό δεν έχει ολοκληρωθεί μέχρι τώρα. Η ύπαρξη διαφορετικών λογαριασμών ανά χρήστη και η κάθε μια με διαφορετικό ρόλο δεν επιτρέπει την εύκολη υποστήριξη πολλαπλών ρόλων σε κάθε χρήστη της πύλης που προτείνουμε. Η αναλυτική περιγραφή της πύλης και της παρούσας διατριβής παρουσιάζονται στην επόμενη ενότητα.

1.3 Κίνητρα

Το βασικό κίνητρο για την μεταπτυχιακή διατριβή αυτή ήταν η επαγγελματική ιδιότητα του γράφοντος ως στέλεχος πληροφορικής του ΤΕΙ Κρήτης. Σε προσωπικό επίπεδο και σε όλο το διάστημα της επαγγελματικής μου καριέρας, αναζητούσα την ευκαιρία για ενασχόληση με νέα εργαλεία ανάπτυξης και τη συνεχή εκπαίδευση και εξοικείωση με τεχνολογίες αιχμής. Η παρούσα διατριβή υπόσχεται ανάπτυξη τέτοιων ικανοτήτων.

Επίσης, κύριο κίνητρο ήταν η προσδοκία της δημιουργίας καινοτόμων εφαρμογών που είναι χρήσιμες στο κοινωνικό σύνολο και η συνακόλουθη ικανοποίηση από την αναγνώριση.

Η διαφαινόμενη καλλιέργεια μιας συγκεκριμένης φιλοσοφίας ανάπτυξης, που βασίζεται, σε πρότυπα και προδιαγραφές που πρέπει να ακολουθηθούν για λόγους συμβατότητας και επεκτασιμότητας των εφαρμογών και η τεχνογνωσία για την ενσωμάτωση και αρμονική συνεργασία διαφόρων APIs για την λειτουργία της πύλης αποτελέσει κίνητρο, αρχικά, και σημαντική γνωστική κατάκτηση της συγκεκριμένης διατριβής.

1.4 Σκοπός και Επιμέρους Στόχοι

Η παρούσα διατριβή έχει ως σκοπό να αναλύσει, να σχεδιάσει και να υλοποιήσει μια διαδικτυακή πύλη (portal) που ενσωματώνει τις υπηρεσίες που λειτουργούν σε διαδικτυακό περιβάλλον και απαιτούν κωδικό πρόσβασης (μοναδικό για κάθε χρήστη) στο ΤΕΙ Κρήτης. Η προσωποποιημένη παροχή υπηρεσιών στους χρήστες του συστήματος επιτυγχάνεται μέσω της κατηγοριοποίησης των χρηστών σε ομάδες (user groups) και την ανάπτυξη των αντίστοιχων προφίλ (profile) ομάδας και χρηστών.

Ο παραπάνω κύριος στόχος κατευθύνει τους ακόλουθους επιμέρους στόχους:

1. Να ολοκληρωθεί βιβλιογραφική επισκόπηση σε θέματα και ανάλυσης, σχεδίασης και αξιολόγησης διαδικτυακών πυλών καθώς και των υποστηρικτικών τεχνολογιών τους.
2. Να καταγραφεί η παρούσα κατάσταση στο ΤΕΙ Κρήτης και να γίνει ανάλυση απαιτήσεων της νέας διαδικτυακής πύλης.

3. Να περιγραφούν τα εργαλεία ανάπτυξης και τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της.
4. Να μελετηθούν τα επιμέρους θέματα που θα επιτρέψουν την σχεδίαση και υλοποίηση της εφαρμογής.
5. Να εξεταστεί η υποστήριξη & εφαρμογή των διεθνώς αναγνωρισμένων κανόνων και οδηγιών προσβασιμότητας του W3C και συγκεκριμένα των Web Content Accessibility Guidelines (WAI/WCAG), ώστε να αποφεύγεται ο αποκλεισμός ειδικών ομάδων πολιτών (AMEA).
6. Να μελετηθεί η πολυκαναλική διάθεση των προσφερόμενων υπηρεσιών, όπου αυτό είναι δυνατόν, με προτεραιότητα στην προσαρμογή των προσφερόμενων υπηρεσιών για αποδοτική διάθεσή τους μέσω διαδικτύου, κινητού τηλεφώνου ή άλλων έξυπνων συσκευών.
7. Να σχεδιαστούν και να υλοποιηθούν οι μικροεφαρμογές (portlets) της πύλης.
8. Να μελετηθούν οι διαδικασίες ενοποίησης των διαφορετικών εφαρμογών υποδομής που θα χρησιμοποιηθούν (νέων και παλιών), κυρίως ως προς τον τρόπο επικοινωνίας τους με το κοινό περιβάλλον πρόσβασης κι εκτέλεσης που θα προσφέρει το portal.
9. Να γίνει αξιολόγηση της πύλης από τους ίδιους τους χρήστες της και να δοκιμαστεί σε ποιο βαθμό είναι συμβατή με διεθνώς αναγνωρισμένα πρότυπα σχεδίασης ιστοχώρων .

Οι παραπάνω επιμέρους στόχοι θα αναπτυχθούν στα επόμενα κεφάλαια αυτής της μεταπτυχιακής διατριβής. Πριν εστιάσουμε σε κάθε τέτοιο στόχο ξεχωριστά, είναι ενδιαφέρον να αναφερθούν οι διαφαινόμενες μεθοδολογικές προσεγγίσεις, τα αναμενόμενα αποτελέσματα και οι περιορισμοί που έχουν εντοπισθεί.

1.5 Αναγκαιότητα και Σπουδαιότητα

Η συγκεκριμένη υποδομή θεωρείται ιδιαίτερου ενδιαφέροντος στην ακαδημαϊκή κοινότητα (από την οποία εξάλλου έχει ξεκινήσει), διότι αποτελεί προαπαιτούμενο για την ενοποίηση υπηρεσιών που υποστηρίζονται εντός αλλά και μεταξύ των ακαδημαϊκών ιδρυμάτων, τόσο για

λειτουργίες της μορφής DEN (Directory Enabled Networking) όσο, κυρίως, για ενοποίηση διοικητικών και διαχειριστικών διαδικασιών. Εν δυνάμει, η διάχυση της ακολουθούμενης προσέγγισης στο σύνολο των ακαδημαϊκών ιδρυμάτων της χώρας συνεπάγεται ενιαία διαχείριση/μοντελοποίηση των χρηστών και ενοποίηση όλων των πηγών πληροφορίας για τους χρήστες σε κοινή βάση δεδομένων [33,34].

1.6 Μεθοδολογική Προσέγγιση

Το σύνολο των υποσυστημάτων που θα αναπτυχθούν στο παρόν υπόεργο θα βασιστούν σε λογισμικό ανοικτού κώδικα και σε τεχνολογίες και πρότυπα διαλειτουργικότητας (portlets, web services[60]), ώστε να είναι διαθέσιμα και να μπορούν να λειτουργήσουν με το δυνατόν λιγότερους τεχνολογικούς περιορισμούς. Στο project αυτό θα ακολουθήσουμε τα εξής βήματα:

- Ανάλυση απαιτήσεων / σύνταξη προδιαγραφών του συστήματος

Στη φάση αυτή, θα μελετηθούν λεπτομερώς οι ανάγκες που πρέπει να καλύψει η προσωποποιημένη πύλη. Για τον λόγο αυτό από την μελέτη απαιτήσεων θα πρέπει να προσδιοριστούν οι κατηγοριών χρηστών, ο κατάλογος προσφερόμενων εφαρμογών/υπηρεσιών και του περιεχομένου ανά κατηγορία χρηστών, να περιγραφεί η λειτουργικότητα του συστήματος και του τρόπου υποστήριξης των χρηστών.

- Σχεδιασμός του συστήματος

Στη φάση αυτή, θα γίνει ο σχεδιασμός της γενικότερης αρχιτεκτονικής του συστήματος, η περιγραφή του μηχανισμού πιστοποίησης χρηστών, ο προσδιορισμός του τρόπου ενσωμάτωσης, εμφάνισης και λειτουργίας των υπηρεσιών/εφαρμογών (microsites) στα αντίστοιχα περιβάλλοντα εργασίας. Επιπλέον, θα γίνει ο σχεδιασμός των διεπαφών για όλες τις κατηγορίες χρηστών και τους διαχειριστές.

- Ανάπτυξη μηχανισμού πρόσβασης στο σύστημα και ενσωμάτωση υπηρεσιών

Στη φάση αυτή, θα γίνει η δημιουργία του μηχανισμού πρόσβασης των χρηστών στις υπηρεσίες της πύλης. Η ταυτοποίηση χρηστών θα γίνεται σύμφωνα με την εγκατεστημένη υπηρεσία καταλόγου του Ιδρύματος, πιστοποίηση χρηστών σε επίπεδο ιδρύματος (σύστημα κεντρικής πιστοποίησης) και διασύνδεση με την αντίστοιχη υπηρεσία καταλόγου του ΕΔΕΤ (Shibboleth).

Επιπλέον, θα γίνει η ανάπτυξη του περιβάλλοντος εργασίας για κάθε κατηγορία χρηστών του συστήματος (φοιτητές, καθηγητές, προσωπικό) και δημιουργία βασικών υπηρεσιών προσωποποίησης όπως αλλαγή στοιχείων χρήστη και ενεργοποίηση επιθυμητών εφαρμογών/υπηρεσιών που θα εμφανίζονται στο περιβάλλον εργασίας (επιλογή από κατάλογο προσφερόμενων). Ακόμα θα γίνουν αλλαγές σε λογισμικό εφαρμογών/υπηρεσιών διαδικτύου που έχουν αναπτυχθεί και ήδη λειτουργούν στο ίδρυμα, ώστε να είναι δυνατή η ενσωμάτωση αυτών στο υπό κατασκευή σύστημα.

1.7 Οφέλη

Τα οφέλη από την ολοκλήρωση της διπλωματικής εργασίας είναι πολλαπλά.

Για την ακαδημαϊκή κοινότητα του ΤΕΙ Κρήτης, το έργο έχει θετική αναγνώριση και αποδοχή πριν, καν, ολοκληρωθεί και τεθεί σε λειτουργία. Η δημιουργία μιας εξατομικευμένης πύλης ως σημείο αναφοράς και εισόδου στο εκπαιδευτικό ίδρυμα απλοποιεί την αναζήτηση της επιθυμητής πληροφορίας. Η συγκέντρωση και ομαδοποίηση των υπηρεσιών ανάλογα το ρόλο του χρήστη ως καθηγητή, διοικητικό υπάλληλο ή φοιτητή βοηθά στην αποτελεσματικότερη πρόσβαση και την άμεση ενημέρωση των χρηστών για διαθεσιμότητα νέων υπηρεσιών στο μέλλον καθώς και γενικότερο καλύτερο έλεγχο πρόσβασης στις προσφερόμενες υπηρεσίες.

Ενδεικτικό στοιχείο της αποδοχής και αναγνώρισης της χρηστικότητας της πύλης στην πιλοτική εφαρμογή της είναι η θετικότερη αποδοχή της από το προσωπικό και τους φοιτητές του ιδρύματος. Ήδη, έχει δοθεί στόχευση από το Τμήμα Πληροφορικής της Διοίκησης για ανάπτυξη νέων διαδικτυακών υπηρεσιών αλλά και η τροποποίηση κάποιων παλαιότερων, ώστε να ενσωματωθούν στη διαδικτυακή πύλη μας.

Σε προσωπικό επίπεδο, η διατριβή αυτή έδωσε την ευκαιρία στον γράφοντα να ασχοληθεί και να γνωρίσει μια πληθώρα από εργαλεία και βιβλιοθήκες ανάπτυξης εφαρμογών και να εμβαθύνει τις γνώσεις του σε ένα ευρύ φάσμα θεμάτων. Η αποκτηθείσα γνώση και εμπειρία είναι πολύτιμη και αναμφίβολα χρήσιμη τόσο για μελλοντικά έργα όσο και ως τρόπος σκέψης και προσέγγισης σε σχέση με τη σχεδίαση διαδικτυακών εφαρμογών.

Ενδεικτικά, αναφέρουμε ότι με την ολοκλήρωση των υποχρεώσεων μας στο μεταπτυχιακό πρόγραμμα του ΑΠΚΥ, έχει δρομολογηθεί η ενεργή συμμετοχή μας στη περαιτέρω ανάπτυξη της

εκπαιδευτικής πύλης και της διασύνδεσης του ΤΕΙ Κρήτης με την ΕΔΕΤ μέσω Shibboleth. Επιπλέον, προσανατολιζόμαστε προς την αυτόνομη διαχείριση του κεντρικού LDAP και του CAS του ιδρύματος ώστε να ανεξαρτητοποιηθεί από την ανάγκη, μέχρι στιγμής, πληρωμής εξωτερικού συνεργάτη καθώς και η δημιουργία της νέας κεντρικής ιστοσελίδας του με ενσωμάτωση της λειτουργικότητας μιας πύλης. Η αναγνώριση του έργου μας είναι τελικά η μεγαλύτερη ηθική ανταμοιβή και ό,τι πιο σημαντικό.

1.8 Περιορισμοί

Στο συγκεκριμένο ΙΤΕ υπάρχουν δύο διαχειριστικές αρχές για τις υπηρεσίες πληροφορικής. Παρότι έχει αποφασιστεί η συγχώνευση τους, αυτή καθυστερεί. Κατά την διάρκεια της διατριβής, παρατηρήθηκε ακόμα και έλλειψη συνεργασίας για τροποποίηση υφιστάμενων υπηρεσιών ώστε να γίνουν συμβατές με το νέο καθεστώς.

Από τεχνικής άποψης, η διαδικασία ενοποιήσεων τομέων που προαναφέραμε, προϋπέθετε ένα ξεκαθάρισμα λογαριασμών χρηστών, ώστε να υπάρχει ένας μοναδικός λογαριασμός για κάθε χρήστη. Αυτό δεν έχει ολοκληρωθεί, μέχρι τώρα. Η ύπαρξη διαφορετικών λογαριασμών ανά χρήστη με επιπλέον χαρακτηριστικό ότι η κάθε μια έχει διαφορετικό ρόλο δεν επιτρέπει την εύκολη υποστήριξη πολλαπλών ρόλων σε κάθε χρήστη της πύλης που προτείνουμε. Την παρούσα στιγμή δεν είναι διαθέσιμος ο ρόλος του «τελειόφοιτου φοιτητή» που θα επέτρεπε την μεγαλύτερη εξατομίκευση υπηρεσιών.

Στη πορεία, πρόεκυψε και θέμα ενοποίησης με το πρόγραμμα διαχείρισης φοιτητών και μαθημάτων, του οποίου κάποιες λειτουργίες είναι διαθέσιμες διαδικτυακά. Η αρχική του σχεδίαση, δεν προέβλεπε ταυτοποίηση χρηστών μέσω κεντρικού συστήματος πιστοποίησης. Παρότι ζητήθηκε από τους προγραμματιστές του η ενσωμάτωση αυτής της δυνατότητας, αυτό δεν έγινε εξαιτίας φόρτου εργασίας τους.

Ταυτόχρονα, το εργαλείο ανάπτυξης, η πλατφόρμα πάνω στην οποία έγινε η ανάπτυξη της πύλης μας, πρόσφατα άλλαξε κύριο αριθμό έκδοσης. Αφενός, αυτό έχει ως αποτέλεσμα την αναβάθμιση και την παροχή νέων δυνατοτήτων, αφετέρου να υπόκειται σε μια σειρά από bugs που δυσκόλεψαν τόσο τη σχεδίαση όσο και της ανάπτυξης της πύλης. Ένα χαρακτηριστικό παράδειγμα ήταν η επιλογή της συνδεδεμένης βάσης δεδομένων στο σύστημα, αφού η αρχική

μας επιλογή (MySQL) δεν ήταν δυνατόν να υλοποιηθεί και καταφύγαμε σε μια άλλη (HSQLDB) που έχει περιορισμούς και δεν συνιστάται για τελικές εγκαταστάσεις.

Οι παραπάνω λόγοι και η μετέπειτα απόφαση του ιδρύματος για παράλληλη δημιουργία του νέου κεντρικού ιστοχώρου έχουν καθυστερήσει την ολοκλήρωση της διαδικασίας αξιολόγησης της πύλης και τα αποτελέσματα από τις απαντήσεις των χρηστών είναι ενδεικτικά λόγω περιορισμένου αριθμού συμπληρωμένων ερωτηματολογίων.

1.9 Σύνοψη

Το εισαγωγικό αυτό κεφάλαιο παρουσίασε σε γενικές γραμμές μια σειρά από θέματα που σχετίζονται με την εκπόνηση της διατριβής αυτής. Στην ενότητα 1.1 δόθηκε μια αδρή περιγραφή του προβλήματος. Στην ενότητα 1.2 παρουσιάστηκε ο οργανισμός για τον οποίο σχεδιάστηκε η εφαρμογή. Τα κύρια και δευτερεύοντα κίνητρα του συντάκτη περιγράφηκαν στην ενότητα 1.3. Ο κύριος σκοπός και οι επιμέρους στόχοι της διατριβής αναλύθηκαν στην ενότητα 1.4 ενώ η επόμενη ενότητα εξήγησε την ανάγκη που οδήγησε στην διατριβή αυτή. Η μεθοδολογική προσέγγιση που ακολουθήθηκε παρουσιάστηκε με συντομία στην ενότητα 1.6 ενώ τα οφέλη της διατριβής συνοψίζονται στην ενότητα 1.7. Όπως σε κάθε έργο, αντιμετωπίσαμε μια σειρά από περιορισμούς που προέκυψαν από πίεση χρόνου και διοικητικές αγκυλώσεις. Οι περιορισμοί αυτοί εξηγούνται στην ενότητα 1.8. Τέλος, στην παρούσα ενότητα παρουσιάζεται η σύνοψη του κεφαλαίου αυτού.

Κεφάλαιο 2

Βιβλιογραφική Επισκόπηση

2.1 Εισαγωγή

Στο κεφάλαιο αυτό θα γίνει καταγραφή της βιβλιογραφικής επισκόπησης σε θέματα ανάλυσης και σχεδίασης διαδικτυακών πυλών.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Στην ενότητα 2.2 γίνεται η παρουσίαση μιας διαδικτυακής πύλης (portal) ξεκινώντας με την ιστορική διαδρομή και εξέλιξη της, τη λειτουργικότητα που προσφέρει, κυρίως με την εξατομίκευση του πληροφοριακού υλικού και των υπηρεσιών και τελικά τα κέρδη που αποκομίζουμε. Στην ενότητα 2.3 περιγράφουμε το κύριο συστατικό-αντικείμενο μιας πύλης, το portlet και εστιάζουμε περισσότερο στη λειτουργικότητα και αρχιτεκτονική του και τις διαφορές του σε σχέση με ένα servlet. Η ενότητα 2.4 αναφέρεται στα τεχνολογικά πρότυπα για ανάπτυξη και συμβατότητα των portlets. Στην ενότητα 2.5 περιγράφεται η αρχιτεκτονική μιας διαδικτυακής πύλης. Οι επόμενες ενότητες 2.6 και 2.7 περιγράφουν τις κύριες μεθόδους πιστοποίησης ενός χρήστη μέσω LDAP και Shibboleth που θα χρησιμοποιηθούν και στην δικιά μας υλοποίηση. Η ενότητα 2.8 περιγράφει το πιο διαδεδομένο πρωτόκολλο ασφαλούς μεταφοράς δεδομένων μεταξύ εξυπηρετητών ιστοσελίδων και προγραμμάτων πλοήγησης του χρήστη στο διαδίκτυο. Η ενότητα 2.9 αναφέρεται στη κρυπτογράφηση και το πώς η πληροφορία μετατρέπεται από την αρχική της αναγνώσιμη

μορφή (plain-text) σε ακατάληπτα δεδομένα, με σκοπό την προστασία από υποκλοπή ενώ η ενότητα 2.10 περιγράφει τη μέθοδο επικύρωσης ταυτότητας του ίδιου του εξυπηρετητή ιστοσελίδων με τη χρήση πιστοποιητικών. Η ενότητα 2.11 αναφέρεται στις οδηγίες προσβασιμότητας του W3C δηλαδή των Web Content Accessibility Guidelines (WAI/WCAG), ώστε να αποφεύγεται ο αποκλεισμός ειδικών ομάδων πολιτών (AMEA). Στην ενότητα 2.12 γίνεται αναφορά στις μεθόδους αξιολόγησης ιστοχώρων. Τέλος η ενότητα 2.13 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

2.2 Portal (Διαδικτυακή Πύλη)

2.2.1 Εισαγωγή

Από λειτουργική σκοπιά, μια διαδικτυακή πύλη ενσωματώνει υπηρεσίες από ένα σύνολο ιστοσελίδων και πηγών και τις παρουσιάζει με ένα ενοποιημένο τρόπο. Συνήθως, κάθε πηγή πληροφοριών έχει το δικό της αποκλειστικό χώρο στην ιστοσελίδα για την εμφάνιση των δεδομένων με τη χρήση ενός portlet (συστατικό στοιχείο πύλης) και ο χρήστης μπορεί να ρυθμίσει ποια από αυτά τα στοιχεία θα εμφανίζονται[05].

Ένα portal εμφανίζει ταυτόχρονα στην ίδια ιστοσελίδα μια πληθώρα υπηρεσιών, όπως, για παράδειγμα μια τυπική μηχανή αναζήτησης, το ηλεκτρονικό ταχυδρομείο, ειδήσεις, τιμές μετοχών, πρόσβαση σε βάσεις δεδομένων κλπ. Οι διαδικτυακές πύλες παρέχουν ενοποιημένη σε εμφάνιση και αισθητική διεπαφή και κατόπιν έλεγχου ταυτοποίηση του χρήστη και πρόσβαση σε πολλαπλές εφαρμογές και βάσεις δεδομένων, οι οποίες διαφορετικά θα ήταν εντελώς διαφορετικές οντότητες και σε διαφορετικούς ιστοχώρους. Συνήθως έχουν δηλαδή το ρόλο του κεντρικού σημείου αναφοράς (gateway) όλων των υποστηριζόμενων ηλεκτρονικών υπηρεσιών, για αυτό και ο όρος portal δηλώνει πύλη, πόρτα, είσοδο [59].

Μια πύλη ορίζεται γενικότερα ως μια πλατφόρμα λογισμικού για την δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Σύγχρονες πύλες έχουν προσθέσει πολλαπλά χαρακτηριστικά που τις καθιστούν την καλύτερη επιλογή για ένα ευρύ φάσμα των εφαρμογών διαδικτύου. Μια πύλη υποστηρίζει χαρακτηριστικά όπως η εξατομίκευση, η συνάθροιση περιεχόμενου, ο έλεγχος γνησιότητας χρήστη, παραμετροποίηση και λειτουργεί ως μια βιβλιοθήκη από ταξινομημένο και εξατομικευμένο περιεχόμενο, κάτι σαν προσωπικό φίλτρο στο διαδίκτυο[08].

Τελικά, είναι ένα σύστημα που παρέχει τις λειτουργίες και τα χαρακτηριστικά για την αναγνώριση και την επικύρωση ταυτοποίησης των χρηστών ώστε να τους παρέχει, μέσω ρυθμιζόμενης από τον ίδιο το χρήστη, μιας κοινής διαδικτυακής διεπαφής (web interface) πρόσβαση σε εξατομικευμένες πληροφορίες και υπηρεσίες ανάλογα με τα ενδιαφέροντα, τα δικαιώματα και τις ανάγκες των χρηστών.

2.2.2 Η εξέλιξη των portals

Ο όρος πύλη (portal) αρχικά χρησιμοποιήθηκε με διαφορετική ερμηνεία: Στα μέσα της δεκαετίας του 90, αναφερόταν ως η αρχική σελίδα στα προγράμματα πλοήγησης ή ως αρχικό σημείο αναφοράς στις μηχανές αναζήτησης. Η πρακτικότητα της αναζήτησης πληροφοριών και ιστοσελίδων με τη πληκτρολόγηση απλών λέξεων και η εύρεση τους μέσα από μια βάση δεδομένων του δημοσιευμένου περιεχομένου στο διαδίκτυο, ήταν ό,τι χρειαζόταν ο κόσμος, για να ξεκινά τη πλοήγηση του με αυτές τις ιστοσελίδες. Παραδείγματα εκείνης της εποχής οι ιστοσελίδες Yahoo, Excite, AOL κλπ[20].

Στη συνέχεια, στις αρχές της δεκαετίας του 2000, ο όρος χρησιμοποιήθηκε για την υπηρεσία του καταλόγου, δηλαδή ιστοχώρων με περιεχόμενο ταξινομημένου ιεραρχικά σε κατηγορίες, αναζήτηση μιας ιστοσελίδας με ένα πιο δομημένο και οργανωμένο τρόπο και τελικά η εύρεση της ιστοσελίδας από τον ίδιο το χρήστη. Παράδειγμα εκείνης της εποχής το Forthnet Directory.

Ταυτόχρονα με την αλματώδη ανάπτυξη του Internet, υπήρξε ανάγκη διαχείρισης του πληροφοριακού υλικού ενός ιστότοπου, με δυνατότητες ταξινόμησης, εύκολης δημοσίευσης και συγκέντρωσης του από διάφορες εξωτερικές πηγές, συνήθως μέσω RSS feeds και τελικά εμφάνισης του σε μια μόνο ιστοσελίδα με εύκολο σε μορφοποίηση γραφικό περιβάλλον. Αυτό επιτυγχάνεται με Content Management Systems (CMS), που χρησιμοποιείται από την πλειοψηφία των ιστοτόπων σήμερα.

Η χρήση του όρου portal, σήμερα, είναι η εξέλιξη του CMS. Οι πύλες δεν είναι πλέον ιστόχωροι που υποστηρίζουν μια απλή δημοσίευση πληροφοριών, αλλά επιτρέπουν εξατομίκευση, κοινοποίηση και την ενσωμάτωση-ενοποίηση υπηρεσιών και πληροφοριών.

Με αυτή τη λογική είναι υλοποιημένα τα ειδησεογραφικά sites, όπως π.χ. in.gr, newsit.gr που εκτός από την ανάρτηση των νέων και την αναζήτηση πληροφοριών, παρέχουν υπηρεσίες προστιθέμενης αξίας, όπως το «περίπτερο» που μπορεί κάποιος να διαβάσει τους τίτλους των

εφημερίδων και περιοδικών, να ενημερωθεί για την πορεία των μετοχών στο χρηματιστήριο, ή να επικοινωνήσει με τρίτους χωρίς επιπλέον κόστος με υπηρεσίες μέσω ηλεκτρονικού ταχυδρομείου (email) ή σημείων δημόσιας συζήτησης (forum).

2.2.3 Λειτουργικότητα

Ο κύριος σκοπός ενός Portal είναι η συγκέντρωση μεγάλου όγκου πληροφοριών και υπηρεσιών, διαθέσιμων από πολλές πηγές και η διαθεσιμότητα τους στους χρήστες με αποδοτικό και αποτελεσματικό τρόπο, -"Many to many in an effective manner". Προκειμένου να γίνει κάτι τέτοιο, οι πληροφορίες και οι υπηρεσίες ομαδοποιούνται με βάση την κατανομή των υπηρεσιών ανά τμήμα και ομάδα χρηστών.

Μία πύλη μπορεί να υποστηρίξει τις περισσότερες από τις ακόλουθες λειτουργίες και χαρακτηριστικά[63]:

- Αναζήτηση πληροφοριών και υπηρεσιών με βάση την ιδιότητα και το ρόλο του χρήστη, συμπεριλαμβανομένων και εκείνων των πληροφοριών και υπηρεσιών που ενδεχομένως ο χρήστης να μην γνωρίζει καν ότι έχει πρόσβαση.
- Απλοποίηση της δημιουργίας και διαχείρισης σελίδων για κάθε τύπο τελικού χρήστη με σύστημα διαχείρισης όλου του κύκλου ζωής των ιστοσελίδων και έχοντας, ενδεχομένως, διάκριση του ρόλου του δημιουργού του περιεχομένου και αυτού που εγκρίνει αν το περιεχόμενο θα δημοσιευτεί ή όχι.
- Πολυκαναλική συνοχή, δηλαδή εμφάνιση του περιεχομένου τους σε πολλαπλές συσκευές (H/Y, κινητά τηλέφωνα tablets κλπ).
- Πρόσβαση και διαχείριση δεδομένων, δομημένων ή μη.
- Διαχείριση εργασιών και ροής εργασιών.
- Κατηγοριοποίηση πληροφοριών και των υπηρεσιών προσαρμοσμένη ειδικά στις ομάδες χρηστών του, σύμφωνα με τα κοινά χαρακτηριστικά των ομάδων (προφίλ), όπως οι ρόλοι και οι αρμοδιότητες τους, αλλά και τις ειδικές ανάγκες αυτών των ομάδων ή χρηστών.

- Υποστήριξη δημιουργίας και σύνδεσης με προσωπικό λογαριασμό.
- Παρουσίαση διαφορετικού περιεχομένου ανάλογα αν ο χρήστης έχει συνδεθεί με το λογαριασμό του ή όχι.
- Εμφάνιση των πληροφοριών και των υπηρεσιών που ένας χρήστης χρειάζεται σύμφωνα με το προφίλ του την κατάλληλη χρονική στιγμή.
- Εξατομικευμένη ρύθμιση του διαθέσιμου περιεχομένου και προσαρμογής της εμφάνισης του ανάλογα την ιδιότητα-ρόλο του χρήστη, τα ενδιαφέροντα, τη τοποθεσία και το πρόγραμμα πλοήγησης του επισκέπτη-χρήστη.
- Αποθετήριο εγγράφων (document repository). Όπως η ιστοσελίδα, έτσι και ένα έγγραφο μπορεί να δημοσιευτεί και να είναι διαθέσιμο σε χρήστες μόνο αν αυτοί διαθέτουν τα κατάλληλα δικαιώματα.
- Συνεργασία ομάδων χρηστών μέσω εφαρμογών, κοινοποίηση περιεχομένου και εγγράφων.
- Υποστήριξη της "Single-sign-on" λειτουργίας, δηλαδή της σύνδεσης του χρήστη με το λογαριασμό του μια φορά μόνο και δυνατότητα πρόσβασης σε διαφορετικές πηγές πληροφοριών και υπηρεσιών που υποστηρίζονται από διαφορετικά πληροφοριακά συστήματα.
- Ενσωμάτωση και ταυτόχρονη εμφάνιση τους στην ίδια ιστοσελίδα των εφαρμογών επιχειρηματικής ευφυΐας και λοιπών συστημάτων-εφαρμογών.
- Ασφάλεια δεδομένων, πρόσβασης και συναλλαγών.
- Ενσωμάτωση πληροφοριών από διαφορετικούς τύπους πηγών, για να μπορούν οι χρήστες να τις χρησιμοποιήσουν με το καλύτερο δυνατό τρόπο. Υπάρχουν αρκετοί μηχανισμοί, για να γίνει κάτι τέτοιο. Συνήθως τα portals παρέχουν τη δική τους διεπαφή προγραμματισμού εφαρμογών (Application Programming Interface - API) για δημιουργία ενοποιημένου περιεχομένου και επιτάχυνση της ανάπτυξης τέτοιων εφαρμογών, συνήθως διαθέσιμων για γλώσσες όπως JAVA, C++, ActiveX.

2.2.4 Εξατομίκευση πύλης

Στις πύλες το πιο βασικό χαρακτηριστικό τους είναι η εμφάνιση περιεχομένου ανάλογα αν ο επισκέπτης έχει συνδεθεί (είναι χρήστης-user) ή όχι (anonymous user/pages-authenticated user/pages).

Ένα παράδειγμα είναι η ιστοσελίδα μιας τράπεζας όπου ο επισκέπτης του ιστότοπου βλέπει ιστοσελίδες που περιγράφουν τις υπηρεσίες, τις προσφορές και τρόπους επικοινωνίας, σελίδες που είναι διαθέσιμες σε όλους, ενώ μετά τη σύνδεση με το λογαριασμό του πλέον έχει επιπλέον ενημέρωση σχετικά με τους λογαριασμούς ή/και τα δάνειά του.

Επιπλέον, οι πύλες απλοποιούν την ανάπτυξη ιστοχώρων και εμφανίζουν διαφορετικά δεδομένα ανάλογα με το ρόλο του χρήστη. Στο παραπάνω παράδειγμα μπορεί να υπάρχει διαφορετικό διαθέσιμο περιεχόμενο ανάλογα με το τύπο του πελάτη (ιδιώτης ή επιχείρηση).

Υπάρχουν διάφοροι τύποι εξατομίκευσης:

- Εξατομίκευση της πλοήγησης, συντομεύσεις για συγκεκριμένες πληροφορίες, γνωστές ως σελιδοδείκτες, ή τα αγαπημένα.
- Εξατομίκευση των δεδομένων/περιεχομένου, π.χ. ποιες μετοχές παρακολουθεί ο κάθε χρήστης στο πρόγραμμα μετοχών χρηματιστηρίου.
- Εξατομίκευση της διάταξης περιεχομένου, π.χ. τι είδους πληροφορία θα εμφανιστεί σε ποιο σημείο της οθόνης, σε τι μορφή, το χρώμα, το μέγεθος γραμματοσειράς.
- Εξατομίκευση ανάλογα με την ιδιότητα του χρήστη: η ιδιότητα του χρήστη καθορίζει την αρχική παραμετροποίηση αλλά ενδέχεται να τροποποιηθεί ανάλογα με τις επιλογές που θα κάνει ο ίδιος ο χρήστης.
- Εξατομίκευση πρόσβασης σε εφαρμογές, όπου οι χρήστες με βάση το ρόλο τους, έχουν άμεση πρόσβαση.
- Δημιουργία προσωπικών σελίδων από τους ίδιους τους τελικούς χρήστες με προσθήκη περιεχόμενου και προκαθορισμένων portlets. Επίσης, οι χρήστες μπορούν να ορίσουν ποιοι θα έχουν πρόσβαση στη σελίδα τους.

- Δυνατότητα για μεταγλώττιση του περιεχομένου τους ανάλογα τη γλώσσα που θα επιλέξει ο χρήστης.

2.2.5 Διαχωρισμός Portal

Ανάλογα τις απαιτήσεις και το σκοπό που εξυπηρετεί μια πύλη μπορεί να διαχωριστεί σε οριζόντια ή κάθετη πύλη[62].

Η οριζόντια πύλη (Horizontal Portal) είναι αυτή που έχει στόχο να εξυπηρετήσει μια μεγαλύτερη κατηγορία χρηστών ή ακόμα και ολόκληρη την κοινότητα του internet, ενώ η κάθετη πύλη (Vertical Portal) στοχεύει σε συγκεκριμένες κατηγορίες χρηστών.

Για παράδειγμα ένα ειδησεογραφικό site όπως το newsit.gr είναι οριζόντιου τύπου γιατί προσπαθεί να καλύψει τις ανάγκες όλων των επισκεπτών, ενώ το zappit.gr είναι κάθετου τύπου, βρίσκει κάποιος ειδήσεις που αφορούν μόνο μια κατηγορία, αυτή των ΜΜΕ.

2.2.6 Κέρδη από τη χρήση ενός portal

Επιχειρήσεις και οργανισμοί μπορεί να έχουν μια πλειάδα από εταιρικούς ιστόχωρους και διαδικτυακές εφαρμογές στις οποίες οι χρήστες έχουν πρόσβαση σε τακτά χρονικά διαστήματα. Για παράδειγμα, μια εταιρία μπορεί να έχει την κεντρική ιστοσελίδα, ιστοσελίδα για τους πελάτες, ιστοσελίδα για τους συνεργάτες της και ιστοσελίδα για το προσωπικό-υπαλλήλους της. Είναι προφανή τα ωφέληματα από ενδεχόμενη αντικατάσταση τους με μια πύλη που θα περιλαμβάνει ενοποιημένο όλο το περιεχόμενο και τις υπηρεσίες [63].

Η χρήση portals συντείνει στη μείωση λειτουργικών εξόδων στην ευκολότερη διαχείριση και συντήρηση πόρων καθώς και στην αύξηση της απόδοσης και της συνολικής παραγωγικότητας των ανθρώπινων πόρων. Ο χρήστης λειτουργεί κάτω από ένα κοινό ενοποιημένο γραφικό περιβάλλον και δεν είναι υποχρεωμένος να επισκεφτεί κάθε μια διαδικτυακή εφαρμογή σε διαφορετική διεύθυνση, αλλά από ένα ενιαίο σημείο εισόδου που έχει πρόσβαση στις υπηρεσίες της και έχει ταυτόχρονα την ευελιξία προσαρμογής και διαμόρφωσης της [28].

Η ενσωμάτωση μπορεί να περιλαμβάνει επίσης εξωτερικά συστήματα που δεν έχουν παρουσία μέσω διαδικτύου από τη φύση τους. Η δημιουργία portlets ενημέρωσης από διάφορα συστήματα back-end, όπως ενός ERP, ή μιας βάσης δεδομένων με τη μορφή dashboard που

εμφανίζουν αναφορές, στατιστικά, δείχνοντας την κατάσταση λειτουργίας τους και άλλες ειδοποιήσεις.

Οι πύλες διευκολύνουν τη δημιουργία ιστοσελίδων ή και ολόκληρων ιστοχώρων με την συναρμολόγηση διαφορετικών συστατικών στοιχείων πύλης (portlets).

Τα portals συνδυάζουν ένα θέμα παρουσίασης (μορφοποίηση και αισθητική ιστοχώρου) με το διαχωρισμό μια ιστοσελίδας σε μικρότερα κομμάτια όπως την κεφαλίδα το υποσέλιδο, μια συλλογή από σελίδες, μενού επιλογών και μια συλλογή από portlets και gadgets (μικροεφαρμογών). Οι διαχειριστές της πύλης μπορούν να κατασκευάσουν νέες σελίδες με τη επαναχρησιμοποίηση υπάρχοντων portlets & gadgets, να αναπτύξουν και να προσθέσουν νέα portlets κτλ.

Οι πύλες επιτρέπουν την ομαδική συνεργασία και οργάνωση ομάδων με τη δυνατότητα δημιουργίας ιστοχώρου με εφαρμογές συνεργασίας όπως ιστολόγιο (blogs), βιβλιοθήκη-εγκυκλοπαίδεια γνώσεων, ημερολόγιο, διαχείριση εργασιών, προειδοποιήσεων, ανταλλαγή εγγράφων.

Οι πύλες διευκολύνουν τη κοινωνική συνεργασία αφού μπορούν να παίξουν το ρόλο των υπηρεσιών κοινωνικής δικτύωσης με υπηρεσίες όπως βαθμολογία και ψηφοφορία, έγκριση ενεργειών, αποστολή μηνυμάτων, τοίχος δραστηριότητας φίλων και επιτρέπουν σε άτυπες ομάδες χρηστών να συνεργάζονται πιο εύκολα και γρήγορα.

2.3 Portlets

2.3.1 Εισαγωγή

Μια πύλη αποτελείται από πολλές ιστοσελίδες ακριβώς όπως μια διαδικτυακή εφαρμογή. Η διαφορά είναι ότι στις πύλες κάθε ιστοσελίδα αποτελείται από μια συγκεκριμένη διάταξη μικροεφαρμογών και ο χρήστης μπορεί να προσθέσει ή να αφαιρέσει μέσα από μια συλλογή που έχει στη διάθεση της η πύλη. Αυτές οι μικροεφαρμογές ονομάζονται portlets.

Portlet ή συστατικό στοιχείο πύλης είναι ένα κομμάτι κώδικα που αποτελεί μια διεπαφή και παρέχει ειδικό περιεχόμενο ενώ μπορεί να προστεθεί ή να αφαιρεθεί κατά επιθυμία του χρήστη.

Τα portlets παρέχουν τη λειτουργικότητα διασύνδεσης και πρόσβασης των χρηστών σε διαφορετικές εφαρμογές ή πηγές δεδομένων και δημιουργούν διακριτά τμήματα σε κάθε σελίδα για εμφάνιση του περιεχομένου τους. Αυτό επιτυγχάνεται συνήθως εξαιτίας της κύριας ευθύνης τους για τη δημιουργία του απαραίτητου κώδικα σε html, XML ή WML ο οποίος εμφανίζεται μέσα σε ένα παράθυρο (portlet window).

Μια πύλη, συνήθως, εμφανίζει πολλαπλά portlets σε διακριτά παράθυρα, το καθένα με το δικό του τίτλο, τρόπο αλλαγής εμφάνισης, προτιμήσεων του χρήστη, την μεγιστοποίηση ή ελαχιστοποίηση ή ακόμα και την αφαίρεση του. Αυτές οι επιλογές μπορούν να διαφέρουν ανάλογα τα δικαιώματα του χρήστη και τις ρυθμίσεις του portlet στη πύλη

Ένα portlet συνήθως δίνει πρόσβαση σε ορισμένες βασικές δυνατότητες μιας εφαρμογής και ταυτόχρονα τη δυνατότητα για μετάβαση στη διεύθυνση της ιστοσελίδας της για πρόσβαση στις υπόλοιπες. Για παράδειγμα, μια μικροεφαρμογή ηλ. ταχυδρομείου μπορεί να επιτρέπει την ανάγνωση των μηνυμάτων, αλλά όχι την δημιουργία νέου. Σε αυτή τη περίπτωση, θα έχει τη δυνατότητα να μεταβαίνει ο χρήστης στο κανονικό πρόγραμμα ηλ. ταχυδρομείου. Στο πλαίσιο αυτό, οι μικροεφαρμογές δεν μπορούν να είναι υποκατάστατο των διαδικτυακών εφαρμογών αλλά να προσθέτουν ή να συμπληρώνουν στη λειτουργικότητα τους.

2.3.2 Λειτουργικότητα

Το συστατικό στοιχείο portlet ενεργεί ως μια παραθυρική εφαρμογή, σε μια πύλη το κάθε portlet έχει το δικό του παράθυρο και ο συνδυασμός αυτών των παραθύρων δημιουργεί τις ιστοσελίδες του portal. Κάθε παράθυρο στην ιστοσελίδα του portal αποτελεί και ένα portlet

Χαρακτηριστικό παράδειγμα πύλης είναι το iGoogle στη διεύθυνση <http://www.google.com/ig> όπου εμφανίζεται συλλογή από παράθυρα με το καθένα παρουσιάζει κάτι διαφορετικό, όπως ηλεκτρονικό ταχυδρομείο, κυριότερες ειδήσεις από ενημερωτικούς ιστοχώρους, video από το youtube κτλ. Τα παράθυρα αυτά, δηλαδή η κάθε μια ξεχωριστή μικροεφαρμογή, μπορεί να εξατομικευτούν από τους χρήστες και τις προτιμήσεις τους, π.χ. πόσα μηνύματα εισερχομένης αλληλογραφίας να εμφανίζει η εφαρμογή ηλεκτρονικού ταχυδρομείου, από ποιες ιστοσελίδες να διαβάζονται τα RSS τροφοδοσίας κτλ. Οι χρήστες έχουν τη δυνατότητα να σύρουν ένα παράθυρο και να το τοποθετήσουν σε οποιοδήποτε σημείο της οθόνης. Μπορούν να προσθέσουν περισσότερα portlets και να αφαιρέσουν αυτά που δεν τους ενδιαφέρουν. Το τελικό αποτέλεσμα είναι να παίρνει ο χρήστης μια ενοποιημένη προβολή των πληροφοριών από

διαφορετικές πηγές. Είναι σαν μια τηλεόραση που ταυτόχρονα βλέπεις εικόνα από πολλά κανάλια μαζί [17].

Τα portlets είναι δυναμικά δηλαδή μπορούν να αλληλεπιδρούν μεταξύ τους ώστε να δείχνουν πληροφορίες σε απάντηση ενεργειών του χρήστη. Για παράδειγμα, σε μια μετεωρολογική πύλη όταν ο χρήστης αλλάξει τη πόλη σε ένα portlet τοποθεσίας, σε ένα άλλο portlet πρόγνωσης καιρού θα εμφανίζονται οι πληροφορίες για αυτή τη συγκεκριμένη πόλη. Αυτό έρχεται σε αντίθεση με το συνήθη τρόπο επικοινωνίας μεταξύ ξεχωριστών διαδικτυακών εφαρμογών που είναι με τη απευθείας χρήσης των βάσεων δεδομένων τους και το αποτέλεσμα αυτό δεν είναι άμεσα αντιληπτό στο χρήστη και πρέπει να συνδεθεί ξεχωριστά στις εφαρμογές.

Σε κάθε δεδομένη χρονική στιγμή, όλα τα συστατικά στοιχεία της πύλης χαρακτηρίζονται με βάση δύο μοναδικά χαρακτηριστικά, τι εμφανίζει αυτή τη στιγμή και πόσο εμβαδόν της συνολικής ιστοσελίδας καλύπτει. Η λεγόμενη κατάσταση (mode) λειτουργίας του συστατικού στοιχείου πύλης καθορίζει τι εμφανίζει την παρούσα στιγμή. Επιπλέον η κατάσταση του παράθυρου στο οποίο ανήκει καθορίζει πόσο χώρο καλύπτει σε μια συγκεκριμένη ιστοσελίδα.

Για παράδειγμα, θα μπορούσαμε να είχαμε γράψει ένα μήνυμα βοήθειας και να εμφανίζεται όταν ένας χρήστης κάνει κλικ στο κουμπί «Βοήθεια». Στην περίπτωση αυτή, το portlet είναι στην κατάσταση βοήθειας. Επίσης, θα μπορούσε να εμφάνιζε αναλυτικές οδηγίες εφόσον το παράθυρο του ήταν σε κατάσταση μεγιστοποίησης. Βλέπουμε, λοιπόν, ότι το portlet υποστηρίζει μηχανισμούς για το τι θα εμφανίζει κάθε στιγμή ανάλογα με την κατάσταση στην οποία βρίσκεται αυτό και το παράθυρο του. Για αυτό, είναι απαραίτητο η πύλη να γνωρίζει ποιες είναι οι διαθέσιμες αυτές καταστάσεις του κάτι που γίνεται κυρίως παραμετρικά από την καταγραφή τους μέσα στα αρχεία παραμετροποίησης που συνοδεύουν το κάθε portlet.

2.3.3 Αρχιτεκτονική

Τα portlets έχουν τη παρακάτω λειτουργικότητα:

- Δυνατότητα αποθήκευσης των προτιμήσεων, τα portlets παρέχουν ένα αντικείμενο (PortletPreferences) για την αποθήκευση των προτιμήσεων των χρηστών. Αυτές οι προτιμήσεις αποθηκεύονται σε ένα αποθετήριο δεδομένων συνήθως μια βάση δεδομένων. Με τον τρόπο αυτό είναι διαθέσιμες και μετά την επανεκκίνηση του

διακομιστή ή της πύλης. Επιπλέον, οι προγραμματιστές δεν ασχολούνται καθόλου για τον τρόπο με τον οποίο αποθηκεύονται.

- Επεξεργασία αίτησης: τα portlets παρέχουν ένα εκλεπτυσμένο χειρισμό αιτήματος. Ένα portlet μπορεί να δεχτεί μια αίτηση όταν ο χρήστης εκτελεί μια ενέργεια σε αυτό ή επειδή ο χρήστης ανέλαβε δράση σε κάποιο άλλο συστατικό στοιχείο πύλης και η ιστοσελίδα πρέπει να ανανεωθεί. Ο Portal Server παρέχει διαφορετικούς μεθόδους για το χειρισμό κλήσης των δύο περιπτώσεων.
- Καταστάσεις λειτουργίας (mode) portlets. Τα portlets χρησιμοποιούν την έννοια της κατάστασης με σκοπό να αναγνωρίσουν τι ο χρήστης θέλει να κάνει. Μια εφαρμογή ηλεκτρονικού ταχυδρομείου μπορεί να χρησιμοποιηθεί για την ανάγνωση, τη σύνθεση, τον έλεγχο μηνυμάτων ηλεκτρονικού ταχυδρομείου, αυτή είναι η αναμενόμενη λειτουργικότητα μιας εφαρμογής ηλεκτρονικού ταχυδρομείου. Τα portlets παρέχουν, σε κανονικές συνθήκες, αυτή τη λειτουργικότητα σε κατάσταση προβολής (View). Αλλά, υπάρχουν και άλλες λειτουργίες, όπως ο καθορισμός ενός χρόνου ανανέωσης ή εισαγωγή διαπιστευτηρίων όπως το όνομα χρήστη και ο κωδικός πρόσβασης. Οι δραστηριότητες αυτές επιτρέπουν στον χρήστη να ρυθμίσει τη συμπεριφορά της εφαρμογής, σε αυτή τη περίπτωση το portlet βρίσκεται σε κατάσταση τροποποίησης (Edit). Η εμφάνιση των οδηγιών χρήσης της εφαρμογής ηλεκτρονικού ταχυδρομείου γίνεται σε κατάσταση βοήθειας (Help). Η χρήση της κατάστασης λειτουργίας δεν αποτελεί καινοτομία, αλλά ο διακριτός διαχωρισμός διευκολύνει τους προγραμματιστές στην διαδικασία ανάπτυξης και την πύλη στη διαχείριση του κάθε portlet. Επιτρέπει σε έναν διαχειριστή της πύλης να ορίσει την πρόσβαση στο κάθε portlet σε επίπεδο χρήστη και κατάσταση λειτουργίας. Για παράδειγμα, ένας χρήστης της εφαρμογής ηλεκτρονικού ταχυδρομείου έχει νόημα να έχει δικαίωμα να ορίσει το όνομα χρήστη και τον κωδικό πρόσβασής του, άρα να έχει πρόσβαση στο portlet όταν λειτουργεί σε κατάσταση edit. Όμως, σε μια άλλη περίπτωση όταν το portlet εμφανίζει ενημερώσεις ειδήσεων επιτρέπει, μεν, στο χρήστη να καθορίσει τη διεύθυνση URL από όπου μπορεί να ανακτήσει ενημερώσεις, αλλά σε αυτή την αλλαγή πρέπει να προχωρεί ο διαχειριστής της πύλης και όχι ο κάθε χρήστης. Ταυτόχρονα, η ιστοσελίδα γίνεται πιο ελκυστική για τον τελικό χρήστη, επειδή τα portlets θα λειτουργούν σε ένα ομοιογενές περιβάλλον εργασίας. Αν θέλει να διαβάσει τις πληροφορίες βοήθειας σχετικά με οποιαδήποτε από τις εφαρμογές, μπορεί να κάνει κλικ στο κουμπί «Βοήθεια». Θα γνωρίζει, επίσης, ότι κάνοντας κλικ στο κουμπί «Επεξεργασία»

θα έχει πρόσβαση στην οθόνη ρυθμίσεων του portlet. Η τυποποίηση της κοινής επαφής χρήστη σε όλα τα portlets κάνει χρήση τους πιο ελκυστική και αποδοτική.

- Η κατάσταση (state) παράθυρου καθορίζει πόσο χώρος πρέπει να δοθεί στο περιεχόμενο που παράγεται από το portlet σε μια σελίδα του portal. Πατώντας το κουμπί μεγιστοποίησης, το portlet θα καταλάβει όλη την οθόνη και θα γίνει το μοναδικό portlet που θα είναι στη διάθεση του χρήστη. Σε κατάσταση ελαχιστοποίησης, το portlet θα πρέπει να εμφανίζεται μόνο ως μια γραμμή τίτλου. Έτσι οι προγραμματιστές μπορούν να προσαρμόσουν το περιεχόμενό του portlet με βάση το χώρο στη διάθεσή τους.
- Πληροφορίες για τον χρήστη, συνήθως τα portlets παρέχουν εξατομικευμένο περιεχόμενο στο χρήστη που υποβάλλει ένα αίτημα. Προκειμένου να γίνει αυτό αποτελεσματικά, μπορούν να ζητήσουν πρόσβαση στα χαρακτηριστικά-ιδιότητες του χρήστη, όπως όνομα, email, τηλέφωνο, κλπ. Το API των portlets δίνει αυτή τη δυνατότητα. Ένας προγραμματιστής μπορεί να έχει πρόσβαση σε αυτά τα χαρακτηριστικά με ένα τυποποιημένο τρόπο και είναι η ευθύνη του διαχειριστή της πύλης να ορίσει αυτές τις ιδιότητες του χρήστη σε ένα κατάλογο δικτύου (συνήθως ένα διακομιστή LDAP).

Στην αρχιτεκτονική προσανατολισμένων υπηρεσιών (service-oriented architecture - SOA)[50] είναι δυνατή η ανάπτυξη εφαρμογών από υπάρχουσες υπηρεσίες. Επειδή όσα portlets είναι συμβατά με τη 2.0 έκδοση των προδιαγραφών μπορούν να επικοινωνούν μεταξύ τους σε επίπεδο διεπαφής παίζουν βασικό ρόλο στην ανάπτυξη τέτοιων εφαρμογών με τη χρήση γεγονότων και δημοσίων παραμέτρων απόδοσης. Το δοχείο (container) των portlets είναι υπεύθυνο για τον χειρισμό της επικοινωνίας μεταξύ των portlet. Αυτή η σύνδεση μεταξύ portlets είναι μόνο σε επίπεδο παρουσίασης. Οι υπηρεσίες που εκπροσωπούν δεν επικοινωνούν απευθείας μεταξύ τους αλλά μόνο μέσω του επιπέδου παρουσίασης του που παρέχουν τα portlets.

2.3.4 Διαφορές και ομοιότητες με τα servlets

Η βασική διαφορά των portlets σε σχέση με τα servlets, είναι ότι το portlet είναι υπεύθυνο μόνο για το κώδικα ενός τμήματος και όχι ολόκληρης της ιστοσελίδας όπως τα servlets. Την τελική ευθύνη για την σύνθεση και εμφάνιση όλης της ιστοσελίδας σε μια πύλη έχει ο διακομιστής.

Αυτή η διάκριση ευθυνών επιτρέπει στο portlet να εστιάσει στη δημιουργία της δικής του διεπαφής [17].

Στη δημιουργία διαδικτυακών εφαρμογών με τη χρήση servlets, ο web container παρέχει το περιβάλλον που διαχειρίζεται και εξυπηρετεί τα servlets και ο διακομιστής του web είναι υπεύθυνος για τη αποστολή της ιστοσελίδας στο πρόγραμμα πλοήγησης. Παρομοίως τα portlets τα διαχειρίζεται ο portlet container, τα μορφοποιεί και έτσι συνθέτει το περιεχόμενο των ιστοσελίδων της πύλης και στη συνέχεια ο διακομιστής στέλνει τις σελίδες στο πρόγραμμα πλοήγησης του χρήστη.

Η βασική ομοιότητα με τα servlets είναι ότι τα portlets αναπτύσσονται μέσα σε ένα δοχείο (container) που παράγει το δυναμικό περιεχόμενο. Από τεχνικής πλευράς, ένα portlet είναι μια κλάση που υλοποιεί το interface javax.portlet, κατόπιν συσκευάζεται σε ένα αρχείο .WAR και στη συνέχεια εγκαθίσταται στο portlet container.

Τα portlets είναι παρόμοια με τα servlets στο ότι:

- Τα διαχειρίζεται ένα εξειδικευμένο δοχείο (container).
- Παράγουν δυναμικό περιεχόμενο.
- Έχουν συγκεκριμένο κύκλο ζωής.
- Αλληλεπιδρούν με τον πελάτη-πρόγραμμα μέσω της μορφής αίτημα & απάντησης (request & respond).

Τα portlets είναι διαφορετικά από τα servlets στο ότι:

- Παράγουν μόνο τμήματα του κώδικα της ιστοσελίδας και όχι ολόκληρες ιστοσελίδες.
- Δεν είναι άμεσα προσπελάσιμα μέσω διεύθυνσης URL, αλλά μόνο μέσω της διεύθυνσης URL της ιστοσελίδας της πύλης που περιέχει το συγκεκριμένο portlet.
- Δεν μπορούν να ορίσουν το τύπο του περιεχομένου που θα δημιουργήσουν, δεδομένου ότι το περιεχόμενο που παράγεται είναι μέρος της πύλης της σελίδας. Εάν ένας διακομιστής πύλης ζητά html/text, τότε όλα τα portlets θα πρέπει να δημιουργήσουν

html/text περιεχόμενο, αν ο διακομιστής πύλης ζητά WML, τότε το κάθε portlet πρέπει να παράγει περιεχόμενο σε μορφή WML.

2.4 JSRs-Java Specification Requests

Η κοινότητα της Java γνωστή και σαν Java Community Process (JCP) υποστηρίζει τη διαδικασία ανταλλαγής προτάσεων για βελτίωση και προσθήκη νέων δυνατοτήτων και καταλήγει στη δημοσίευση των προτύπων και προδιαγραφών ανάπτυξης, τα γνωστά Java Specification Requests (JSR) που πρέπει να ακολουθούν όλοι για λόγους συμβατότητας.[42]

Όπως όφειλε, η κοινότητα έχει ορίσει συγκεκριμένες προδιαγραφές που ορίζουν τρόπο ανάπτυξης των διαδικτυακών εφαρμογών. Οι προδιαγραφές ειδικότερα των java portlets ορίζουν στη πραγματικότητα κάτι σαν συμβόλαιο μεταξύ των portlets και του java container, δηλαδή του περιβάλλοντος που τρέχουν και παρέχουν το προγραμματιστικό μοντέλο για την ανάπτυξη των portlets.

Η πρώτη προδιαγραφή v1.0 τυποποιήθηκε και εκδόθηκε τον Οκτώβριο του 2003 και είναι γνωστή με τη κωδική ονομασία JSR 168. Σε αυτή τη πρώτη προδιαγραφή έγινε η περιγραφή του βασικού μοντέλου ανάπτυξης portlets με κύρια χαρακτηριστικά:

- Ξεχωριστές και ανεξάρτητες διαδικασίες ανάπτυξης για την επεξεργασία των δεδομένων και την δημιουργία της διεπαφής με υποστήριξη του μοντέλου Model-View-Controller (MVC).
- Το κάθε portlet υποστηρίζει διάφορες καταστάσεις (modes) επιτρέποντας τη μερική επικοινωνία μεταξύ πύλης και portlet αφού ανάλογα την κλήση που λαμβάνει το portlet και την κατάσταση λειτουργίας που ζητά η πύλη, το portlet εκτελεί τις σχετικές εργασίες και γνωρίζει τι περιεχόμενο τελικά θα παρουσιάσει.
- Καταστάσεις παραθύρων, που ορίζουν το μέγεθος σε αναλογία με ολόκληρη τη σελίδα της πύλης θα καταλάβει το portlet.
- Μοντέλο δεδομένων, επιτρέπει την αποθήκευση και ανάκτηση πληροφοριών και παραμέτρων που χρησιμοποιεί το portlet για την εμφάνιση του περιεχομένου του, τη διαχείριση της περιόδου λειτουργίας του και τις προτιμήσεις του κάθε χρήστη.

- Συγκεκριμένο τρόπο συσκευασίας των portlets και άλλων αντικειμένων, απαραίτητων για την εγκατάστασή τους σε ένα διακομιστή πύλης.

Στη συνέχεια η προδιαγραφή v2.0 τυποποιήθηκε τον Ιούνιο του 2008 και αναπτύχθηκε παράλληλα με την ανανεωμένη έκδοση 2.0 του WSRP (Web Services for Remote Portlets) και καλύπτει όλες τις ελλείψεις που είχαν παρατηρηθεί στη πρώτη προδιαγραφή, αντιμετωπίζει θέματα λειτουργικότητας και φυσικά είναι συμβατή με τη πρώτη με βασικές δυνατότητες:

- Επικοινωνία μεταξύ portlets μέσω γεγονότων και δημοσίων παραμέτρων απεικόνισης
- Εισαγωγή των φίλτρων και των ακροατών (listeners)
- Εξυπηρέτηση δυναμικά δημιουργούμενων πόρων απευθείας από τα portlets
- Υποστήριξη ασύγχρονης Javascript AJAX και δεδομένων JSON απευθείας από τα portlets

2.4.1 Web Services for Remote Portlets

Διαδικτυακές Υπηρεσίες απομακρυσμένων portlets (Web Services for Remote Portlets -WSRP) είναι το εγκεκριμένο πρότυπο για ένα πρωτόκολλο δικτύου που προορίζεται για την επικοινωνία με απομακρυσμένα portlets. Επιτρέπει την ενσωμάτωση portlets που έχουν σχεδιαστεί για άλλα πλαίσια Portal Servers και την ολοκλήρωση διαφορετικών υπηρεσιών [61].

Το WSRP κύριο στόχο έχει την επίτευξη της διαλειτουργικότητας των portlets. Το WSRP βρίσκεται πάνω από την υπάρχουσα στοίβα υπηρεσίας Web, χρησιμοποιώντας WSDL για τον καθορισμό ενός συνόλου διεπαφών. Τυποποιεί τις διεπαφές προγραμματισμού εφαρμογών (API) μεταξύ των καταναλωτών και των δημιουργών των portlets, το πρωτόκολλο επικοινωνίας, και ορισμένες πτυχές του μοντέλου των εξαρτημάτων όπως τους τρόπους και τις περιγραφές εξατομίκευσης[02].

2.5 Υποδομή πύλης

Η υποδομή μιας πύλης που αποτελείται από το container των portlets, τον διακομιστή της πύλης, το ενιαίο αισθητικά περιβάλλον διεπαφής, τη διαχείριση όλου του κύκλου ζωής ενός portlet, τη

προσωρινή αποθήκευση του περιεχομένου (content caching), την παρεχομένη ασφάλεια και υπηρεσία ταυτοποίησης χρήστη είναι αυτή τελικά που δίνει όλα τα πλεονεκτήματα της.

2.5.1 Το δοχείο του συστατικού στοιχείου πύλης (portlet container)

Ένα portlet σε μια ιστοσελίδα πύλης αντιπροσωπεύεται από ένα στιγμιότυπο (instance) του. Στις ευθύνες του portlet container, συγκαταλέγονται η διαχείριση του στιγμιότυπων και η δημιουργία των κώδικα συνήθως σε html που αντιστοιχεί στα portlets που συνθέτουν κάθε ιστοσελίδα της πύλης.

Ο portlet container είναι υπεύθυνος επίσης για την κλήση μεθόδων ανάλογα στη φάση του κύκλου ζωής που βρίσκεται το κάθε portlet και παρέχει το απαιτούμενο περιβάλλον λειτουργίας. Τα portlets έχουν πρόσβαση στο περιβάλλον λειτουργίας τους χρησιμοποιώντας το PortletContext αντικείμενο (παρόμοια με ServletContext), έχοντας τη δυνατότητα να μοιραστούν πληροφορίες.

Ο portlet container πρέπει να είναι συμβατός με τις απαιτήσεις που προβλέπονται από τις προδιαγραφές του portlet ώστε να λειτουργούν και να μπορούν να μεταφερθούν σε άλλο container χωρίς να χάσουν τη λειτουργικότητα τους.

Τα περισσότερα containers παρέχουν επιπλέον δυνατότητες σε σχέση με τις απαιτήσεις που ορίζονται στις προδιαγραφές όπως δυνατότητα προσαρμοσμένων από το χρήστη καταστάσεων λειτουργίας και είναι η επιλογή του προγραμματιστή αν θα χρησιμοποιήσει αυτά τα χαρακτηριστικά ή όχι. Χρησιμοποιώντας αυτές τις επεκτάσεις στη δημιουργία ενός portlet το πιθανότερο θα προκαλέσει τη μη φορητότητα του σε άλλο container αλλά η χρήση αυτών των δυνατοτήτων συνήθως μειώνει το χρόνο ανάπτυξης των portlets, αφού απλοποιεί το προγραμματισμό τους.

2.5.2 Ο διακομιστής πύλης (portal server)

Ο διακομιστής πύλης είναι υπεύθυνος για την υποβολή των αιτήσεων των χρηστών που έλαβε από την σελίδα της πύλης στο portlets container, για τη συγκέντρωση και τις απαντήσεις που παράγονται από αυτά. Ο portal server είναι υπεύθυνος για τη δημιουργία της ιστοσελίδας της πύλης και της πρόσβασης από το χρήστη μέσα από μια ενιαία εμφάνιση και διεπαφή.

Το portal server μπορεί να θεωρηθεί ως ένα στοιχείο που βρίσκεται μεταξύ των αιτημάτων των χρηστών προς την ιστοσελίδα της πύλης και του portlet container. Ο portlet container δεν είναι υπεύθυνος για τη δημιουργία της ιστοσελίδας. Παραδίδει το περιεχόμενο που παράγεται από τα portlets στο Portal Server, ο οποίος συνθέτει το περιεχόμενο και εμφανίζει τη τελική ιστοσελίδα της πύλης. Υπάρχουν πολλοί portal servers ανοιχτού κώδικα (όπως το Liferay Portal και το GateIn Portal) αλλά και αδειοδοτικά (όπως το WebSphere Portal της IBM) και παρέχουν ένα ευρύ φάσμα χαρακτηριστικών: διαχείριση πύλης, διαχείριση περιεχομένου, αναζήτηση, υποστήριξη single sign-on σύνδεσης. Η χρησιμοποίηση των portal servers διευκολύνει τους προγραμματιστές να δημιουργήσουν γρήγορα μια πλήρως λειτουργική διαδικτυακή πύλη. Η επιλογή του Portal Server διαδραματίζει σημαντικό ρόλο στην υλοποίηση της διαδικτυακής πύλης, επειδή μπορεί να βοηθήσει στη μείωση του χρόνου ανάπτυξης με τη παροχή ενσωματωμένων portlets, και τη διασύνδεση με εξωτερικά συστήματα όπως LDAP.

2.6 LDAP

Ένας κατάλογος είναι ένας τύπος μιας βάσης δεδομένων σχεδιασμένη να περιέχει δεδομένα που να επιτρέπουν στους χρήστες να βρίσκουν αντικείμενα και δεδομένα που έχουν συσχετιστεί με αυτούς. Η δομή του καταλόγου επιτρέπει να περιέχει πληροφορίες μικρές σε μέγεθος και όχι συχνά τροποποιήσιμες.

Οι κατάλογοι χρησιμοποιούνται ευρέως για πληροφορίες σχετικά με χρήστες και εφαρμογές. Το πρόβλημα είναι ότι υπάρχουν πολλοί διαφορετικοί κατάλογοι σε διάφορα μέρη ενός δικτύου και αρκετοί από αυτούς αποθηκεύουν τις ίδιες πληροφορίες. Αυτό κάνει δύσκολο την επικαιροποίηση, το συγχρονισμό και τον έλεγχό τους. Είναι επιθυμητό να υπάρχει μια δομή καταλόγου, σε μια κεντρική βάση, όπου οι πληροφορίες για οτιδήποτε στο δίκτυο να είναι προσβάσιμες μέσω ενός ευρέως διαδεδομένου πρωτοκόλλου πρόσβασης, όπως το LDAP (Lightweight Directory Access Protocol) [18].

2.6.1 Ιστορική ανάδρομη

Η πρώτη προσπάθεια για ένα νέο τύπο καταλόγου, που να μην είναι ειδικά μόνο για μια εφαρμογή ή ένα λειτουργικό σύστημα ήταν το X.500. Ορίστηκε από το OSI/ITTF, δημοσιεύτηκε το 1988 και επικαιροποιήθηκε το 1993. Το X.500 είναι μια σειρά από πρότυπα που καθορίζει πως η πληροφορία αποθηκεύεται και ανακτάται από ένα γενικό κατάλογο διάρθρωσης

υπηρεσιών. Λόγω της πολυπλοκότητας του, ποτέ δεν υιοθετήθηκε από τους κατασκευαστές λογισμικού. Το X.500 χρησιμοποιεί μεταξύ άλλων ένα πρωτόκολλο που ονομάζεται Directory Access Protocol, DAP ως μέθοδο πρόσβασης στο κατάλογο. Το DAP είναι πολύ δύσκολο και πολύπλοκο και είχε κακή απόδοση σε όλες τις υλοποιήσεις του. Έγιναν 2 προσπάθειες αντικατάστασης του με τα πρωτοκόλλα Directory Assistance Service, DAS, and Directory Interface to X.500 Implemented Efficiently, DIXIE όπου οι πελάτες επικοινωνούσαν με αυτά τα TCP/IP πρωτοκόλλα και στη συνέχεια μεταφραζόντουσαν σε αιτήσεις του X.500. Αυτά είχαν μεγαλύτερη υιοθέτηση από τη κοινότητα και στη συνέχεια ενοποιηθήκαν και προχώρησε ο IETF σε τυποποίηση τους το 1992. Αποτέλεσμα ήταν το νέο πρωτόκολλο με όνομα Lightweight Directory Access Protocol, LDAP [35].

2.6.2 Πλεονεκτήματα

Η άμεση υιοθέτηση του LDAP είναι κατανοητή αφού είχε απλοποιήσει θέματα που αντιμετώπιζε το DAP

- Λειτουργικότητας (ευκολότερη υλοποίηση διακομιστών και πελατών).
- Παρουσίασης δεδομένων (τα περισσότερα δεδομένα μεταφέρονται με τη μορφή απλού κειμένου-συμβολοσειρών κάτι που βελτιώνει επιπλέον και την απόδοση).
- Κωδικοποίησης (όσα μηνύματα κωδικοποιούνται, αυτό γίνεται με ένα μόνο υποσύνολο των κανόνων του X.500).
- Επικοινωνίας (τρέχει απευθείας στο TCP/IP και δεν απαιτεί την πολυεπίπεδη στοίβα δικτύωσης OSI).

Η καθιέρωση του και η συνεχή αναβάθμιση του πρωτοκόλλου LDAP οδήγησε από εκεί που ήταν ένα μέσο μετάφρασης κλήσεων για σύνδεση σε X.500 καταλόγου, τελικά να αντικαταστήσει εντελώς το X.500. Αυτό πρώτη φορά επιτεύχθηκε το 1995 από το πανεπιστήμιο του Μίσιγκαν και ονομάστηκε slapd (standalone LDAP daemon). Το LDAP παραμένει αρκετά συμβατό με το X.500 αλλά πλέον υποστηρίζει πολλές νέες δυνατότητες.

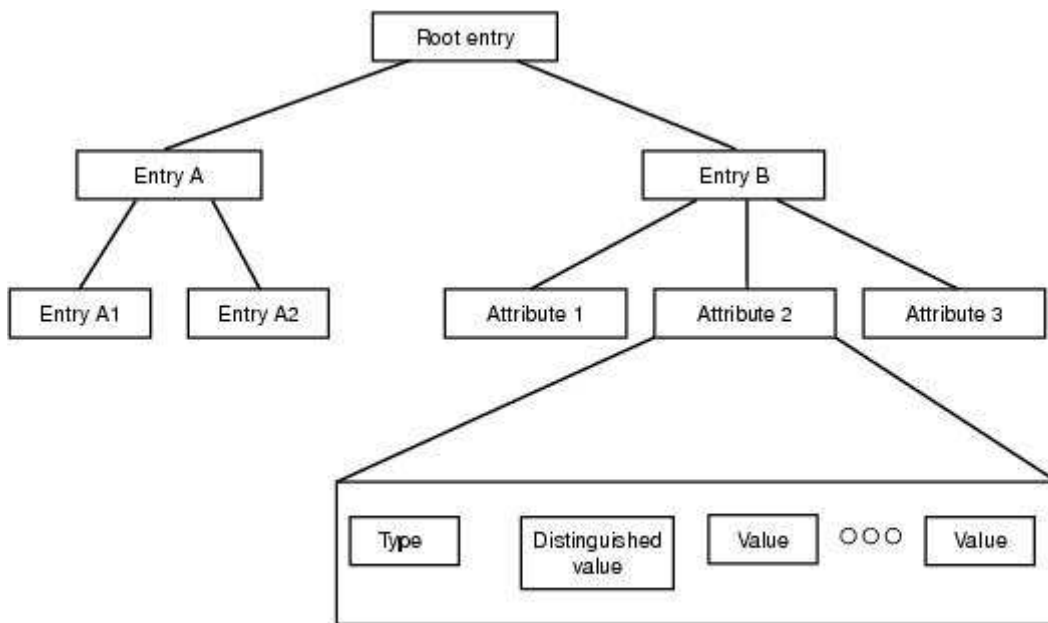
2.6.3 Προδιαγραφές

Η τρέχουσα 3η έκδοση του LDAP έχει λειτουργίες-προδιαγραφές που παρέχουν ένα αποδοτικό τρόπο οργάνωσης, ανάκτησης και συντήρησης μιας όσο μεγάλης ή πολύπλοκης δομής καταλόγου [43].

Το μοντέλο πληροφοριών ορίζει την δομή του τρόπου που τα δεδομένα συντηρούνται και αναφέρονται στο κατάλογο. Ο κατάλογος αποτελείται από μια συλλογή από εγγραφές. Μια εγγραφή αναφέρεται σε ένα αντικείμενο και όλες τις πληροφορίες σχετικές με αυτό. Η εγγραφή αποτελείται από ιδιότητες-χαρακτηριστικά με τιμές για κάθε αντικείμενο. Παράδειγμα μιας εγγραφής είναι το παρακάτω:

```
cn: Henrik Jaakkola  
hja  
sn: Jaakkola  
telephoneNumber: +46 (0)8 7267582  
uid: Hja  
mail: Henrik.Jaakkola@ausys.se  
employeeNumber: 924
```

Το μοντέλο ονομασίας είναι οργανωμένο ιεραρχικά σαν μια δενδρική δομή και με μια εγγραφή σαν την ρίζα του δέντρου. Αυτή η ρίζα συνήθως είναι το όνομα του Οργανισμού π.χ. ΤΕΙΚρητης. Κάθε εγγραφή έχει ένα μοναδικό όνομα το DN (distinguished names) κάτι σαν κλειδί της εγγραφής. Κάθε DN αποτελείται από σχετικά μοναδικά ονόματα (relative distinguished names) χωρισμένα με κώματα πχ 'cn=hja, ou=users, ou=Stockholm, dc=ausys, dc=se'



Εικόνα 2.1: Η δομή της πληροφορίας στο καταλογο[33]

Κάθε εγγραφή στο κατάλογο έχει μια ειδική ιδιότητα που λέγεται *objectclass*. Η τιμή της ιδιότητας αυτής ορίζει το τύπου του αντικείμενου. Το σχήμα ορίζει ποιες ιδιότητες πρέπει ή μπορεί να έχει ένα τύπος αντικείμενου. Κάθε αντικείμενο κληρονομεί τις ιδιότητες των προγονών του όπως στο αντικειμενοστραφή προγραμματισμό[24].

Το μοντέλο λειτουργικότητας ορίζει πως μπορεί να έχει κάποιος πρόσβαση στα δεδομένα του καταλόγου. Στην τρίτη έκδοση του LDAP υπάρχουν 9 διαφορετικές λειτουργίες[35]:

- 1 εγκατάλειψη, επιτρέπει στο πελάτη να αιτηθεί τη διακοπή μιας προηγούμενης λειτουργίας σε εξέλιξη
- 2 προσθήκη, επιτρέπει στο πελάτη να αιτηθεί την προσθήκη μιας εγγραφής
- 3 ταυτοποίηση, επιτρέπει τη ταυτοποίηση του πελάτη στο διακομιστή
- 4 σύγκριση, επιτρέπει τη σύγκριση μιας εγγραφής του καταλόγου με ένα όρισμα
- 5 διαγραφή, επιτρέπει ο χρήστης να αιτηθεί την διαγραφή μιας εγγραφής
- 6 τροποποίηση, επιτρέπει στο πελάτη να αιτηθεί την τροποποίηση μιας εγγραφής

- 7 τροποποίηση μοναδικού ονόματος εγγραφής
- 8 αναζήτηση, επιτρέπει στο πελάτη να αναζητήσει με διαφόρους τρόπους μια ή περισσότερες εγγραφές μέσα στη δενδρική μορφή του καταλόγου
- 9 λύση ταυτοποίησης, αποδεσμεύει προηγούμενη σύνδεση ταυτοποίησης μεταξύ πελάτη και διακομιστή

Το μοντέλο ασφαλείας φροντίζει 2 σημαντικά θέματα ασφαλείας. Το πρώτο θέμα είναι να περιορίζει ποιος και σε τι έχει πρόσβαση μέσα στο κατάλογο. Το δεύτερο θέμα είναι να διασφαλίζει τα δεδομένα που μεταφέρονται μεταξύ πελάτη και διακομιστή.

Η τρέχουσα έκδοση του LDAP δεν ορίζει ένα κοινό υποχρεωτικό μοντέλο πρόσβασης και έτσι υπάρχουν διάφορα μοντέλα ανάλογα την υλοποίηση του διακομιστή. Υπάρχει βέβαια μια προτεινόμενη μέθοδος πάλι από το IETF και συνήθως όλες οι υλοποιήσεις τουλάχιστον υποστηρίζουν αυτή.

Όταν ένας πελάτης θέλει να μιλήσει στο LDAP διακομιστή ανοίγει μια σύνδεση. Μπορεί να ζητήσει ταυτοποίηση ώστε να έχει τα προνόμια ανάλογα με τα δικαιώματα του. Η διαδικασία της ταυτοποίησης γίνεται με τη λειτουργία bind όπου ο πελάτης συσχετίζεται με ένα μοναδικό όνομα και ένα κωδικό πρόσβασης. Ο διακομιστής ελέγχει την ιδιότητα "userpassword" και την τιμή του αντικείμενου και ανάλογα επικυρώνει την σύνδεση με το πελάτη και ταυτόχρονα ξέρει τι δικαιώματα πάνω στα δεδομένα του καταλόγου επιτρέπεται να έχει ο πελάτης.

Υπάρχουν διάφοροι τρόποι ταυτοποίησης. Ο πιο απλός τρόπος είναι να γίνει η σύνδεση με αποστολή του κωδικού σε μη κωδικοποιημένη μορφή ενώ υποστηρίζονται κωδικοποιημένοι τρόποι αποστολής, όπως SSL και SASL.

Η μορφή ανταλλαγής δεδομένων (LDAP Data Interchange Format, LDIF) είναι ένας τυποποιημένος τρόπος περιγραφής εγγραφών στο κατάλογο βασισμένος σε απλές συμβολοσειρές. Αυτό επιτρέπει την εύκολη εξαγωγή και εισαγωγή εγγραφών στο κατάλογο. Στη πράξη υποστηρίζεται η τροποποίηση και διαγραφή εγγραφών. Συνήθως οι εγγραφές είναι αποθηκευμένες σε ένα αρχείο κειμένου που ο διακομιστής αναλύει και ξεχωρίζει τις εγγραφές προς τροποποίηση.

Το LDAP χρησιμοποιεί το TCP ως πρωτόκολλο μεταφοράς και το πρωτόκολλο LDAP ορίζει πως μεταφέρονται οι αιτήσεις από τους πελάτες και οι απαντήσεις από τους διακομιστές.

Οι υπεύθυνοι ανάπτυξης του LDAP δημιούργησαν ένα τυποποιημένο API με σκοπό να διευκολύνουν την ανάπτυξη εφαρμογών με υποστήριξη καταλόγου. Το αρχικό API υλοποιήθηκε στο πανεπιστήμιο του Μίσιγκαν και ήταν σε C. Πλέον υπάρχουν διαθέσιμα σε διάφορες γλώσσες όπως C++, Java Perl.

2.7 Shibboleth

2.7.1 Εισαγωγή

Το Shibboleth [49] είναι ένα έργο ανάπτυξης αρχιτεκτονικής και υλοποίησης με χρήση ανοιχτού κώδικα της αναγνώρισης χρηστών μεταξύ ομοσπονδίας τομέων με χρήση λιστών έλεγχου πρόσβασης (Access control list-ACL). Επιτρέπει στις πληροφορίες των χρηστών από ένα τομέα να είναι διαθέσιμες σε όλους τους τομείς που αποτελούν τη ομοσπονδία. Επιτρέπει σε ένα χρήστη να συνδέεται σε ένα τομέα ενός οργανισμού ενώ τα διαπιστευτήρια του βρίσκονται σε ένα άλλο τομέα του οργανισμού αποκτώντας πρόσβαση στις υπηρεσίες του τομέα που συνδέεται.

Η προσπάθεια αυτή ξεκίνησε το 2000 από το Middleware Architecture Committee for Education (MACE) [44] ώστε να αντιμετωπισθούν θέματα κοινής χρήσης πόρων μεταξύ ιδρυμάτων που είχαν μέχρι τότε διαφορετικούς τρόπους ταυτοποίησης και εξουσιοδότησης χρηστών. Η πρώτη έκδοση του ήταν το 2004 και τώρα βρίσκεται στην έκδοση 2 που πρωτοκυκλοφόρησε το 2008 [48].

Είναι μια τεχνολογία για το διαδίκτυο που υλοποιεί πρωτόκολλο HTTP/POST μαζί με αντικείμενα και ιδιότητες των χαρακτηριστικών προφίλ του Security Assertion Markup Language (SAML) και απαιτεί την υποστήριξη του τόσο από το πάροχο ταυτότητας Identity Provider (IdP) όσο και από τον πάροχο της υπηρεσίας Service Provider (SP). Ανάλογα με το ποιο προφίλ SAML χρησιμοποιεί, υπάρχουν οι έξης τρέχουσες υλοποιήσεις: Shibboleth 1.3 για SAML 1.1, προδιαγραφές και Shibboleth 2.0 για SAML 2.0 προδιαγραφές[21].

2.7.2 Έλεγχος πρόσβασης

Ο έλεγχος πρόσβασης του Shibboleth γίνεται με το ταίριασμα ιδιοτήτων που δίνονται από τον πάροχο ταυτότητας και εσωτερικών κανόνων που ορίζονται από τον πάροχο υπηρεσίας. Μια ιδιότητα είναι μια πληροφορία για τον χρήστη, όπως "μέλος αυτής της κοινότητας", "Alice Smith", ή "σύμφωνα με την άδεια της σύμβασης A". Η ταυτότητα του χρήστη θεωρείται ακόμα μια ιδιότητα και δίνεται μόνο όταν απαιτείται ρητώς. Έτσι προφυλάσσεται η προστασία των δεδομένων των χρηστών. Οι ιδιότητες μπορούν να γραφτούν σε Java ή να ανακτηθούν από τους καταλόγους δικτύου και βάσεις δεδομένων. Οι ιδιότητες του X.520 χρησιμοποιούνται πιο συχνά, αλλά και νέα χαρακτηριστικά μπορούν να οριστούν αυθαίρετα εφόσον είναι κατανοητά και μπορούν να ερμηνευτούν κατά παρόμοιο τρόπο από τον IdP και τον SP σε μια συναλλαγή.

Όταν ένας χρήστης προσπαθεί να αποκτήσει πρόσβαση σε υπηρεσία που προστατεύεται από Shibboleth, συνήθως ανακατευθύνεται σε ένα διακομιστή ταυτοποίησης, οποίος συνήθως έχει τη κωδική ονομασία Where Are You From (WAYF) και ζητάει από το χρήστη να επιλέξει το δικό του Πάροχο Ταυτότητας-Identity Provider (IdP) από μια λίστα γνωστών και αξιόπιστων τοποθεσιών. Ο δικτυακός τόπος παροχής υπηρεσιών έχει ήδη προεδραιωμένη σχέση εμπιστοσύνης με όλους αυτούς τους πάροχους ταυτότητας και τους εμπιστεύεται για τον έλεγχο ταυτότητας των χρηστών του.

Αφού ο χρήστης έχει επιλέξει το πάροχο ταυτότητας του, το πρόγραμμα πλοήγησης ανακατευθύνεται στο διακομιστή ελέγχου ταυτότητας της περιοχής του, π.χ. ένα αποθετήριο LDAP και ο χρήστης καλείται να συνδεθεί. Μετά την επιτυχή πιστοποίηση, η κεντρική τοποθεσία ανακατευθύνει το χρήστη πίσω στο SP και το μήνυμα μεταφέρει μια ψηφιακή υπογραφή με κωδική ονομασία Security Assertion Markup Language (SAML) [21], διαβεβαιώνοντας ότι ο χρήστης έχει πιστοποιηθεί με επιτυχία ή μη, με ένα συγκεκριμένο τρόπο. Ο πραγματικός μηχανισμός ελέγχου ταυτότητας που χρησιμοποιείται είναι στην επιλογή του IdP.

Εάν η ψηφιακή υπογραφή SAML επαληθεύει την πιστοποίηση του χρήστη, τότε ο SP έχει λάβει μαζί με το μήνυμα ένα προσωρινό ψευδώνυμο για τον χρήστη (handle), τη τοποθεσία της αρχής που διαχειρίζεται τα χαρακτηριστικά του χρήστη και τη διεύθυνση URL της υπηρεσίας που ο χρήστης είχε προηγουμένως προσπαθήσει να αποκτήσει πρόσβαση. Στη συνέχεια, ο SP στέλνει στον IdP ερώτημα αν ο χρήστης έχει κάποια συγκεκριμένα χαρακτηριστικά-ιδιότητες που χρειάζεται για την σύνδεση του στην υπηρεσία και ο IdP απαντά θετικά ή αρνητικά με νέο SAML μήνυμα. Το μοντέλο εμπιστοσύνης Shibboleth είναι ότι ο SP εμπιστεύεται τον IdP να διαχειριστεί

τα χαρακτηριστικά του κάθε χρήστη σωστά με όποιο τρόπο επιθυμεί. Τα χαρακτηριστικά σε αυτό στον ισχυρισμό μπορούν στη συνέχεια να χρησιμοποιηθούν για να επιτρέψει στο χρήστη πρόσβαση σε συγκεκριμένες περιοχές της υπηρεσίας. Έτσι, ουσιαστικά γίνεται η σύνδεση με την υπηρεσία χωρίς ο SP να γνωρίζει σε κανένα βήμα την ταυτότητα του χρήστη[19].

Η εμπιστοσύνη μεταξύ των διαφόρων τομέων υλοποιείται χρησιμοποιώντας κρυπτογραφία δημόσιου κλειδιού (συντά απλά πιστοποιητικά SSL server) και τα μεταδεδομένα που περιγράφουν τους παρόχους. Η χρήση των πληροφοριών που διακινούνται ελέγχεται μέσω συμφωνιών. Οι Ομοσπονδίες συνήθως έχουν απλοποιήσει αυτές τις σχέσεις με τη συγκέντρωση μεγάλου αριθμού παρόχων που συμφωνούν να χρησιμοποιήσουν τους κοινούς κανόνες και συμβάσεις.

Ομοσπονδίες έχουν διαμορφωθεί σε πολλές χώρες για την οικοδόμηση εμπιστοσύνης δομών για την ανταλλαγή πληροφοριών με τη χρήση SAML και Shibboleth λογισμικού. Πολλοί μεγάλοι πάροχοι περιεχομένου υποστηρίζουν το Shibboleth για την πρόσβαση σε αυτούς.

Η χρήση του έχει υιοθετηθεί από μεγάλους οργανισμούς και πολυεθνικές καθώς και από κοινότητες κυρίως εκπαιδευτικές. Τα ακαδημαϊκά ιδρύματα της Ελλάδας για παράδειγμα έχουν αποφασίσει τη διασύνδεση τους μέσω κοινής πρόσβασης με τη χρήση Shibboleth.

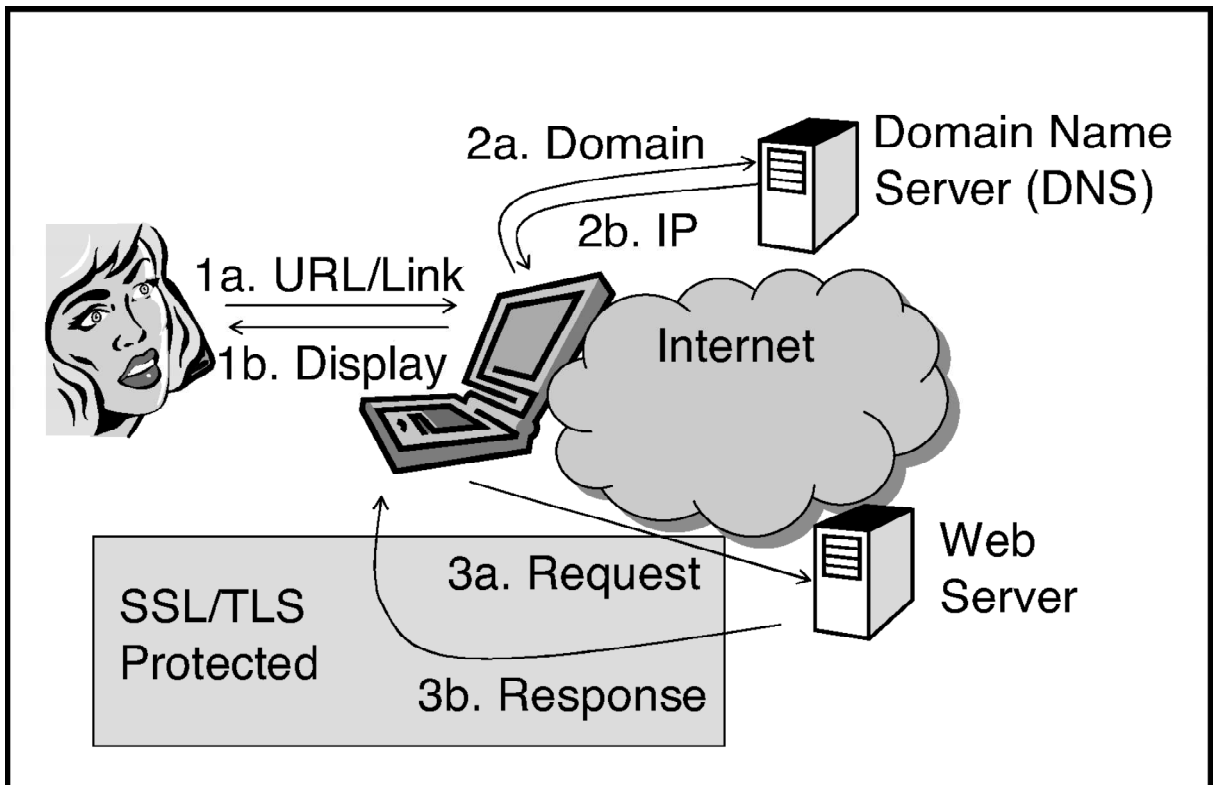
2.8 Hypertext Transfer Protocol Secure (HTTPS)

2.8.1 Εισαγωγή

Hypertext Transfer Protocol Secure (HTTPS) είναι ένας συνδυασμός του Hypertext Transfer Protocol (HTTP) με το πρωτόκολλο SSL/TLS. Παρέχει κρυπτογραφημένη επικοινωνία και ασφαλή ταυτοποίηση ενός διακομιστή ιστοσελίδων. Οι συνδέσεις HTTPS χρησιμοποιούνται κυρίως για τις συναλλαγές πληρωμής στο World Wide Web και για άλλες ευαίσθητες συναλλαγές στα εταιρικά συστήματα πληροφοριών, όπως εξασφάλιση των λογαριασμών, τη ιδιωτική διατήρηση των επικοινωνιών των χρηστών, του ιστορικού περιήγησης και την εγγύηση της αυθεντικότητας όλων των τύπων των διαδικτυακών τόπων [10].

Το HTTPS είναι ένα σύστημα που έχει ίδια σύνταξη με τη συνήθη μέθοδο HTTP, εκτός από την ένδειξη του συστήματος. Ωστόσο, το HTTPS υποχρεώνει το πρόγραμμα περιήγησης να

χρησιμοποιήσει ένα επιπλέον επίπεδο κρυπτογράφησης SSL/TLS για την προστασία της κυκλοφορίας. Το SSL είναι ειδικά σχεδιασμένο για το HTTP, δεδομένου ότι μπορεί να παρέχει κάποια προστασία, ακόμη και αν μόνο η μία πλευρά της επικοινωνίας είναι επικυρωμένη. Αυτή είναι η περίπτωση με http συναλλαγές μέσω του Internet, όπου συνήθως μόνο ο διακομιστής είναι επικυρωμένος (από τον πελάτη εξετάζοντας το πιστοποιητικό του διακομιστή)[38].



Εικόνα 2.2: Διαδικασία αίτησης και απόκρισης πρωτοκόλλου HTTP με ασφάλεια SSL[10]

Η κύρια ιδέα του HTTPS είναι να δημιουργηθεί ένα ασφαλές κανάλι σε ένα ανασφαλές δίκτυο. Αυτό εξασφαλίζει ικανοποιητική προστασία από κακόβουλες ενέργειες παρακολούθησης κίνησης δικτύου, υπό την προϋπόθεση ότι το πιστοποιητικό του διακομιστή έχει ελεγχθεί και είναι αξιόπιστο [03].

Προγράμματα περιήγησης στο Web εμπιστεύονται HTTPS ιστοσελίδες που βασίζονται σε πιστοποιητικά, εγκεκριμένα από Αρχές Έκδοσης Πιστοποιητικών (π.χ. η VeriSign, Microsoft κλπ.).

Ο χρήστης πρέπει να εμπιστευθεί μια σύνδεση HTTPS σε μια ιστοσελίδα, αν και μόνο αν όλα τα παρακάτω ισχύουν:

- Ο χρήστης εμπιστεύεται το λογισμικό πρόγραμμα περιήγησης ότι εφαρμόζει σωστά το HTTPS και τα προεγκατεστημένα πιστοποιητικά.
- Ο χρήστης εμπιστεύεται την αρχή έκδοσης πιστοποιητικών που υπογράφει και εγγυάται την αυθεντικότητα του ιστοχώρου.
- Η ιστοσελίδα παρέχει ένα έγκυρο πιστοποιητικό.
- Το πιστοποιητικό προσδιορίζει σωστά την ιστοσελίδα (π.χ., όταν ο browser επισκεφθεί "https://example.com", το πιστοποιητικό που θα λάβει είναι για την για την διεύθυνση αυτή και όχι για κάποια άλλη.
- Ο χρήστης εμπιστεύεται ότι η κρυπτογράφηση του πρωτοκόλλου (TLS/SSL) είναι αρκετά ασφαλής.

Το http λειτουργεί στο υψηλότερο επίπεδο του μοντέλου OSI, στο στρώμα εφαρμογών. Αλλά το πρωτόκολλο ασφαλείας που λειτουργεί σε χαμηλότερο υπόστρωμα, κρυπτογραφεί το μήνυμα http πριν από τη διαβίβαση και αποκρυπτογραφεί το μήνυμα HTTP κατά την άφιξή του. Τα πάντα στο μήνυμα HTTPS είναι κρυπτογραφημένα, συμπεριλαμβανομένων των κεφαλίδων και του αιτήματος/απόκρισης. Σε περίπτωση πιθανής επίθεσης αποκρυπτογράφησης, ο εισβολέας μπορεί να γνωρίζει μόνο ότι η μια σύνδεση πραγματοποιείται μεταξύ δύο μερών στο συγκεκριμένο τομέα (domain name) και σε ποιες συγκεκριμένες διευθύνσεις IP.

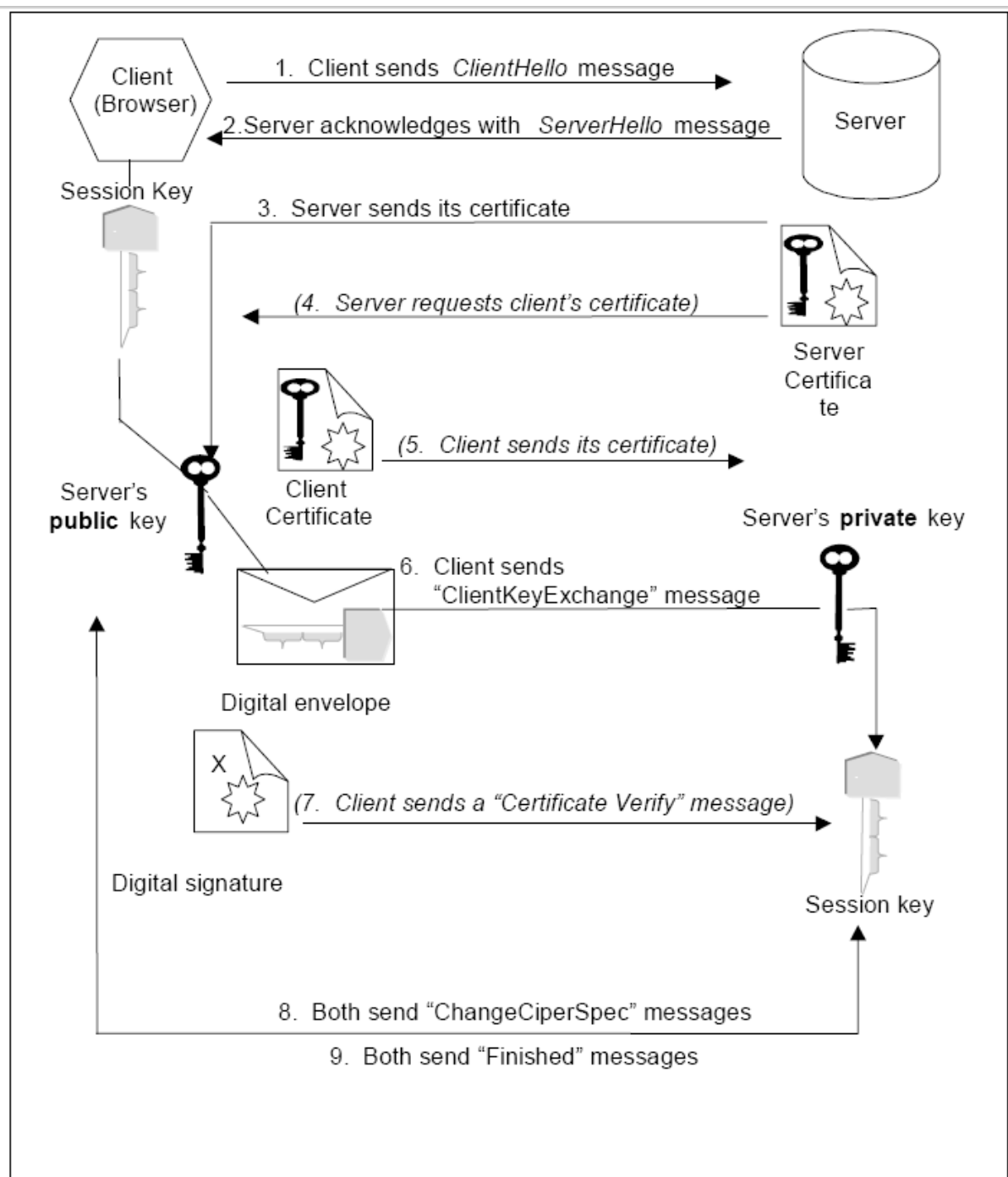
2.8.2 Ρύθμιση διακομιστή

Προκειμένου να προετοιμάσει ένα web server να δεχτεί συνδέσεις HTTPS, ο διαχειριστής πρέπει να δημιουργήσει ένα πιστοποιητικό με δημόσιο κλειδί για τον web server. Αυτό το πιστοποιητικό πρέπει να υπογραφεί από μια αξιόπιστη αρχή έκδοσης πιστοποιητικών ώστε ο web browser του επισκέπτη να τη δεχθεί χωρίς προειδοποίηση. Η αρχή πιστοποιεί ότι ο κάτοχος του πιστοποιητικού είναι ο διαχειριστής του web server που το παρουσιάζει. Τα προγράμματα περιήγησης διανέμονται συνήθως με μια λίστα με τις σημαντικότερες αρχές έκδοσης πιστοποιητικών, έτσι ώστε να μπορούν να ελέγχουν τα πιστοποιητικά που υπογράφονται από αυτές.

2.8.3 SSL

Το Secure Socket Layer (SSL) είναι μια τεχνολογία που επιτρέπει στα προγράμματα πλοήγησης και τις ιστοσελίδες των διακομιστών να επικοινωνούν μέσω μιας ασφαλούς σύνδεσης. Αυτό σημαίνει ότι τα δεδομένα που αποστέλλονται κρυπτογραφούνται από τη μία πλευρά, μεταδίδονται και τελικά αποκρυπτογραφούνται από την άλλη πλευρά πριν τη επεξεργασία. Αυτή είναι μια αμφίδρομη διαδικασία, τόσο ο διακομιστής όσο και το πρόγραμμα περιήγησης κρυπτογραφούν όλη τη κίνηση πριν από την αποστολή δεδομένων.

Μια άλλη σημαντική πτυχή του πρωτοκόλλου SSL είναι η πιστοποίηση ταυτότητας. Αυτό σημαίνει ότι κατά την αρχική σας προσπάθεια να επικοινωνήσετε με έναν web server μέσω ασφαλούς σύνδεσης, ο διακομιστής θα παρουσιάσει στο web browser ένα σύνολο διαπιστευτηρίων του, με τη μορφή ενός «πιστοποιητικού», σαν απόδειξη ότι το site είναι αυτό που ισχυρίζεται ότι είναι. Σε ορισμένες περιπτώσεις, ο server μπορεί επίσης να ζητήσει ένα πιστοποιητικό από το web browser σας, σαν απόδειξη ταυτοποίησης. Αυτό είναι γνωστό ως «έλεγχος ταυτότητας πελάτη», αν και στην πράξη αυτό χρησιμοποιείται περισσότερο στις επιχειρήσεις και στις business-to-business(B2B) συναλλαγές. Οι περισσότεροι web servers με δυνατότητα SSL δεν ζητούν έλεγχο ταυτότητας πελάτη.



Εικόνα 2.3: Διαδικασία έναρξης SSL/TLS σύνδεσης [03]

Ενώ το πρωτόκολλο SSL έχει σχεδιαστεί να είναι αποτελεσματικό σε θέματα ασφάλειας μέσω κρυπτογράφησης/αποκρυπτογράφησης, είναι γενικά υπολογιστικά ακριβής διαδικασία από την άποψη των επιδόσεων. Δεν είναι απαραίτητο να τρέξει μια ολόκληρη διαδικτυακή εφαρμογή μέσω SSL και ένας προγραμματιστής μπορεί να επιλέξει ποιες σελίδες απαιτούν μια ασφαλή σύνδεση και ποιες όχι. Για ένα αρκετά πολυσύχναστο ιστοχώρο, είναι σύνηθες να χρησιμοποιείται το SSL σε ορισμένες μόνο σελίδες, δηλαδή όσες σελίδες απαιτούν ανταλλαγή

ευαίσθητων πληροφοριών όπως ιστοσελίδες σύνδεσης, προσωπικές σελίδες πληροφοριών, και ταμεία σε καλάθι αγορών ενός eshop, όπου τα στοιχεία της πιστωτικής κάρτας θα μπορούσαν ενδεχομένως να υποκλαπούν [03]. Κάθε ιστοσελίδα διαδικτυακής εφαρμογής μπορεί να αιτηθεί σύνδεση SSL απλά με τη αντικατάσταση του προθέματος http με https στη διεύθυνση URL της. Οι σελίδες που απαιτούν υποχρεωτικά ασφαλή σύνδεση θα πρέπει να ελέγχουν το είδος του πρωτοκόλλου που συνδέεται ο χρήστης στο αίτημα της σελίδας και να λάβουν τα κατάλληλα μέτρα, εάν το https δεν έχει καθοριστεί.

2.9 Κρυπτογράφηση

Κρυπτογράφηση είναι η διαδικασία κατά την οποία η πληροφορία μετατρέπεται από την αρχική της αναγνώσιμη μορφή (plain-text) σε ακατάληπτα δεδομένα, με σκοπό την προστασία από υποκλοπή. Η αντίστροφη διαδικασία καλείται αποκρυπτογράφηση. Τόσο η κρυπτογράφηση, όσο και η αποκρυπτογράφηση, βασίζονται σε έναν αλγόριθμο, ο οποίος ως είσοδο παίρνει την αρχική πληροφορία και ένα κλειδί (key) και ως έξοδο βγάζει την κρυπτογραφημένη (κρυπτογράφηση) ή την αποκρυπτογραφημένη (αποκρυπτογράφηση) πληροφορία. Ο όρος κρυπτογραφία περιλαμβάνει το σύνολο των πρακτικών κρυπτογράφησης και αποκρυπτογράφησης.

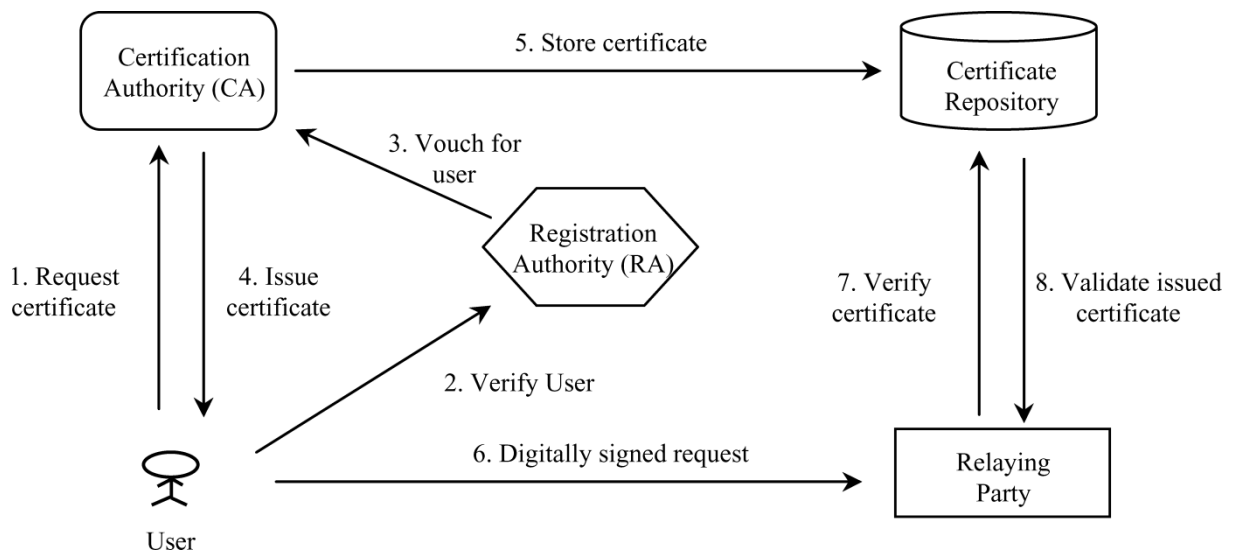
Μια ειδική κατηγορία αλγορίθμων κρυπτογράφησης και αποκρυπτογράφησης είναι οι ασύμμετροι αλγόριθμοι κρυπτογράφησης ή αλγόριθμοι κρυπτογράφησης δημόσιου κλειδιού (public key). Στους αλγορίθμους αυτούς, το κλειδί κρυπτογράφησης είναι διαφορετικό από το κλειδί αποκρυπτογράφησης. Ότι κρυπτογραφείται από το δημόσιο κλειδί μπορεί να αποκρυπτογραφηθεί μόνο από το ιδιωτικό κλειδί και το αντίστροφο [03].

Η ασυμμετρία καθιστά τις μεθόδους αυτές κρυπτογράφησης ιδανικές για χρήση σε επισφαλή μέσα μετάδοσης πληροφοριών:

- Ο παραλήπτης κρατάει μυστικό το ιδιωτικό κλειδί του (private key). Παράλληλα διανέμει ελεύθερα σε όποιον το επιθυμεί το δημόσιο κλειδί του (public key).
- Ο αποστολέας χρησιμοποιεί το δημόσιο κλειδί του παραλήπτη με σκοπό να κρυπτογραφήσει την πληροφορία που θέλει να του στείλει. Κατ' αυτόν τον τρόπο, η

πληροφορία μπορεί να αναγνωστεί μόνο από τον παραλήπτη αφού αυτός είναι ο μόνος που διαθέτει το ιδιωτικό κλειδί.

Η Υποδομή Δημόσιου Κλειδιού (Public Key Infrastructure, PKI) αποτελεί ένα συνδυασμό λογισμικού, τεχνολογιών ασύμμετρης κρυπτογραφίας και διαδικασιών, ο οποίος κατά βάση πιστοποιεί την εγκυρότητα κάθε εμπλεκόμενου σε μια ψηφιακή συναλλαγή [11].



Εικόνα 2.4: Υποδομή Δημόσιου Κλειδιού [12]

Η ΥΔΚ πιστοποιεί την ταυτότητα μιας πιστοποιημένης οντότητας υπογράφοντας το δημόσιο κλειδί της και δημοσιεύοντάς το, μαζί με πληροφορίες σχετικά με την ταυτότητα της οντότητας, σε ένα πιστοποιητικό. Παράλληλα, διατηρεί καταλόγους με τα έγκυρα, τα ληγμένα αλλά και τα ανακληθέντα πιστοποιητικά.

Οι κυριότεροι μηχανισμοί ασφάλειας τους οποίους καλύπτει η Υποδομή Δημόσιου Κλειδιού είναι η εξής:

- Απόρρητο της επικοινωνίας (Confidentiality): Τα δεδομένα προστατεύονται από μη εξουσιοδοτημένη πρόσβαση με μηχανισμούς ελέγχου πρόσβασης στην περίπτωση αποθήκευσης δεδομένων και μέσω κρυπτογράφησης κατά την αποστολή τους.
- Ακεραιότητα (Integrity): Τα δεδομένα προστατεύονται από μη εξουσιοδοτημένη τροποποίηση μέσω μηχανισμών κρυπτογράφησης όπως οι ηλεκτρονικές υπογραφές.

- Πιστοποίηση (Authentication): Πραγματοποιείται επιβεβαίωση της ταυτότητας ενός ατόμου ή της πηγής αποστολής δεδομένων.
- Μη Άρνηση Αποδοχής (Non-Repudiation): Η Μη Άρνηση Αποδοχής συνδυάζει τις υπηρεσίες της Πιστοποίησης και της Ακεραιότητας και διασφαλίζει ότι μία συναλλαγή η οποία πραγματοποιήθηκε ηλεκτρονικά δε μπορεί να αμφισβητηθεί από τα συμβαλλόμενα μέρη. Για παράδειγμα, ο αποστολέας δεδομένων δεν μπορεί να αρνηθεί ότι δημιούργησε και απέστειλε το μήνυμα.

Το X.509 είναι ένα διεθνές πρότυπο που καθορίζει τον τρόπο λειτουργίας των Υποδομών Δημοσίου Κλειδιού (Public Key Infrastructure/PKI). Προδιαγράφει τις μορφές διάθεσης της σχετικής πληροφορίας (κλειδιά, πιστοποιητικά, λίστες ανάκλησης), καθώς και τους αλγορίθμους επαλήθευσης του κύρους ενός πιστοποιητικού[11].

Στο X.509, η Αρχή Πιστοποίησης (Certification Authority CA) εκδίδει ένα πιστοποιητικό, το οποίο περιλαμβάνει πληροφορίες για μια οντότητα-υποκείμενο (subject). Το υποκείμενο του πιστοποιητικού μπορεί να είναι ένα όνομα (common name) ή κάποια άλλη οντότητα (alternative name), όπως μια διεύθυνση e-mail, μια διεύθυνση IP ή ένα όνομα DNS.

Ένα πιστοποιητικό X.509 περιέχει τις ακόλουθες πληροφορίες:

- Πιστοποιητικό
- Έκδοση X.509 (π.χ. 3)
- Αύξων Αριθμός
- Αλγόριθμος
- Εκδίδουσα Αρχή Πιστοποίησης
- Περίοδος εγκυρότητας
- Όχι νωρίτερα από
- Όχι αργότερα από

- Υποκείμενο
- Πληροφορίας δημόσιου κλειδιού υποκειμένου
- Αλγόριθμος δημόσιου κλειδιού
- Τιμή δημόσιου κλειδιού
- Προαιρετικές πληροφορίες...
- Επεκτάσεις (extensions)
- Αλγόριθμος υπογραφής πιστοποιητικού

Ας υποθέσουμε ότι συνδεόμενοι σε έναν Εξυπηρετητή (server), παρουσιάζεται ένα πιστοποιητικό υπογεγραμμένο από την αρχή πιστοποίησης «CA 1». Η «CA 1» είναι ενδιάμεση αρχή πιστοποίησης και δεν υπάρχει εγκατεστημένη στο λειτουργικό σύστημα, όμως ο εξυπηρετητής μας παρέχει το πιστοποιητικό της μαζί με το δικό του. Ελέγχοντας την υπογραφή, διαπιστώνεται ότι, σύμφωνα με τη «CA 1», το πιστοποιητικό του εξυπηρετητή είναι έγκυρο, ωστόσο η «CA 1» δεν είναι έμπιστη Αρχή Πιστοποίησης. Το πιστοποιητικό της «CA 1» φέρει υπογραφή από την «CA 2», η οποία βρίσκεται στον κατάλογο με τις έμπιστες Αρχές Πιστοποίησης του λειτουργικού συστήματος που χρησιμοποιούμε. Επομένως, επαληθεύοντας το πιστοποιητικό της «CA 1» χρησιμοποιώντας το δημόσιο κλειδί από το πιστοποιητικό της «CA 2», καταλήγουμε σε μια έμπιστη αρχή πιστοποίησης. Κατ' αυτόν τον τρόπο σχηματίζεται η αλυσίδα πιστοποίησης (certification chain).[11]

Επομένως το δημόσιο κλειδί που βρίσκεται στο πιστοποιητικό του εξυπηρετητή μπορεί να χρησιμοποιηθεί με ασφάλεια για την κρυπτογράφηση δεδομένων προς αυτόν, αφού η έμπιστη «CA 2» έχει επιβεβαιώσει (μέσω της «CA 1») ότι το πιστοποιητικό αυτό (και επομένως και το δημόσιο κλειδί) όντως ανήκει στον εξυπηρετητή.

2.10 Πιστοποιητικά

2.10.1 Εισαγωγή

Για την υλοποίηση του SSL, ένας web server πρέπει να έχει ένα σχετικό πιστοποιητικό για κάθε εξωτερικό περιβάλλον (διεύθυνση IP) που δέχεται ασφαλείς συνδέσεις. Η λογική πίσω από αυτό το σχέδιο είναι ότι ένας διακομιστής πρέπει να παρέχει κάποια εύλογη ασφάλεια ότι είναι ο ιδιοκτήτης αυτού που δηλώνει ότι είναι, ιδιαίτερα πριν από τη λήψη οποιασδήποτε ευαίσθητης πληροφορίας. Ένα πιστοποιητικό είναι μια «ψηφιακή άδεια οδήγησης» για μια διεύθυνση στο Internet. Αναφέρει πληροφορίες για την εταιρία που είναι κάτοχος της ιστοσελίδας. Αυτό έχει τη μορφή κρυπτογραφημένης υπογραφής από τον ιδιοκτήτη του ιστότοπου και είναι εξαιρετικά δύσκολο για οποιονδήποτε άλλον να την πλαστογραφήσει. Για τους διαδικτυακούς τόπους που εμπλέκονται στο ηλεκτρονικό εμπόριο, ή οποιαδήποτε άλλη εμπορική συναλλαγή όπου η εξακρίβωση της ταυτότητας είναι σημαντική, ένα τέτοιο πιστοποιητικό αγοράζεται συνήθως από μια γνωστή αρχή έκδοσης πιστοποιητικών (CA), όπως η VeriSign ή Thawte. Τα πιστοποιητικά αυτά μπορεί να ελεγχθούν ηλεκτρονικά, η Αρχή Πιστοποίησης θα εγγυηθεί για τη γνησιότητα των πιστοποιητικών που χορηγεί. Στη πραγματικότητα το πιστοποιητικό ισχύει αν εμπιστεύεστε την αρχή πιστοποίησης που το χορήγησε.

Σε πολλές περιπτώσεις ένας διαχειριστής μπορεί να θέλει μόνο την εξασφάλιση ότι τα δεδομένα που μεταδίδονται και λαμβάνονται από το διακομιστή είναι ιδιωτικά και δεν μπορούν να κατασκοπευτούν από όποιον κάνει υποκλοπές. Η Java παρέχει ένα σχετικά απλό εργαλείο γραμμής εντολών, που ονομάζεται keytool και μπορεί εύκολα να δημιουργήσει ένα "αυτο-υπογεγραμμένο" πιστοποιητικό. Τα αυτουπογεγραμμένα πιστοποιητικά δεν έχουν καταγραφεί επίσημα σε οποιαδήποτε γνωστή ΑΠ και επομένως δεν θεωρούνται αυθεντικά.

2.10.2 Λειτουργικότητα

Την πρώτη φορά που ένας χρήστης προσπαθεί να αποκτήσει πρόσβαση σε μια ασφαλή σελίδα, συνήθως παρουσιάζεται ένα παράθυρο που περιέχει τις λεπτομέρειες του πιστοποιητικού (όπως η εταιρεία και το όνομα της επαφής) και ρωτά αν ο επισκέπτης επιθυμεί να αποδεχθεί το πιστοποιητικό ως έγκυρο και να συνεχίσει με τη συναλλαγή. Ορισμένα προγράμματα περιήγησης προσφέρουν τη δυνατότητα για μόνιμη αποδοχή ενός συγκεκριμένου πιστοποιητικού, σαν έγκυρα, οπότε ο επισκέπτης δεν θα ενοχληθεί με νέα προτροπή κάθε φορά που επισκέπτεται το site. Άλλοι browsers δεν παρέχουν αυτή την επιλογή. Αφού εγκριθεί από το χρήστη, ένα πιστοποιητικό θα θεωρηθεί έγκυρο για τουλάχιστον ολόκληρη την σύνοδο του browser.

2.10.3 Υπογραφή πιστοποιητικού

Η Αρχή Πιστοποίησης υπογράφει ψηφιακά τις πληροφορίες του πιστοποιητικού, ώστε όποιος διαθέτει το δημόσιο κλειδί της, να μπορεί να επαληθεύσει την ισχύ των πληροφοριών που φέρει το πιστοποιητικό. Κάθε Αρχή Πιστοποίησης διαθέτει με τη σειρά της ένα δικό της πιστοποιητικό, υπογεγραμμένο από κάποια άλλη αρχή πιστοποίησης (σε αυτή την περίπτωση μιλάμε για ενδιάμεση αρχή πιστοποίησης intermediate CA) ή από τον εαυτό της (οπότε μιλάμε για αρχική αρχή πιστοποίησης root CA). Επομένως η επαλήθευση της εγκυρότητας ενός πιστοποιητικού μπορεί να γίνει ακολουθώντας μια αλυσίδα Αρχών Πιστοποίησης, μέχρι να φτάσουμε σε μια έμπιστη Αρχή Πιστοποίησης (trusted CA).

Κάθε λειτουργικό σύστημα ή πρόγραμμα περιήγησης του διαδικτύου (browser), διανέμεται με έναν κατάλογο έμπιστων αρχικών και ενδιάμεσων Αρχών Πιστοποίησης, πράγμα το οποίο σημαίνει ότι μπορεί να επαληθεύσει την εγκυρότητα ενός μεγάλου αριθμού πιστοποιητικών.

2.11 Οδηγίες για την Προσβασιμότητα του Περιεχομένου του Ιστού

2.11.1 Εισαγωγή

Άτομα με ειδικές ανάγκες βασίζονται στο διαδίκτυο για καθημερινές εργασίες καθώς και για την απασχόληση, τη μάθηση και ψυχαγωγία τους. Τα καταστήματα on-line αγορών επιτρέπουν στους ανθρώπους με προβλήματα κινητικότητας να ψωνίζουν χωρίς να ταξιδεύουν στη φυσική τοποθεσία του καταστήματος ή να χρειάζονται βοήθεια από άλλους. Ευκαιρίες μάθησης που παρέχονται μέσω του διαδικτύου προσφέρουν περαιτέρω εκπαίδευση για τους ανθρώπους από όλα τα κοινωνικά στρώματα και ικανότητες. Εικονικές κοινότητες, κοινωνικές και ψυχαγωγικές περιοχές μπορούν να παρέχουν όλες τις σημαντικές ευκαιρίες αλληλεπίδρασης, κοινωνικοποίησης και απασχόλησης σε μεγάλο αριθμό ανθρώπων. Ενώ το Web 2.0 μπορεί να προσφέρει τεράστια οφέλη, όλα τα νέα πρότυπα αλληλεπίδρασης δεν είναι άμεσα προσβάσιμα [09].

Το Web έχει γίνει αρκετά προσιτό σε άτομα με ειδικές ανάγκες, αλλά αυτό δεν ήταν πάντα η περίπτωση. Αρχικά οι άνθρωποι χρησιμοποιούσαν βοηθητικές τεχνολογίες, όπως αναγνώστες οθόνης, μεγεθυντές οθόνης, και εναλλακτικές συσκευές εισόδου και είχαν δυσκολία να αλληλεπιδρούν με τις υπηρεσίες στο διαδίκτυο. Οι απαιτήσεις από τις κυβερνήσεις έχουν

αναγκάσει τις εταιρείες να αντιμετωπίσουν την προσβασιμότητα και να ακολουθήσουν κατευθυντήριες γραμμές, όπως οι Οδηγίες για την Προσβασιμότητα του Περιεχομένου του Ιστού προκειμένου να κάνουν το Web προσβάσιμο[04].

Οι οδηγίες για την Προσβασιμότητα του Περιεχομένου του Ιστού (Web Content Accessibility Guidelines-WCAG) περιγράφουν πως το περιεχόμενο του παγκόσμιου ιστού είναι προσβάσιμο σε ανθρώπους με αναπηρίες. Η προσβασιμότητα ασχολείται με ένα ευρύ φάσμα αναπηριών, οπτικών, ακουστικών, φυσικών, ομιλίας, μαθησιακών, γλωσσολογικών, και νευρολογικών. Παρά το γεγονός ότι αυτές οι κατευθυντήριες γραμμές καλύπτουν ένα ευρύ φάσμα θεμάτων, δεν είναι σε θέση να αντιμετωπίσουν τις ανάγκες των ατόμων με όλους τους τύπους, βαθμούς, και συνδυασμούς αναπηρίας [56].

Αυτές οι κατευθυντήριες γραμμές κάνουν το περιεχόμενο του ιστού πιο εύχρηστο και για άτομα μεγαλύτερης ηλικίας με τις μεταβαλλόμενες ικανότητες τους λόγω της γήρανσης.

Το πρότυπο WCAG 2.0 έχει αναπτυχθεί μέσω της διαδικασίας του W3C σε συνεργασία με άτομα και οργανισμούς σε όλο τον κόσμο, με στόχο την παροχή ενός κοινού προτύπου για την προσβασιμότητα του περιεχομένου Web που να ανταποκρίνεται στις ανάγκες των ατόμων, οργανισμών και κυβερνήσεων διεθνώς. Το WCAG 2.0 έχει σχεδιαστεί να εφαρμόζεται ευρέως σε διάφορες τεχνολογίες στο Web τώρα και στο μέλλον και είναι ελέγξιμο με ένα συνδυασμό αυτόματων δοκιμών και ανθρώπινης αξιολόγησης[03] [58].

Η προσβασιμότητα στον Παγκόσμιο Ιστό δεν εξαρτάται μόνο από το πόσο προσβάσιμο είναι το περιεχόμενο, αλλά και από την προσβασιμότητα που υποστηρίζουν οι Web browsers των χρηστών. Συγγραφικά εργαλεία δημιουργίας περιεχομένου έχουν επίσης σημαντικό ρόλο στην προσβασιμότητα στον Παγκόσμιο Ιστό.

Βασικά συστατικά της προσβασιμότητας του Ιστού:

2.11.2 Επίπεδα WCAG 2.0 Προσανατολισμού

Τα άτομα και οι οργανισμοί που χρησιμοποιούν το WCAG ποικίλλουν και περιλαμβάνουν σχεδιαστές ιστοσελίδων και προγραμματιστές, τους υπεύθυνους χάραξης πολιτικής, εκπαιδευτικούς και μαθητές. Προκειμένου να καλυφτούν οι διάφορες ανάγκες τους, έχουν οριστεί συγκεκριμένες γενικές αρχές, γενικές κατευθυντήριες γραμμές, ελέγξιμα κριτήρια

επιτυχίας και μια πλούσια συλλογή τεχνικών επάρκειας, συμβουλευτικής, καταγραφή προηγούμενων αποτυχημένων προσπαθειών και μια συλλογή από πηγες πληροφόρησης αλλά και κώδικα για την ενημέρωση των ενδιαφερόμενων και τελικά την βελτίωση προσβασιμότητας των ατόμων με αναπηρίες στο διαδίκτυο [53].

2.11.3 Γενικές αρχές καθοδήγησης

Το πληροφοριακό υλικό πρέπει να καλύπτει πρώτα από όλα τους παρακάτω τέσσερις κανόνες. Ο κάθε κανόνας επιτυγχάνεται αν ακολουθηθούν συγκεκριμένες κατευθυντήριες γραμμές. Έτσι, το δημοσιευμένο περιεχόμενο πρέπει να είναι [57]:

- 1 Αντιληπτό, οι πληροφορίες και τα στοιχεία που αποτελούν τη διεπαφή του χρήστη πρέπει να παρουσιάζονται με τρόπους που μπορούν να αντιληφθούν οι χρήστες. Πρέπει να γίνεται αντιληπτό τουλάχιστον από μια από τις αισθήσεις του ατόμων. Οι κατευθυντήριες γραμμές ώστε να επιτυγχάνεται αυτό είναι οι παρακάτω:
 - Εναλλακτική χρήση κειμένου: Παροχή εναλλακτικής εμφάνισης κείμενου για το περιεχόμενο που δεν έχει τη μορφή κειμένου, έτσι ώστε να μπορεί να μετατραπεί σε άλλες μορφές που οι χρήστες αντιλαμβάνονται, όπως μεγάλα τυπογραφικά στοιχεία, γραφή Μπράιγ, εκφώνηση του κειμένου, άλλα σύμβολα ή πιο απλή γλώσσα.
 - Εναλλακτική παρουσίαση με άλλες μεθόδους σε περιεχόμενο πολυμέσων. όπως υπότιτλοι σε αρχεία video και ηχητική περιγραφή του video.
 - Προσαρμογή περιεχομένου: Δημιουργία περιεχομένου που μπορεί να παρουσιαστεί με διάφορους τρόπους (για παράδειγμα με απλούστερη διάταξη), χωρίς απώλεια πληροφοριών ή δομής του.
 - Διακρινόμενου περιεχομένου ώστε να είναι ευκολότερο για τους χρήστες να δουν και να ακούσουν, συμπεριλαμβανομένης του ξεκάθਾਰου διαχωρισμού του πίσω και του μπροστινού μέρους της ιστοσελίδας με τη χρήση σωστών χρωμάτων, αντίθεσης χρωμάτων (contrast), μεταβολής μεγέθους γραμματοσειράς, αλλά και ευκολότερης πρόσβασης στα πολυμέσα της ιστοσελίδας.

- 2 Εύχρηστο, τα στοιχεία της διεπαφής χρήστη και πλοήγησης πρέπει να είναι λειτουργικά, η διεπαφή δεν πρέπει να απαιτεί αλληλεπίδραση όταν ο χρήστης δεν μπορεί να εκτελέσει. Αυτό επιτυγχάνεται με τις παρακάτω κατευθυντήριες γραμμές;
- Πρόσβαση μέσω πληκτρολόγιου, όλη η λειτουργικότητα της διεπαφής με τη χρήση μόνο πληκτρολόγιου.
 - Επαρκής Χρόνος, οι χρήστες να έχουν το απαραίτητο χρόνο να διαβάσουν και να χρησιμοποιήσουν το περιεχόμενο.
 - Αποφυγή επιληπτικών κρίσεων: Αποφυγή παρουσίασης περιεχόμενου με τρόπο που είναι γνωστό ότι προκαλεί επιληπτικές και άλλες κρίσεις
 - Εύκολη πλοήγηση, μηχανισμοί που βοηθούν τους χρήστες να πλοηγηθούν, να βρίσκουν περιεχόμενο, και να προσδιορίζουν πού βρίσκονται.
- 3 Κατανοητό υλικό, οι πληροφορίες και η λειτουργία της διεπαφής θα πρέπει να είναι κατανοητές στο χρήστη. Αυτό επιτυγχάνεται με τις παρακάτω κατευθυντήριες γραμμές;
- Αναγνωσιμότητα, το περιεχόμενο να είναι αναγνώσιμο και κατανοητό όπως με τη χρήση επιλογής γλώσσας χρήστη.
 - Δημιουργία προβλέψιμου περιεχόμενο, οι ιστοσελίδες να εμφανίζονται και να λειτουργούν με προβλέψιμους τρόπους όπως με συνεπή μηχανισμό πλοήγησης χρήστη, κοινό σε όλες τις ιστοσελίδες, αλλά και συνεπή επιλογή χρωμάτων π.χ. η χρήση κόκκινου κειμένου είναι μόνο για όλα τα μηνύματα σφάλματος.
 - Εγκυρότητα εισαγωγής δεδομένων, μηχανισμοί έλεγχου και αυτόματης διόρθωσης εισαγωγής δεδομένων για αποφυγή λαθών από τους χρήστες.
- 4 Το περιεχόμενο να ερμηνεύεται το ίδιο αξιόπιστα από μια μεγάλη ποικιλία προγραμμάτων-πελατών από τα προγράμματα πλοήγησης μέχρι τις εξειδικευμένες τεχνολογίες υποστήριξης ατόμων με αναπηρίες. Κατευθυντήρια γραμμή:

- Συμβατότητα: Μεγιστοποίηση της συμβατότητας με σύγχρονα και πρόβλεψη για μελλοντικά προγράμματα-πελάτες, συμπεριλαμβανομένων των ειδικών υποστηρικτικών τεχνολογιών.

Εάν οποιαδήποτε από τις παραπάνω αρχές και οδηγίες δεν ακολουθούνται στη συγγραφή περιεχομένου για το διαδίκτυο, οι χρήστες με αναπηρίες δεν θα είναι σε θέση να το χρησιμοποιήσουν.

Οι 12 παραπάνω κατευθυντήριες γραμμές παρέχουν τους βασικούς στόχους που οι συγγραφείς πρέπει να εργαστούν προς την κατεύθυνση προκειμένου να καταστεί το περιεχόμενο πιο προσιτό στους χρήστες με διαφορετικές αναπηρίες. Οι κατευθυντήριες γραμμές δεν μπορούν να ελεγχτούν, αλλά παρουσιάζουν τους στόχους και το συνολικό πλαίσιο ώστε να βοηθήσει τους συγγραφείς να κατανοήσουν τα κριτήρια επιτυχίας και να εφαρμόσουν καλύτερα τις απαραίτητες τεχνικές.

2.11.4 Κριτήρια επιτυχίας

Για κάθε κατευθυντήρια γραμμή, έχουν οριστεί συγκεκριμένα κριτήρια επιτυχίας που επιτρέπουν στους συγγραφείς περιεχομένου, να χρησιμοποιούν μόνο αυτές τις γραμμές που απαιτούνται να επιτύχουν το βαθμό προσβασιμότητας που τους ζητείται συνήθως από τις συμβατικές συμφωνίες τους. Προκειμένου να γνωρίζουμε αν το περιεχόμενο μπορεί να ανταποκριθεί στις ανάγκες ομάδων με διαφορετικές αναπηρίες και διαφορετικές καταστάσεις, έχουν οριστεί τρία επίπεδα συμμόρφωσης τα ονομαζόμενα κριτήρια επιτυχίας: A (χαμηλότερη), AA, AAA και (υψηλότερο).

2.11.5 Τεχνικές Επάρκειας και Συμβουλευτικής

Για κάθε μία από τις κατευθυντήριες γραμμές και τα κριτήρια επιτυχίας στο έγγραφο WCAG 2.0 η ίδια ομάδα εργασίας έχει καταγράψει επίσης μια μεγάλη ποικιλία τεχνικών. Οι τεχνικές αυτές είναι ενημερωτικές και εμπίπτουν σε δύο κατηγορίες, αυτές που είναι επαρκείς για την ικανοποίηση των κριτηρίων επιτυχίας και εκείνες που είναι συμβουλευτικές. Οι συμβουλευτικές τεχνικές προχωράνε ένα επίπεδο παραπάνω από τα επιμέρους κριτήρια επιτυχίας και επιτρέπουν στους συντάκτες να αντιμετωπίσουν καλύτερα τις οδηγίες. Μερικές τεχνικές

συμβουλευτικής αφορούν την αντιμετώπιση των εμποδίων πρόσβασης που δεν καλύπτονται από τα ελέγξιμα κριτήρια επιτυχίας.

Όλα αυτά τα επίπεδα αναφοράς (αρχές, κατευθυντήριες γραμμές, κριτήρια επιτυχίας, επαρκείς και συμβουλευτικές τεχνικές) συνεργάζονται να παρέχουν καθοδήγηση για το πώς να διαμορφώνεται το περιεχόμενο ώστε να είναι πιο προσιτό. Οι συγγραφείς καλούνται να εφαρμόσουν όλα τα παραπάνω ώστε να αντιμετωπίζουν καλύτερα τις ανάγκες του ευρύτερου δυνατού φάσματος χρηστών.

Ακόμα όμως και το περιεχόμενο που συμμορφώνεται με το υψηλότερο επίπεδο (AAA) δεν θα είναι προσβάσιμο σε όλα τα άτομα με όλους τους τύπους, βαθμούς ή συνδυασμούς αναπηρίας, ιδιαίτερα με προβλήματα στο γνωστικό και στην εκμάθηση γλωσσών. Οι συγγραφείς καλούνται να εξετάσουν το πλήρες φάσμα των τεχνικών, συμπεριλαμβανομένων των συμβουλευτικών τεχνικών, καθώς και να αναζητήσουν την τρέχουσα βέλτιστη πρακτική με σκοπό να εξασφαλιστεί ότι το περιεχόμενο Web είναι προσβάσιμο, στο μέτρο του δυνατού, σε αυτή την κοινότητα.

2.11.6 WCAG 2.0 έγγραφα

Τα έγγραφα WCAG 2.0 καλύπτουν τις ανάγκες εκείνων που χρειάζονται ένα σταθερό τεχνικό πρότυπο ως αναφορά στην δημιουργία περιεχομένου προσβάσιμου σε άτομα με ειδικές ανάγκες.

Το WCAG 2.0 έχει σχεδιαστεί και λειτουργεί ως τεχνικό πρότυπο υλοποίησης. Ενώ όμως υπάρχει το πραγματικό τεχνικών προδιαγραφών έγγραφο, οι περισσότεροι προτιμούν να χρησιμοποιούν τους συνοδευτικούς οδηγούς, που έχουν αναφορές στο βασικό έγγραφο. Αναλυτικά οι τίτλοι:

- Εισαγωγή στο WCAG 2.0 (How to Meet WCAG 2.0), είναι ο οδηγός που διαβάζεται κυρίως και έχει το σκοπό της γρήγορης πρόσβασης στις μεθόδους και τις τεχνικές συγγραφής περιεχομένου που καλύπτουν τις κατευθυντήριες γραμμές και τα κριτήρια επιτυχίας. Αυτός ο οδηγός έχει συνέχεια αναφορές στο πραγματικό τεχνικό έγγραφο. Έχει τη δυνατότητα φιλτραρίσματος του περιεχομένου του, ώστε να εντοπίζεται κάτι συγκεκριμένο πιο γρήγορα.
- Κατανόηση WCAG 2.0 (Understanding WCAG 2.0), ένας οδηγός για την κατανόηση και την εφαρμογή του WCAG 2.0. Στη πραγματικότητα είναι ένας πιο λεπτομερής οδηγός και

συνήθως διαβάζεται σε συνέχεια του πρώτου οδηγού. Εννοείται ότι έχει ακόμα πιο πολλές αναφορές στο πραγματικό τεχνικό έγγραφο.

- Τεχνικές για το WCAG 2.0 (Techniques for WCAG 2.0), μια συλλογή από τεχνικές υλοποίησης του πρότυπου. Περιγράφει συγκεκριμένους τρόπους για τη δημιουργία προσβάσιμου περιεχομένου. Περιλαμβάνει γενικότερα αλλά και πιο συγκεκριμένα παραδείγματα υλοποίησης με τη χρήση HTML/XHTML, CSS, Javascript, και της εξειδικευμένης πλατφόρμας WAI-ARIA δημιουργίας περιεχομένου προσβάσιμου σε άτομα με ειδικές ανάγκες. Ταυτόχρονα, περιέχει και παραδείγματα προς αποφυγή.

2.12 Μέθοδοι αξιολόγησης ευχρηστίας

2.12.1 Εισαγωγή

Σε μεγάλο βαθμό, αυτό που κάνει κάτι εύχρηστο είναι η απουσία της απογοήτευσης στη χρήση του. Έτσι ο ορισμός της ευχρηστίας μπορεί να οριστεί ως «μια υπηρεσία πραγματικά είναι εύχρηστη, όταν ο χρήστης μπορεί να κάνει ότι θέλει, με τον τρόπο που αναμένει να το κάνει, χωρίς εμπόδια, δισταγμούς, ή ερωτήσεις. Για να είναι εύχρηστη μια υπηρεσία θα πρέπει να είναι ταυτόχρονα χρήσιμη, αποτελεσματική, αποδοτική, προσβάσιμη απ' όλους, εύκολη στην εκμάθηση της και να προσφέρει ικανοποίηση στη χρήση της [69].

Η χρησιμότητα (Usefulness) μιας υπηρεσίας υπολογίζεται σε ποιο βαθμό, επιτρέπει σε ένα χρήστη να επιτύχει τους στόχους του και εκτιμάται η προθυμία του χρήστη να την χρησιμοποιεί. Ένα σύστημα μπορεί να είναι εύκολο στη χρήση και στην εκμάθηση του, ακόμη και να προσφέρει ικανοποίηση στη χρήση του, αλλά αν δεν βοηθά στην επίτευξη των συγκεκριμένων στόχων, του συγκεκριμένου χρήστη, είναι αποτυχημένο.

Η αποδοτικότητα (Efficiency) είναι η ταχύτητα με την οποία ο στόχος του χρήστη μπορεί να επιτευχθεί πλήρως και με ακρίβεια και είναι συνήθως μετρήσιμη με το χρόνο. Για παράδειγμα, μπορεί να οριστεί ένα σημείο αναφοράς στις δοκιμές ευχρηστίας που λέει " 95 τοις εκατό από όλους τους χρήστες θα είναι σε θέση να φορτώσει το λογισμικό μέσα σε 10 λεπτά."

Η αποτελεσματικότητα (Effectiveness) αφορά το βαθμό στον οποίο η υπηρεσία συμπεριφέρεται με τον τρόπο που οι χρήστες αναμένουν και την ευκολία με την οποία οι χρήστες μπορούν να τη

χρησιμοποιήσουν για να κάνουν αυτό που σκοπεύουν. Αυτό συνήθως μετράται ποσοτικά με ποσοστό σφάλματος. Η επίτευξη αποτελεσματικότητας μιας υπηρεσίας στις δοκιμές, θα πρέπει να συνδέεται με κάποιο ποσοστό του συνόλου των χρηστών. Επεκτείνοντας το παράδειγμα από την απόδοση, το σημείο αναφοράς θα μπορούσε να εκφραστεί ως " 95 τοις εκατό από όλους τους χρήστες θα είναι σε θέση να φορτώσετε το λογισμικό σωστά με την πρώτη προσπάθεια.

Η Εκμάθηση (Learnability) είναι ένα μέρος της αποτελεσματικότητας και έχει να κάνει με την ικανότητα του χρήστη να λειτουργήσει το σύστημα σε κάποιο καθορισμένο επίπεδο επάρκειας μετά από κάποια προκαθορισμένη περίοδο εκπαίδευσης. Μπορεί, επίσης, να αναφέρεται στην ικανότητα των όχι συχνών χρηστών να ξαναμάθουν το σύστημα μετά από περιόδους μη χρήσης του.

Η Ικανοποίηση (Satisfaction) αναφέρεται στα συναισθήματα και τις γνώμες των χρηστών για μια υπηρεσία και συνήθως εκφράζονται μέσω γραπτών και προφορικών ερωτήσεων. Οι χρήστες είναι πιο πιθανό να αποδίδουν καλά σε ένα προϊόν που ανταποκρίνεται στις ανάγκες τους και να παρέχει ικανοποίηση από ένα που δεν το κάνει. Συνήθως, οι χρήστες καλούνται να βαθμολογήσουν και να κατατάξουν τις υπηρεσίες που δοκιμάζουν, και αυτό μπορεί να αποκαλύψει συχνά αίτια και λόγους για τα προβλήματα που προκύπτουν.

Η προσβασιμότητα (Accessibility) όπως έχουμε ήδη αναφέρει, εξετάζει σε ποιο βαθμό μια υπηρεσία μπορεί να χρησιμοποιηθεί από τους ανθρώπους που έχουν ειδικές ανάγκες. Αν μια εφαρμογή μπορεί να χρησιμοποιηθεί από άτομα με ειδικές ανάγκες, αυτό πάντα ωφελεί και τους ανθρώπους που δεν έχουν αναπηρίες. Λαμβάνοντας υπόψη την προσβασιμότητα, η σχεδίαση της διεπαφής απλοποιείται και βοηθά στη μεγαλύτερη ευχρηστία της υπηρεσίας.

Όλες οι παραπάνω ιδιότητες είναι απαραίτητες, για να είναι μια υπηρεσία εύχρηστη. Το κακό είναι ότι η αντίληψη και οι ικανότητες κάθε ατόμου είναι μοναδικές και είναι αδύνατον να μετρηθεί απόλυτα ο βαθμός ευχρηστίας για όλους τους χρήστες. Αυτό που μπορεί να μετρηθεί, πιο αντικειμενικά, είναι πόσο μη εύχρηστη είναι μια υπηρεσία και η ενσωμάτωση των μεθόδων αξιολόγησης, όπως οι δοκιμές ευχρηστίας σε όλη την επαναληπτική διαδικασία σχεδιασμού του λογισμικού, είναι δυνατόν, να κάνει τις υπηρεσίες πιο εύχρηστες για περισσότερα άτομα.

2.12.2 Μέθοδοι αξιολόγησης

Η συνεχής αξιολόγηση της ευχρηστίας του συστήματος κατά τη σχεδίαση, την ανάπτυξη πρωτοτύπων και τη λειτουργία του αποτελεί βασική αρχή του ανθρωποκεντρικού σχεδιασμού (user-centered design).

Ανάλογα με τη φάση ανάπτυξης του προϊόντος η αξιολόγηση διακρίνεται [67]:

- διαμορφωτική αξιολόγηση (formative evaluation), η οποία έχει ως στόχο τη διαμόρφωση, δηλαδή τη βελτίωση της ευχρηστίας του συστήματος και γίνεται κατά τη φάση ανάπτυξης του προϊόντος
- συμπερασματική αξιολόγηση (summative evaluation) που αφορά το ολοκληρωμένο προϊόν και γίνεται συνήθως πριν τεθεί σε λειτουργία το σύστημα, ώστε να μετρηθεί η ευχρηστία του είτε σε σύγκριση με άλλα ανταγωνιστικά συστήματα είτε σε σχέση με τις προδιαγραφές ευχρηστίας

Παράλληλα, υπάρχει μια σειρά από διάφορες τεχνικές που μπορούν να χρησιμοποιηθούν για τη μέτρηση της ευχρηστίας και διακρίνονται σε:

- αναλυτικές, στηρίζονται σε θεωρητικά μοντέλα τα οποία προσομοιώνουν τη συμπεριφορά του χρήστη και την συμβατότητα της εφαρμογής με αναγνωρισμένα πρότυπα και κανόνες και για το λόγο αυτό μπορούν να χρησιμοποιηθούν κατά τη φάση σύνταξης προδιαγραφών, χωρίς να απαιτείται η συμμετοχή των χρηστών,
- εμπειρικές, στηρίζονται στην αξιολόγηση της συμπεριφοράς ή των χαρακτηριστικών κατά την πιλοτική λειτουργία του συστήματος. Στη διαδικασία μπορούν να συμμετέχουν αντιπροσωπευτικοί χρήστες και ειδικοί ευχρηστίας λογισμικού

2.12.3 Μέθοδος ελέγχου συμβατότητας με κανόνες και πρότυπα

Στη μέθοδο αυτή, το μέσο αξιολόγησης, είναι λεπτομερείς κανόνες σχεδιασμού εύχρηστων συστημάτων (design guidelines) που είναι γνωστοί στους αξιολογητές. Οι κανόνες παίρνουν τη μορφή ενός καταλόγου ελέγχου (checklist). Ο κάθε αξιολογητής αποφαινεται αν ικανοποιείται ο κανόνας από το σύστημα. Στον κατάλογο επισυνάπτονται, επίσης, σχόλια και υποδείξεις βελτίωσης του συστήματος, ώστε να καταστεί δυνατή η ικανοποίηση του αντίστοιχου κανόνα [67].

Οι κανόνες και τα διεθνή πρότυπα (standards) που χρησιμοποιούνται είτε είναι γενικής φύσεως για την σχεδίαση μιας οποιαδήποτε εφαρμογής, είτε αναφέρονται στην κατηγορία εφαρμογών όπως διαδικτυακές εφαρμογές.

Υπάρχουν στη βιβλιογραφία [70] πολλές πλήρεις συλλογές κανόνων και προτύπων, οι οποίες προέρχονται είτε από τη συσσωρευμένη εμπειρία ομάδων ανάπτυξης λογισμικού είτε από οδηγίες που έχουν δημιουργήσει εταιρείες κατασκευής λογισμικού ανάπτυξης συστημάτων.

2.12.4 Διερευνητικοί μέθοδοι

Οι μέθοδοι αυτές αφορούν τη διερεύνηση των χαρακτηριστικών ευχρηστίας ενός συστήματος μέσω καταγραφής των απόψεων του χρήστη. Οι διαδικασίες αξιολόγησης της κατηγορίας αυτής συνήθως γίνονται εκτός του εργαστηρίου, στο φυσικό χώρο του χρήστη, και δεν απαιτούν ιδιαίτερο εξοπλισμό. Χρησιμοποιούνται ως τεχνικές συμπερασματικής αξιολόγησης, αφού το προϊόν βρίσκεται ήδη σε λειτουργία. Οι μέθοδοι αυτές απαιτούν τη συμμετοχή μεγάλου συνήθως αριθμού χρηστών. Οι κύριες μέθοδοι της κατηγορίας αυτής είναι [67]:

- οι συνεντεύξεις χρηστών (User interviews),
- η συμπλήρωση ερωτηματολογίων από τους χρήστες (user questionnaires),
- η σύσταση και λειτουργία ομάδων αξιολόγησης του συστήματος από τους χρήστες (focus groups),
- οι αναφορές συμβάντων εκ μέρους των χρηστών (self reporting),
- οι εθνογραφικές παρατηρήσεις πεδίου (field observation).

2.12.5 Συμπλήρωση ερωτηματολογίων

Η μέθοδος συμπλήρωσης ερωτηματολογίων (questionnaires) είναι μια δοκιμασμένη μέθοδος καταγραφής της αντίδρασης των χρηστών στη χρήση ενός λογισμικού. Σε αντίθεση με τη συνέντευξη, το ερωτηματολόγιο είναι λιγότερο ευέλικτο, αφού οι ερωτήσεις είναι καταγραμμένες και δε μπορούν να αλλάξουν κατά περίπτωση, όμως, για ακριβώς αυτόν το λόγο η διαχείριση ερωτηματολογίων και η συναγωγή συμπερασμάτων είναι πολύ πιο εύκολη [67].

Το ερωτηματολόγιο είναι καλό να διαχυθεί σε μεγάλο πλήθος χρηστών. Η ευκρίνεια στη διατύπωση των ερωτήσεων είναι βασικό χαρακτηριστικό των ερωτηματολογίων, όπως περιγράφεται και στην επόμενη ενότητα, στην οποία περιέχονται οδηγίες για τη συμπλήρωση ερωτηματολογίων.

Ένα από τα μειονεκτήματα της μεθόδου είναι ότι τα συμπεράσματα που συνάγονται στηρίζονται μόνο στην υποκειμενική άποψη των χρηστών για το σύστημα. Πολλές φορές οι χρήστες φαίνεται να εκφράζουν θετικές ή αρνητικές κρίσεις, επειδή αυτό κατά τη γνώμη τους πρέπει να ειπωθεί και όχι επειδή αυτό προκύπτει από την προσωπική τους εμπειρία.

Ένα καλό ερωτηματολόγιο θα πρέπει να έχει τα εξής χαρακτηριστικά [69]:

- Το ερωτηματολόγιο θα πρέπει να αποτελείται από 20 ως 40 ερωτήσεις.
- Βάθος ερωτήσεων: Η απάντηση σε κάθε ερώτηση πρέπει να δίνει αρκετή πληροφορία για κάθε σημείο που θέλουμε να αξιολογήσουμε.
- Πληρότητα: Το σύνολο όλων των απαντήσεων πρέπει να δίνει αρκετή πληροφορία για όλους τους στόχους της αξιολόγησης.
- Ανεξαρτησία: Πρέπει κάθε διαφορετική απάντηση να δίνει διαφορετικό είδος πληροφορίας.
- Ισορροπία: Το σύνολο των ερωτήσεων πρέπει να καλύπτει όλα τα ερωτήματα που θέλουμε να ερευνήσουμε.

Τα ερωτηματολόγια πρέπει να είναι όσο πιο απλά και κατανοητά γίνεται και παράλληλα κατά το δυνατόν η συμπλήρωσή τους να μην είναι χρονοβόρα. Οι ερωτήσεις πρέπει να είναι ευκρινείς και, αν απαιτείται, πρέπει να παρέχονται διευκρινήσεις για τη σημασία των απαντήσεων, π.χ. δεν πρέπει να ζητήσουμε την αξιολόγηση του δείκτη ικανοποίησης ενός χρήστη από το σύστημα σε κλίμακα 1–5 και να μην είναι σαφές σε τι αντιστοιχεί η τιμή «5». Μια καλή πρακτική είναι η πειραματική εφαρμογή του ερωτηματολογίου σε ένα μικρό δείγμα, έτσι ώστε να εξασφαλίσουμε ότι είναι ισορροπημένο και εύστοχο.

2.13 Σύνοψη

Στο κεφάλαιο αυτό περιλαμβάνεται η βιβλιογραφική επισκόπηση σε θέματα ανάλυσης, σχεδίασης και αξιολόγησης διαδικτυακών πυλών.

Στην ενότητα 2.2 έγινε παρουσιάστηκαν θέματα που αφορούν στη λειτουργικότητα και στις θετικές επιπτώσεις της εξατομίκευσης του πληροφοριακού υλικού και των υπηρεσιών για τις διαδικτυακές πύλες. Στην ενότητα 2.3 έγινε περιγραφή του portlet, του βασικού αντικείμενου μιας πύλης, καθώς κάθε ιστοσελίδα μιας πύλης αποτελείται από ένα ή περισσότερα portlets. Η ενότητα 2.4 αναφέρθηκε στα τεχνολογικά πρότυπα που έχουν οριστεί για την ανάπτυξη των portlets και προσφέρουν συμβατότητα στη χρήση τους σε διαφορετικές πύλες. Εστίασαμε στη αρχιτεκτονική μιας διαδικτυακής πύλης στην ενότητα 2.5 ενώ οι ενότητες 2.6 και 2.7 έγινε περιγραφή των κύριων μεθόδων πιστοποίησης και εξουσιοδότησης ενός χρήστη. Η ενότητα 2.8 αναφέρθηκε στο HTTPS, το πρωτόκολλο ασφαλούς μεταφοράς δεδομένων μεταξύ εξυπηρετητών ιστοσελίδων, η ενότητα 2.9 στη κρυπτογράφηση δεδομένων με σκοπό την προστασία από υποκλοπή και η ενότητα 2.10 περιγράφει τη μέθοδο επικύρωσης ταυτότητας του ίδιου του εξυπηρετητή ιστοσελίδων με τη χρήση πιστοποιητικών. Η ενότητα 2.11 περιέγραψε τις οδηγίες προσβασιμότητας Web Content Accessibility Guidelines (WAI/WCAG) του W3C ώστε να αποφεύγεται ο αποκλεισμός ειδικών ομάδων πολιτών (AMEA). Στην ενότητα 2.12 περιγράψαμε τις μεθόδους αξιολόγησης ιστοχώρων. Τέλος η ενότητα 2.13 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τον επιμέρους στόχο #1 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Το επόμενο κεφαλαίο θα εξετάσει τη παρούσα κατάσταση στο ΤΕΙ Κρήτης και θα γίνει ανάλυση απαιτήσεων του νέου συστήματος που θα μας βοηθήσουν να τεκμηριώσουμε το επιμέρους στόχο #2 όπως αυτός αναφέρεται στην εισαγωγή.

Κεφάλαιο 3

Ανάλυση Απαιτήσεων

3.1 Εισαγωγή

Στο κεφάλαιο αυτό θα γίνει καταγραφή της παρούσας κατάστασης στο ΤΕΙ Κρήτης και θα γίνει ανάλυση απαιτήσεων για τη προτεινόμενη διαδικτυακή μας πύλη.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Η ενότητα 3.2 αναφέρεται στην εκπαιδευτική δομή του ιδρύματος. Η ενότητα 3.3 περιγράφει την τρέχουσα κατάσταση υποστήριξης χρηστών στο ίδρυμα και η ενότητα 3.4 καταγράφει την δομή των υπολοίπων τμημάτων και υπηρεσιών του ΤΕΙ Κρήτης. Η ενότητα 3.5 ορίζει τις υπηρεσίες και ιστοσελίδες που θα καλύπτει το νέο σύστημα ενώ στην ενότητα 3.6 ορίζονται οι κατηγορίες χρηστών που θα αναγνωρίζει το νέο σύστημα. Η ενότητα 3.7 περιγράφει τη απαιτούμενη λειτουργικότητα της πύλης με την εξατομικευμένη προσωπική σελίδα κάθε χρήστη. Τέλος η ενότητα 3.8 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

3.2 Επισκόπηση ΤΕΙ Κρήτης

Το Τ.Ε.Ι. Κρήτης περιλαμβάνει τις παρακάτω σχολές:

- Η Σχολή Τεχνολογικών Εφαρμογών (Σ.Τ.Ε.Φ) του Τ.Ε.Ι. Κρήτης είναι ανεπτυγμένη σε τρεις πόλεις της Κρήτης, το Ηράκλειο, το Ρέθυμνο και τα Χανιά. Τα τμήματα της Σ.Τ.Ε.Φ που εδρεύουν στο Ηράκλειο είναι τα εξής: Ηλεκτρολογίας, Μηχανολογίας, Πολιτικών Δομικών Έργων, Εφαρμοσμένης Πληροφορικής και Πολυμέσων, Γενικό Τμήμα Θετικών Επιστημών. Στο Παράρτημα Ρεθύμνου λειτουργεί το τμήμα Μουσικής Τεχνολογίας & Ακουστικής. Στο Παράρτημα Χανίων εδρεύουν τα τμήματα Ηλεκτρονικής και Φυσικών Πόρων & Περιβάλλοντος.
- Η Σχολή Διοίκησης και Οικονομίας (Σ.Δ.Ο.) του ΤΕΙ Κρήτης είναι ανεπτυγμένη σε τρεις πόλεις της Κρήτης, το Ηράκλειο, τον Άγιο Νικόλαο και την Ιεράπετρα. Τα τμήματα που εδρεύουν στο Ηράκλειο είναι τα εξής: Λογιστικής, Τουριστικών Επιχειρήσεων και Διοίκησης Επιχειρήσεων. Στον Άγιο Νικόλαο λειτουργούν τα τμήματα Χρηματοοικονομικής και Ασφαλιστικής και το Επιχειρηματικού Σχεδιασμού και Πληροφοριακών Συστημάτων και τέλος στην Ιεράπετρα το τμήμα Εμπορίας και Διαφήμισης.
- Η Σχολή Επαγγελματιών Υγείας & Πρόνοιας (Σ.Ε.Υ.Π.) του ΤΕΙ Κρήτης είναι ανεπτυγμένη σε δύο πόλεις της Κρήτης το Ηράκλειο και τη Σητεία. Στο Ηράκλειο εδρεύουν τα τμήματα Κοινωνικής Εργασίας και Νοσηλευτικής και στη Σητεία το τμήμα Διατροφής & Διαιτολογίας.
- Η Σχολή Τεχνολογίας Γεωπονίας (Σ.Τε.Γ.) του ΤΕΙ Κρήτης εδρεύει στο Ηράκλειο και απαρτίζεται από τα Τμήματα Φυτικής Παραγωγής και Βιολογικών Θερμοκηπιακών Καλλιέργειών και Ανθοκομίας.

3.3 Τρέχουσα κατάσταση υποστήριξης χρηστών στο ΤΕΙ Κρήτης

Μέχρι πριν ένα χρόνο υπήρχαν κάτω από το κεντρικό τομέα TEICRETE.GR διάφοροι άλλοι, όπως τομείς (κατάλογοι χρηστών) ανά σχολή (stef.teicrete.gr, sdo.teicrete.gr, steg.teicrete.gr, seyp.teicrete.gr), τομέας διοικητικών υπαλλήλων (admin.teicrete.gr), κάποιοι τομείς τμημάτων

όπως app.teicrete.gr & cs.teicrete.gr, ακόμα και γραφείων ή προγραμμάτων όπως το Γρ. Διασύνδεσης (career.teicrete.gr).

Ταυτόχρονα, υπήρχαν διαδικτυακές υπηρεσίες, κάθε μια με τη δική της βάση χρηστών. Αυτό σημαίνει ότι ο χρήστης αυτής της υπηρεσίας συνδεόταν δίνοντας τα διαπιστευτήρια (username/password) που είχε ειδικά για αυτήν. Η πλειοψηφία των χρηστών κάθε φορά είχε ανοίξει λογαριασμούς με διαφορετικά διαπιστευτήρια σε κάθε υπηρεσία με τη λογική της μεγαλύτερης ασφάλειας. Βέβαια, κάτι τέτοιο είχε το αυτονόητο πρόβλημα υποστήριξης, ο χρήστης προφανώς δεν μπορούσε να θυμάται τα διαπιστευτήρια αυτά ανά υπηρεσία και καθημερινά το helpdesk του τμήματος πληροφορικής είχε λειτουργική επιβάρυνση από τις κλήσεις υπενθύμισης και επαναφοράς των διαπιστευτηρίων λησμονούντες χρήστες.

Πριν ένα χρόνο, όμως, και με τελικό στόχο την διασύνδεση των διαπιστευτηρίων όλων των συνδεδεμένων ιδρυμάτων (Shibboleth) άρχισε η επίμονη και δύσκολη εργασία μεταφοράς των λογαριασμών των χρηστών σε ένα ενιαίο κεντρικό κατάλογο. Έτσι, έγινε ενοποίηση των τομέων των σχολών με το τομέα διοικητικών υπαλλήλων κάτω από ένα νέο τομέα staff.teicrete.gr. Κάποιες υπηρεσίες, όπως το ηλ. ταχυδρομείο που είναι ίσως η πιο βασική ακολούθησαν, όπως ήταν λογικό, άμεσα την υποστήριξη του νέου τομέα.

Παραμένουν, όμως, ακόμα για διάφορους λόγους κάποιοι ξεχωριστοί τομείς σε διάφορα τμήματα και γραφεία.

Το θέμα της ενοποίησης όλων των τομέων καθυστερεί υπερβολικά και ο λόγος είναι ότι στο ίδρυμα υπάρχουν δυο διαχειριστικές αρχές για τις υπηρεσίες πληροφορικής, το Κέντρο Ελέγχου & Διαχείρισης Δικτύων (ΚΕΕΔ) και το τμήμα πληροφορικής της διοίκησης (Τ.Π.Δ), το καθένα με διαφορετική φιλοσοφία και διαφορετικές τεχνολογίες καταλόγων δικτύου (Τ.Π.Δ.: microsoft-active directory, ΚΕΕΔ: open source-linux & openldap) και με διαφορετικές υπηρεσίες [46].

Η ενοποίηση των διαχειριστικών αρχών έχει αποφασιστεί να γίνει μελλοντικά αλλά αυτό δεν έπρεπε να εμποδίζει την διασύνδεση μας με το Shibboleth. Έτσι, αποφασίστηκε η χρήση ενός κοινού LDAP το οποίο θα ενημερώνεται καθημερινά από τους επιμέρους καταλόγους δικτύων και θα έχει κεντρικά τη βάση όλων των χρηστών του ιδρύματος ώστε να μπορέσουμε να συνδεθούμε άμεσα με το Shibboleth. Υπενθυμίζεται ότι η δυνατότητα ανταλλαγής δεδομένων με την τυποποιημένη προδιαγραφή (LDAP Data Interchange Format, LDIF) που υποστηρίζουν όλοι οι κατάλογοι δικτύου επιτρέπει αυτή την προσέγγιση.

Όλη η παραπάνω ανάλυση καταδεικνύει ότι η διαδικασία των ενοποιήσεων τομέων υπήρξε ένα βασικό πρόβλημα που ακόμα και τώρα δεν έχει επιλυθεί. Η ενοποίηση τομέων προϋπόθετε ένα ξεκαθάρισμα λογαριασμών χρηστών ώστε να υπάρχει ένας μοναδικός λογαριασμός για κάθε χρήστη όπως είναι και η λογική του ενός και μοναδικού διαπιστευτηρίου για κάθε χρήστη και κύρια συνθήκη λειτουργίας του Shibboleth. Αυτό δεν έχει ολοκληρωθεί μέχρι τώρα. Υπάρχουν, μεν, userid με τη μορφή του email π.χ. username@teicrete.gr, όπως έχει συμφωνηθεί σε επίπεδο Shibboleth, αλλά η χρήση ενός κοινού «προς τα έξω» LDAP, λειτουργεί σαν υπερσύνολο βάσης λογαριασμών και έτσι ο κάθε χρήστης εξακολουθεί να έχει διάφορους λογαριασμούς τύπου ggeorge@cs.teicrete.gr, ή ggeorge@career.teicrete.gr ανάλογα σε ποιους τομείς έχει πρόσβαση, ο κάθε λογαριασμός χαρακτηρίζεται με βάση το ρόλο που έχει στο συγκεκριμένο τομέα.

Η ύπαρξη διαφορετικών λογαριασμών ανά χρήστη σε συνδυασμό με το γεγονός ότι ο κάθε ένας έχει διαφορετικό ρόλο δεν επιτρέπει την εύκολη υποστήριξη πολλαπλών ρόλων σε κάθε χρήστη της πύλης μας.

Εμείς, όμως, έπρεπε σχεδιαστικά να προβλέψουμε την μελλοντική αυτή δυνατότητα και για αυτό προχωρήσαμε στην υλοποίηση μιας τέτοιας λειτουργίας. Έτσι, αν ένας χρήστης έχει περισσότερους από ένα ρόλους θα έχει πρόσβαση και σε όλες τις υπηρεσίες που έχουν συσχετιστεί με αυτούς τους ρόλους.

3.4 Καταγραφή τμημάτων & υπηρεσιών

- Κεντρικές Υπηρεσίες
 - Ειδικός Λογαριασμός
 - Γραφείο Διασύνδεσης
 - Κεντρική Βιβλιοθήκη
 - Τμήμα Διεθνών Σχέσεων
 - Τμήμα Ερευνητικών Προγραμμάτων
 - Τμήμα Παιδικού Σταθμού

- Τμήμα Περιθαλψης & Κοινωνικής Μέριμνας
- Τμήμα Πληροφορικής
- Τμήμα Σπουδαστικής Μέριμνας
- Τμήμα Σπουδών & Πρακτικής Άσκησης
- Διευθύνσεις
 - Διεύθυνση Διοικητικού
 - Τμήμα Διοικητικού & ΕΤΠ
 - Τμήμα Διοικητικής Μέριμνας & Πρωτοκόλλου
 - Τμήμα Εκπαιδευτικού
 - Τμήμα Σπουδών & Πρακτικής Άσκησης
 - Τμήμα Συλλογικών & Ατομικών Οργάνων & Επιτροπών
 - Γραφείο ΠΣΕΑ
 - Διεύθυνση Οικονομικού
 - Τμήμα Δαπανών & ΚΠΣ
 - Τμήμα Μισθοδοσίας
 - Τμήμα Προμηθειών
 - Τμήμα Προϋπολογισμού
 - Διεύθυνση Τεχνικών Υπηρεσιών
 - Τμήμα Μελετών - Κατασκευών

- Τμήμα Συντήρησης
- Διεύθυνση Αγροκτήματος
 - Τμήμα Εκμετάλλευσης - Προγ/σμού
- Διοίκηση Παραρτημάτων
 - Διοίκηση Παραρτήματος Λασιθίου
 - Διοίκηση Παραρτήματος Ρεθύμνου
 - Διοίκηση Παραρτήματος Χανίων
- Επιπλέον υπηρεσίες
 - Βιβλιοθήκη
 - Γραφείο Δια Βίου Εκπαίδευσης
 - Γραφείο Διαμεσολάβησης
 - Γραφείο Διασύνδεσης
 - Γραφείο Πρακτικής Άσκησης
 - Δομή Απασχόλησης και Σταδιοδρομίας
 - Ειδικός Λογαριασμός
 - Κέντρο Διαχείρισης Δικτύων
 - Σπουδαστική Μέριμνα
 - Τμήμα Δημοσίων & Διεθνών Σχέσεων
 - Τμήμα Πληροφορικής - Κεντρική Διοίκηση

Κατά την διάρκεια της συλλογής πληροφοριών ήταν φανερό ότι λόγω του μεγέθους του ιδρύματος δεν θα μπορούσε να γίνει ενσωμάτωση όλων των πολλών δεκάδων ιστοσελίδων στη πύλη μας και προσπαθήσαμε να συλλέξουμε πληροφοριακά στοιχεία επισκεψιμότητας των παραπάνω ιστοτόπων. Δεν μπορέσαμε να βρούμε αρκετές πληροφορίες αφού η επισκεψιμότητα δεν καταγράφεται τις περισσότερες φορές και οι διεθνώς αναγνωρισμένες υπηρεσίες μέτρησης αυτής όπως η alexa.com δεν επιμερίζει τη συλλογική μέτρηση ανά τμήματα. Έτσι προχωρήσαμε σε ενσωμάτωση περιεχομένου και υπηρεσιών ιστοσελίδων του ιδρύματος σύμφωνα με την εμπειρία μας.

3.5 Κατάλογος ιστοσελίδων προς υλοποίηση

Από τους δεκάδες ιστοχώρους που βρίσκονται κάτω από την ομπρέλα του domain teirete.gr αποφασίσαμε αρχικά σε ενσωμάτωση και πρόσβαση των παρακάτω ιστοσελίδων μέσα από τη διαδικτυακή πύλη.

- Ιστοσελίδες Σχολών
- Ιστοσελίδες Τμημάτων
- Κεντρική ιστοσελίδα ιδρύματος
- Αναζήτηση βιβλίου από Κατάλογος Βιβλιοθήκης του Τ.Ε.Ι. Κρήτης
- Ιστοσελίδα τμήματος Σπουδαστικής Μέριμνας
- Τμήμα Δημοσίων & Διεθνών Σχέσεων ΤΕΙ Κρήτης
- ΔΑΣΤΑ ΤΕΙ Κρήτης (Μονάδα Καινοτομίας και Επιχειρηματικότητας, Γραφείο Πρακτικής Άσκησης, Γρ. Διασύνδεσης
- Πληροφοριακό σύστημα Γρ. Διασύνδεσης
- Προκηρύξεις διαγωνισμών-θέσεων εργασίας

- Αναζήτηση προσωπικού
- Κέντρο Ελέγχου και Διαχείρισης Δικτύων (ΚΕΔΔ)
- Γενικό Τμήμα Θετικών Επιστημών - ΤΕΙ Κρήτης
- Email
- Πρόγραμμα γραμματειών τμημάτων
- E-class

3.6 Κατηγορίες χρηστών

Σύμφωνα και με τα παραπάνω έχουμε κάποιους ξεκάθαρους ρόλους χρηστών:

- Επισκέπτης χωρίς λογαριασμό
- Χρήστης (σύνδεση με λογαριασμό)
- Φοιτητής
- Πτυχιούχος
- Διοικητικός Υπάλληλος
- Καθηγητής
- Διαχειριστής Συστήματος

3.7 Περιγραφή λειτουργικότητας του συστήματος

3.7.1 Εισαγωγή

Πρόσβαση στην πύλη θα έχει κάποιος από την κεντρική ιστοσελίδα του εκπαιδευτικού ιδρύματος και από τις βασικές επιλογές του κεντρικού μενού συνδέσμων και με την ονομασία myTEI. Η πύλη θα είναι διαθέσιμη σε ιστοσελίδα της μορφής <http://myTEI.teicrete.gr>.

Στη αρχική σελίδα της πύλης ενημερώνεται ο επισκέπτης για το πώς λειτουργεί και πως αποκτά πρόσβαση στις προσωποποιημένες υπηρεσίες ενώ ταυτόχρονα θα του προτείνει σύνδεση με το λογαριασμό του.

Σε κάθε ιστοσελίδα και ανάλογα την σύνδεση ή όχι του χρήστη, τα δικαιώματα και την ιδιότητα του, προτείνεται η λίστα με τα διαθέσιμα portlets που έχει πρόσβαση. Ανάλογα υπάρχουν προεπιλεγμένα λίγα βασικά κατά την γνώμη μας portlets που η πλειοψηφία των χρηστών χρησιμοποιεί. Αργότερα αυτά τα portlets θα τυποποιηθούν βασιζόμενοι σε πραγματικά δεδομένα χρήσης τους από τους επισκέπτες της πύλης με σκοπό να εμφανίζονται τα πιο «αγαπημένα» από τους ίδιους τους χρήστες.

3.7.2 Προσωπική σελίδα χρήστη

Ο κάθε χρήστης αφού συνδεθεί επιτυχημένα στο σύστημα έχει πρόσβαση στην εξατομικευμένη σελίδα του. Αν δεν έχει δημιουργήσει προσωπική σελίδα ή συνδεθεί για πρώτη φορά βλέπει αυτήν που έχει σχεδιαστεί και με προτεινόμενα portlets σύμφωνα με το ρόλο ή τους ρόλους που του αναγνωρίζει το σύστημα και τα δικά του δικαιώματα.

Ο χρήστης έχει τη δυνατότητα να αλλάξει την εμφάνιση του περιεχομένου της σελίδας του, τόσο από πλευράς αισθητικής (αλλαγή CSS θέματος) όσο και λειτουργικότητας με αναδιάταξη περιεχομένου και portlets σε μια, δυο ή τρεις στήλες με τη χρήση drag & drop για την μετακίνηση των portlets σε συγκεκριμένη στήλη ή σειρά ή ανταλλαγή θέσης μεταξύ 2 portlets.

Η διαδικτυακή πύλη uportal προσφέρει τη δυνατότητα εμφάνισης της λίστας των διαθέσιμων portlets, ανάλογα το ρόλο και τα δικαιώματα του χρήστη που έχει συνδεθεί. Ο χρήστης έχει την επιλογή μέσα από τον κατάλογο των προσφερόμενων, να προσθέσει μόνιμα το επιθυμητό portlet στη εξατομικευμένη προσωπική του σελίδα, ώστε να είναι άμεσα διαθέσιμο κάθε φορά που συνδέεται ή να το επιλέξει για να το χρησιμοποιήσει εκείνη τη στιγμή μόνο. Στη τελευταία περίπτωση, το παράθυρο του portlet εμφανίζεται σε μεγιστοποιημένη μορφή, καταλαμβάνει δηλαδή όλο το χώρο της σελίδας του χρήστη. Για λόγους σωστής λειτουργίας, κάποια

πληροφοριακό υλικό, όπως η λίστα με τα διαθέσιμα portlets δεν θα έχει τη δυνατότητα να το αφαιρέσει.

Προκειμένου να γίνει ο διαχωρισμός μεταξύ των υπηρεσιών-portlets που προσφέρει το σύστημα και έχουν αναπτυχτεί ειδικά για τις ανάγκες του ιδρύματος, ώστε να τα ξεχωρίζει ο χρήστης, τα portlets αυτά έχουν το δικό τους εικονίδιο με τη μορφή ενός ταύρου που είναι και το έμβλημα του ιδρύματος του ΤΕΙ Κρήτης.

Σε κάθε σελίδα θα είναι δυνατή η επιλογή αλλαγής γλώσσας (αρχικά μεταξύ ελληνικών και αγγλικών) ώστε τα portlets που υποστηρίζουν και έχουν μεταγλωττισμένο υλικό να εμφανίζουν ανάλογα το περιεχόμενό τους.

Η πύλη βοηθά την πρόσβαση ατόμων με ειδικές ανάγκες με τη χρήση εξειδικευμένων γραφικών θεμάτων με πιο έντονο κοντράστ και λιγότερα χρώματα και με δυνατότητα πλοήγησης με τη χρήση πληκτρολόγιου.

Όλες οι παραπάνω αλλαγές αποθηκεύονται στο ατομικό λογαριασμό του χρήστη, ώστε την επόμενη φορά που θα συνδεθεί, να εμφανίσει την προσωπική του σελίδα σύμφωνα με τις επιλογές του.

Επιπλέον, κάθε φορά που το επιθυμεί, έχει τη δυνατότητα επαναφοράς της προσωπικής του σελίδας, στη μορφή της προεπιλεγμένης που έχει σχεδιαστεί, σύμφωνα με το ρόλο και τη ιδιότητα του.

3.8 Σύνοψη

Στο κεφάλαιο έγινε καταγραφή της παρούσας κατάστασης στο ΤΕΙ Κρήτης ορισμός των υπηρεσιών και των χρηστών που θα υποστηρίζει η διαδικτυακή μας πύλη και περιγράφηκε η ζητούμενη λειτουργικότητα της.

Έτσι στην ενότητα 3.2 έγινε μια εισαγωγή στην εκπαιδευτική δομή του ιδρύματος. Η ενότητα 3.3 περιέγραψε την τρέχουσα κατάσταση υποστήριξης χρηστών στο ίδρυμα και η ενότητα 3.4 εξήγησε την δομή των υπολοίπων τμημάτων και υπηρεσιών του ΤΕΙ Κρήτης. Στην ενότητα 3.5 ορίστηκαν οι υπηρεσίες και ιστοσελίδες που θα καλύπτει το νέο σύστημα ενώ στην ενότητα 3.6 δηλώθηκαν οι κατηγορίες χρηστών που θα αναγνωρίζει το νέο σύστημα. Η ενότητα 3.7

περιέγραψε την απαιτούμενη λειτουργικότητα της πύλης. Τέλος, η ενότητα 3.8 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τον επιμέρους στόχο #2 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Το επόμενο κεφαλαίο θα εξετάσει τα εργαλεία ανάπτυξης και τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της ως τεκμηρίωση του επιμέρους στόχου #3 όπως αυτός αναφέρεται στην εισαγωγή.

Κεφάλαιο 4

Μεθοδολογική Προσέγγιση & Εργαλεία Ανάπτυξης

4.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζονται η μεθοδολογική προσέγγιση, τα εργαλεία ανάπτυξης και οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της διαδικτυακής πύλης και των προσφερόμενων διαδικτυακών υπηρεσιών.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Η ενότητα 4.2 αναφέρεται για λόγους πληρότητας στη μεθοδολογία ICONIX και στα διαγράμματα που χρησιμοποιήθηκαν για το βασικό use case της σύνδεσης του χρήστη με το σύστημα. Η ενότητα 4.3 αναφέρεται στην πλατφόρμα ανοιχτού κώδικα η οποία είναι το κεντρικό εργαλείο ανάπτυξης για την δημιουργία της εκπαιδευτική πύλης μας. Η ενότητα 4.4 περιγράφει το επιλεγμένο σύστημα ταυτοποίησης χρηστών, ενώ η ενότητα 4.5 περιγράφει την εφαρμογή ανοιχτού κώδικα που βασιστήκαμε για την υποστήριξη φορητών συσκευών όπως smartphones. Η ενότητα 4.6 αναφέρεται στο επιλεγμένο API και την

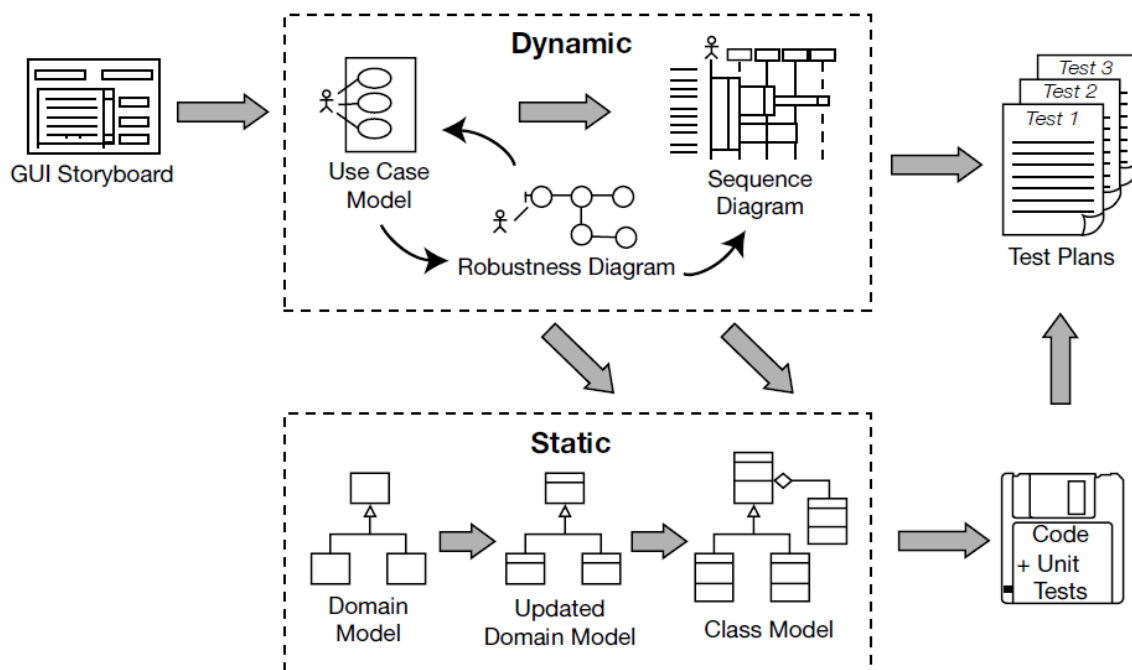
ιδιαίτερη φιλοσοφία του που χρησιμοποιήθηκε για την ανάπτυξη των “μικροεφαρμογών” μας (portlets). Η ενότητα 4.7 περιγράφει το δομικό μοντέλο αρχιτεκτονικής εφαρμογών που είναι πλέον η πιο διαδεδομένη στη ανάπτυξη διαδικτυακών εφαρμογών. Η ενότητα 4.8 αναφέρεται στη βάση δεδομένων που επιλέξαμε για την λειτουργία της πύλης. Οι ενότητες 4.9 και 4.10 περιγράφουν τις τεχνολογίες υλοποίησης της αισθητικής (εμφάνισης) της πύλης με τη δημιουργία των κατάλληλων templates και stylesheets. Στις ενότητες 4.11 και 4.12 αναφέρονται στα επικουρικά εργαλεία ανάπτυξης για την διαχείριση και συντήρηση του συνολικού έργου αλλά και την ενσωμάτωση σε αυτό των portlets που δημιουργήσαμε. Τέλος, η ενότητα 4.13 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

4.2 Μεθοδολογία ICONIX

4.2.1 Εισαγωγή

Θεωρητικά, κάθε πτυχή της UML είναι δυνητικά χρήσιμη, αλλά στην πράξη, δεν φαίνεται να υπάρχει αρκετός χρόνος για να γίνει μοντελοποίηση, ανάλυση, και σχεδιασμός. Πάντα υπάρχει πίεση για γρήγορη μετάβαση στην υλοποίηση, με την έναρξη του πρόωρου προγραμματισμού, επειδή η πρόοδος των έργων λογισμικού τείνει να μετράται με το ποσοστό υλοποίησης του κώδικα [73].

Η μεθοδολογία ICONIX, όπως φαίνεται στο παρακάτω διάγραμμα, είναι μια μινιμαλιστική, ορθολογιστική προσέγγιση που εστιάζει σε αυτή την περιοχή που βρίσκεται ανάμεσα στις περιπτώσεις χρήσης (use cases) και του κώδικα προγραμματισμού [74].



Εικόνα 4.1: χάρτης διαδικασιών ICONIX

Η ICONIX είναι μια μεθοδολογία ανάπτυξης λογισμικού η οποία προϋπήρχε τόσο της Rational Unified Process (RUP), Extreme Programming (XP) και της Agile ανάπτυξης λογισμικού. Η ICONIX βασίζεται σε περιπτώσεις χρήσης με διαγράμματα UML, όπως η RUP, χωρίς όμως την ιδιαίτερη έμφαση, που η δεύτερη μεθοδολογία δίνει. Σε αντίθεση με τις προσεγγίσεις XP και Agile, η ICONIX παρέχει επαρκή τεκμηρίωση σε θέματα απαιτήσεων και σχεδιασμού. Η ICONIX χρησιμοποιεί μόνο τέσσερα UML διαγράμματα σε μια διαδικασία μετατροπής τεσσάρων βημάτων της χρήσης κειμένου (περιπτώσεις χρήσης) σε κώδικα προγραμματισμού [75].

Μια διαδικασία που μόνο η μεθοδολογία της ICONIX υιοθετεί είναι η χρήση της ανάλυσης ευρωστίας, μια μεθόδου για τη γεφύρωση του χάσματος μεταξύ της ανάλυσης και του σχεδιασμού. Η ανάλυση ευρωστίας μειώνει την ασάφεια στις περιγραφές των περιπτώσεων χρήσεων, εξασφαλίζοντας ότι έχουν συνταχθεί στο πλαίσιο ενός συνοδευτικού μοντέλο Ανάλυσης Κλάσεων (domain model). Αυτή η διαδικασία διευκολύνει πολύ την σχεδίαση τη δοκιμή και την εκτίμηση των περιπτώσεων χρήσεων. Ουσιαστικά, η ICONIX περιγράφει τον πυρήνα της λογικής ανάλυσης και της διαδικασίας μοντελοποίησης του σχεδιασμού μιας εφαρμογής [75].

Στη συνέχεια, με τη βασική περίπτωση χρήσης, «σύνδεσης χρήστη με το λογαριασμό του», μέσω της μεθοδολογίας ICONIX , παρουσιάζουμε το τρόπο που μας βοηθά στην ανάλυση και την σχεδίαση του συστήματος.

4.2.1 Use Case: Σύνδεση χρήστη με το λογαριασμό του

Βασική Ροή.

1. Ο χρήστης επιλέγει σύνδεση με το λογαριασμό του από την αρχική οθόνη επισκέπτη του συστήματος.
2. Το σύστημα μεταφέρει τον έλεγχο στο CAS.
3. Το CAS πραγματοποιεί έλεγχο, για να διαπιστώσει αν ο χρήστης έχει πιστοποιηθεί ξανά.
4. Το CAS διαπιστώνει ότι ο χρήστης δεν έχει πιστοποιηθεί ξανά.
5. Το CAS εμφανίζει την οθόνη εισαγωγής στοιχείων χρήστη.
6. Ο χρήστης κάνει εισαγωγή διαπιστευτηρίων στην οθόνη εισαγωγής στοιχείων χρήστη.
7. Το CAS διενεργεί έλεγχο διαπιστευτηρίων από όλες τις διαθέσιμες πηγές D.E.N. (LDAP, Active Directory, DB).
8. Το CAS πιστοποιεί το χρήστη.
9. Το CAS μεταφέρει τον έλεγχο πίσω στο σύστημα με αναφορά επιτυχούς πιστοποίησης και επιστρέφοντας το username του χρήστη.
10. Το σύστημα αιτείται την παραλαβή τιμών ζητούμενων ιδιοτήτων χρήστη από όλες τις διαθέσιμες πηγές D.E.N.
11. Το σύστημα επιτελεί κατηγοριοποίηση χρήστη με βάση τις τιμές ιδιοτήτων.
12. Το σύστημα διενεργεί έλεγχο αν ο χρήστης έχει ξανασυνδεθεί.

13. Το σύστημα αναγνωρίζει ότι ο χρήστης έχει ξανασυνδεθεί..
14. Το σύστημα εμφανίζει την εξατομικευμένη οθόνη με τα portlets που έχει επιλέξει ο χρήστης από προηγούμενη σύνδεση του με το σύστημα.

Εναλλακτική ροή 1: Ανεπιτυχής Πιστοποίηση Χρήστη.

8.1 Το CAS εμφανίζει μήνυμα ανεπιτυχούς σύνδεσης.

8.2 Το CAS επιστρέφει στο βήμα 5 της βασικής ροής.

Εναλλακτική ροή 2: ο χρήστης δεν έχει ξανασυνδεθεί στο παρελθόν.

13.1 Το σύστημα αναγνωρίζει ότι ο χρήστης δεν έχει ξανασυνδεθεί στο παρελθόν.

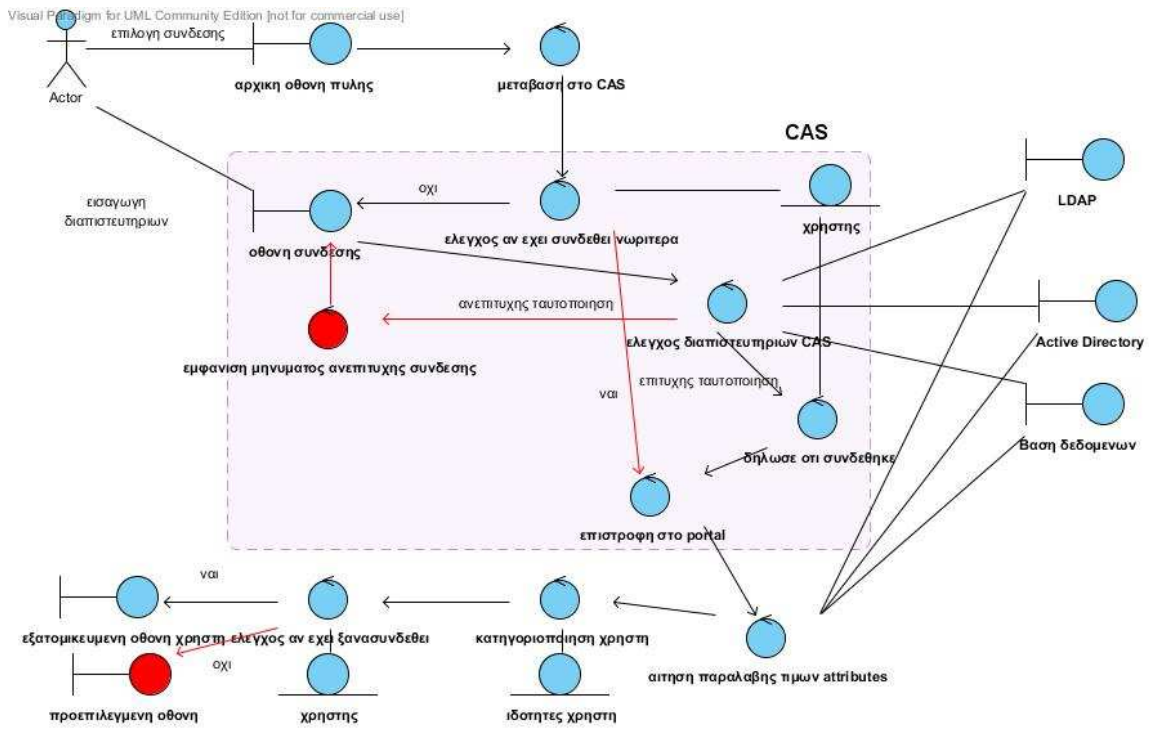
13.2 Το σύστημα επιστρέφει στο βήμα 14 της βασικής ροής.

Εναλλακτική ροή 3: ο χρήστης έχει ήδη πιστοποιηθεί από το CAS.

5.1 Το σύστημα εμφανίζει μήνυμα ότι ο χρήστης έχει ήδη πιστοποιηθεί από το CAS.

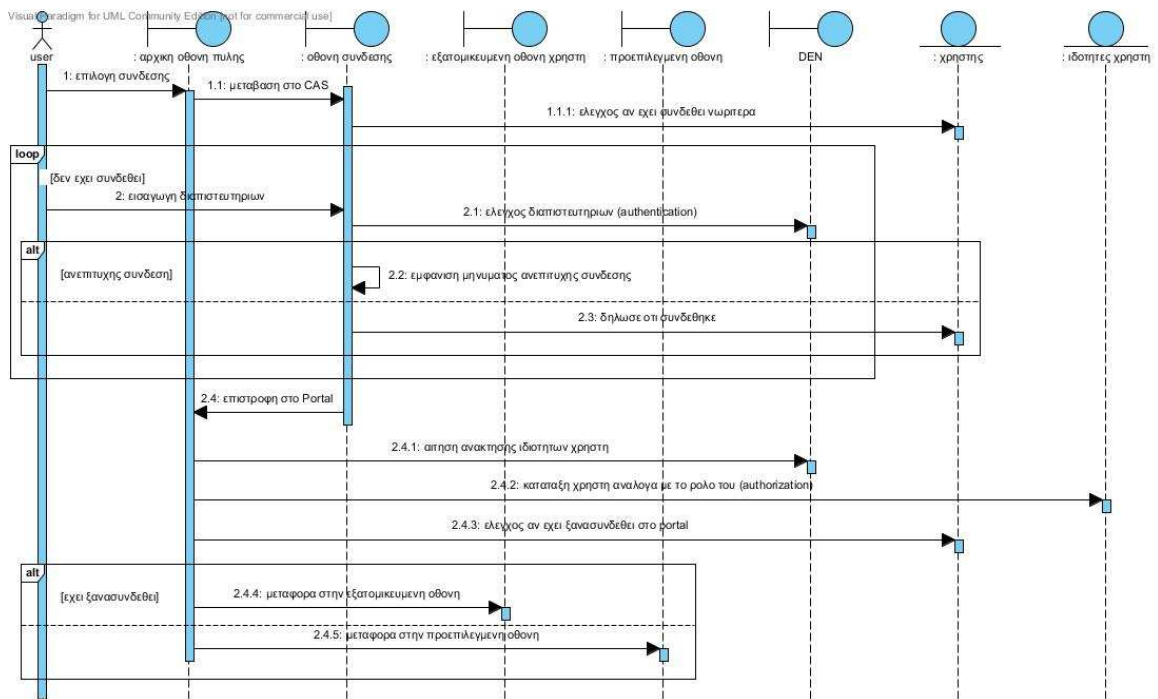
4.2 Το σύστημα επιστρέφει στο βήμα 9 της βασικής ροής.

4.2.2 Use Case: Σύνδεση με το λογαριασμό, Robustness diagram



Εικόνα 4.2: Robustness Diagram

4.2.3 Use Case: Σύνδεση με το λογαριασμό, Sequence Diagram



Εικόνα 4.3: Sequence Diagram

4.3 uPortal

4.3.1 Εισαγωγή

Το uPortal είναι ένα πλαίσιο ανάπτυξης πύλης ανοιχτού κώδικα που έχει υλοποιηθεί και συνεχίζει να εξελίσσεται από την κοινότητα της τριτοβάθμιας εκπαίδευσης σε παγκόσμιο επίπεδο και υποστηρίζεται οικονομικά από επιχορηγήσεις, δωρεές και κυρίως συμμετοχές από ακαδημαϊκά ιδρύματα, από εμπορικές επιχειρήσεις και από μη κερδοσκοπικά ιδρύματα.

Το uPortal βασίζεται σε ανοιχτά πρότυπα και τεχνολογίες όπως η Java και XML και επιτρέπει την εύκολη ολοκλήρωση εφαρμογών μέσω προτύπων, με έλεγχο ταυτότητας και υποδομών ασφαλείας, με περιεχόμενο που βασίζεται στο διαδίκτυο και επιτρέπει προσαρμογή σύμφωνα με τις προτιμήσεις του τελικού χρήστη[16]. Το uPortal μπορεί εύκολα να ενσωματωθεί σε άλλα συστήματα επιχειρήσεων και μπορεί να προσαρμοστεί για συγκεκριμένες άλλες ανάγκες. Το ισχυρό σύστημα διαχείρισης διάταξης περιεχομένου επιτρέπει μια δυναμική, προσωπική προβολή για χρήστες με πολλαπλούς ρόλους. Η εμπειρία του χρήστη είναι περαιτέρω ενισχυμένη και προσβάσιμη μέσω της χρήσης του συστήματος fluid. Το uPortal είναι ένα από τα πιο ευρέως διαδεδομένα ανοιχτού κώδικα λογισμικού πύλης, έχει εγκριθεί από εκατοντάδες ιδρύματα τριτοβάθμιας & δευτεροβάθμιας εκπαίδευσης και ερευνητικές κοινότητες παγκοσμίως.

4.3.2 Δυνατότητες

Πλαίσιο του πυρήνα

Το ευέλικτο API του uPortal επιτρέπει την επέκταση της πύλης με σκοπό να ανταποκρίνεται στις τοπικές ανάγκες και να ενσωματώνεται με άλλες υπηρεσίες. Μια αρθρωτή αρχιτεκτονική διατηρεί το διαχωρισμό της πύλης και του portlet κώδικα και έτσι προκύπτει μια ανάπτυξη με χωριστά αναβαθμιζόμενα στοιχεία.

Τα τεχνικά χαρακτηριστικά του περιλαμβάνουν:

- Δημιουργία portlets με τα πρότυπα JSR-168 (Portlet 1.0), JSR-268 (Portlet 2.0).
- Υποστήριξη πλαισίου Spring v3.0.

- Ολοκληρωμένο σύστημα παρουσίασης περιεχομένου με εξαιρετική απόδοση λόγω χρήσης προσωρινής αποθήκευσης (cache pages).
- Υποστήριξη πολλαπλών γλωσσών.
- Βιβλιοθήκη διαχείρισης Web υπηρεσιών (REST), για πρόσβαση σε χρήστες, ομάδες του uPortal και ανάλογα με τα δικαιώματά τους.
- Ευέλικτο σύστημα δημιουργίας θέματος εμφάνισης και προσαρμογής διεπαφής.
- Εγγραφή του χρήστη σε προδιαμορφωμένες καρτέλες περιεχομένου.
- Υποστήριξη συμπλεγμάτων Η/Υ (clustering) και φιλοξενία πολλών πυλών από μία μόνο εγκατάσταση.
- Υποστήριξη για πολλές δημοφιλείς βάσεις δεδομένων όπως Oracle, MySQL, PostgreSQL και Microsoft SQL Server.

4.3.3 Έλεγχος ταυτότητας

Το uPortal υποστηρίζει μια ποικιλία από τα πιο δημοφιλή συστήματα ελέγχου ταυτότητας καθιστώντας εύκολη τη σύνδεση με όποιο σύστημα χρησιμοποιεί ένας οργανισμός. Άμεσα με την αρχική εγκατάσταση του υποστηρίζει την πρόσβαση συστημάτων μέσω ενιαίας και μιας φοράς σύνδεσης (SSO). Το uPortal παρέχει ακόμη ένα API ασφάλειας που επιτρέπει τη δημιουργία προσαρμοσμένου ελέγχου ταυτότητας εφαρμογής.

Ενδεικτικά, οι πηγές ελέγχου ταυτότητας περιλαμβάνουν: Jsig υπηρεσία κεντρικού έλεγχου ταυτότητας (CAS), Internet2 Shibboleth, PubCookie, υπηρεσία καταλόγου Active Directory, OpenLDAP[46], Novell eDirectory και JDBC. Αυτό επιτυγχάνεται με τη χρήση του ειδικού API ενσωματωμένου καταλόγου με κωδική ονομασία "πρόσωπο" που επιτρέπει στην πύλη να προσαρμόζεται και να χρησιμοποιεί χαρακτηριστικά-ιδιότητες του χρήστη από πολλές πηγές, είτε καταλόγους δικτύου ή προσαρμοσμένων βάσεων δεδομένων.

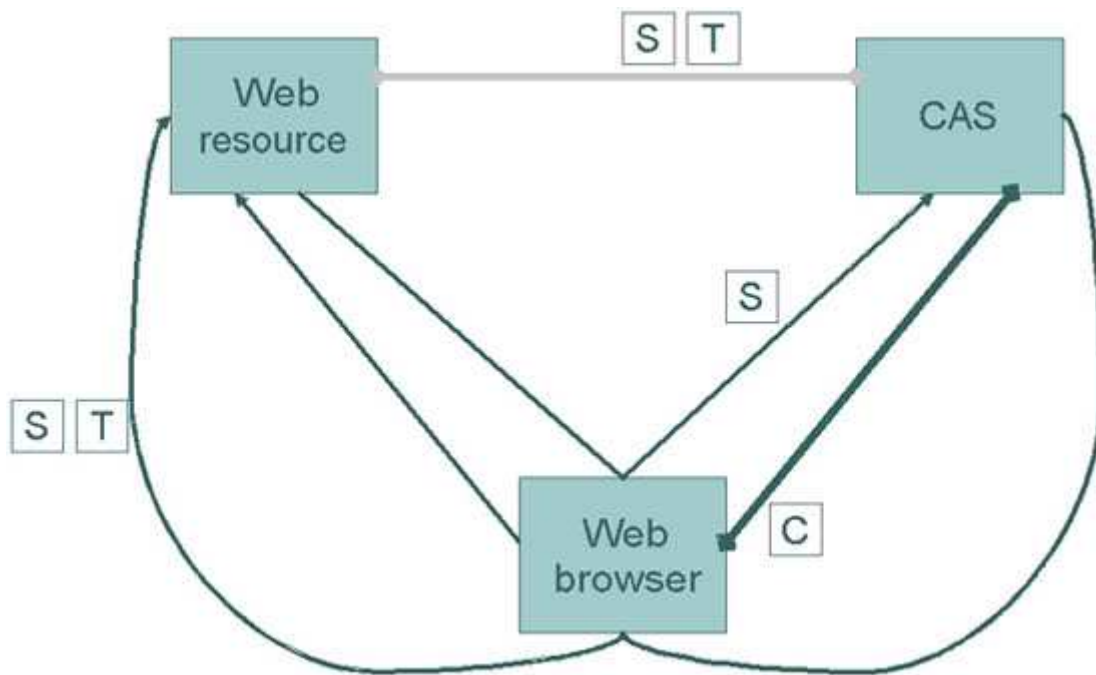
4.4 Central Authentication System, CAS

CAS είναι ένα σύστημα ταυτοποίησης που σχεδιάστηκε αρχικά στο πανεπιστήμιο Yale προκειμένου να παρέχει ένα ασφαλή τρόπο σε μια εφαρμογή να πιστοποιεί την αυθεντικότητα του χρήστη [29]. Πλέον, είναι ένα έργο από την ίδια κοινοπραξία πανεπιστημίων που υλοποιεί το uPortal. Μερικά από τα πλεονεκτήματά του:

- Ανοιχτό και καλά τεκμηριωμένο πρωτόκολλο.
- Μια εφαρμογή διακομιστή ανοιχτού κώδικα σε java.
- Με μια πληθώρα από διαθέσιμα πλέον προγράμματα-πελάτες σε Java, .Net, PHP, Perl και άλλα.
- Ολοκλήρωση με uPortal, Liferay, BlueSocket, TikiWiki, Mule, Moodle και άλλα.
- Αρκετά καλή τεκμηρίωση και τεχνική υποστήριξη στην υλοποίησή του.
- Μεγάλη αποδοχή από την κοινότητα με πολλές εγκαταστάσεις σε όλο τον κόσμο.

Είναι ένα πρωτόκολλο μιας μοναδικής σύνδεσης για το διαδίκτυο. Σκοπός του είναι να επιτρέπει στο χρήστη να έχει πρόσβαση σε πολλαπλές εφαρμογές ενώ συνδέεται μόνο μια φορά στο σύστημα δίνοντας τα διαπιστευτήρια του όπως όνομα χρήστη ή ηλ. διεύθυνση και ένα κωδικό πρόσβασης. Επιτρέπει σε πολλές διαδικτυακές εφαρμογές να ταυτοποιούν χρήστες χωρίς αυτές να έχουν πρόσβαση στο κωδικό του χρήστη [30].

Το πρωτόκολλο αυτό συνεργάζεται με το πρόγραμμα περιήγησης του χρήστη, τη διαδικτυακή εφαρμογή που αιτείται πιστοποίηση και το διακομιστή. Όταν ο χρήστης βρίσκεται σε μια εφαρμογή και αιτείται ταυτοποίηση, γίνεται ανακατεύθυνση στο CAS, υπεύθυνου για την επικύρωση αυθεντικότητας του χρήστη, συγκρίνοντας τα προτεινόμενα διαπιστευτήρια του με αυτά που είναι αποθηκευμένα σε μια βάση δεδομένων ή κατάλογο δικτύου (Directory Enabled Networking) όπως Active directory ή Κέρβερος. Εφόσον η ταυτοποίηση είναι επιτυχής, το CAS στέλνει ένα εισιτήριο ασφαλείας στην εφαρμογή. Αυτή, στη συνέχεια, επικυρώνει το εισιτήριο επικοινωνώντας και πάλι με το CAS, μέσω ασφαλούς σύνδεσης, αποστέλλοντας το αναγνωριστικό της εφαρμογής και ξανά το εισιτήριο. Στην συνέχεια, το CAS επιστρέφει την πληροφορία στην εφαρμογή αν ο χρήστης έχει ταχτοποιηθεί επιτυχώς και ο χρήστης επιστρέφει στο περιβάλλον της αρχικής εφαρμογής με νέα ανακατεύθυνση [12].



Εικόνα 4.4: Κλήση διαδικτυακής υπηρεσίας μέσω CAS []

Το CAS υποστηρίζει, την πολλαπλών επιπέδων, ταυτοποίηση μέσω διεύθυνσης proxy. Με αυτή την διεύθυνση, μπορεί οποιαδήποτε υπηρεσία όπως μια βάση δεδομένων ή ένας διακομιστής μηνυμάτων να συνεργάζεται με το CAS.

4.5 uMobile

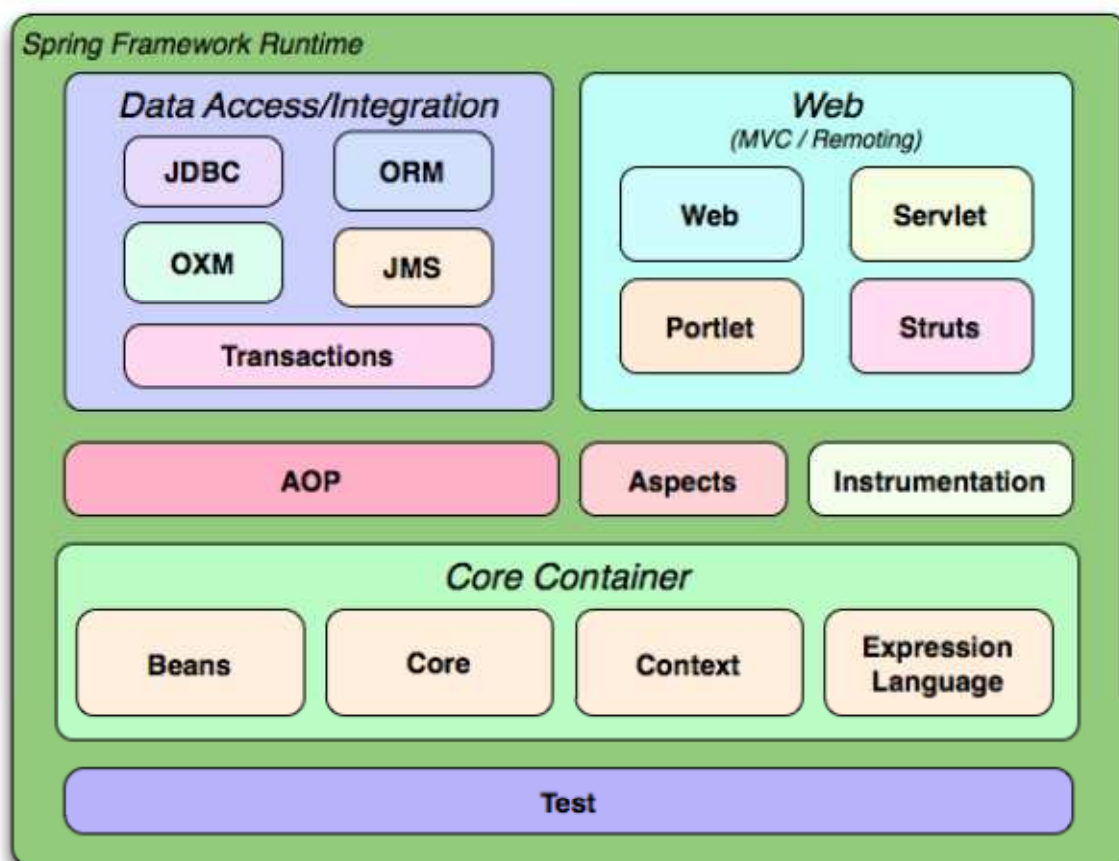
Το uMobile είναι μια νέα εφαρμογή ανοιχτού κώδικα με σκοπό να προσφέρει πρόσβαση στις διαδικτυακές εφαρμογές ενός οργανισμού, το περιεχόμενο και τα δεδομένα του σε κινητές συσκευές. Το έργο αποτελείται από εφαρμογές ειδικά για συσκευές iPhone ή με λειτουργικό σύστημα Android, καθώς και σε άλλα έξυπνα τηλέφωνα μέσω πρόσβασης από πρόγραμμα πλοήγησης. Το σημαντικό είναι ότι αυτή η λειτουργικότητα επιτυγχάνεται από τον ίδιο ενιαίο κώδικα ανάπτυξης [54].

Οι βασικές ενότητες του uMobile περιλαμβάνουν μια σειρά από εφαρμογές με χρήση χαρτών, καταλόγου, ανακοινώσεων, αναζήτηση, μαθημάτων, νέων-ανακοινώσεων/RSS feeds και ημερολόγιου.

4.6 Spring

4.6.1 Εισαγωγή

Το Spring είναι ένα μη “παρεμβατικό” πλαίσιο ανάπτυξης εφαρμογών JAVA που έχει αποκτήσει εξέχουσα θέση τα τελευταία χρόνια εξαιτίας της POJO (Plain Old Java Object) προσέγγισης του για την ανάπτυξη εταιρικών εφαρμογών και έχει οδηγήσει σε αυξημένη παραγωγικότητα και ευκολία ανάπτυξης δοκιμών καλής λειτουργίας του λογισμικού (JUnit Testing) [64].



Εικόνα 4.5: Επισκόπηση του πλαισίου ανάπτυξης Spring

Η φιλοσοφία των διαδικτυακών εφαρμογών με χρήση Spring έχουν τις παρακάτω βασικές αρχές με τελικό στόχο τελικά την ευκολία ανάπτυξης εφαρμογών σε Java [23].

- Dependency injection (DI) είναι η βασική ιδέα που διέπει όλα τα προγράμματα σε Spring. Συνήθως τα προγράμματα σε Java αλλά γενικότερα όλες οι γλώσσες αντικειμενοστραφή προγραμματισμού (object oriented) έχουν σοβαρά θέματα φορητότητας, Αποτελούνται από μια συλλογή από αντικείμενα, κλάσεις που εξαιτίας της κληρονομικότητας και της εξάρτησης τους από άλλα αντικείμενα ή βιβλιοθήκες, είναι δύσκολο αναλόγως και το μέγεθος να μεταφερθούν αυτούσια σε ένα άλλο περιβάλλον. Συνήθως, αν και ο στόχος

είναι να χρησιμοποιηθεί ένα μέρος μόνο του κώδικα, τελικά καταλήγουμε να μεταφέρουμε ολόκληρο το project. Το Spring πλαίσιο δημιουργήθηκε, ώστε να αντιμετωπίσει την πολυπλοκότητα της ανάπτυξης εφαρμογών των επιχειρήσεων και καθιστά δυνατή τη χρήση απλών JavaBeans για την επίτευξη πραγμάτων που προηγουμένως ήταν δυνατά μόνο με Enterprise JavaBeans. Αλλά η χρησιμότητα του δεν περιορίζεται στην ανάπτυξη server-side εφαρμογών καθώς κάθε εφαρμογή Java μπορεί να επωφεληθεί, όσον αφορά την απλότητα και την χαλαρή σύνδεση μεταξύ των αντικειμένων-εξαρτημάτων.

- **Application Context:** Με το DI, οι εξαρτήσεις που έχει ο κώδικας, ορίζονται σε ένα εξωτερικό αρχείο ρυθμίσεων, συνήθως σε αρχείο XML ή εντός του πηγαίου κώδικα (με σχόλια) και αυτές οι εξαρτήσεις «εγχέονται» στα εξαρτώμενα αντικείμενα από το δοχείο IoC (Inversion of Control). Ένα δοχείο IoC είναι υπεύθυνο για τη δημιουργία και τη διαχείριση των αντικειμένων αλλά και για την προσθήκη των απαραίτητων εξαρτήσεων που έχουν με βάση τις πληροφορίες από το αρχείο ρυθμίσεων της εφαρμογής. Έτσι, οι προγραμματιστές δεν ασχολούνται με την ευθύνη της διαχείρισης των αντικειμένων και κυρίως των εξαρτήσεων που μπορούν να έχουν [01].

4.6.2 Φιλοσοφία

Το Spring απλοποιεί την ανάπτυξη λογισμικού με java και αυτό επιτυγχάνεται με 4 στρατηγικές:

- 1 Ελαφριά και ελάχιστα παρεμβατική ανάπτυξη με τη χρήση των POJO (Plain Old Java Object), επιτρέπει ο κώδικας να είναι συνοπτικός, απλός και σπονδυλωτός (modular).
- 2 Χαλαρή σύνδεση μέσω της έγχυσης εξάρτησης και την χρήση όπου είναι δυνατόν των java interfaces επιτρέπει τη φορητότητα δηλαδή τη δυνατότητα μεταφοράς και ενσωμάτωσης και επαναχρησιμοποίησης του κώδικα σε άλλες εφαρμογές, βοηθά στη εύρεση λογικών λαθών και λάθους συμπεριφοράς του κώδικα πριν την δημοσίευση του (JUnit Testing).
- 3 Προγραμματισμός με χρήση δηλώσεων μέσα στο κώδικα των αντικειμένων (aspect programming) και κοινές συμβάσεις, επιτρέπει τη δημιουργία κώδικα που με λίγες γραμμές με τη μορφή σχολίων και ένα αρχείο παραμέτρων μπορεί κάποιος να φτιάξει ολόκληρες εφαρμογές [13].

- 4 Μείωση του μεγέθους κώδικα με τη χρήση διαθέσιμων αντικειμένων και interfaces που παρέχει το API του Spring βοηθά στη αύξηση της παραγωγικότητας και τη μείωση του χρόνου ανάπτυξης.

Δεν είναι υπερβολή να πούμε ότι το spring framework είναι επί σειρά ετών πρωτοπόρο και πολλές τεχνολογίες και κυρίως φιλοσοφίες που έχει αναπτύξει έχουν υιοθετηθεί από τα μετέπειτα πρότυπα και προδιαγραφές της Java κοινότητας. Έννοιες όπως το μοντέλο MVC με τη ξεκάθαρη διάκριση του ρόλων επιτρέπει την ανάπτυξη επεκτάσιμου κώδικα, Webflow με την ευκολία ανάπτυξης εφαρμογών που απαιτούν πολλά βήματα αλληλεπιδράσεων με το χρήστη όπως δημιουργία wizards και διαχείριση πολύπλοκων φορμών, θέματα ασφαλείας, ελέγχου πιστοποίησης και διαχείρισης περιόδου λειτουργίας μιας διαδικτυακής εφαρμογής, έχουν ενσωματωθεί σε άλλα frameworks όπως του uPortal.

4.7 Model-View-Controller (MVC)

Στην τεχνολογία λογισμικού, το Model-View-Controller (MVC) είναι ένα δομικό μοντέλο αρχιτεκτονικής εφαρμογών που χωρίζει τις αλληλεπιδράσεις μεταξύ των χρηστών και των εφαρμογών σε τρεις ρόλους:

- το μοντέλο-Model (επιχειρηματική λογική),
- τη προβολή-View (διεπαφή),
- ο ελεγκτής-Controller (που διαβάζει τις ενέργειες του χρήστη).

Αυτός ο διαχωρισμός διευκολύνει την ανεξάρτητη ανάπτυξη, δοκιμή και συντήρηση του κάθε ρόλου. Είναι το πλέον αναγνωρισμένο μοντέλο ανάπτυξης και βάση του προγραμματισμού διαδικτυακών εφαρμογών και όχι μόνο σε Java. Έχει υιοθετηθεί από το JCP, υπεύθυνου τυποποίησης της Java και από τα περισσότερα frameworks όπως το Spring, 'Επιπλέον, και άλλα εργαλεία ανάπτυξης σε διαφορετικές πλατφόρμες, όπως για .NET (Microsoft ASP.NET MVC) και PHP το έχουν υιοθετήσει. Δεν είναι τυχαίο ότι η προτεινόμενη μέθοδος ανάπτυξης λογισμικού ICONIX που διδαχτήκαμε στη θεματική ενότητα ΠΗΛΣ-61 κατά τη διάρκεια του μεταπτυχιακού προγράμματος, ακολουθεί το μοντέλο MVC ως προς τη φιλοσοφία και την υιοθέτηση των

διαγραμμάτων ευρωστίας (robustness analysis) για την γεφύρωση του κενού μεταξύ ανάλυσης και σχεδίασης [15].

4.8 HSQLDB

Η HSQLDB (HyperSQL DataBase) είναι μια διαδεδομένη σχεσιακή βάση δεδομένων γραμμένη σε Java. Υποστηρίζει σχεδόν εξολοκλήρου το πρότυπο ANSI-92 SQL και εξολοκλήρου το SQL:2008. Είναι εφαρμογή μικρή σε απαιτήσεις μνήμης, γρήγορη με υποστήριξη συναλλαγών. Είναι ευέλικτη καθώς μπορεί να λειτουργήσει τόσο σε ενσωματωμένη κατάσταση με πρόσβαση απευθείας στα αρχεία με τα δεδομένα, με παρόμοιο τρόπο όπως η Microsoft Access και η Sqlite, όσο και σε δικτυακή κατάσταση μέσω TCP/IP σε απομακρυσμένη κατάσταση. Περιλαμβάνει εργαλεία είτε για εκτέλεση ενεργειών μέσω γραμμής εντολών (CLI) ή για πρόσβαση μέσα από γραφικό περιβάλλον εργασίας. Με πάνω από 10 χρόνια ύπαρξης έχει αποδειχτεί σταθερή και αξιόπιστη [37].

4.9 XSL

XSL είναι μια γλώσσα για την έκφραση stylesheets (θεμάτων). Ένα XSL stylesheet είναι, όπως και το CSS, ένα αρχείο που περιγράφει πώς εμφανίζεται ένα έγγραφο XML ενός συγκεκριμένου είδους. Η XSL μοιράζεται τη λειτουργικότητα και είναι συμβατή με το CSS2 (παρόλο που χρησιμοποιεί μια διαφορετική σύνταξη). Ταυτόχρονα, παρέχει και μια γλώσσα για μετασχηματισμό εγγράφων XML, τη XSLT. Αρχικά, επρόκειτο να εκτελεί περίπλοκες εργασίες στυλ, όπως η δημιουργία πινάκων περιεχομένων και ευρετηρίων, αλλά χρησιμοποιείται πλέον ως γλώσσα επεξεργασίας XML για γενικότερους σκοπούς όπως τη δημιουργία HTML ιστοσελίδων από δεδομένα XML [14].

Τα προηγμένα χαρακτηριστικά στυλ αποτυπώνονται σε ένα έγγραφο τύπου XML καθορίζουν ένα σύνολο στοιχείων που ονομάζονται αντικείμενα σχηματοποίησης (Formatting Objects) και ιδιότητες (όπως περίπου οι ιδιότητες της CSS2) και ταυτόχρονα προσθέτουν πιο σύνθετα αντικείμενα. Ένα ή περισσότερα έγγραφα XML περιέχουν τις πληροφορίες τι και πως θα το εμφανίσει το stylesheet (θέμα) [52].

Στο uPortal χρησιμοποιείται ευρέως η τεχνολογία XSLT κυρίως για παραμετροποίηση της πύλης και των portlets που την συνοδεύουν αλλά και για τη δημιουργία του γραφικού θέματος της διεπαφής με μετασχηματισμό αρχείων XML σε XHTML.

4.10 Fluid

Είναι ένα ανοικτό έργο [66] συνεργασίας για τη βελτίωση της εμπειρίας του χρήστη σε διαδικτυακές εφαρμογές ανοιχτού κώδικα [37] και υποστηρίζει τα ακόλουθα:

- Σχεδίαση ευέλικτων, προσαρμόσιμων διεπαφών με επίκεντρο το χρήστη.
- Προώθηση του σχεδιασμού αυτού χωρίς αποκλεισμούς (πχ υποστήριξη ατόμων με ειδικές ανάγκες).
- Ενσωμάτωση αυτών των διεπαφών σε εφαρμογές ανοιχτού κώδικα.
- Δημιουργία έτοιμων αντικειμένων και λειτουργιών όπως drag and drop, επεξεργασία δεδομένων με χρήση AJAX, έξυπνη σελιδοποίηση δεδομένων. Τα αντικείμενα αυτά είναι πλήρως παραμετρικά, υποστηρίζουν χρήση πληκτρολογίου και είναι συμβατά με τα αντικείμενα του jQuery UI.

Τα παραπάνω υποστηρίζονται από ένα συνδυασμό javascript, CSS & HTML.

4.11 Apache Maven

Το Apache Maven είναι ένα λογισμικό διαχείρισης έργων JAVA και εργαλείο κατανόησης. Με βάση την έννοια του μοντέλου αντικειμένων του έργου (POM), το Maven μπορεί να διαχειριστεί την κατασκευή ενός έργου, την δημιουργία συνοδευτικών εκθέσεων και τεκμηρίωσης από ένα μοναδικό αρχείο πληροφοριών [26].

Το Maven καταφέρνει να:

- Διευκολύνει την διαδικασία δημιουργίας jar & war αρχείων (compile & build).

- Παρέχει ένα κοινό σύστημα σύνθεσης για συνεργαζόμενα έργα μέσα από ένα μοναδικό κοινό αρχείο πληροφοριών σε μορφή XML, με την ονομασία project object model (POM).
- Παρέχει υπηρεσίες ποιότητας λογισμικού με την αυτόματη δημιουργία αναφορών εξαρτήσεων μεταξύ έργων και δοκιμών έλεγχου καλής λειτουργίας συμπεριλαμβανομένης της επικάλυψης μεταξύ αντικειμένων.
- Δίνει κατευθυντήριες γραμμές για τις βέλτιστες πρακτικές ανάπτυξης λογισμικού.
- Διαχειρίζεται την αυτόματη μεταφόρτωση όλων των απαραίτητων εργαλείων και βιβλιοθηκών και τον έλεγχο καλής λειτουργίας μιας εφαρμογής.

4.12 Apache Ant

Το Apache Ant είναι μια βιβλιοθήκη της Java και εργαλείο εκτέλεσης εντολών σε CLI (MSDOS, bash κτλ) με αποστολή να οδηγεί τις διαδικασίες διακρινόμενες ως στόχους, όπως περιγράφονται σε αρχεία μορφής απλού κειμένου, για τη δημιουργία εκτελέσιμων αρχείων από τη μεταγλώττιση του πηγαίου κώδικα εφαρμογών σε Java, C και C++. Υποστηρίζει επιπλέον εργασίες για συναρμολόγηση και έλεγχο καθώς και εκτέλεση εφαρμογών. Γενικότερα, το Apache Ant μπορεί να χρησιμοποιηθεί για το χειρισμό κάθε είδους διαδικασίας, η οποία μπορεί να περιγραφεί σαν μια σειρά στόχων και καθηκόντων [25].

Το Apache Ant είναι γραμμένο σε Java και είναι εξαιρετικά ευέλικτο και δεν επιβάλλει υιοθέτηση κάποιου συγκεκριμένου τρόπου γραφής κώδικα ή συγκεκριμένης δημιουργίας και διάταξης υποκαταλόγων του έργου και αυτός είναι ο βασικός λόγος της διάδοσής του.

4.13 Σύνοψη

Στο κεφάλαιο αυτό έγινε καταγραφή των εργαλείων ανάπτυξης και των επικουρικών τεχνολογιών που χρησιμοποιήθηκαν για την υλοποίηση της εκπαιδευτικής πύλης.

Η ενότητα 4.2 αναφέρεται στη μεθοδολογία ICONIX και στα διαγράμματα που χρησιμοποιήθηκαν στο βασικό use case της σύνδεσης του χρήστη με το σύστημα. Η ενότητα 4.3 αναφέρεται στην πλατφόρμα ανοιχτού κώδικα uPortal που πάνω σε αυτό βασιστήκαμε για την

δημιουργία της εκπαιδευτική πύλης μας. Η ενότητα 4.4 περιέγραψε το κεντρικό σύστημα ταυτοποίησης χρηστών CAS ενώ η ενότητα 4.5 περιέγραψε την εφαρμογή ανοιχτού κώδικα uMobile, υπεύθυνη για την υποστήριξη φορητών συσκευών όπως smartphones. Η ενότητα 4.6 αναφέρθηκε στο Spring Framework και την ιδιαίτερη φιλοσοφία του που χρησιμοποιήθηκε για την ανάπτυξη των “μικροεφαρμογών” μας (portlets). Η ενότητα 4.7 περιέγραψε το δομικό μοντέλο αρχιτεκτονικής εφαρμογών MVC για την ανάπτυξη διαδικτυακών εφαρμογών. Η ενότητα 4.8 αναφέρθηκε στη βάση δεδομένων που επιλέξαμε για την λειτουργία της πύλης. Οι ενότητες 4.9 και 4.10 περιέγραφαν τις τεχνολογίες υλοποίησης των αισθητικής (εμφάνισης) της πύλης XSL & Fluid για τη δημιουργία των θεμάτων (templates και stylesheets). Στις ενότητες 4.11 και 4.12 έγινε αναφορά στα επικουρικά εργαλεία ανάπτυξης Maven & Ant για την διαχείριση και συντήρηση του συνολικού έργου αλλά και την ενσωμάτωση σε αυτό των portlets που δημιουργήσαμε. Τέλος, στην παρούσα ενότητα παρουσιάζεται η σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τον επιμέρους στόχο #3 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Το επόμενο κεφαλαίο θα εξετάσει την αντιμετώπιση θεμάτων στην υλοποίηση του συστήματος μας, ως τεκμηρίωση των επιμέρους στόχων #4 και #5 όπως αυτοί αναφέρονται στην εισαγωγή.

Κεφάλαιο 5

Υλοποίηση και Αντιμετώπιση Θεμάτων

5.1 Εισαγωγή

Στο κεφάλαιο αυτό, θα παρουσιαστεί η υλοποίηση της εφαρμογής και η επίλυση επιμέρους θεμάτων για τον σχεδιασμό του συστήματος και την ανάπτυξη του μηχανισμού πρόσβασης καθώς και η ενσωμάτωση των υπηρεσιών.

Η δομή αυτού του κεφαλαίου είναι ως εξής: στην ενότητα 5.2 αναλύουμε το πώς επιλέχτηκε η πλατφόρμα uPortal ως βάση για την ανάπτυξη της εκπαιδευτικής μας πύλης. Η ενότητα 5.3 εξηγεί πως οργανώθηκαν οι ρόλοι-κατηγορίες των χρηστών του συστήματος και πως επιτυγχάνεται η ταυτοποίηση τους. Η ενότητα 5.4 αναφέρεται στην κατηγοριοποίηση των εφαρμογών/υπηρεσιών της πύλης. Στην ενότητα 5.5 γίνεται περιγραφή των επιπλέον μέτρων που πήραμε για την ασφάλεια της τόσο σε επίπεδο προστασίας δεδομένων όσο και ασφάλειας συναλλαγών. Η ενότητα 5.6 είναι αφιερωμένη στο μηχανισμό πιστοποίησης χρηστών. Η ενότητα 5.7 εξετάζει τις τεχνικές σχεδίασης για πρόσβαση σε άτομα με ειδικές ανάγκες και η ενότητα 5.8 αναφέρεται στην πρόσβαση μέσω έξυπνων συσκευών όπως τηλεφώνων. Τέλος, η ενότητα 5.9 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

5.2 Επιλογή της πύλης

Ένα από τα πρώτα ερωτήματα που έπρεπε να απαντηθούν στην αρχή της μεταπτυχιακής διατριβής ήταν ποια πλατφόρμα και εργαλεία ανάπτυξης θα χρησιμοποιήσουμε για την υλοποίηση του έργου μας.

Με μια μικρή έρευνα στο διαδίκτυο, ο καθένας μπορεί να βρει μια πληθώρα λύσεων για ανάπτυξη πυλών με διάφορα εργαλεία ανάπτυξης. Υπάρχουν λύσεις που προτείνουν οι μεγαλύτεροι κατασκευαστές λογισμικού όπως η Microsoft (Sharepoint, .NET), IBM (WebSphere, JAVAE), RedHat(Jboss, JAVAE), Oracle(Application Server, JAVAE). Όλα τα παραπάνω είναι διαθέσιμα μετά από αγορά τους ή ετήσιας συνδρομής στους κατασκευαστές τους.

Η στρατηγική επιλογή μας για open source εργαλεία και τη JAVA σαν τη γλώσσα προγραμματισμού ανάπτυξης ήταν πλέον μονόδρομος. Ταυτόχρονα πιστεύουμε ότι τα ΙΤΕ σαν ακαδημαϊκά κέντρα πρέπει να προτρέπουν τη χρήση τέτοιων εργαλείων καθώς έτσι συνεισφέρουν στην ανάπτυξη νέων τεχνολογιών χωρίς περιορισμούς και δικαιώματα επί ευρεσιτεχνιών που τόσο έχουν πληγώσει τη κοινότητα της πληροφορικής.

Στη συνέχεια αφού αναζητήσαμε περισσότερα εργαλεία ανάπτυξης στη κορυφή της λίστας μπήκαν 2 λύσεις, αυτές του Liferay και του uPortal. Το καθένα από αυτά τα εργαλεία είχαν μοναδικές δυνατότητες και βρεθήκαμε σε ευχάριστα δύσκολη θέση ποιο να επιλέξουμε.

Το Liferay είναι πραγματικά ένα εργαλείο που απολαμβάνει να δουλεύει κανείς μαζί του, βρίσκεται στην έκτη έκδοση, με δυνατότητες που ξεπερνούν κατά πολύ τις απαιτήσεις μας. Με μεγάλη κοινότητα χρηστών, εξαιρετική υποστήριξη και πλούσια βιβλιογραφία τόσο σε έντυπη όσο και σε ηλεκτρονική μορφή είναι το κορυφαίο εργαλείο ανάπτυξης για οποιοδήποτε τύπου διαδικτυακών εφαρμογών. Με μια μεγάλη συλλογή έτοιμων portlets και θεμάτων για κάθε περίπτωση και γρήγορη προσαρμογή, εντυπωσιακό πελατολόγιο, και ενσωματωμένο δυνατό υποσύστημα CMS πραγματικά κάποιος θα το διάλεγε χωρίς πολύ σκέψη για οποιαδήποτε επιχειρησιακή πύλη.

Από την άλλη το uPortal είναι ένα εργαλείο δημιουργίας πύλης που σχεδιάστηκε με αποκλειστικό στόχο την κάλυψη των αναγκών της εκπαιδευτικής κοινότητας και αναπτύχθηκε από μια κοινοπραξία εκπαιδευτικών ιδρυμάτων. Το uPortal είναι ακριβώς αυτό που λέει, μια πύλη για – κυρίως- την τριτοβάθμια εκπαίδευση. Στερείται βιβλιογραφικής υποστήριξης, έχει ελλιπή τεκμηρίωση συγκεντρωμένη μόνο σε ένα WIKI της κοινοπραξίας, και λίγα διαθέσιμα και ενσωματωμένα portlets, χωρίς συλλογή από γραφικά θέματα αλλαγής εμφάνισης της πύλης. Τα πράγματα ήταν ακόμα πιο δύσκολα, γιατί αυτή τη περίοδο, έχει αλλάξει βασική έκδοση και υπάρχουν αρκετά προβλήματα –bugs- ακόμα και τώρα που γράφεται αυτό το κείμενο, τόσο σε θέματα εγκατάστασης όσο και λειτουργικότητας, κάτι που θα δυσκόλευε πολύ το έργο μας με το ανεβασμένο βαθμό δυσκολίας του.

Τελικά, όμως, προχωρήσαμε μετά από πολύ σκέψη και προβληματισμό στην υλοποίηση της πύλης μας με βάση το uPortal για τους παρακάτω λόγους:

Το uPortal σε υποχρεώνει να υλοποιήσεις τα portlets ακολουθώντας τις προδιαγραφές των προτύπων JSR 168, & JSR 268. Αυτό σημαίνει ότι τα portlets του απλά το μεταφέρεις σε οποιαδήποτε άλλη πύλη ή έστω container που είναι συμβατό με τις προδιαγραφές αυτές και το portlet έχει την ίδια λειτουργικότητα χωρίς καμιά αλλαγή σε κώδικα ή ακόμα και σε παραμετροποίηση. Ταυτόχρονα, βέβαια σημαίνει ότι η τεχνογνωσία που αποκτά κάποιος μπορεί να την χρησιμοποιήσει και σε άλλα portal frameworks. Το Liferay μπορεί να είναι συμβατό με τις προδιαγραφές, αλλά η συντριπτική πλειοψηφία των portlets του δεν είναι, οι προγραμματιστές για λόγους ευκολίας και γρηγορότερης ανάπτυξης ενσωματώνουν λειτουργίες που είναι διαθέσιμες μόνο για αυτό.

Το uPortal προτείνει και γι αυτό έχει ενσωματωμένη τη βιβλιοθήκη javascript jQuery ενώ το Liferay έρχεται αρχικά με την javascript της yahoo YUI3. Η λειτουργικότητα της jQuery και η πληθώρα τεκμηρίωσης, παραδειγμάτων και modules ειδικά για αυτήν είναι μεγάλο πλεονέκτημα του uPortal.

Το uPortal είναι προϊόν της ίδιας κοινοπραξίας που έχει δημιουργήσει το CAS (Central Authentication Service), ένα έργο καθιερωμένης πλέον αναγνώρισης και χρήσης από τη πλειοψηφία των εκπαιδευτικών ιδρυμάτων και στην Ελλάδα (και το ΤΕΙ Κρήτης μεταξύ άλλων) που επιτρέπει την γρήγορη και μιας φορές μόνο πιστοποίηση ενός χρήστη ώστε να έχει πρόσβαση σε μια πληθώρα διαδικτύων εφαρμογών που το υποστηρίζουν. Στη πραγματικότητα το uPortal έρχεται με ενσωματωμένο και προεπιλεγμένο στη χρήση του το CAS.

Το uPortal έχει δημιουργηθεί εξαρχής με τη φιλοσοφία του Spring, του πιο δημοφιλή πλαισίου ανάπτυξης εφαρμογών java με υψηλή απόδοση, εξαιρετικά εύκολου σε παραμετροποίηση με κώδικα που μπορεί να χρησιμοποιηθεί ξανά ή να ενσωματωθεί σε άλλες εφαρμογές γρήγορα και χωρίς περιορισμούς. Η πρόκληση της ευκαιρίας απόκτησης της αυτής τεχνογνωσίας έπαιξε ίσως το κύριο λόγο που καταλήξαμε στο uPortal. Υπενθυμίζεται ότι είναι και η προτεινόμενη φιλοσοφία ανάπτυξης λογισμικού που προτάθηκε και στην ΠΛΗΣ-61 με την μοναδική διάκριση του μοντέλου MVC (model-view-Controller) που προτείνει και η ICONIX διαδικασία στα διαγράμματα ευρωστίας (state diagrams)

Τελικά σαν εκπαιδευτικά ιδρύματα όπως είναι το Ανοιχτό Πανεπιστήμιο Κύπρου και το ΤΕΙ Κρήτης στο οποίο θα γίνει υλοποίηση της πύλης, πρέπει να υποστηρίζουν έργα άλλων Πανεπιστημίων και να είναι προτρέπουν σε χρήση προδιαγραφών και εργαλείων που επιτρέπουν την ανάπτυξη εφαρμογών με μεγαλύτερη συμβατότητα παρά ευκολία.

5.3 Κατηγορίες χρηστών

5.3.1 Εισαγωγή

Κατά την αρχική ανάλυση απαιτήσεων του συστήματος είχαν οριστεί (ενότητα 3.6) οι κατηγορίες χρηστών που θα έπρεπε το σύστημα να αναγνωρίζει. Κατά την μελέτη και επιλογή αντιστοίχισης υπηρεσιών σε κάθε ρόλο ήταν φανερό ότι θα μπορούσαν να υπάρξουν και δυο επιπλέον ρόλοι-ιδιότητες σαν υπερσύνολο κάποιων από τους παραπάνω ρόλους. Υπάρχουν υπηρεσίες που πρέπει να έχουν πρόσβαση φοιτητές, καθηγητές και διοικητικοί υπάλληλοι. Μια τέτοια υπηρεσία είναι αυτή του ηλ. ταχυδρομείου. Έτσι, σχεδιάστηκε ένας επιπλέον ρόλος που τον ονομάσαμε “campus” ώστε αυτές οι κοινές υπηρεσίες να είναι προσβάσιμες για όλους τους χρήστες που έχουν έστω μια ιδιότητα από αυτές του φοιτητή, καθηγητή και Δ. Υπαλλήλου. Επιπλέον, δημιουργήθηκε και ο ρόλος του “academic” που είναι υπηρεσίες κοινές για φοιτητές και καθηγητές.

Κάθε χρήστης του συστήματος μπορεί να ανήκει τουλάχιστον σε έναν από τους παραπάνω ρόλους. Αυτό σημαίνει ότι π.χ. ένας διοικητικός υπάλληλος μπορεί ταυτόχρονα να είναι και ωρομίσθιος καθηγητής σε κάποιο τμήμα αλλά να έχει και την ιδιότητα του πτυχιούχου δηλαδή να έχει σπουδάσει στο παρελθόν στο εκπαιδευτικό ίδρυμα. Το ιδανικό θα ήταν το σύστημα να

αναγνώριζε όλους τους ρόλους του χρήστη ώστε να του έδινε πρόσβαση ανάλογα σε όλες τις υπηρεσίες που έχουν συσχετιστεί με το κάθε ρόλο.

Προσπαθήσαμε να δούμε αν ήταν κάτι τέτοιο εφικτό. Η διαδικτυακή πύλη uPortal είναι σωστά δομημένη και επιτρέπει τη υποστήριξη χρηστών με παραπάνω από έναν ρόλους άρα τεχνικά είναι εφικτό.

5.3.2 Ταυτοποίηση ρόλων

Το κεντρικό LDAP του ιδρύματος στην ανάλυση του, μας δίνει τη δυνατότητα να διαχωρίζουμε τους ρόλους με βάση την τιμή στις παρακάτω ιδιότητες των χρηστών όπως είναι καταγεγραμμένες στο σύστημα:

Υποστηρίζονται ρόλοι ανάλογα τη σχολή του χρήστη:

- TeiCreteUserOrgUnit: TEI of Crete - STEG
- TeiCreteUserOrgUnit: TEI of Crete - STEF
- TeiCreteUserOrgUnit: TEI of Crete - SEYP
- TeiCreteUserOrgUnit: TEI of Crete - SDO

Επιπλέον, υποστηρίζεται ένας ρόλος για τους διοικητικούς υπαλλήλους

- TeiCreteUserOrgUnit: TEI of Crete - Administration

Ακόμα, υποστηρίζονται ρόλοι ανάλογα με το τμήμα του χρήστη

- TeiCreteUserOrgDepartment: Information & Computer Services
- TeiCreteUserOrgDepartment: Floriculture & Greenhouse Crops
- TeiCreteUserOrgDepartment: Biological Floriculture & Greenhouse Crops
- TeiCreteUserOrgDepartment: Sciences

- TeiCreteUserOrgDepartment: Electronics
- TeiCreteUserOrgDepartment: Electrical Engineering
- TeiCreteUserOrgDepartment: Mechanical Engineering
- TeiCreteUserOrgDepartment: Civil Engineering
- TeiCreteUserOrgDepartment: Music Technology & Acoustics
- TeiCreteUserOrgDepartment: Natural Resource Systems Engineering
- TeiCreteUserOrgDepartment: Crop Science (ΦΠ)
- TeiCreteUserOrgDepartment: Social Work
- TeiCreteUserOrgDepartment: Human Nutrition & Dietetics
- TeiCreteUserOrgDepartment: Nursing
- TeiCreteUserOrgDepartment: Business Administration
- TeiCreteUserOrgDepartment: Marketing (Commerce & Advertising)
- TeiCreteUserOrgDepartment: Accounting
- TeiCreteUserOrgDepartment: Business Planning & Information Systems
- TeiCreteUserOrgDepartment: Finance & Insurance
- TeiCreteUserOrgDepartment: Tourism & Hotel Management

Είναι εφικτό, επίσης, να διακρίνουμε ρόλους ανάλογα με τη θέση του χρήστη στο ίδρυμα

- TeiCreteUserOrgType: Administrative - support staff

- TeiCreteUserOrgType: Academic full time staff
- TeiCreteUserOrgType: Academic part time staff
- TeiCreteUserOrgType: Graduate Student
- TeiCreteUserOrgType: Undergraduate student

Ο συνδυασμός των παραπάνω διαμορφώνει του διαθέσιμους ρόλους.

Βλέπουμε ότι η αρχική μας σχεδίαση για τους ρόλους των χρηστών διαχωρίζεται χωρίς πρόβλημα. Θα μπορούσαμε να διαχωρίσουμε τις υπηρεσίες ανά τμήμα αλλά στην ανάλυση απαιτήσεων που προηγήθηκε ήταν φανερό ότι δεν έχει νόημα καθώς οι υπηρεσίες-portlets που το portal στην πιλοτική του μορφή υποστηρίζει και είναι μοναδικές για το κάθε τμήμα είναι ελάχιστες, είναι αυτές των ανακοινώσεων νέων από τις ιστοσελίδες των τμημάτων. Επιπλέον υπάρχουν φοιτητές και πτυχιούχοι που έχουν σπουδάσει σε παραπάνω από ένα τμήματα και θα ήταν λάθος να τους περιορίσουμε να βλέπουν μόνο υπηρεσίες σε επίπεδο τμήματος. Ταυτόχρονα υπάρχουν καθηγητές που διδάσκουν σε περισσότερα από ένα τμήματα. Υπενθυμίζεται το τρέχον πρόβλημα ότι οι λογαριασμοί του κεντρικού LDAP προς το παρόν δεν δίνουν αυτή τη επιπλέον πληροφορία.

Έτσι, αποφασίστηκε ο διαχωρισμός των καθηγητών και φοιτητών-πτυχιούχων σε επίπεδο σχολής. Ο κάθε χρήστης θα μπορεί ανάλογα τα ενδιαφέροντα του να επιλέξει ποιες υπηρεσίες επιπλέον θα προσθέτει στην εξατομικευμένη σελίδα του.

5.3.3 Τεχνική υποστήριξη χρηστών πύλης

Για τις ανάγκες υποστήριξης των χρηστών και για την καλή λειτουργία της πύλης θα υπάρχει τεχνική υποστήριξη των χρηστών από το τμήμα πληροφορικής της διοίκησης και το τμήμα δικτύων του ιδρύματος. Για οποιαδήποτε απορία, πληροφορία και πρόβλημα θα μπορεί να επικοινωνεί είτε μέσω ηλεκτρονικού μηνύματος ή τηλεφωνικά με το helpdesk των ηλεκτρονικών υπηρεσιών του ιδρύματος.

Μελλοντικά, έχει προγραμματιστεί η υλοποίηση και ενσωμάτωση portlet τεχνικής υποστήριξης στην πύλη. Ο χρήστης με τη συμπλήρωση μιας φόρμας θα μπορεί να ενημερώνει τους διαχειριστές για τα ερωτήματα ή αιτήματα του.

5.4 Κατηγορίες εφαρμογών/υπηρεσιών Πύλης

Οι εφαρμογές/υπηρεσίες που έχουν επιλεχτεί για ενσωμάτωση στη διαδικτυακή πύλη, ανάλογα τον τρόπο ενσωμάτωσης τους, μπορούν να διαχωριστούν στις παρακάτω κατηγορίες:

- Αυτές που απαιτούν σύνδεση με λογαριασμό χρήστη και χρειάζονται αλλαγή στον κώδικα του λογισμικού τους για υποστήριξη του CAS. Ένα τέτοιο παράδειγμα είναι το πληροφοριακό σύστημα του Γρ. Διασύνδεσης που παρέχει στους φοιτητές-πτυχιούχους να αναζητήσουν εργασία ή θέσεις πρακτικής άσκησης ανάλογα με το βιογραφικό τους και στις επιχειρήσεις να αναρτήσουν και να διαχειριστούν προσφερόμενες θέσεις εργασίας ή πρακτικής άσκησης. Το σύστημα αυτό έχει αναπτυχτεί σε php, jQuery και είχε μέχρι τώρα τη δική του βάση με τους λογαριασμούς των χρηστών. Έγιναν οι απαραίτητες αλλαγές στο κώδικα του και με την ενσωμάτωση του προγράμματος-πελάτη phpCAS που παρέχει το CAS. Τώρα, πλέον, ο χρήστης μπορεί να συνδεθεί απευθείας στην εφαρμογή με τα διαπιστευτήρια που έχει εγκρίνει το CAS και να έχει άμεση πρόσβαση στα εξατομικευμένα RSS feeds που προσφέρει το σύστημα μέσα από το uPortal
- Αυτές που απαιτούν σύνδεση με λογαριασμό χρήστη και μπορούν να συνεργαστούν με το CAS χωρίς αλλαγή στον κώδικα του λογισμικού τους. Σε αυτή τη κατηγορία υπάγεται η πλατφόρμα ασύγχρονης εκπαίδευσης e-class του Ακαδημαϊκού Διαδικτύου GUnet που χρησιμοποιεί το ίδρυμα μας. Για την ανάπτυξη του είναι υπεύθυνη η Ομάδα Ασύγχρονης Τηλεκπαίδευσης του GUnet και διανέμεται ελεύθερα ως λογισμικό ανοικτού κώδικα. Στις τελευταίες εκδόσεις υποστηρίζει σύνδεση χρήστη μέσω του CAS. Σε κάποιες από αυτές τις εφαρμογές ανοικτού κώδικα, προχωρήσαμε σε τροποποίηση τους ώστε να εκμεταλλευτούν τη λειτουργικότητα της πύλης. Για παράδειγμα το eclass, καλό θα ήταν να επέτρεπε στο κάθε χρήστη να έχει μια συνολική εικόνα των μαθημάτων του και να βλέπει συγκεντρωτικά τις ανακοινώσεις-νέα όλων των μαθημάτων στα οποία είναι εγγεγραμμένος.

- Αυτές που δεν απαιτούν σύνδεση με λογαριασμό χρήστη. Σε αυτή τη κατηγορία είναι τα portlets που έχουν ενημερωτικό χαρακτήρα, με δημοσίευση ανακοινώσεων-νέων από διάφορες ιστοσελίδες αλλά και κάποιες υπηρεσίες που λειτουργούν μέσα στο intranet του ιδρύματος, όπως η αναζήτηση βιβλίου από τη βάση δεδομένων της κεντρικής βιβλιοθήκης.

Για τις ανάγκες αυτών δημιουργήθηκαν κάποιοι νέοι τύποι portlets που μπορούν να χωριστούν σε 4 επιμέρους κατηγορίες.

- RSS portlet για ανάγνωση νέων από ιστοσελίδες που παρέχουν RSS feeds. Σημείωση: το uPortal έχει ενσωματωμένο ένα τέτοιο τύπο αλλά έχει αρκετούς περιορισμούς και δεν υποστηρίζει όλους του διαθέσιμους RSS feeders. Δημιουργήθηκε ένα νέο portlet για ανάγνωση όλων των παραλλαγών RSS feeds. Για τις ανάγκες του χρησιμοποιήθηκε το API του YQL (Yahoo! Query Language) μιας γλώσσας προγραμματισμού, με σύνταξη που θυμίζει sql, που επιτρέπει την δημιουργία ερωτημάτων φίλτρων και ένωση δεδομένων από διάφορες διαδικτυακές υπηρεσίες μεταξύ άλλων και RSS feeds.
- Parsing html Portlet κυρίως για ανάγνωση νέων από ιστοσελίδες που δεν παρέχουν RSS feeds, όπως είναι δυστυχώς αρκετές ιστοσελίδες του ιδρύματος λ.χ. η κεντρική σελίδα του ιδρύματος.
- HTML portlets που ενσωματώνουν μέρος του κώδικα HTML των αρχικών υπηρεσιών, όπως η υπηρεσία αναζήτηση εργαζομένου.
- Web Proxy portlet για ενσωμάτωση εφαρμογών δύσκολων περιπτώσεων όπως αυτές που απαιτούν Windows authentication, χρησιμοποιούν διαφορετική κωδικοποίηση. Παράδειγμα τέτοιας περίπτωσης το πρόγραμμα γραμματειών.

5.5 Ασφάλεια πύλης

Τα θέματα ασφάλειας της πύλης μπορούν να χωριστούν σε 2 επίπεδα, το ένα είναι προστασία δεδομένων εφαρμογής, το δεύτερο είναι ασφάλεια συναλλαγών και γενικότερα προστασία από κακόβουλες ενέργειες.

5.5.1 Προστασία δεδομένων uPortal

Η προστασία των δεδομένων εφαρμογής επιτυγχάνεται από τα μέτρα που παίρνονται προκειμένου να μην αποκτήσει κάποιος πρόσβαση στις πληροφορίες της βάσης δεδομένων[22].

Στη συγκεκριμένη εγκατάσταση χρησιμοποιούμε σαν βάση δεδομένων την προεπιλεγμένη βάση σε Hypersonic SQL, η οποία έρχεται με ενσωματωμένο λογαριασμό για πρόσβαση με πλήρη δικαιώματα στους πίνακες της: ένας λογαριασμός χωρίς κωδικό πρόσβασης. Ταυτόχρονα, το uPortal έχει ενσωματωμένους μια σειρά από δικούς του χρήστες-λογαριασμούς με διάφορα δικαιώματα και γνωστούς κωδικούς πρόσβασης για τη διαχείριση της πύλης αλλά και την επίδειξη των δυνατοτήτων της.

Προκειμένου να μην έχουμε τα παραπάνω κενά ασφαλείας, έγιναν οι παρακάτω τυπικές ενέργειες:

- Άλλαξαν οι κωδικοί πρόσβασης σε ορισμένους λογαριασμούς του uPortal που είναι απαραίτητοι, όπως του διαχειριστή.
- Διεγράφησαν ή απενεργοποιήθηκαν οι υπόλοιποι λογαριασμοί προς επίδειξη.
- Ο λογαριασμός διαχείρισης της βάσης απέκτησε κωδικό πρόσβασης.
- Ταυτόχρονα, αλλάξαμε tcp θύρα στην οποία εξυπηρετεί η Hypersonic SQL ώστε αν προσπαθήσει κακόβουλος να ανιχνεύσει τις ενεργές υπηρεσίες στο διακομιστή μας να μην μπορεί να βγάλει τόσο εύκολα συμπεράσματα για το τι είδους υπηρεσία είναι αυτή.
- Το πιο βασικό όμως είναι ότι το μηχάνημα στο οποίο είναι εγκατεστημένο το uPortal και το CAS βρίσκεται σε NAT δίκτυο κάτω από την επίβλεψη linux firewall το οποίο έχουμε εγκαταστήσει για τις ευρύτερες ανάγκες του ιδρύματος και η αποτελεσματικότητά του οποίου έχει δοκιμαστεί σε περίοδο ετών.

5.5.2 Ασφάλεια συναλλαγών

Το uPortal έρχεται με προεπιλεγμένες ρυθμίσεις να δουλεύει σε πρωτόκολλο http. Το ίδιο ισχύει και για το CAS. Για το uPortal είναι λογικό να λειτουργεί έτσι, αλλά για το CAS που

χρησιμοποιείται για ταυτοποίηση χρηστών και γίνεται αποστολή διαπιστευτηρίων δεν είναι. Έγιναν οι απαραίτητες ρυθμίσεις στα αρχεία ρυθμίσεων ώστε το CAS να λειτουργεί υποχρεωτικά και μόνο με το πρωτόκολλο https.

Είναι σημαντικό να σημειωθεί ότι η ρύθμιση του Tomcat ώστε να επωφεληθεί από Secure Sockets είναι συνήθως απαραίτητη μόνο όταν λειτουργεί ως ένα αυτόνομος web server. Κατά την εκτέλεση του Tomcat κυρίως ως ένα Servlet/JSP container πίσω από ένα άλλο web server, όπως ο Apache ή ο Microsoft IIS, είναι συνήθως απαραίτητο να ρυθμιστεί ο κύριος web server να διαπραγματευτεί τις συνδέσεις SSL από τους χρήστες και στη συνέχεια να προωθεί τα αιτήματα που προορίζονται για τον Tomcat μόνο μετά την αποκρυπτογράφηση αυτών των αιτημάτων. Ομοίως, ο Tomcat θα επιστρέψει το περιεχόμενο του και θα κρυπτογραφηθεί από το web server πριν επιστρέψει στον browser του χρήστη.

5.6 Περιγραφή του μηχανισμού πιστοποίησης χρηστών

Όπως περιγράψαμε παραπάνω, σαν κεντρικό κατάλογο για την εύρεση και πιστοποίηση των χρηστών της πύλης μας, πήραμε το κεντρικό LDAP του ιδρύματος που ενώνει τους περισσότερους καταλόγους δικτύου του ΤΕΙ. Το LDAP αυτό εξηγήσαμε, ότι έχει διάφορους περιορισμούς και κυρίως είναι κατάλογος μόνο για ανάγνωση, δεν μπορούμε να προσθέσουμε ή να τροποποιήσουμε τους λογαριασμούς που περιλαμβάνει, Μελλοντικά έχει προγραμματιστεί να έχει καλύτερη λειτουργικότητα και κυρίως είναι αυτό που χρησιμοποιείται για τη σύνδεση με το Shibboleth και τα άλλα εκπαιδευτικά ιδρύματα.

Επιπλέον με σκοπό να δείξουμε τη λειτουργικότητα υποστήριξης πολλαπλών καταλόγων δικτύου, προσθέσαμε σαν δεύτερη πηγή το LDAP του γρ. Διασύνδεσης, το οποίο έχει κυρίως δυο λειτουργίες:

1. την διαχείριση των χρηστών και των διαθέσιμων πόρων του γρ Διασύνδεσης αλλά και των συγγενών γραφείων στο Πρόγραμμα Δομής απασχόλησης και Σταδιοδρομίας (Δ.Α.ΣΤΑ.).
2. συμβατότητα και σύνδεση με την υπηρεσία Google Apps for Education. Το γρ. Διασύνδεσης είναι το πρώτο γραφείο που χρησιμοποιεί πιλοτικά αυτή την υπηρεσία και μελλοντικά υπάρχει η σκέψη μεταφοράς σε αυτό και άλλων τμημάτων του ιδρύματος.

Ταυτόχρονα κρατήσαμε σαν τρίτη πηγή για εύρεση χρηστών τη προεπιλεγμένη βάση δεδομένων σε HSQLDB που έχει ενσωματωμένους δικούς της χρήστες. Σημειώνεται ότι θα τη χρησιμοποιούσαμε και ως βάση δεδομένων των λειτουργιών της πύλης και ως μέσο αποθήκευσης των εξατομικευμένων επιλογών των χρηστών.

Να σημειώσουμε ότι τα 2 παραπάνω LDAP δεν είναι συμβατά μεταξύ τους. Αυτό σημαίνει ότι όχι μόνο οι χρήστες, αλλά τα περισσότερα αντικείμενα και οι ιδιότητες τους είναι διαφορετικά. Το ίδιο ισχύει και για τη τρίτη πηγή της βάσης δεδομένων. Στόχος ήταν να δείξουμε ότι το CAS και κατά συνέπεια το uPortal μπορεί να διαβάζει από πολλαπλές και ασύμβατες πηγές καταλόγων και βάσεων δεδομένων και παρόλα αυτά έχει τη δυνατότητα να αντιστοιχεί τους χρήστες με τις διαφορετικές ιδιότητες τους στους σχεδιασμένους ρόλους που έχουμε ορίσει.

5.7 Πρόσβαση Διαδικτυακής Πύλης σε ΑΜΕΑ

Η επιλεγμένη πλατφόρμα uPortal στη προηγούμενη έκδοση της ήταν συμβατή με το επίπεδο 1 του WCAG 2, ακολουθούσε δηλαδή τους διεθνώς αναγνωρισμένους κανόνες και οδηγίες Web Content Accessibility Guidelines προσβασιμότητας του W3C ώστε να αποφεύγεται ο αποκλεισμός ειδικών ομάδων πολιτών (ΑΜΕΑ).

Στη παρούσα έκδοση της έχει επιλέξει να μην συμμορφώνεται, αυστηρά πλέον, σε κανένα συγκεκριμένο σύνολο κατευθυντήριων γραμμών καθώς αναγνωρίζει ότι υπάρχουν πολλά πρότυπα και οδηγίες για την τήρηση της προσβασιμότητας σε παγκόσμιο επίπεδο, με την κάθε χώρα να θέτει τις δικές τις πολιτικές προσβασιμότητας και υιοθέτησης συμμόρφωσης σε διαφορετικούς κανόνες.

Το uPortal χρησιμοποιεί τις ευρέως αποδεκτές τεχνικές για την πρόσβαση στο διαδίκτυο και την παροχή εξατομίκευσης και προσαρμογής και επιτρέπει στις διάφορες υλοποιήσεις του την προσαρμογή πλήρως του γραφικού θέματος του, εξασφαλίζοντας ότι μια ποικιλία από πρότυπα προσβασιμότητας μπορούν να φιλοξενηθούν.

Παρόλα αυτά το uPortal (έκδοση 4) παρέχει μια σειρά από σημαντικές δυνατότητες προσβασιμότητας όπως οι εξής:

1. Υποστηρικτική Τεχνολογία

- Καλή σημασιολογική σήμανση που λειτουργεί καλά ακόμα και όταν είναι απενεργοποιημένη CSS.
- Υποστήριξη αλλαγής μεγέθους του κειμένου με κλιμάκωση από 50% έως και 200% του αρχικού μεγέθους επιτρέποντας μια σειρά από μεγεθύνσεις οθόνης,
- Χρήση εναλλακτικών περιγραφών για τις εικόνες και τις συνδέσεις.
- Η χρήση JavaScript και η υποστήριξη AJAX είναι παραμετροποιήσιμη και προαιρετική.

2. Πλοήγηση μέσω πληκτρολογίου

- Μετάβαση σε συνδέσεις μέσω συντομεύσεων πληκτρολογίου.
- Η αποτελεσματική χρήση του πλήκτρου tab για μετακίνηση σε διαφορετικά αντικείμενα της ιστοσελίδας.
- Παραμετρική επιλογή πλήκτρων πρόσβασης.

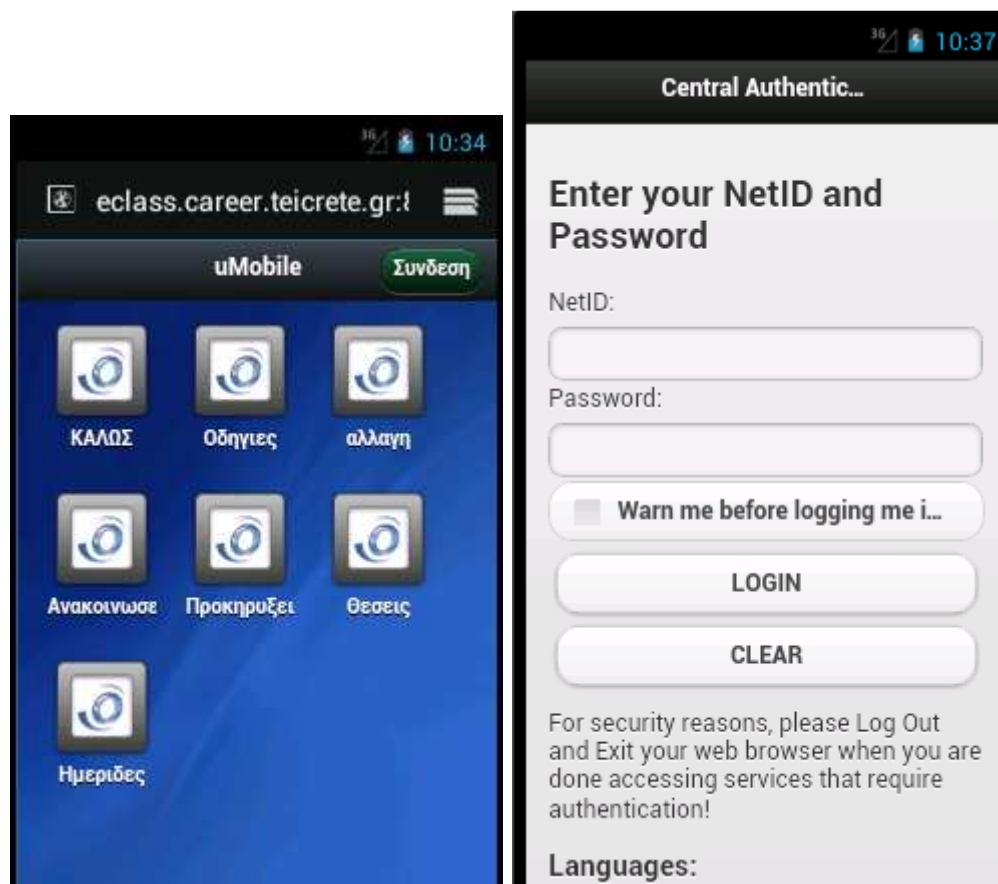
Κυρία φροντίδα μας κατά την σχεδίαση του δικού μας γραφικού θέματος διεπαφής καθώς και την ανάπτυξη όλων των portlets που δημιουργήθηκαν για τις ανάγκες της πύλης μας να είναι συμβατές με όλα τα παραπάνω.

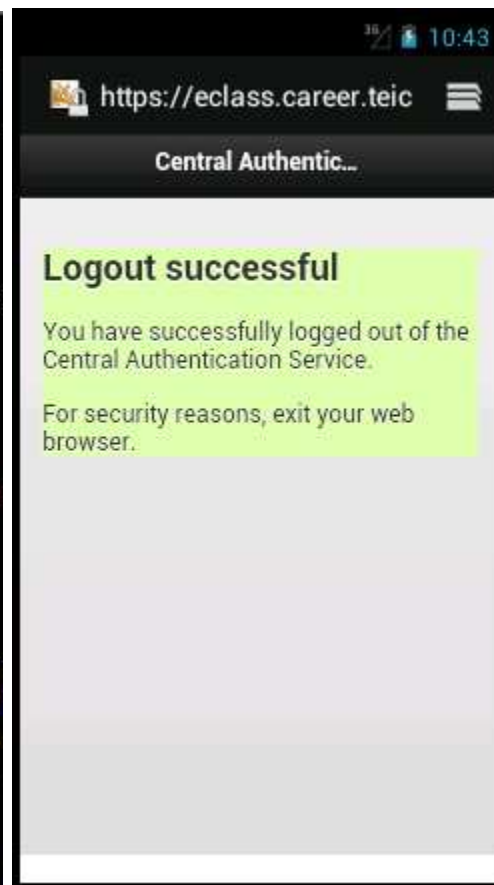
5.8 Πολυκαναλική διάθεση των προσφερόμενων υπηρεσιών

Η πλατφόρμα uPortal έχει αναπτύξει και προσφέρει το πλαίσιο uMobile το οποίο επιτρέπει πρόσβαση σε αυτό μέσω κινητού τηλεφώνου ή άλλων έξυπνων συσκευών. Αυτό επιτυγχάνεται με την προσαρμογή των προσφερόμενων υπηρεσιών στις διαφορετικές απαιτήσεις οθόνης της κάθε συσκευής για αποδοτική διάθεσή τους μέσω διαδικτύου.

Όταν ο χρήστης μέσω του κινητού τηλεφώνου και του προγράμματος πλοήγησης επισκεφτεί την ιστοσελίδα της πύλης μας αυτόματα βλέπει μια τροποποιημένη αρχική σελίδα. Η κάθε υπηρεσία και portlet που είναι διαθέσιμα έχουν τη μορφή εικονιδίου. Ο χρήστης επιλέγοντας το

εικονίδιο που επιθυμεί αποκτά πρόσβαση στο αντίστοιχο portlet. Ταυτόχρονα βέβαια έχει τη δυνατότητα σύνδεσης και ταυτοποίησης του ώστε να αποκτήσει πρόσβαση σε εξατομικευμένο πληροφοριακό υλικό και υπηρεσίες ανάλογα το ρόλο και ιδιότητα του.





Εικόνα 5.1:Ενδεικτικές οθόνες εμφάνισης της πύλης μέσω smartphone

5.9 Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκε η επίλυση επιμέρους θεμάτων για τον σχεδιασμό του συστήματος και την ανάπτυξη του μηχανισμού πρόσβασης καθώς και η ενσωμάτωση των υπηρεσιών.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Στην ενότητα 5.2 αναλύσαμε το πώς επιλέχτηκε η πλατφόρμα uPortal ως βάση για την ανάπτυξη της εκπαιδευτικής μας πύλης. Η ενότητα 5.3 εξήγησε πως οργανώθηκαν οι ρόλοι-κατηγορίες των χρηστών του συστήματος και πως επιτυγχάνεται η ταυτοποίηση τους. Η ενότητα 5.4 αναφέρθηκε στην κατηγοριοποίηση των εφαρμογών/υπηρεσιών της πύλης. Στην ενότητα 5.5 έγινε περιγραφή των επιπλέον μέτρων πάρθηκαν για την ασφάλεια της τόσο σε επίπεδο προστασίας δεδομένων όσο και ασφάλειας συναλλαγών. Η ενότητα 5.6 ήταν αφιερωμένη στο μηχανισμό πιστοποίησης χρηστών. Η ενότητα 5.7 εξέτασε τις τεχνικές σχεδίασης για πρόσβαση σε άτομα με ειδικές ανάγκες και η ενότητα 5.8 αναφέρθηκε στη πρόσβαση μέσω έξυπνων συσκευών όπως τηλεφώνων. Τέλος η ενότητα 5.9 παρουσίασε μια σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τους επιμέρους στόχους #3, #4, #5 και #6 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Το επόμενο κεφαλαίο θα εξετάσει τη σχεδίαση των μικροεφαρμογών που αναπτύχθηκαν ως τεκμηρίωση των επιμέρους στόχων #7 και #8 όπως αυτοί αναφέρονται στην εισαγωγή.

Κεφάλαιο 6

Σχεδίαση των Μικροεφαρμογών-Portlets

6.1 Εισαγωγή

Στο κεφάλαιο αυτό θα γίνει περιγραφή της μεθοδολογίας και ανάπτυξης των portlets που δημιουργήθηκαν εξολοκλήρου από μας καθώς και η ενσωμάτωση των ήδη υφισταμένων διαδικτυακών υπηρεσιών για τις ανάγκες της εκπαιδευτικής πύλης.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Στην ενότητα 6.2 γίνεται η παρουσίαση των portlets και των υπηρεσιών που παρέχει η πύλη μας ανά κατηγορία χρήστη. Η ενότητα 6.3 αναφέρεται στις προδιαγραφές των portlets που δημιουργήθηκαν. Η ενότητα 6.4 εξετάζει τη σχεδίαση τους. Η ενότητα 6.5 περιγράφει τη λειτουργικότητα των portlets ανάγνωσης πολλαπλών πηγών δεδομένων. Στην ενότητα 6.6 περιγράφεται ο τρόπος ενσωμάτωσης των διαδικτυακών υπηρεσιών του ιδρύματος. Τέλος, η ενότητα 6.7 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

6.2 Υπηρεσίες-εφαρμογές ανά χρήστη

Το uPortal, που αποτελεί το εργαλείο ανάπτυξης της πύλης μας, έχει ενσωματωμένα κάποια portlets τα περισσότερα από αυτά είναι για τον διαχειριστή του συστήματος. Ταυτόχρονα, δημιουργήθηκαν και μια σειρά από portlets για την υποστήριξη των διαδικτυακών υπηρεσιών που προσφέρει το ίδρυμα και έχουν επιλεχτεί να βρίσκονται στη διαδικτυακή μας πύλη.

Στο παρακάτω αναλυτικό κατάλογο σε κάθε υπηρεσία–portlet υπάρχει ένδειξη αν είναι portlet ενσωματωμένο (built-in), αν το προσθέσαμε εμείς αλλά δεν είναι δημιουργία δικιά μας (import) και τέλος αν κατασκευάστηκε για τις ανάγκες του ιδρύματος (created).

Οι υπηρεσίες που προσφέρει το σύστημα μπορούν να διαχωριστούν ανά ρόλο ως εξής:

Διαχειριστής συστήματος (Admin)

- Audit DLM Fragments (builtin), διαχείριση των σελίδων με βάση το ρόλο του χρήστη
- Cache Manager (builtin), διαχείριση της προσωρινής μνήμης της πύλης
- Exit Fragment Administration (builtin) διαχείριση των σελίδων με βάση το ρόλο του χρήστη
- Groups Administration (builtin), διαχείριση των ομάδων χρηστών
- IFramePortlet (builtin), προσθήκη portlet τύπου iframe
- Import Export Portlet (builtin), εισαγωγή/εξαγωγή portlets
- Most Popular Apps (builtin), ποια portlets προτιμούν οι χρήστες
- Permissions Administration (builtin), διαχείριση δικαιωμάτων
- Portal Administration (builtin), διαχείριση της πύλης
- Portlet Administration (builtin), διαχείριση των portlets
- Translator (builtin), μεταγλώττιση όρων σε διαφορετικές γλώσσες

- User Administration (builtin), διαχείριση χρηστών συστήματος
- Who's Online (builtin), ποιοι χρήστες είναι συνδεδεμένοι

Campus (ρόλοι φοιτητές, πτυχιούχοι, διοικητικοί υπάλληλοι και καθηγητές)

- Ηλ. Ταχυδρομείο (import) πρόσβαση και ανάγνωση των εισερχομένων μηνυμάτων από τους διάφορους εξυπηρετητές email του ιδρύματος
- Πρόσβαση στο ιστολόγιο (blog) του ιδρύματος (created)
- Εύρεση βιβλίου από τον κατάλογο τίτλων της κεντρικής βιβλιοθήκης του ιδρύματος (created)
- Πληροφοριακό σύστημα Γρ. Διασύνδεσης (created)
- Πρόσβαση στο PKI (GrNet), Υπηρεσία αίτηση Προσωπικών Ψηφιακών Πιστοποιητικών (created)
- Πρόσβαση στο Pithos (GrNet), Υπηρεσία απομακρυσμένης αποθήκευσης αρχείων (created)
- Πρόσβαση στο myNetLab (GrNet), Υπηρεσία εργαστηρίου δικτύων (created)
- Πρόσβαση στο IPTV (GrNet), Υπηρεσία ψηφιακής τηλεόρασης (created)
- Πρόσβαση στο RTS (GrNet), Υπηρεσία Voip & τηλεδιασκέψεων (created)
- Πρόσβαση στο Megaron Plus (GrNet), Υπηρεσία ζωντανής σύνδεσης με το Μέγαρο Αθήνας (created)
- Πρόσβαση στο Forrester (GrNet), (created)

- Πρόσβαση στο Scopus, Βάση δεδομένων δημοσιεύσεων και επιστημονικών άρθρων (created)

Academic (ρόλοι φοιτητές, πτυχιούχοι και καθηγητές)

- eclass (import), Πλατφόρμα Ασύγχρονης Τηλεκπαίδευσης
- πρόσβαση στο MSDNAA & DreamSpark, Microsoft λογισμικο (created)

Φοιτητές, Πτυχιούχοι

- σύνδεση με πρόγραμμα γραμματειών (import)
- πρόσβαση στο Eudoxus (GrNet), διανομή βιβλίων (created)
- πρόσβαση στην έκδοση φοιτητικής ταυτότητας (created)

Guest-(όλοι οι χρήστες του συστήματος)

- google-maps (import)
- google-search (import)
- google-translate (import)
- Dictionary (import)
- Search (builtin)
- Site map (builtin)
- Weather (import)
- User Locales Selector επιλογή γλώσσας διεπαφής (builtin)
- Ανακοινώσεις-νέα από κεντρική ιστοσελίδα TEI Κρήτης (created)

- Προκηρύξεις διαγωνισμών από κεντρική ιστοσελίδα ΤΕΙ Κρήτης (created)
- Προκηρύξεις θέσεων εργασίας από κεντρική ιστοσελίδα ΤΕΙ Κρήτης (created)
- Ανακοινώσεις-νέα από ιστοσελίδες σχολών προγραμμάτων σπουδών (created)
- Ανακοινώσεις-νέα από ιστοσελίδες τμημάτων σχολών προγραμμάτων σπουδών (created)
- Ανακοινώσεις-νέα από ιστοσελίδα ΚΕΕΔ (created)
- Bookmarks (builtin), αγαπημένων προσωπικών συνδέσμων
- Reset My Layout (builtin), ο χρήστης μπορεί να επαναφέρει τη διάταξη περιεχομένου της σελίδας του στην αρχική που σχεδιάστηκε ειδικά για το ρόλο του.
- Ανακοινώσεις-νέα από ιστοσελίδα Γρ. Δημοσίων & Διεθνών Σχεσεων (created)
- Ανακοινώσεις-νέα από ιστοσελίδα τμήματος μέριμνας φοιτητών (created)
- Ανακοινώσεις-νέα από ιστοσελίδα ΔΑΣΤΑ (created)
- Ανακοινώσεις-νέα από ιστοσελίδα Γρ. Διασύνδεσης (created)
- Ανακοινώσεις-νέα από ιστοσελίδα Πρακτικής άσκησης (created)
- Αναζήτηση εργαζομένου στο ΤΕΙ Κρήτης (created).

Η δυνατότητα πλήρης διαχείρισης των κατηγοριών-ρόλων χρηστών για την πρόσβαση σε όλα τα παραπάνω portlets μέσα από την πύλη είναι πολύ εύκολη και γρήγορη.

6.3 Προδιαγραφές των portlets

Τα portlets που δημιουργήθηκαν έχουν σχεδιαστεί να ακολουθούν συγκεκριμένες προδιαγραφές:

- Συμβατότητα με άλλες διαδικτυακές πύλες.
- Υποστήριξη μεταγλώττισης, ώστε να εμφανίζονται το περιεχόμενό τους στη γλώσσα επιλογής του χρήστη.
- Ευέλικτη παραμετροποίηση χωρίς τροποποίηση του πηγαίου κώδικά τους.
- Υποστήριξη JavaScript και χρήση πρότυπου CSS2 για το αισθητικό αποτέλεσμα - εμφάνιση χωρίς προβλήματα συμβατότητας με τα άλλα portlets και τη πύλη.
- Αναγνώριση όλων των ορισμένων ιδιοτήτων - ρόλων που υποστηρίζει η διαδικτυακή πύλη, ώστε το πληροφοριακό υλικό τους να μεταβάλλεται ανάλογα με το χρήστη που έχει συνδεθεί.
- Πλήρη λειτουργικότητα ανάγνωσης-παρουσίασης πολλαπλών πηγών δεδομένων από άλλες ιστοσελίδες (RSS, XML κλπ.).

6.4 Σχεδιασμός των portlets

6.4.1 Συμβατότητα-επικοινωνία με άλλες διαδικτυακές πύλες

Τα portlets είναι πλήρως συμβατά με το πρότυπο JSR168. Ακολουθώντας τις συγκεκριμένες προδιαγραφές, τα portlets αυτά δεν περιορίζονται έτσι σε καμιά διαδικτυακή πύλη ή σε κανένα Java Container. Μπορούν να εγκατασταθούν σε όλες τις πύλες που είναι συμβατές με αυτό το πρότυπο.

Ένα μεγάλο πλεονέκτημα του πρότυπου αυτού είναι ότι δεν επιτρέπει μόνο την εγκατάσταση αλλά και τη πλήρη ενσωμάτωση του portlet ακόμα και αν η πύλη δεν διαχειρίζεται τους ίδιους ρόλους-ιδιότητες των χρηστών με αυτούς που αναγνωρίζει το portlet. Μέσα από το αρχείο παραμέτρων του που δομείται με XML αρκεί να γίνει μια αντιστοίχιση του ρόλου πύλης με το ρόλο που ξέρει το portlet.

Γνωρίζοντας τους διαθέσιμους ρόλους και ανακτώντας τις ιδιότητες και το ρόλο του συνδεδεμένου χρήστη από το portal, μπορούν να προσαρμόζουν το πληροφοριακό υλικό που εμφανίζουν. Οι ιδιότητες που αναγνωρίζουν, ορίζονται μέσα από το αρχείο παραμέτρων τους

6.4.2 Συμβατότητα με τα υπόλοιπα portlets

Ένα portlet είναι μόνο ένα αντικείμενο της πύλης και κάθε ιστοσελίδα αποτελείται συνήθως από πολλαπλά portlets. Αυτό αποτελεί σχεδιαστικό πρόβλημα γιατί τόσο η εμφάνιση τους όσο και η λειτουργικότητα τους μέσω javascript μπορεί εύκολα να προκαλέσει πρόβλημα συμβατότητας με τα υπόλοιπα στην ίδια σελίδα ή ακόμα και με το ίδιο portal που για τις ανάγκες του χρησιμοποιεί δικές του JavaScript βιβλιοθήκες και συγκεκριμένο αισθητικό θέμα παρουσίασης (template).

Η πρόβλεψη σχεδίασης και επίλυσης του συγκεκριμένου προβλήματος είναι επιτακτική. Τα portlets που υλοποιήθηκαν για αυτό το λόγο, έχουν σχεδιαστεί με συγκεκριμένη μεθοδολογία:

- Χρησιμοποιούν όλα jQuery JavaScript με την ειδική προτεινόμενη σύνταξη, ώστε να μην εμποδίζουν την συνύπαρξη τους στην ίδια σελίδα με άλλα portlets με χρήση ίδιου ή διαφορετικού JavaScript library.
- Είναι συμβατά με το πρότυπο CSS2 με δικό τους προεπιλεγμένο θέμα και συντακτικά είναι έτσι δομημένα ώστε να μην εμποδίζουν την αισθητική εμφάνιση άλλων portlets. Αυτό επιτυγχάνεται όταν όλα τα HTML αντικείμενα τους έχουν το δικό τους μοναδικό όνομα-id. Έτσι οποιαδήποτε αναφορά σε CSS αρχεία και τροποποίηση εμφάνισης αυτών των αντικειμένων δεν επηρεάζει τα υπόλοιπα portlets ή την ίδια την πύλη.

6.4.3 Παραμετροποίηση

Τα portlets είναι πλήρως παραμετροποιημένα, με όλες τις προεπιλογές τους εύκολα τροποποιήσιμες, μέσα από το τυπικό αρχείο παραμέτρων που έχουν τα portlets.

Ταυτόχρονα, η επιλογή ανάπτυξης των portlets με χρήση του spring framework και ειδικότερα του Spring Portlet MVC δίνει εξαιρετικές δυνατότητες στη συντήρηση και επεκτασιμότητα τους.

6.4.4 Υποστήριξη μεταγλώττισης

Τα portlets υποστηρίζουν μεταγλώττιση. Έρχονται όλα με μετάφραση των λεκτικών τους όρων σε ελληνικά και αγγλικά σε ξεχωριστά ASCII αρχεία αλλά πολύ εύκολα και χωρίς τροποποίηση του πηγαίου κώδικα τους μπορούν να υποστηρίξουν και άλλες γλώσσες μόνο με τη προσθήκη

των αναλόγων αρχείων μεταγλώττισης. Η επιλογή των διαθέσιμων γλωσσών που αναγνωρίζει το portlet γίνεται από το αρχείο παραμέτρων της εφαρμογής.

6.4.5 Σχεδιασμός Λειτουργικότητας

Για την ανάπτυξη των portlets ακολουθήθηκε πιστά η λογική model-view-controller (MVC) που επιτρέπει το διαχωρισμό επεξεργασίας δεδομένων και εμφάνισης στη σελίδα. Αυτό επιτρέπει το διαχωρισμό των αρχείων που είναι υπεύθυνα για την εμφάνιση του περιεχομένου με τη δημιουργία της html, με αυτά που είναι υπεύθυνα με τη λογική της εφαρμογής και της μοντελοποίησης των τυχόν αντικειμένων που χρειάζεται το portlet (java classes). Ειδικότερα, για την εμφάνιση του περιεχομένου χρησιμοποιήθηκαν JavaServer Pages και JSP Standard Tag Library.

Για την εύκολη ενσωμάτωση τρίτων προγραμμάτων στη λογική και τελικά τη λειτουργικότητα των portlets, υποστηρίζουν τη χρήση των λεγόμενων φίλτρων, μέσω προσθήκης λίγων γραμμών στο αρχείο παραμέτρων του κάθε portlet ενώ με αυτό το τρόπο και συντηρούν ιστορικό καταγραφής ενεργειών τους σε αρχείο .log με τη χρήση της βιβλιοθήκης common-logging & log4j της κοινοπραξίας apache.

6.5 Σχεδιασμός portlet ανάγνωσης πολλαπλών πηγών δεδομένων

Οι ιστόχωροι των διάφορων σχολών και τμημάτων έχουν αναπτυχτεί σε βάθος μεγάλου χρονικού διαστήματος και χρησιμοποιούν είτε μια πληθώρα διάφορων συστημάτων διαχείρισης πληροφοριακού υλικού (Content Management Systems) ή ακόμα και καθαρά στατικών σελίδων. Τα περισσότερα από αυτά τα συστήματα, είτε υποστηρίζουν διαφορετικά πρότυπα καναλιών τροφοδοσίας (RSS) ή δεν παρέχουν καθόλου τέτοια δυνατότητα. Το πρόβλημα συγκέντρωσης πληροφοριακού υλικού ήταν σοβαρό αφού κανένα διαθέσιμο portlet δεν μπορούσε να διαβάσει όλα τα παραπάνω κανάλια. Ταυτόχρονα, έπρεπε να βρεθεί τρόπος συγκέντρωσης πληροφοριακού υλικού και από καθαρά στατικές σελίδες.

Έτσι αποφασίσαμε να προχωρήσουμε στη δημιουργία ενός portlet που θα μπορούσε να ξεπεράσει όλα τα παραπάνω προβλήματα. Αρχικά η σκέψη ήταν να γίνει η ανάκτηση δεδομένων με ασύγχρονο τρόπο με χρήση JavaScript & Ajax άλλα πρόεκυπτε το αζεπέραστο πρόβλημα ότι η

τέτοια κλήση Ajax επιτρέπεται για λόγους ασφάλειας μόνο σε άλλες ιστοσελίδες του ίδιου ιστόχωρου. Αυτό σημαίνει ότι αν η πύλη μας π.χ. βρισκόταν στη διεύθυνση <http://mytei.teicrete.gr> δεν θα μπορούσε να ανακτήσει html δεδομένα από την ιστοσελίδα <http://www.teicrete.gr>.

Έτσι προχωρήσαμε στη ενσωμάτωση μιας web based υπηρεσίας που παρέχει το Yahoo! με τη χρήση της Yahoo Query Language [65]. Το δικό μας “universal portlet” γνωρίζοντας, από το αρχείο παραμέτρων του, είτε ποιο κανάλι τροφοδοσίας θέλουμε να ανακτήσουμε ή ποια στατική σελίδα html θέλουμε να διαβάσουμε, στέλνει ασύγχρονα τη διεύθυνση στην υπηρεσία της Yahoo!. Αυτή αφού συλλέξει τα δεδομένα τα επιστρέφει στο portlet με μορφή JSON και το portlet αναλαμβάνει το κατάλληλο parsing των δεδομένων και την εμφάνιση τους στην ιστοσελίδα.

Για να γίνει κάτι τέτοιο όμως, σε στατικές ιστοσελίδες, ήταν λίγο πιο πολύπλοκο αφού έπρεπε σε κάποιες από αυτές να «μαρκάρουμε» (bookmark) το μέρος της ιστοσελίδας που επιθυμούσαμε να ανακτήσουμε. Υποχρεωθήκαμε, λοιπόν, να τροποποιήσουμε κάποιες ιστοσελίδες διάφορων ιστοχώρων, όπως του κεντρικού, για να έχουμε το επιθυμητό αποτέλεσμα.

6.6 Ενσωμάτωση διαδικτυακών υπηρεσιών του Ιδρύματος

Μια βασική δυνατότητα της πλατφόρμας uPortal είναι η ενσωμάτωση άλλων παρεχομένων διαδικτυακών υπηρεσιών που βρίσκονται σε διαφορετικούς ιστοχώρους του ιδρύματος. Αυτές οι υπηρεσίες μπορεί να έχουν γραφτεί με μια πληθώρα διαφορετικών εργαλείων ανάπτυξης και γλώσσες προγραμματισμού (java, php, .NET) , η κάθε μια έχει τη δική της βάση δεδομένων και τους δικούς της χρήστες. Το πρόβλημα γίνεται ακόμα πιο μεγάλο αν υπολογιστεί ότι έχουν και διαφορετικούς τρόπους πιστοποίησης και εξουσιοδότησης των χρηστών τους.

Οι εφαρμογές αυτές απαιτούν ειδική τροποποίηση του πηγαίου κώδικα τους ώστε να επιτρέψουν τη επικύρωση ταυτότητας από το CAS που χρησιμοποιεί και η πύλη μας. Για το λόγο αυτό η ομάδα ανάπτυξης του CAS παρέχει τα αντίστοιχα API που επιτρέπουν τη σύνδεση, επικοινωνία και τελικά πιστοποίηση του χρήστη μέσω του CAS. Στη συνέχεια η εφαρμογή μπορεί ανάλογα να προχωρήσει σε εξουσιοδότηση του χρήστη είτε ανακτώντας τους ρόλους και τις ιδιότητες του από το CAS είτε με τους δικούς της αυτόνομους μηχανισμούς έγκρισης δικαιωμάτων.

Τέτοιο παράδειγμα εφαρμογής είναι το πληροφοριακό σύστημα του Γρ. Διασύνδεσης που παρέχει στους φοιτητές-πτυχιούχους να αναζητήσουν εργασία ή θέσεις πρακτικής άσκησης ανάλογα με το βιογραφικό τους και στις επιχειρήσεις να αναρτήσουν και να διαχειριστούν προσφερόμενες θέσεις εργασίας ή πρακτικής άσκησης. Το σύστημα αυτό έχει αναπτυχθεί σε php, jQuery και είχε μέχρι τώρα τη δική του βάση με τους λογαριασμούς των χρηστών. Έγιναν οι απαραίτητες αλλαγές στο κώδικα του και με την ενσωμάτωση του προγράμματος-πελάτη phpCAS που παρέχει το CAS. Τώρα πλέον ο χρήστης μπορεί να συνδεθεί απευθείας στην εφαρμογή με τα διαπιστευτήρια που έχει εγκρίνει το CAS και να έχει άμεση πρόσβαση στα εξατομικευμένα RSS feeds που προσφέρει το σύστημα, να τροποποιήσει το βιογραφικό του και να αναζητήσει θέσεις εργασίας ή πρακτικής άσκησης μέσα από την ιστοσελίδα της πύλης μας.

Ταυτόχρονα, το εκπαιδευτικό ίδρυμα μας, σαν μέλος του Gnet της ΕΔΕΤ, παρέχει στα μέλη της ακαδημαϊκής κοινότητας του πρόσβαση σε μια σειρά υπηρεσιών που υποστηρίζουν το CAS. Έτσι προχωρήσαμε σε ενσωμάτωση αυτών των υπηρεσιών και πρόσβασης τους μέσα από την πύλη μας.

6.7 Σύνοψη

Στο κεφάλαιο αυτό περιγράψαμε την μεθοδολογία ανάπτυξης όλων των portlets που υλοποιήθηκαν εξολοκλήρου από μας καθώς και η ενσωμάτωση των ήδη υφισταμένων διαδικτυακών υπηρεσιών για τις ανάγκες της εκπαιδευτικής πύλης για τις ανάγκες της κοινότητας του ΤΕΙ Κρήτης.

Ξεκινήσαμε στην ενότητα 6.2 με τον κατάλογο των διαθέσιμων portlets. Στην ενότητα 6.3 περιγράψαμε τις προδιαγραφές που έπρεπε να έχουν τα καινούργια portlets. Στην ενότητα 6.4 αναλύσαμε την σχεδίαση που ακολουθήσαμε για να καλύψουμε διάφορα θέματα είτε συμβατότητας με την διαδικτυακή πύλη και τα υπόλοιπα portlets, είτε υποστήριξη μεταγλώττισης και δυνατότητα φιλικής παραμετροποίησης. Ταυτόχρονα είδαμε και άλλα σχεδιαστικά ζητήματα για την παρεχόμενη γενική λειτουργικότητα τους. Στην ενότητα 6.5 δώσαμε ειδική έμφαση στη σχεδίαση των portlets ανάγνωσης πολλαπλών πηγών δεδομένων. Επίσης στην ενότητα 6.6 είδαμε με ποιο τρόπο ενσωματώσαμε διαδικτυακές υπηρεσίες του ιδρύματος στη εκπαιδευτική πύλη μας. Τέλος, στην παρούσα ενότητα παρουσιάζεται η σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τους επιμέρους στόχους #7 και #8 όπως αυτοί αναφέρονται στην εισαγωγή της παρούσας διατριβής. Το επόμενο κεφαλαίο θα εξετάσει τις μεθόδους αξιολόγησης ευχρηστίας της πύλης ως τεκμηρίωση του επιμέρους στόχου #9 όπως αυτός αναφέρεται στην εισαγωγή.

Κεφάλαιο 7

Αξιολόγηση της Πύλης

7.1 Εισαγωγή

Στο κεφάλαιο αυτό, θα γίνει περιγραφή της μεθοδολογίας αξιολόγησης της εκπαιδευτικής πύλης. Όπως είδαμε στο δεύτερο κεφάλαιο υπάρχουν διάφορες τεχνικές και μέθοδοι για την μέτρηση ευχρηστίας μιας εφαρμογής ή υπηρεσίας. Για τις ανάγκες αξιολόγησης της διαδικτυακής πύλης χρησιμοποιήσαμε την αναλυτική μέθοδο αξιολόγησης συμβατότητας με κανόνες και πρότυπα και την εμπειρική-διερευνητική μέθοδο συμπλήρωσης ερωτηματολογίων.

Η δομή αυτού του κεφαλαίου είναι ως εξής: Στην ενότητα 7.2 γίνεται περιγραφή των αποτελεσμάτων της αξιολόγησης συμβατότητας ως μιας αναλυτικής μεθόδου, βασισμένης σε θεωρητικά μοντέλα τα οποία προσομοιώνουν τη συμπεριφορά του χρήστη και την συμβατότητα της εφαρμογής με αναγνωρισμένα πρότυπα και κανόνες. Η ενότητα 7.3 αναφέρεται στην διερευνητική μέθοδο συμπλήρωσης ερωτηματολογίων και τα αποτελέσματα της αξιολόγησης από τους ίδιους χρήστες της πύλης. Τέλος, η ενότητα 7.4 παρουσιάζει μια σύνοψη του κεφαλαίου αυτού.

7.2 Μέθοδος αξιολόγησης συμβατότητας.

Κατά την διαδικασία αξιολόγησης χρησιμοποιήσαμε online εργαλεία συμβατότητας με διεθνώς αναγνωρισμένους κανόνες και πρότυπα, τα αποτελέσματα των οποίων περιγράφονται παρακάτω. Σκοπός μας ήταν να μετρηθούν, από ανεξάρτητες διαφορετικές πηγές, οι βαθμοί αποδοτικότητας και προσβασιμότητας της πύλης καθώς και ο βαθμός συμβατότητας με γνωστά πρότυπα και η αποτελεσματικότητα της σχεδίασης διεπαφής της.

7.2.1 Χρόνος φόρτωσης σελίδας

Σύμφωνα με το pingdom.com η βαθμολογία της κεντρικής ιστοσελίδας της πύλης είναι 82/100 κατατάσσοντας την, γρηγορότερη από το 81% των ιστοσελίδων του συνόλου δοκιμασμένων ιστοχώρων.



Εικόνα 7.1:Καταταξη σύμφωνα με το pingdom.com

Σύμφωνα με το PageSpeed Insights της Google η βαθμολογία της κεντρικής ιστοσελίδας της πύλης είναι 95/100.

- Overview
- Critical Path Explorer
- Low priority (7)**
- Avoid landing page redire...
- Leverage browser caching
- Avoid CSS @import
- Optimize images
- Minify JavaScript
- Minify CSS
- Specify a Vary: Accept-...
- Experimental rules (1)**
- Already done! (21)**

Overview

The page T.E.I. of Cretes Portal: Welcome got an overall PageSpeed Score of **95** (out of 100). [Learn more](#)

This PageSpeed Report is generated for this page as it appears in desktop browsers. To get suggestions on how to optimize the performance of this page for mobile devices, generate a mobile report.



Suggestion Summary

Click on the rule names to see suggestions for improvement.

- **High priority.** These suggestions represent the largest potential performance wins for the least development effort. However, there are no high priority suggestions for this site. Good job!
- **Medium priority.** These suggestions may represent smaller wins or much more work to implement. However, there are no medium priority suggestions for this site. Good job!
- **Low priority.** These suggestions represent the smallest wins. You should only be concerned with these items after you've handled the higher priority items. [Avoid landing page redirects](#), [Leverage browser caching](#), [Avoid CSS @import](#), [Optimize images](#), [Minify JavaScript](#), [Minify CSS](#), [Specify a Vary: Accept-encoding header](#)
- **Experimental rules.** These suggestions are experimental, but do not affect the overall PageSpeed Score. Consider this item as a pointer to explore, but your mileage might vary: [Eliminate unnecessary reflows](#)
- **Already done!** There are no suggestions for these rules, since this page already follows these best practices. Good job!

Εικόνα 7.2: Καταταξη σύμφωνα με το pagespeed insights

Στο pagespeed insights είχαμε την ευκαιρία να δοκιμάσουμε την ταχύτητα και της έκδοσης για κινητές συσκευές αποσπώντας ξανά ένα 95/100 βαθμολογία

PageSpeed Insights — class.career.teicrete.gr:8080/uPortal/normal/render.uP [Edit](#) Desktop **Mobile**

- Overview
- Critical Path Explorer
- Medium priority (1)**
- Avoid landing page redire...
- Low priority (3)**
- Leverage browser caching
- Minify JavaScript
- Specify a Vary: Accept-...
- Experimental rules (2)**
- Already done! (26)**

Overview

The page uMobile got an overall PageSpeed Score of **95** (out of 100). [Learn more](#)

This PageSpeed Report is generated for this page as it appears on mobile devices. To get a PageSpeed Report for desktop clients, view the [desktop report](#) instead.

Suggestion Summary

Click on the rule names to see suggestions for improvement.

- **High priority.** These suggestions represent the largest potential performance wins for the least development effort. However, there are no high priority suggestions for this site. Good job!
- **Medium priority.** These suggestions may represent smaller wins or much more work to implement. You should address this item next: [Avoid landing page redirects](#)
- **Low priority.** These suggestions represent the smallest wins. You should only be concerned with these items after you've handled the higher priority items. [Leverage browser caching](#), [Minify JavaScript](#), [Specify a Vary: Accept-encoding header](#)
- **Experimental rules.** These suggestions are experimental, but do not affect the overall PageSpeed score. Consider these items as points to explore, but your mileage might vary: [Eliminate unnecessary reflows](#), [Specify a viewport for mobile browsers](#)
- **Already done!** There are no suggestions for these rules, since this page already follows these best practices. Good job!

Εικόνα 7.3: Καταταξη σύμφωνα με το pagespeed insights για φορητές συσκευές

7.2.2 Συμβατότητα με κανόνες και πρότυπα

Το powermapper.com προσφέρει έλεγχο συμβατότητας με διάφορους κανόνες και πρότυπα ευχρηστίας και προσβασιμότητας καθώς και προστασίας προσωπικών δεδομένων και σχεδίασης ιστοσελίδων.

Σύμφωνα με το powermapper.com δεν είχαμε κανένα απολύτως πρόβλημα και η πύλη μας είναι συμβατή με όλα τα προγράμματα πλοήγησης σε Η/Υ και smartphones.

This tab shows pages that exhibit browser-specific behavior, or trigger browser bugs.

Browser Version	Internet Explorer					Firefox			Safari	Opera	Chrome		iPhone			Android		BlackBerry		
	6.0	7.0	8.0	9.0	10.0	≤ 3.6	≤ 15.0	16.0	≤ 6.0	≤ 12.0	≤ 21.0	22.0	4.0	5.0	6.0	3.0	4.0	5.0	6.0	7.0
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Major Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minor Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Key

- Missing content or functionality on some browsers
- Major layout or performance problems on some browsers
- Minor layout or performance problems on some browsers

Priority URL / Description Guideline All Line

0 issues on 0 pages

Εικόνα 7.4: έλεγχος συμβατότητας με προγράμματα πλοήγησης

Στην διαδικτυακή μας πύλη εντοπίστηκε πρόβλημα συμβατότητας με τους κανόνες προστασίας προσωπικών δεδομένων της Ευρωπαϊκής “Ένωσης καθώς δεν υπήρχε σύνδεσμος με όνομα «προστασία προσωπικών δεδομένων» που να εξηγεί πως χρησιμοποιούνται τα cookies και πως μπορούν να απενεργοποιηθούν. Προχωρήσαμε στην προσθήκη του συνδέσμου που εξηγεί το τρόπο χρήσης των cookie και τον λόγο μη δυνατότητας απενεργοποίησης τους εξαιτίας του CAS.

Εντοπιστήκαν, όπως ήταν αναμενόμενο. λίγα προβλήματα συμμόρφωσης μας με τους οδηγούς προσβασιμότητας του W3C στο επίπεδο 2 και 3 που δεν ήταν εύκολο να ξεπεραστούν μέχρι την πιλοτική λειτουργία της πύλης.

Τέλος, εντοπίστηκαν μικρά θέματα συμβατότητας με το HTML5 και λήφθησαν τα απαραίτητα μέτρα.

7.3 Διερευνητική μέθοδος συμπλήρωσης ερωτηματολογίων

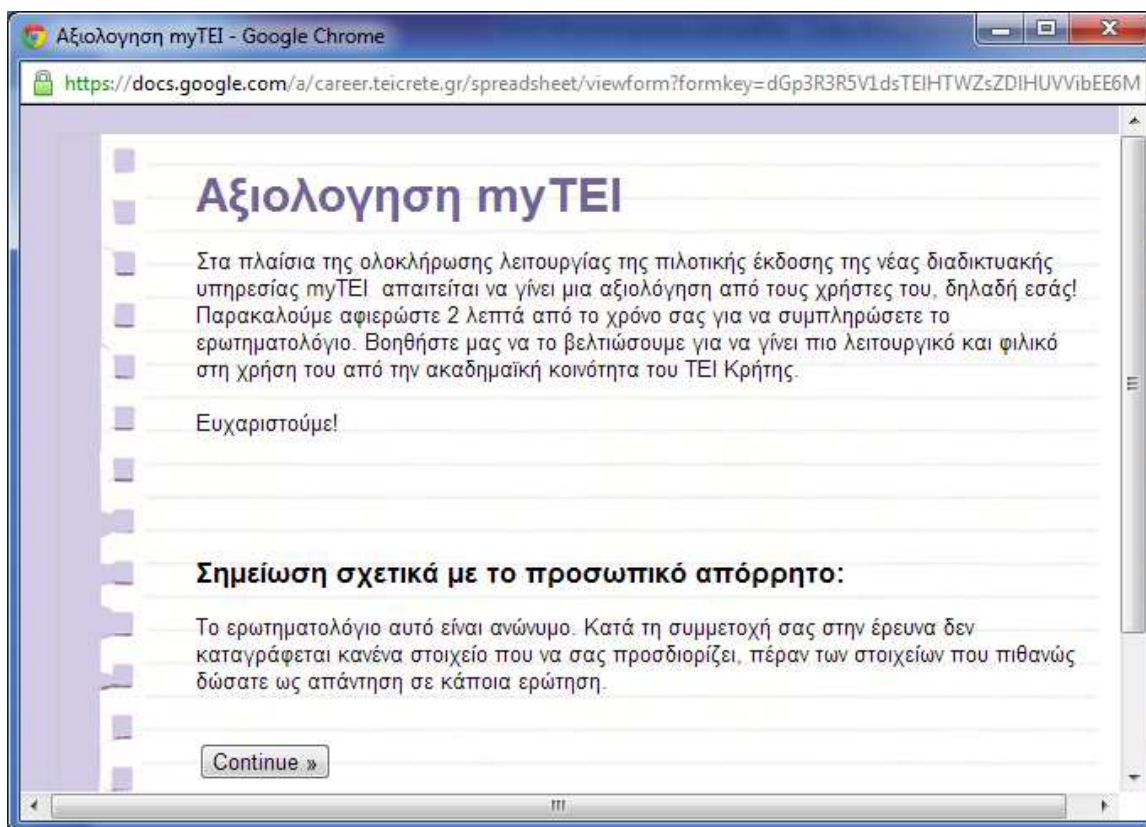
Κατά την πιλοτική λειτουργία της πύλης μας ήταν φυσικό να προχωρήσουμε σε αξιολόγηση της πύλης από τους ίδιους τους χρήστες της. Ο πιο διαδεδομένος τρόπος αξιολόγησης από ένα μεγάλο αριθμό χρηστών είναι η συμπλήρωση ερωτηματολογίου.

Σύμφωνα με τη βιβλιογραφία [69], έπρεπε να προχωρήσουμε σε διαχωρισμό των χρηστών σε focus groups. Στη πύλη μας αυτός ο διαχωρισμός είναι ήδη προκαθορισμένος αφού έχουμε τρεις μεγάλες κατηγορίες: καθηγητές, διοικητικό προσωπικό και φοιτητές. Προχωρήσαμε επίσης σε μια ειδική κατηγορία χρηστών, τους «εμπειρογνώμονες», που αποτελείται από τους συναδέλφους του τμήματος της πληροφορικής και τους καθηγητές του τμήματος εφαρμοσμένης πληροφορικής και πολυμέσων.

Το ερωτηματολόγιο που συντάξαμε αποτελείται από 20 ερωτήματα για αξιολόγηση σε 4 κατηγορίες:

1. Προσφερόμενης βοήθειας στο myTEI
2. Λειτουργικότητας
3. Ευκολίας στη χρήση και σχεδιαστικό-αισθητικό αποτέλεσμα
4. Γενικότερης αξιολόγηση του συστήματος

Το ερωτηματολόγιο αναρτήθηκε στο διαδίκτυο [71] και σε αυτό είχε ο χρήστης πρόσβαση μετά από την σύνδεση του με το σύστημα ώστε να καταγράφεται ο ρόλος του, άρα και το focus group που ανήκει.



Εικόνα 7.5: online ερωτηματολόγιο αξιολόγησης διαδικτυακής πύλης

Στο περιορισμένο αριθμό συμπληρωμένων ερωτηματολογίων μέχρι στιγμής, τα αποτελέσματα είναι πολύ θετικά.. Το αξιοσημείωτο είναι ότι δεν υπάρχει σημαντική διακύμανση τόσο μεταξύ των απαντήσεων των διαφορετικών ομάδων χρηστών όσο και των «εμπειρογνομώνων». Οι τελευταίοι ήταν πιο συγκεκριμένοι και αναλυτικοί στις προσθήκες και αλλαγές που θα θελαν να δουν στην τελική μορφή της πύλης. Έτσι παραθέτουμε τα αποτελέσματα παρακάτω ενοποιημένα με τη σημείωση ότι η κλίμακα βαθμολόγησης είναι αναλογική (1...5), και όσο μεγαλύτερο βαθμό δίνει ο ερωτώμενος τόσο και θετικότερη η αξιολόγηση.

1. On line Βοήθεια στο myTEI:

- Αξιολόγηση Εισαγωγικών Κείμενων για μη συνδεδεμένο χρήστη: 4,4
- Αξιολόγηση επεξηγηματικών κείμενων μετά από συνδεση: 4,4

2. Λειτουργικότητα πύλης:

- βαθμός ικανοποίησης από τις προσφερόμενες υπηρεσίες: 4,5

- διαδικασία σύνδεσης με το λογαριασμό πόσο κατανοητή και γρήγορη σας φάνηκε: 4,7
- Η διαδικασία πρόσβασης και προσθήκης ή χρήσης μιας υπηρεσίας πόσο κατανοητή και γρήγορη σας φάνηκε: 4,5
- πόσο χρήσιμη σας φαίνεται τελικά η πύλη: 4,6



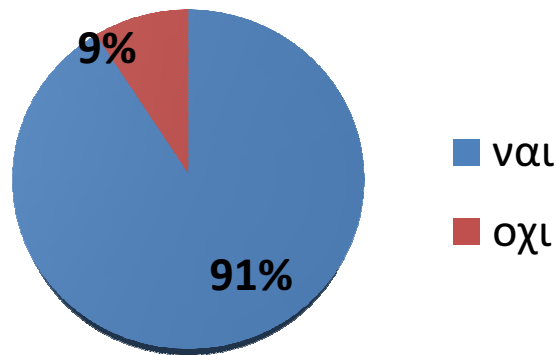
Εικόνα 7.6: πόσο αξιόπιστη σας φαίνεται η πύλη, (χωρίς προβλήματα, λάθη ή καθυστερήσεις)

3. ευκολία στη χρήση-αισθητικό αποτέλεσμα

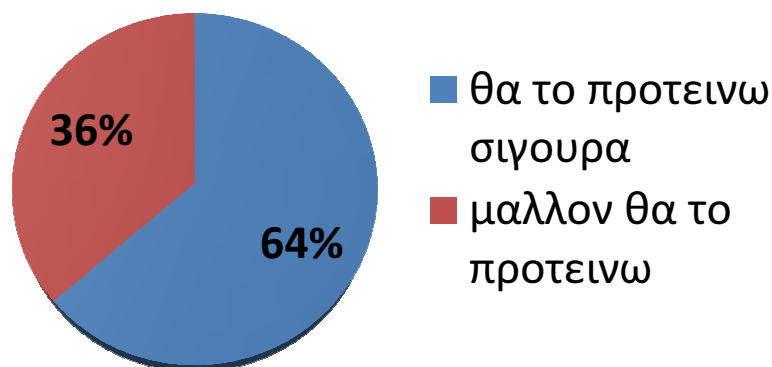
- πόσο γρήγορα μάθατε να το χρησιμοποιείτε: 4,6
- ποσό φιλικό-εύκολο είναι στη χρήση του: 4,5
- Αισθητικό αποτέλεσμα: 4,2

4. Τελική αξιολόγηση

- Γενικότερα τι τελική βαθμολογία θα δίνετε στο σύστημα: 4,3



Εικόνα 7.7: σε σύγκριση με τη κεντρική σελίδα του ΤΕΙ Κρήτης το προτιμάτε σαν αρχική σας σελίδα σε κάθε επίσκεψη στο ίδρυμα μας;



Εικόνα 7.8: πόσο πιθανόν είναι να προτείνετε τη πύλη σε κάποιον που γνωρίζετε:

7.4 Σύνοψη

Στο κεφάλαιο αυτό περιγράψαμε την μεθοδολογία αξιολόγησης της εκπαιδευτικής πύλης. Στην ενότητα 7.2 έγινε περιγραφή των αποτελεσμάτων της αξιολόγησης συμβατότητας της εφαρμογής με αναγνωρισμένα πρότυπα και κανόνες. Η ενότητα 7.3 ασχολήθηκε με την μέθοδο συμπλήρωσης ερωτηματολογίων και τα αποτελέσματα της αξιολόγησης από τους ίδιους τους χρήστες της πύλης. Τέλος, στην παρούσα ενότητα παρουσιάζεται η σύνοψη του κεφαλαίου αυτού.

Με τα παραπάνω τεκμηριώσαμε τον επιμέρους στόχο #9 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Σε αυτό το σημείο ολοκληρώθηκε η εξέταση των θεμάτων

και των στόχων της παρούσας διατριβής. Στο επόμενο κεφάλαιο θα γίνει μια σύνοψη της διπλωματικής εργασίας.

Κεφάλαιο 8

Σύνοψη

8.1 Εισαγωγή

Στο κεφάλαιο αυτό γίνεται μια επισκόπηση της παρούσας διατριβής. Στην ενότητα 8.2 παρατίθεται η συμβολή της διπλωματικής εργασίας και στην ενότητα 8.3 περιγράφονται οι μελλοντικές επεκτάσεις της.

8.2 Συμβολή

Στο εισαγωγικό κεφάλαιο παρουσιάσαμε σε γενικές γραμμές τα πιο σημαντικά θέματα που σχετίζονται με την εκπόνηση της διατριβής αυτής. Στην ενότητα 1.1 δόθηκε μια αδρή περιγραφή του προβλήματος. Στην ενότητα 1.2 παρουσιάστηκε ο οργανισμός για τον οποίο σχεδιάστηκε η εφαρμογή. Τα κύρια και δευτερεύοντα κίνητρα του συντάκτη περιγράφηκαν στην ενότητα 1.3. Ο κύριος σκοπός και οι επιμέρους στόχοι της διατριβής αναλύθηκαν στην ενότητα 1.4 ενώ η

επόμενη ενότητα εξήγησε την ανάγκη που οδήγησε στην διατριβή αυτή. Η μεθοδολογική προσέγγιση που ακολουθήθηκε παρουσιάστηκε με συντομία στην ενότητα 1.6 ενώ τα οφέλη της διατριβής συνοψίζονται στην ενότητα 1.7. Όπως σε κάθε έργο, αντιμετωπίσαμε μια σειρά από περιορισμούς που προέκυψαν από πίεση χρόνου και διοικητικές αγκυλώσεις. Οι περιορισμοί αυτοί εξηγούνται στην ενότητα 1.8.

Στο δεύτερο κεφάλαιο προχωρήσαμε στην βιβλιογραφική επισκόπηση σε θέματα ανάλυσης, σχεδίασης και αξιολόγησης διαδικτυακών πυλών για να τεκμηριώσουμε τον επιμέρους στόχο #1, όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Πιο συγκεκριμένα, στην ενότητα 2.2 έγινε παρουσίαση μιας διαδικτυακής πύλης (portal) εστιάζοντας στη λειτουργικότητα και τα κέρδη που έχει ο χρήστης με την εξατομίκευση του πληροφοριακού υλικού και των υπηρεσιών. Στην ενότητα 2.3 έγινε περιγραφή του portlet, του βασικού αντικείμενου μιας πύλης, καθώς κάθε ιστοσελίδα μιας πύλης αποτελείται από ένα ή περισσότερα portlets. Η ενότητα 2.4 αναφέρθηκε στα τεχνολογικά πρότυπα που έχουν οριστεί για την ανάπτυξη των portlets και προσφέρουν συμβατότητα στη χρήση τους σε διαφορετικές πύλες. Εστίασαμε στη αρχιτεκτονική μιας διαδικτυακής πύλης στην ενότητα 2.5 ενώ οι ενότητες 2.6 και 2.7 έγινε περιγραφή των κύριων μεθόδων πιστοποίησης και εξουσιοδότησης ενός χρήστη. Η ενότητα 2.8 αναφέρθηκε στο HTTPS, το πρωτόκολλο ασφαλούς μεταφοράς δεδομένων μεταξύ εξυπηρετητών ιστοσελίδων, η ενότητα 2.9 στη κρυπτογράφηση δεδομένων με σκοπό την προστασία από υποκλοπή και η ενότητα 2.10 περιγράφει τη μέθοδο επικύρωσης ταυτότητας του ίδιου του εξυπηρετητή ιστοσελίδων με τη χρήση πιστοποιητικών. Η ενότητα 2.11 περιέγραψε τις οδηγίες προσβασιμότητας Web Content Accessibility Guidelines (WAI/WCAG) του W3C ώστε να αποφεύγεται ο αποκλεισμός ειδικών ομάδων πολιτών (AMEA). Στην ενότητα 2.12 περιγράψαμε τις μεθόδους αξιολόγησης ιστοχώρων.

Στο τρίτο κεφάλαιο έγινε καταγραφή της παρούσας κατάστασης στο ΤΕΙ Κρήτης καθώς ο ορισμός των υπηρεσιών και των χρηστών που θα υποστηρίξει η διαδικτυακή μας πύλη και περιγράφηκε η ζητούμενη λειτουργικότητα της ως τεκμηρίωση του επιμέρους στόχου #2 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Έτσι, στην ενότητα 3.2 έγινε μια εισαγωγή στην εκπαιδευτική δομή του ιδρύματος. Η ενότητα 3.3 περιέγραψε την τρέχουσα κατάσταση υποστήριξης χρηστών στο ίδρυμα και η ενότητα 3.4 εξήγησε την δομή των υπολοίπων τμημάτων και υπηρεσιών του ΤΕΙ Κρήτης. Στην ενότητα 3.5 ορίστηκαν οι υπηρεσίες και ιστοσελίδες που θα καλύπτει το νέο σύστημα ενώ στην ενότητα 3.6 δηλώθηκαν οι κατηγορίες χρηστών που θα αναγνωρίζει το νέο σύστημα. Η ενότητα 3.7 περιέγραψε την απαιτούμενη λειτουργικότητα της πύλης.

Στο τέταρτο κεφάλαιο αυτό παρουσιάστηκε η μεθοδολογία και τα εργαλεία ανάπτυξης και οι επικουρικές τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εκπαιδευτικής πύλης ως τεκμηρίωση του επιμέρους στόχου #3 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής. Έτσι, η ενότητα 4.2 αναφέρεται στη μεθοδολογία ICONIX και η ενότητα 4.3 αναφέρεται στην πλατφόρμα ανοιχτού κώδικα uPortal που πάνω σε αυτό βασιστήκαμε για την δημιουργία της εκπαιδευτικής πύλης μας. Η ενότητα 4.4 περιέγραψε το κεντρικό σύστημα ταυτοποίησης χρηστών CAS, ενώ η ενότητα 4.5 περιέγραψε την εφαρμογή ανοιχτού κώδικα uMobile, υπεύθυνη για την υποστήριξη φορητών συσκευών όπως smartphones. Η ενότητα 4.6 αναφέρθηκε στο Spring Framework και την ιδιαίτερη φιλοσοφία του που χρησιμοποιήθηκε για την ανάπτυξη των “μικροεφαρμογών” μας (portlets). Η ενότητα 4.7 περιέγραψε το δομικό μοντέλο αρχιτεκτονικής εφαρμογών MVC για την ανάπτυξη διαδικτυακών εφαρμογών. Η ενότητα 4.8 αναφέρθηκε στη βάση δεδομένων που επιλέξαμε για την λειτουργία της πύλης. Οι ενότητες 4.9 και 4.10 περιέγραφαν τις τεχνολογίες υλοποίησης των αισθητικής (εμφάνισης) της πύλης XSL & Fluid για τη δημιουργία των θεμάτων (templates και stylesheets). Στις ενότητες 4.11 και 4.12 έγινε αναφορά στα επικουρικά εργαλεία ανάπτυξης Maven & Ant για την διαχείριση και συντήρηση του συνολικού έργου αλλά και την ενσωμάτωση σε αυτό των portlets που δημιουργήσαμε.

Στο πέμπτο κεφάλαιο αυτό έγινε τεκμηρίωση των επιμέρους στόχων #3, #4, #5 και #6 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής με την παρουσίαση της επίλυσης των θεμάτων για τον σχεδιασμό του συστήματος και την ανάπτυξη του μηχανισμού πρόσβασης καθώς και η ενσωμάτωση των υπηρεσιών. Η δομή αυτού του κεφαλαίου είναι ως εξής: στην ενότητα 5.2 αναλύσαμε το πώς επιλέχτηκε η πλατφόρμα uPortal ως βάση για την ανάπτυξη της εκπαιδευτικής μας πύλης. Η ενότητα 5.3 εξήγησε πως οργανώθηκαν οι ρόλοι-κατηγορίες των χρηστών του συστήματος και πως επιτυγχάνεται η ταυτοποίησή τους. Η ενότητα 5.4 αναφέρθηκε στην κατηγοριοποίηση των εφαρμογών/υπηρεσιών της πύλης. Στην ενότητα 5.5 έγινε περιγραφή των επιπλέον μέτρων πάρθηκαν για την ασφάλεια της τόσο σε επίπεδο προστασίας δεδομένων όσο και ασφάλειας συναλλαγών. Η ενότητα 5.6 ήταν αφιερωμένη στο μηχανισμό πιστοποίησης χρηστών. Η ενότητα 5.7 εξέτασε τις τεχνικές σχεδίασης για πρόσβαση σε άτομα με ειδικές ανάγκες και η ενότητα 5.8 αναφέρθηκε στη πρόσβαση μέσω έξυπνων συσκευών όπως τηλεφώνων.

Στο έκτο κεφάλαιο περιγράψαμε την μεθοδολογία ανάπτυξης όλων των portlets που υλοποιήθηκαν εξολοκλήρου από μας καθώς και η ενσωμάτωση των ήδη υφισταμένων

διαδικτυακών υπηρεσιών για τις ανάγκες της εκπαιδευτικής πύλης για τις ανάγκες της κοινότητας του ΤΕΙ Κρήτης.

Ξεκινήσαμε στην ενότητα 6.2 με τον κατάλογο των διαθέσιμων portlets. Στην ενότητα 6.3 περιγράψαμε τις προδιαγραφές που έπρεπε να έχουν τα καινούργια portlets. Στην ενότητα 6.4 αναλύσαμε την σχεδίαση που ακολουθήσαμε για να καλύψουμε διάφορα θέματα είτε συμβατότητας με την διαδικτυακή πύλη και τα υπόλοιπα portlets, είτε υποστήριξη μεταγλώττισης και δυνατότητα φιλικής παραμετροποίησης. Ταυτόχρονα είδαμε και άλλα σχεδιαστικά ζητήματα για την παρεχόμενη γενική λειτουργικότητα τους. Στην ενότητα 6.5 δώσαμε ειδική έμφαση στη σχεδίαση των portlets ανάγνωσης πολλαπλών πηγών δεδομένων. Επίσης στην ενότητα 6.6 είδαμε με ποιο τρόπο ενσωματώσαμε διαδικτυακές υπηρεσίες του ιδρύματος στη εκπαιδευτική πύλη μας

Με τα παραπάνω τεκμηριώσαμε τους επιμέρους στόχους #7 και #8 όπως αυτοί αναφέρονται στην εισαγωγή της παρούσας διατριβής.

Στο έβδομο κεφάλαιο αυτό περιγράψαμε την μεθοδολογία αξιολόγησης της εκπαιδευτικής πύλης. Στην ενότητα 7.2 έγινε περιγραφή των αποτελεσμάτων της αξιολόγησης συμβατότητας της εφαρμογής με αναγνωρισμένα πρότυπα και κανόνες. Η ενότητα 7.3 ασχολήθηκε με την μέθοδο συμπλήρωσης ερωτηματολογίων από τους ίδιους τους χρήστες της πύλης. Με τα παραπάνω, τεκμηριώσαμε τον επιμέρους στόχο #9 όπως αυτός αναφέρεται στην εισαγωγή της παρούσας διατριβής, ολοκληρώνοντας την εξέταση των θεμάτων όπως αυτοί αναφέρονται στην εισαγωγή της παρούσας διατριβής.

8.2 Μελλοντικές επεκτάσεις

Η πιλοτική εφαρμογή της πύλης και η αποδοχή της από την ακαδημαϊκή κοινότητα του ΤΕΙ Κρήτης έχουν δρομολογήσει τα επόμενα βήματα:

Η απόφαση για την δημιουργία του νέου κεντρικού ιστοχώρου του ΤΕΙ Κρήτης με την ενοποίηση όλων των ιστοσελίδων των σχολών και τμημάτων του ιδρύματος θα γίνει με την ενσωματωμένη λειτουργικότητα του myTEI. Όλοι οι χρήστες θα έχουν καθορισμένα δικαιώματα και πρόσβαση στις υπηρεσίες ανάλογα το ρολό τους που θα αναγνωρίζεται κατά την σύνδεση τους με το CAS.

Για το λόγο αυτό θα γίνει το συντομότερο δυνατόν το ξεκαθάρισμα των ιδιοτήτων και των διπλοεγγραφών των χρηστών του κεντρικού LDAP, ώστε να προχωρήσουμε σε ακόμα περισσότερους συνδυασμούς ρόλων και να φτάσουμε σε επίπεδο του τμήματος που ανήκει ο χρήστης είτε είναι καθηγητής είτε είναι φοιτητής ή υπάλληλος του ιδρύματος.

Έχει παρθεί η απόφαση όλες οι νέες διαδικτυακές υπηρεσίες του ιδρύματος να έχουν υποχρεωτικά σύνδεση με το CAS, ώστε να γίνεται η ενσωμάτωση τους στη διαδικτυακή μας πύλη. Το ίδιο ισχύει και για όλες τις υπάρχουσες υπηρεσίες.

Ενδεικτικά να αναφέρουμε, ότι το έργο που έχει ξεκινήσει στο χώρο του ΤΕΙ και είναι η πλειοψηφία των αρχείων που βρίσκονται σε κοινόχρηστους δικτυακούς καταλόγους, να μεταφερθούν σε διαδικτυακό web περιβάλλον, θα υποστηρίζει πλήρως το CAS και θα ενσωματωθεί σαν υπηρεσία στη πύλη μας.

Βιβλιογραφία

- [01]. A Basman, C. Lewis, C. Clark, "To Inclusive Design Through Contextually Extended IoC", SPLASH'11 Companion, October 22–27, 2011
- [02]. F. Bellas, D. Fernández, A. Muiño, "A Flexible Framework for Engineering My Portals", WWW 2004, May 17–22, 2004
- [03]. R. J. Boncella, "Web Security for E-Commerce", Communications of the Association for Information Systems: Vol. 4
- [04]. D. A. Bradbard, C. Peters, "Web Accessibility: A Tutorial for University Faculty", Communications of the Association for Information Systems: Vol. 22
- [05]. D. Chohan, A Akram, R. Allan, "Grid Middleware Portal Infrastructure",
- [06]. O. Díaz, J. Iturrioz, A. Irastorza, "Improving Portlet Interoperability Through Deep Annotation", WWW 2005, May 10-14, 2005
- [07]. O. Diaz, J. J. Rodriguez, "Portlet Syndication: Raising Variability Concerns", ACM Transactions on Internet Technology, Vol. 5, No. 4, November 2005
- [08]. J. Gant, D. Gant, "Web Portals and Their Role in E-Government", AMCIS 2001 Proceedings
- [09]. B. Gibson, "Enabling an Accessible Web 2.0", W4A2007 - Keynote
- [10]. Herzberg, A. Jbara, "Security and Identification Indicators for Browsers against Spoofing and Phishing Attacks", ACM Trans. Internet Technol., 8, 4, Article 16 (September 2008)
- [11]. R. Holz, L. Braun, N. Kammenhuber, G. Carle, "The SSL Landscape – A Thorough Analysis of the X.509 PKI Using Active and Passive Measurements", IMC'11, November 2–4, 2011
- [12]. W. Jie, J. Arshad, R. Sinnott, P. Townend, Z. Lei, "A Review of Grid Authentication and Authorization Technologies and Support for Federated Access Control", ACM Computing Surveys, Vol. 43, No. 2, Article 12, Publication date: January 2011

- [13]. N. Kumar, D. Sosale, A. Rathi, S. N. Konuganti "Enabling the Adoption of Aspects - Testing Aspects: A Risk Model, Fault Model and Patterns", AOSD'09, March 2-6, 2009
- [14]. C. Li, P. Bohannon, H. F. Korth, P.P.S. Narayan, "Composing XSL Transformations with XML Publishing Views", SIGMOD 2003, June 9-12, 2003
- [15]. S. F. Morse, C. L. Anderson, "Introducing Application Design and Software Engineering Principles in Introductory CS COURSES: MVC Java Application Framework", JCSC 20, 1 (October 2004)
- [16]. M. Mustica, G. Moraru, C. Grigorescu, D. Mi. Kristaly, "Portal Type Applications for Web Campus"
- [17]. A. Sarin, "Portlets in Action" ©2012 by Manning Publications Co. ISBN: 9781935182542
- [18]. S. Shukla, A. Deshpande, "Tutorial: LDAP Directory Services - Just Another Database Application?", ACM SIGMOD 2000
- [19]. R. O. Sinnott, J. Jiang, J. Watt, O. Ajayi, "Shibboleth-based Access to and Usage of Grid Resources" Grid Computing Conference 2006
- [20]. M. A. Smith, "Portals: Toward an Application Framework for Interoperability," Communications of the ACM October 2004/Vol. 47, No. 10
- [21]. Organization for the Advancement of Structured Information Standards (OASIS), Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 March 2005.
- [22]. D. D. Vecchio, V. Hazlewood, M. Humphrey, "Evaluating Grid Portal Security"
- [23]. C. Walls, "Spring in Action", 3th edition ©2011 by Manning Publications Co. ISBN 9781935182351
- [24]. "An Introduction to LDAP", ανακτήθηκε 1 Απριλίου 2012 από http://ldapman.org/articles/intro_to_ldap.html

- [25]. “Apache Ant”, ανακτήθηκε 1 Απριλίου 2012 από <http://ant.apache.org/>
- [26]. “Apache Maven”, ανακτήθηκε 1 Απριλίου 2012 από <http://maven.apache.org/>
- [27]. “Apache Tomcat”, ανακτήθηκε 1 Απριλίου 2012 από <http://tomcat.apache.org/>
- [28]. “Benefits of Portal Content Management” ανακτήθηκε 1 Απριλίου 2012 από <http://www.content-management-junction.com/portal-cm.html>
- [29]. “C.A.S.”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.jasig.org/cas>
- [30]. “Central Authentication Service”, ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Central_Authentication_Service
- [31]. “Constructing a POUR Website”, 1 Απριλίου 2012 από <http://webaim.org/articles/pour/>
- [32]. “Directory Enabled Network”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.wbemsolutions.com/tutorials/CIM/den.html>
- [33]. “Directory-Enabled Networking”, ανακτήθηκε 1 Απριλίου 2012 από http://docwiki.cisco.com/wiki/Directory-Enabled_Networking
- [34]. “Directory Enabled Networks - The DEN Value Proposition”, ανακτήθηκε 1 Απριλίου 2012 από <http://technet.microsoft.com/en-us/library/cc750196.aspx>
- [35]. “Directory Enabled Networks”, ανακτήθηκε 1 Απριλίου 2012 από http://www.e.kth.se/~e95_hja/tp/reports/DEN_report_A.pdf
- [36]. “Fluid”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.fluidproject.org/>
- [37]. “HSQLDB” ανακτήθηκε 1 Απριλίου 2012 από <http://hsqldb.org/>
- [38]. “Hypertext Transfer Protocol Secure”, ανακτήθηκε 1 Απριλίου 2012 από <http://en.wikipedia.org/wiki/Https>

- [39]. "Installing Apache Tomcat", ανακτήθηκε 1 Απριλίου 2012 από <https://wiki.jasig.org/display/UPM40/Installing+Tomcat>
- [40]. "Internetworking Technology Handbook", ανακτήθηκε 1 Απριλίου 2012 από http://docwiki.cisco.com/wiki/Internetworking_Technology_Handbook
- [41]. "Java SE at a Glance", ανακτήθηκε 1 Απριλίου 2012 από <http://www.oracle.com/technetwork/java/javase/overview/index.html>
- [42]. "Java Portlet Specifications", ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Java_Portlet_Specification
- [43]. "Lightweight Directory Access Protocol", ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol
- [44]. "MACE-Dir Working Group Space", ανακτήθηκε 1 Απριλίου 2012 από <https://spaces.internet2.edu/display/macedir/MACE-Dir+Working+Group+Space>
- [45]. "Middleware Architecture Committee for Education", ανακτήθηκε 1 Απριλίου 2012 από <http://middleware.internet2.edu/dir/>
- [46]. "Openldap", ανακτήθηκε 1 Απριλίου 2012 από <http://www.openldap.org/>
- [47]. "OpenSSL: The Open Source toolkit for SSL/TLS", ανακτήθηκε 1 Απριλίου 2012 από <http://www.openssl.org/>
- [48]. "Shibboleth", ανακτήθηκε 1 Απριλίου 2012 από [http://en.wikipedia.org/wiki/Shibboleth_\(Internet2\)](http://en.wikipedia.org/wiki/Shibboleth_(Internet2))
- [49]. "Shibboleth", ανακτήθηκε 1 Απριλίου 2012 από <http://www.shibboleth.net/>
- [50]. "SOA and Web Services", ανακτήθηκε 1 Απριλίου 2012 από <http://www.roseindia.net/webservices/>
- [51]. "SSL Configuration HOW-TO", ανακτήθηκε 1 Απριλίου 2012 από <http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html>

- [52]. “The Extensible Stylesheet Language Family”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.w3.org/Style/XSL/>
- [53]. “The Web Content Accessibility Guidelines” ανακτήθηκε 1 Απριλίου 2012 από <http://webaim.org/standards/wcag/>
- [54]. “uMobile”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.jasig.org/umobile>
- [55]. “uPortal 4.0 Manual - Jasig Wiki”, ανακτήθηκε 1 Απριλίου 2012 από <https://wiki.jasig.org/display/UPM40/Home>
- [56]. “Web Accessibility Initiative (WAI)”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.w3.org/WAI/>
- [57]. “Web Content Accessibility Guidelines”, ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Web_Content_Accessibility_Guidelines
- [58]. “Web Content Accessibility Guidelines (WCAG) 2.0” ανακτήθηκε 1 Απριλίου 2012 από <http://www.w3.org/TR/WCAG/>
- [59]. “Web Portal”, ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Web_portal
- [60]. “Web Service”, ανακτήθηκε 1 Απριλίου 2012 από http://en.wikipedia.org/wiki/Web_service
- [61]. “Web Services Architecture”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.w3.org/TR/ws-arch/>
- [62]. “What is a Content Management Portal and What Types of Other Web Portal Are There?”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.contentmanager.eu.com/portal.htm>
- [63]. “What is a Portal?” ανακτήθηκε 1 Απριλίου 2012 από <http://www.liferay.com/products/what-is-a-portal/web-platform>
- [64]. “What is Spring?”, ανακτήθηκε 1 Απριλίου 2012 από <http://www.springsource.org/>

- [65]. Yahoo! Query Language ανακτήθηκε 1 Απριλίου 2012 από <http://developer.yahoo.com/yql/console/>
- [66]. Open Source Software ανακτήθηκε 1 Δεκεμβρίου 2012 από http://en.wikipedia.org/wiki/Open_source
- [67]. Ν. Αδούρης, “Επικοινωνία Ανθρώπου-Υπολογιστή” Τόμος Α, ©2003 Ελληνικό Ανοικτό Πανεπιστήμιο. ISBN: 960-538-219-9
- [68]. J. Nielsen, R.L. Mack, Usability Inspection Methods ©1994 Wiley
- [69]. J. Rubin, D Chisnell, “Handbook of Usability Testing”, second edition, ©2008 Wiley ISBN: 978-0-470-18548-3
- [70]. U.S. Department of Health and Human Services, “Research-Based Web Design & Usability Guidelines”, ISBN 0-16-076270-7
- [71]. online ερωτηματολόγιο αξιολογήσης διαδικτυακής πύλης, ανακτήθηκε 2 Ιανουαρίου 2013 από <https://docs.google.com/a/career.teicrete.gr/spreadsheet/viewform?formkey=dGp3R3R5V1dsTEIHTWZsZDIHUVVibEE6MQ>
- [72]. D. Rosenber, M. Stephens, “Use Case Driven Object Modeling with UML”, ©2007, Apress ISBN: 978-1-59059-774-3
- [73]. The ICONIX Process, ανακτήθηκε 2 Ιανουαρίου 2013 από <http://iconixprocess.com/>
- [74]. ICONIX Software Engineering, ανακτήθηκε 2 Ιανουαρίου 2013 από <http://www.iconixsw.com/>
- [75]. ICONIX, ανακτήθηκε 2 Ιανουαρίου 2013 από, <http://en.wikipedia.org/wiki/ICONIX>

Παράρτημα Α

Οδηγίες Πρόσβασης

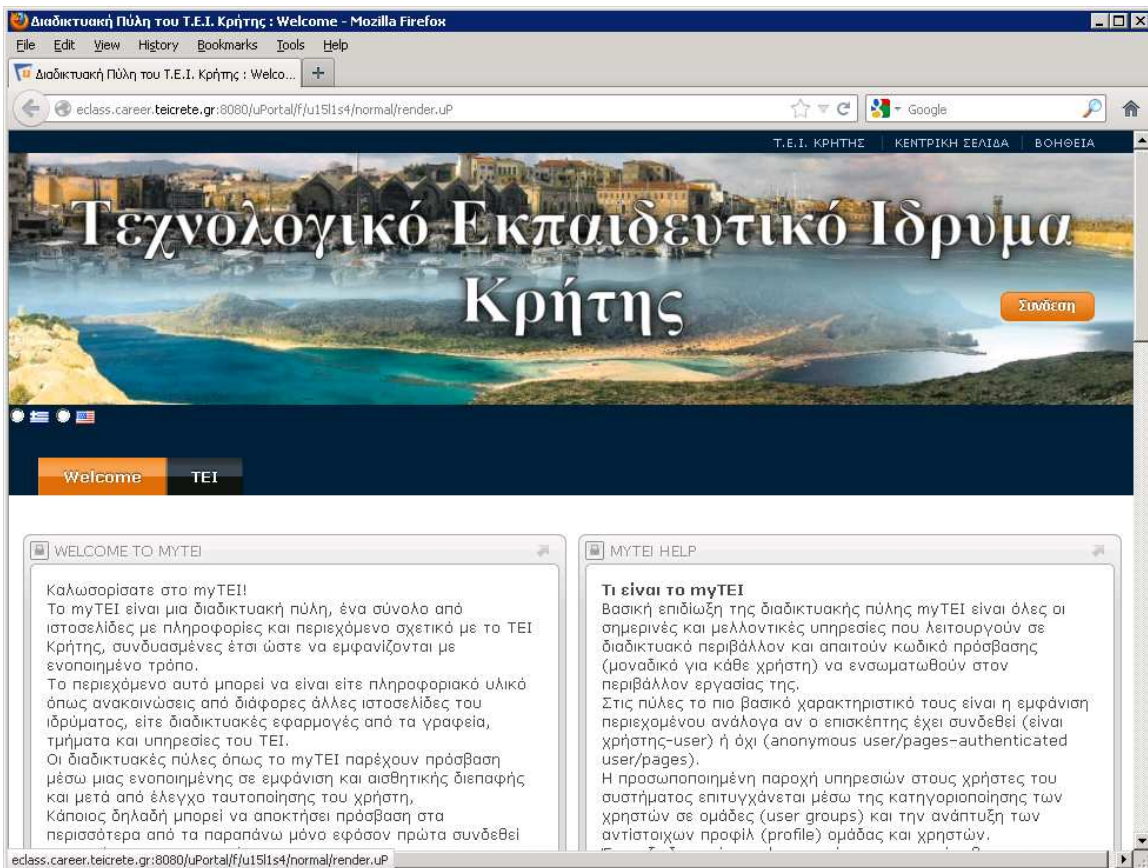
A.1 Πρόσβαση στην εκπαιδευτική πύλη

Για τις ανάγκες της μεταπτυχιακής διατριβής έχει δημιουργηθεί ένας ιστοχώρος στην διεύθυνση <http://eclass.career.teicrete.gr:8080/uPortal> για την φιλοξενία της πιλοτικής έκδοσης της εκπαιδευτικής πύλης.

Διαπιστευτήρια διαχειριστή πύλης είναι:

Username: admin

Password: admin



Εικόνα Α.1: Αρχική σελίδα εκπαιδευτικής πύλης

A.2 Πρόσβαση στο υλικό της μεταπτυχιακής εργασίας

Στη διεύθυνση <http://eclass.career.teicrete.gr/thesis> είναι προσβάσιμο όλο το υπόλοιπο συνοδευτικό υλικό όπως:

- source code (portal & portlets),
- binaries για εγκατάσταση του tomcat με εγκατεστημένη την πύλη,
- JavaSDK και υπόλοιπων απαραίτητων εργαλείων ανάπτυξης
- Αναλυτικές οδηγίες πρόσβασης

Προχωρήσαμε σε μια τέτοια λύση αφού το συνολικό μέγεθος ξεπερνά τα 100 MB και ήταν αδύνατο να μεταφορτωθούν μαζί με την παρούσα μεταπτυχιακή διατριβή.

Παράρτημα Β

Εγκατάσταση

B.1 Απαιτήσεις

Η πλατφόρμα uPortal που επιλέχτηκε για την υλοποίηση της διαδικτυακής πύλης μας έχει συγκεκριμένες απαιτήσεις τόσο σε hardware όσο κυρίως στο εγκατεστημένο λογισμικό στο μηχάνημα εγκατάστασης:[55]

B.1.2 Απαιτήσεις σε hardware:

- GB ελευθέρου δίσκου

- GB RAM
- Κάρτα δικτύου τουλάχιστον 100Mbits

B.1.3 Απαιτήσεις σε εγκατεστημένο λογισμικό:

- Java SE 1.6.21+
- Maven 3.0.3+
- Ant 1.8.2+
- Servlet Container spec 2.5+
- Database

Από τα παραπάνω, είναι φανερό ότι το σύστημα βασίζεται σε εργαλεία και λογισμικό ελεύθερου λογισμικού όπως ήταν και ο αρχικός μας σκοπός και είναι ανεξάρτητου λειτουργικού συστήματος [55].

Η υλοποίηση της διαδικτυακής μας πύλης και η ανάπτυξη λογισμικού έγινε σε Η/Υ με λειτουργικό σύστημα Windows Server 2003.

B.2 Εγκατάσταση και παραμετροποίηση προαπαιτούμενου λογισμικού [55]

B.2.1 Java SE 1.6.21[41]

Μεταφόρτωση από <http://www.oracle.com/technetwork/java/javase/downloads/index.html> και εγκατάσταση από το setup αρχείο.

B.2.2 Maven 3.0.3+

Μεταφόρτωση από <http://maven.apache.org/download.html> και εξαγωγή αρχείων από το συμπίεσμένο αρχείο μορφής (Binary tar.gz)

B.2.3 Ant 1.8.2+

Μεταφόρτωση από <http://ant.apache.org/bindownload.cgi> και εξαγωγή αρχείων από το συμπίεσμένο αρχείο μορφής (Binary tar.gz)

Για λόγους εύκολης συντήρησης επιλέξαμε να εγκαταστήσουμε τα παραπάνω εργαλεία κάτω από ένα κοινό κατάλογο και επειδή όλα είναι εργαλεία με βάση τη Java ονομάσαμε το κοινό κατάλογο c:\java μέσα στον οποίο δημιουργήθηκαν οι υποκατάλογοι C:\Java\apache-ant-1.8.2, C:\Java\apache-maven-3.0.3, C:\Java\jdk1.6.0_29.

B.2.4 Tomcat 6+

Το επόμενο βήμα είναι να διαλέξουμε Servlet Container στο οποίο θα τρέχει το uPortal. Υπάρχουν διάφορες επιλογές άλλες ελεύθερες και ανοιχτού κώδικα όπως Apache Tomcat, Eclipse Jetty, Oracle Glassfish και άλλες που εμπορικές όπως IBM WebSphere Application Server. Προτιμήσαμε να στήσουμε την πιο διαδεδομένη πλατφόρμα και αυτή που προτείνει το uPortal και αυτή είναι ο Tomcat 6+ [27],[39]

Μεταφόρτωση από <http://tomcat.apache.org/download-60.cgi> και εξαγωγή αρχείων από το συμπίεσμένο αρχείο μορφής (Binary tar.gz) σε κατάλογο της επιλογής μας πχ C:\apache-tomcat-6.0.33

Για την εγκατάσταση του uPortal χρειάζεται να παραμετροποιητή το αρχείο ρυθμίσεων του tomcat κατάλληλα. Το αρχείο αυτό %TOMCAT_HOME\conf\server.xml μετά τις αλλαγές πρέπει να είναι όπως παρακάτω:

```
<?xml version='1.0' encoding='utf-8'?>
<Server port="8005" shutdown="SHUTDOWN">
  <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
  <Listener className="org.apache.catalina.core.JasperListener" />
  <Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener" />
  <Listener className="org.apache.catalina.mbeans.ServerLifecycleListener" />
  <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
  <GlobalNamingResources>
```



```

<Resource name="UserDatabase" auth="Container"
  type="org.apache.catalina.UserDatabase"
  description="User database that can be updated and saved"
  factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
  pathname="conf/tomcat-users.xml" />
</GlobalNamingResources>
<Service name="Catalina">
  <Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"      emptySessionPath="true"      compression="on"
    redirectPort="8443" />
  <Connector protocol="org.apache.coyote.http11.Http11Protocol"
    port="8443" maxThreads="200" keystoreType="pkcs12"
    scheme="https" secure="true" SSLEnabled="true"
    keystoreFile="c:\tomcat.p12" keystorePass="changeit"
    clientAuth="false" sslProtocol="TLS" />
  <Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
  <Engine name="Catalina" defaultHost="localhost">
    <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
      resourceName="UserDatabase"/>
    <Host name="localhost" appBase="webapps"
      unpackWARs="true" autoDeploy="true"
      xmlValidation="false" xmlNamespaceAware="false">
      </Host>
    </Engine>
  </Service>
</Server>

```

Επιπλέον, χρειάζεται μια αλλαγή σε ένα άλλο αρχείο ρυθμίσεων του tomcat %TOMCAT_HOME\conf\catalina.properties, το αρχείο μετά τις αλλαγές πρέπει να μοιάζει όπως παρακάτω:

```

package.access=sun.,org.apache.catalina,org.apache.coyote,org.apache.tomcat,org.apache.jasper,
sun.beans.
package.definition=sun.,java.,org.apache.catalina,org.apache.coyote,org.apache.tomcat,org.apach
e.jasper.
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${catalina.home}/lib,${catalina.ho
me}/lib/*.jar
server.loader=shared.loader=${catalina.base}/shared/classes,${catalina.base}/shared/lib/*.jar
tomcat.util.buf.StringCache.byte.enabled=true

```

Προκειμένου να τρέξει ο tomcat αρκεί να εκτελεστεί το batch αρχείο %TOMCAT_HOME/bin/startup.sh για linux/unix ή %TOMCAT_HOME\bin\startup.bat για Windows

B.2.5 Μεταβλητές Περιβάλλοντος

Στο περιβάλλον του συστήματος είναι απαραίτητο να υπάρχουν οι παρακάτω μεταβλητές με τις τιμές ανάλογες με το κατάλογο εγκατάστασης του κάθε προγράμματος, στη συγκεκριμένη εγκατάσταση.

```
ANT_HOME = C:\Java\apache-ant-1.8.2
```

```
JAVA_HOME = C:\Java\jdk1.6.0_29
```

```
M2 = C:\Java\apache-maven-3.0.3\bin
```

```
M2_HOME = C:\Java\apache-maven-3.0.3
```

```
TOMCAT_HOME = C:\apache-tomcat-6.0.33
```

Επιπλέον στη μεταβλητή PATH προσθέσαμε τις παρακάτω τιμές:

```
%JAVA_HOME%\bin; %M2;C:\java\apache-ant-1.8.2\bin;
```

B.2.6 Εγκατάσταση του uPortal

Από την διεύθυνση <http://www.jasig.org/uportal/download>, μπορεί κάποιος να μεταφορτώσει το πηγαίο κώδικα (Full-Source Version) του uPortal. Στη δική μας εγκατάσταση χρησιμοποιήθηκε η έκδοση 4.0.2 η οποία όπως έχει ήδη τονιστεί είναι μια έκδοση με αρκετά σοβαρά bugs που μας δυσκόλεψαν στην υλοποίηση. Μέχρι τώρα, που γράφεται αυτό το κείμενο παρότι έχει κυκλοφορήσει η έκδοση 4.0.4 κάποια σοβαρά θέματα ακόμα επιζητούν επίλυση.

Το αρχείο είναι σε μορφή (source tar.gz) και απαιτείται εξαγωγή αρχείων σε κατάλογο της επιλογής μας. Πχ. C:\uPortal-4.0.2

Σημειώσεις-Παρατηρήσεις

Σε περιβάλλον windows χρειάζεται να εγκατασταθεί επιπλέον για τις ανάγκες εξαγωγής αρχείων μορφής tar.gz ή zip ειδικό πρόγραμμα εξαγωγής. Εμείς εγκαταστήσαμε το winrar από τη διεύθυνση <http://www.rarlab.com/download.htm>

Είναι σημαντικό, αν και δεν το γράφει κανένα manual, για να μην έχετε προβλήματα στην εγκατάσταση αλλά και λειτουργία των εφαρμογών Java να επιλέγεται καταλόγους εγκατάστασης χωρίς κενά στο όνομα, πχ ο κατάλογος C:\Program Files είναι ακατάλληλος! Προτιμήστε καταλόγους της μορφής όπως της δική μας.

B.2.7 Επιλογή βάσης δεδομένων

Το uPortal υποστηρίζει μια μεγάλη γκάμα από βάσεις δεδομένων όπως DB2, Microsoft SQL Server, Oracle Server, PostgreSQL, Sybase SQL Server και MySQL. Αυτονόητο είναι ότι προτιμάμε μια από τις ελεύθερες βάσεις δεδομένων και πρώτη μας επιλογή ήταν η MySQL. Δυστυχώς και μέχρι τη στιγμή αυτή που γράφεται αυτό το κείμενο διάφορα bugs στις εκδόσεις του uPortal 4.0.X μας εμποδίζουν να προχωρήσουμε σε μια τέτοια υλοποίηση. Έτσι καταφύγαμε στη προεπιλεγμένη επιλογή της Hypersonic SQL μιας βάσης δεδομένων σε Java με τυπική υποστήριξη τυποποιημένων SQL και JDBC διαπασών. Πληροφορίες και μεταφόρτωση στη διεύθυνση <http://hsqldb.org/>. Δεν χρειάζεται να την εγκαταστήσετε αφού έρχεται μαζί με την εγκατάσταση του uPortal.

Σε αυτή τη προεπιλεγμένη βάση το uPortal αποθηκεύει όλες τις πληροφορίες για τα διαθέσιμα portlets, κάποιους έτοιμους χρήστες ο καθένας με την διαφορετική ιδιότητα-ρόλο του μαζί με τις προσωπικές επιλογές τους και τα δικαιώματά τους, γενικότερα αποθηκεύεται ότι χρειάζεται να λειτουργήσει η διαδικτυακή μας πύλη.

Σημείωση: έχουν γίνει όλες οι απαραίτητες προσθήκες και αλλαγές στη παραμετροποίηση του uPortal ώστε σε επομένη έκδοση που θα ξεπεραστεί το πρόβλημα με την Mysql πολύ εύκολα με μια μικρή αλλαγή σε ένα μονό αρχείο ρυθμίσεων της εφαρμογής να γίνει η μεταφορά λειτουργίας του κάτω από Mysql.

B.3 Παραμετροποίηση αρχείων

Στη πορεία ανάπτυξης της πύλης χρειάστηκε να τροποποιηθούν ή να δημιουργηθούν πάνω από 30 αρχεία τα περισσότερα από αυτά έχουν να κάνουν με τη παραμετροποίηση της πύλης. Αυτά μπορούν να χωριστούν στις εξής κατηγορίες:

Βασικά αρχεία εγκατάστασης και παραμετροποίησης

Αρχεία διαμόρφωσης της διεπαφής (δημιουργία θέματος διεπαφής-CSS, υποστήριξη ελληνικής γλώσσας.

Δημιουργία των ρόλων χρηστών που έχουμε σχεδιάσει καθώς και την προεπιλεγμένη διάταξη περιεχομένου και portlets για κάθε ένα ρόλο.

Αρχεία υποστήριξης πολλαπλών πηγών εύρεσης χρηστών της πύλης καθώς και τη μέθοδο εύρεσης των χαρακτηριστικών που τους προσδιορίζει το ρόλο και τα δικαιώματά τους.

Αναλυτικότερα:

B.3.1 Εγκατάσταση

Ξεκινώντας με τη διαμόρφωση των παρακάτω αρχείων έχουμε τη βασική παραμετροποίηση εκκίνησης, επιλογής βάσης δεδομένων και σύνδεσης με το CAS:

C:\uPortal-4.0.2\filters\local.properties ώστε να έχει τη τελική παρακάτω μορφή:

```
environment.build.hsql.port=8887
environment.build.hibernate.connection.driver_class=org.hsqldb.jdbc.JDBCdriver
environment.build.hibernate.connection.url=jdbc:hsqldb:hsql://localhost:${environment.build.hsql.port}/uPortal
environment.build.hibernate.connection.username=sa
environment.build.hibernate.connection.password=
environment.build.hibernate.dialect=org.hibernate.dialect.HSQLDialect
environment.build.uportal.server=eclass.career.teicrete.gr:8080
environment.build.uportal.protocol=http
environment.build.uportal.context=/uPortal
environment.build.uportal.email.fromAddress=ggeorge@career.teicrete.gr
environment.build.cas.server=eclass.career.teicrete.gr:8443
environment.build.cas.protocol=https
```

Άλλα αρχεία επικουρικά στην παραμετροποίηση της εγκατάστασης:

C:\uPortal-4.0.2\pom.xml

C:\uPortal-4.0.2\uportal-ear\pom.xml

C:\uPortal-4.0.2\uportal-portlets-overlay\pom.xml

C:\uPortal-4.0.2\portal-portlets-overlay\cas\pom.xml

C:\uPortal-4.0.2\portal-war\src\main\resources\properties\portal.properties

C:\uPortal-4.0.2\portal-war\src\main\resources\properties\rdbm.properties

C:\uPortal-4.0.2\portal-war\src\main\resources\properties\contexts\portletContainerContext.xml

B.4 Δημιουργία Διεπαφής

B.4.1 Γραφικό θέμα

Το uPortal έρχεται με προεγκαταστημένα κάποια βασικά θέματα που ακολουθούν τις προδιαγραφές και τις οδηγίες για την Προσβασιμότητα του Περιεχομένου του Ιστού (WCAG) και διευκολύνουν την πρόσβαση σε άτομα με ειδικές ανάγκες.

Τα προεγκαταστημένα θέματα έχουν γίνει με την ενσωμάτωση τεχνολογιών που έχουν περιγραφεί νωρίτερα όπως Fluid και XSL.

Έτσι σχεδιάσαμε το νέο αισθητικό περιβάλλον της διεπαφής με βάση τα ενσωματωμένα. Σχεδιάστηκε ένα θέμα που να δίνει άμεση πρόσβαση σε άλλες κύριες ιστοσελίδες σχολών, τμημάτων και των κύριων διαδικτυακών υπηρεσιών του ιδρύματος. Ταυτόχρονα επειδή δεν υπήρχε μεταγλώττιση στα ελληνικά της πύλης, προχωρήσαμε σε μετάφραση όλου του λεκτικού περιεχομένου, τόσο της πύλης uPortal όσο και του Κεντρικού Συστήματος Πιστοποίησης (CAS) και η τελική της μορφή θα δοθεί στην κοινοπραξία ανάπτυξης του λογισμικού σαν μια μικρή συνεισφορά μας σ' αυτό το λογισμικό ανοιχτού κώδικα.

Αυτό το νέο θέμα έγινε μετά από σειρά από αλλαγές ή δημιουργία των παρακάτω αρχείων κειμένου:

Κατάλογος C:\uPortal-4.0.2\portal-war\src\main\webapp\media\skins\universality\portal3

Αρχεία: layout.css, skin.xml, portal.css, layout2.css

Κατάλογος C:\uPortal-4.0.2\portal-war\src\main\resources\layout\theme\universality

Αρχεία: universality.xsl, components.xsl, page.xsl

Ταυτόχρονα δημιουργήσαμε και ένα νέο θέμα διεπαφής για την οθόνη σύνδεσης του CAS.

Αρχεία: C:\uPortal-4.0.2\portal-portlets-overlay\cas\src\main\webapp\css\cas.css,
C:\uPortal-4.0.2\portal-portlets-overlay\cas\src\main\webapp\WEB-INF\view\jsp\default\ui\casLoginView.jsp

B.4.2 Δημιουργία αρχείων μεταγλώττισης στα ελληνικά

Κατάλογος C:\uPortal-4.0.2\portal-war\src\main\resources\properties\i18n

Αρχεία: Messages.properties, Messages_el.properties, Messages_el_UTF8.properties

Κατάλογος C:\uPortal-4.0.2\portal-war\src\main\resources\properties\i18n\CAS

Αρχεία: casLoginView.jsp, messages_el.properties, messages_el_UTF8.properties

Το σύστημα μεταγλώττισης των εφαρμογών αυτών επιτρέπει ουσιαστικά τη συλλογή και κωδικοποίηση των λεζάντων που χρειάζονται να μεταφραστούν μέσα σε ένα αρχικό αρχείο όνομα messages.properties. Αυτό είναι και το αρχείο από το οποίο διαβάζει τις λεζάντες εφόσον δεν υπάρχει μετάφραση στη γλώσσα που έχει επιλέξει ο χρήστης. Η μετάφραση της κάθε μια γλώσσας αποθηκεύεται σε ξεχωριστό αρχείο με το χαρακτηριστικό αναγνωριστικό της προσθήκης 2 γραμμάτων που δηλώνουν τη γλώσσα για την οποία είναι υπεύθυνα. Για παράδειγμα για μετάφραση στα ελληνικά το σύστημα αναμένει την ύπαρξη του αρχείου messages_el.properties. Τα αρχεία αυτά είναι καθαρά σε μορφή ASCII. Πως γίνεται λοιπόν η μετάφραση σε γλώσσες που προϋποθέτουν υποστήριξη UTF8 κωδικοποίησης όπως η ελληνική;

Αυτό επιτυγχάνεται με τη μεσολάβηση ενός αρχείου που στη περίπτωση μας το ονομάσαμε messages_el_UTF8.properties και είναι σε κωδικοποίηση UTF8 όπως το περιγράφει και το όνομα του. Στη πραγματικότητα σ' αυτό το αρχείο προχωρήσαμε στη μετάφραση. Στη συνέχεια πρέπει να γίνει η μετατροπή του σε κωδικοποίηση ASCII ώστε να την αναγνωρίζουν οι εφαρμογές με τη

βοήθεια ενός μετατροπέα κωδικοποίησης όπως το πρόγραμμα native2ascii όπου με τη παρακάτω κλήση του γίνεται η δημιουργία του κατάλληλου αρχείου:

```
native2ascii messages_el_UTF8.properties messages_el.properties
```

Σαν τελευταίο βήμα ορίζουμε τις διαθέσιμες γλώσσες με τη σειρά προτίμησης που επιθυμούμε στο αρχείο C:\uPortal-4.0.2\uportal-war\src\main\resources\properties\portal.properties

Για παράδειγμα στη δική μας υλοποίηση, η γραμμή που ορίζει αυτό έχει τη παρακάτω μορφή:

```
org.jasig.portal.i18n.LocaleManager.portal_locales=el_GR,en_US
```

Υπενθυμίζουμε ότι το uPortal αποθηκεύει πάντα την επιλογή γλώσσας του κάθε χρήστη ώστε την επομένη φορά που θα συνδεθεί αυτόματα θα δει την σελίδα του στη γλώσσα της επιλογής του, έτσι η παραπάνω τροποποίηση ισχύει για αυτούς που δεν έχουν λογαριασμό ή δεν έχουν ακόμα συνδεθεί.

B.5 Σύνδεση με καταλόγους δικτύου

Κάτι τέτοιο κατορθώνεται τροποποιώντας μια σειρά από αρχεία xml που καθορίζουν τις εξαρτήσεις αντικειμένων και τις παραμέτρους των εφαρμογών, ακλουθώντας τη σύνταξη που προβλέπει το spring framework, πάνω στο οποίο έχει χτιστεί το uPortal.

Κατάλογος C:\uPortal-4.0.2\uportal-portlets-overlay\cas\src\main\webapp\WEB-INF

Τα βήματα που έγιναν είναι τα παρακάτω:

Τροποποίηση του αρχείου περιγραφής και παραμετροποίησης της εφαρμογής CAS όπου ορίζουμε για κάθε λειτουργία της σε ποια ιστοσελίδα θα απαντά αναλόγως. Αρχείο:web.xml

Ορισμός από πού θα αντλεί πληροφορίες το CAS για την ταυτοποίηση ενός χρήστη. Αρχείο:deployerConfigContext.xml

Εδώ ορίζουμε 2 νέα beans (αντικείμενα μαζί με τις εξαρτήσεις τους), όπου περιγράφουν τις πληροφορίες σύνδεσης με τα LDAP που συνδέεται το CAS.

Για παράδειγμα το κομμάτι του αρχείου που περιγράφει τη σύνδεση με το κεντρικό LDAP του ΤΕΙ Κρήτης έχει την εξής σύνταξη:

```
<bean id="contextSourceT" class="org.springframework.ldap.core.support.LdapContextSource">
<property name="urls" value="ldaps://ldap.teicrete.gr" />
<property name="baseEnvironmentProperties">
  <map>
    <entry key="com.sun.jndi.ldap.connect.timeout" value="2000" />
    <entry key="com.sun.jndi.ldap.read.timeout" value="2000" />
    <entry key="java.naming.security.authentication" value="simple" />
  </map>
</property>
</bean>
```

Ταυτόχρονα, στο ίδιο αρχείο ορίζουμε για κάθε κατάλογο δικτύου μέσα σε ποια αντικείμενα και με βάση ποιες ιδιότητες αναζητούμε το χρήστη, τμήμα του ίδιου αρχείου είναι το παρακάτω:

```
<bean id="bind1" class="org.jasig.cas.adaptors.ldap.BindLdapAuthenticationHandler" >
  <property name="filter" value="mail=%u" />
  <property name="searchBase" value="dc=teicrete,dc=gr" />
  <property name="contextSource" ref="contextSourceT" />
</bean>
```

Στο παραπάνω παράδειγμα, είναι φανερό ότι αναζητούμε τους χρήστες του contextSourceT που έχουμε ορίσει νωρίτερα συσχετίζοντας το όνομα χρήστη που έχει πληκτρολογήσει στη οθόνη σύνδεσης του CAS με τη τιμή της ιδιότητας mail που έχουν τα αντικείμενα τύπου "dc=teicrete,dc=gr" στον συγκεκριμένο κατάλογο δικτύου

B.6 Δημιουργία ρόλων

Το επόμενο βήμα είναι να ορίσουμε τους ρόλους που θα αναγνωρίζει το uPortal. Αυτό γίνεται με τη διαμόρφωση του αρχείου C:\uPortal-4.0.2\uportal-war\src\main\resources\properties\groups\PAGSGroupStoreConfig.xml

Αυτό επιτυγχάνεται με τον ορισμό κάθε επιθυμητού ρόλου και στη συνέχεια την αντιστοίχιση του κάθε χρήστη με έναν ή περισσότερους ρόλους. Δηλώνουμε ποιες τιμές πρέπει να έχουν μια ή περισσότερες ιδιότητες του αντικείμενου με τα στοιχεία του χρήστη, ώστε να αναγνωρίζεται ο κάθε ένας ρόλος.

Στο αρχείο C:\uPortal-4.0.2\uportal-war\src\main\resources\properties\dlm.xml ορίζουμε τις διαθέσιμες εξατομικευμένες αρχικές ιστοσελίδες που αντιστοιχούν στους παραπάνω ρόλους και ο χρήστης τη πρώτη φορά που θα συνδεθεί θα δει ανάλογα με το ρόλο του. Το περιεχόμενο και η διάταξη της κάθε παραπάνω ιστοσελίδας ορίζεται μέσα από το uPortal και τα ενσωματωμένα εργαλεία διαχείρισης της πύλης.

Ορίζουμε ξανά, αλλά για λογαριασμό του uPortal, τους καταλόγους δικτύου ως τις πηγές αναζήτησης των ιδιοτήτων που μπορεί κάθε χρήστης να έχει. Επίσης για κάθε μια πηγή αντιστοιχούμε τα ονόματα των ιδιοτήτων που βρίσκονται στο κατάλογο με τα ονόματα των ιδιοτήτων που θα χρησιμοποιεί το uPortal. Αυτό είναι ίσως το πιο βασικό σημείο, επειδή όπως έχουμε πει, κατάλογοι δικτύου ασύμβατοι μεταξύ τους μπορούν και συνδυάζονται κάτω από την πλατφόρμα του uPortal. Για παράδειγμα μπορούμε να ορίσουμε ότι κάθε χρήστης του uPortal θα έχει μια εσωτερική ιδιότητα με όνομα Email_address. Στη συνέχεια, για κάθε μια πηγή προχωρούμε στην αντιστοίχιση αυτής της ιδιότητας του uPortal με μια ιδιότητα του καταλόγου. Σε ένα κατάλογο η ιδιότητα αυτή μπορεί να έχει το όνομα mail, σε ένα άλλο το όνομα Email αλλά τελικά η εσωτερική ιδιότητα Email_address θα πάρει τη σωστή τιμή από το κατάλογο από τον οποίο θα ανακτήσει τις πληροφορίες για αυτόν το χρήστη.

Στο κατάλογο C:\uPortal-4.0.2\uportal-war\src\main\resources\properties\contexts

personDirectoryContext.xml

ldapContext.xml

B.7 Υλοποίηση https-SSL

Μια σέρα από αρχεία παραμέτρων τροποποιήθηκαν για την ρυθμιση παραμετρων σε θεματα ασφαλειας και υποχρεωτικής λειτουργία του CAS σε πρωτόκολλο https.

C:\uPortal-4.0.2\filters\local.properties

C:\uPortal-4.0.2\uportal-war\src\main\resources\properties\security.properties

Κατάλογος C:\uPortal-4.0.2\uportal-portlets-overlay\cas\src\main\webapp\WEB-INF

deployerConfigContext.xml

web.xml

B.7.1 Μεταφορά πιστοποιητικού

Το Γραφείο Διασύνδεσης για τις ανάγκες μηχανογράφησης του, έχει ήδη αποκτήσει πιστοποιητικό έγκυρο από αρχή έκδοσης πιστοποιητικού. Το πιστοποιητικό καλύπτει τις ανάγκες του ιστότοπου και του πληροφοριακού συστήματος του γραφείου και είναι εγκατεστημένο σε λειτουργικό σύστημα Linux και χρησιμοποιείται από διακομιστής ιστοσελίδων Apache. Είναι ένα πιστοποιητικό που είναι έγκυρο για όλο το τομέα career.teicrete.gr του γραφείου.

Το uPortal είναι εγκατεστημένο σε άλλο διακομιστή και έπρεπε να γίνει η μεταφορά του πιστοποιητικού στο διακομιστή αυτόν και σε λειτουργικό σύστημα Windows Server. Η διαδικασία αυτή είναι αρκετά πολύπλοκη, καθώς τα πιστοποιητικά έχουν τη μορφή ενός ή περισσότερων κωδικοποιημένων αρχείων και έχουν άλλη δομή σε Linux και άλλη σε Windows διακομιστές και μια απλή αντιγραφή των αρχείων δεν αρκεί. Απαιτείται λοιπόν η μετατροπή τους στο νέο τύπο. Αυτό επιτυγχάνεται με τη εκτέλεση μιας επιπλέον εντολής που εξάγει το πιστοποιητικό στη μορφή που ζητείται.

Η παρακάτω εκτέλεση είναι ενδεικτική μόνο για την συγκεκριμένη εγκατάσταση του uPortal και μετατρέπει το πιστοποιητικό .pem του Linux σε τύπο PKCS#12 των Windows:

```
#openssl pkcs12 -export -inkey apollon-key.pem -in apollon-cert.pem -out  
apollon.p12 -name apollon.carrer.teicrete.gr
```

Στη συνέχεια, τα αρχεία με την μορφή που είναι συμβατή με τα Windows αντιγράφονται στο διακομιστή του uPortal και γίνονται εισαγωγή στο tomcat.

B.7.2 Εισαγωγή στο tomcat:

```
keytool -import -alias tomcat -keystore c:\keystore -trustcacerts -file apollon.p12
```

Σε αυτό το σημείο έχουμε ένα τυπικό setup για https.[51]

Το CAS όμως απαιτεί να μην είναι μόνο διαπιστευμένος ο διακομιστής αλλά και το πρόγραμμα-πελάτης. Στη περίπτωση μας πελάτης είναι το uPortal που βρίσκεται εγκατεστημένο στον ίδιο διακομιστή.

Αυτό επιτυγχάνεται με τη εκτέλεση του παρακάτω προγράμματος InstallCert το οποίο εσωκλείεται στο κώδικα με σύνταξη κλήσης:

```
java InstallCert [host]:[port]
```

όπου host η διεύθυνση του διακομιστή που τρέχει το uPortal και port συνήθως είναι 8443 όπου εξυπηρετείται το https πρωτόκολλο.

εναλλακτικά:

```
keytool -exportcert -keystore jssecacerts -storepass changeit -file output.cert
```

```
keytool -importcert -keystore [DESTINATION_KEYSTORE] -file output.cert
```

B.8 Δημιουργία πιστοποιητικού

3.1. Λειτουργικά συστήματα Linux, *BSD, Mac OS X με χρήση της βιβλιοθήκης OpenSSL[47]

```
$ OLD_UMASK=`umask`
```

```
$ umask 077
```

Η εντολή umask 077 εξασφαλίζει ότι το ιδιωτικό κλειδί θα αποθηκευθεί σε αρχείου που θα είναι αναγνώσιμο μόνο από το χρήστη που το δημιούργησε.

```
$ openssl genrsa 2048 > key.pem
```

Η εντολή αυτή δημιουργεί ένα ιδιωτικό κλειδί μεγέθους 2048 bits και το αποθηκεύει στο αρχείο με όνομα key.pem.

Τέλος, επαναφέρουμε την παλιά μάσκα δημιουργίας αρχείων:

```
$ umask $OLD_UMASK
```

Όλη η ασφάλεια του πιστοποιητικού εξαρτάται από την ασφαλή διαφύλαξη του ιδιωτικού κλειδιού. Ποτέ μη δίνετε πρόσβαση σε αυτό σε τρίτους. Όποιος έχει πρόσβαση στο ιδιωτικό κλειδί μπορεί να προσποιηθεί την ταυτότητα του εξυπηρετητή για τον οποίο εκδίδεται το πιστοποιητικό.

Αν το ιδιωτικό κλειδί χαθεί ή σβηστεί, τότε το πιστοποιητικό είναι άχρηστο. Γι' αυτό προτείνεται η τήρηση ενός αντιγράφου ασφαλείας του ιδιωτικού κλειδιού σε ασφαλή τοποθεσία.

B.8.1 Δημιουργία νέου Αιτήματος Υπογραφής Πιστοποιητικού

Προκειμένου να προχωρήσετε στην έκδοση νέου πιστοποιητικού, είναι απαραίτητη η δημιουργία ενός νέου Αιτήματος Υπογραφής Πιστοποιητικού (Certificate Signing Request - CSR). Το Αίτημα περιλαμβάνει τα στοιχεία του Υποκειμένου του πιστοποιητικού (όνομα, χώρα, οργανισμός κλπ.), καθώς και το δημόσιο κλειδί που αντιστοιχεί στο ιδιωτικό. Στη συνέχεια η Αρχή Πιστοποίησης θα υπογράψει τα στοιχεία αυτά, προσδίδοντάς τους κύρος.

Το Αίτημα δημιουργείται χρησιμοποιώντας την εντολή:

```
$ openssl req -new -key key.pem -nodes
```

Θα ερωτηθείτε σχετικά με τα στοιχεία που θέλετε να περιλαμβάνει το πεδίο. Από αυτά υποχρεωτικό είναι μόνο το πεδίο «Common Name» (<1> - κοινό όνομα), το οποίο πρέπει να συμπίπτει με το πλήρες όνομα του εξυπηρετητή (Fully Qualified Domain Name - FQDN) για τον οποίο ζητάτε τον εξοπλισμό. Αν ο εξυπηρετητής έχει περισσότερα του ενός ονόματα (π.χ. www.example.com και mail.example.com), τότε αναφέρετε μόνο το κύριο. Στη συνέχεια θα έχετε τη δυνατότητα να προσθέσετε τα επιπρόσθετα ονόματα του εξυπηρετητή.

Αφού απαντήσετε στις ερωτήσεις, θα τυπωθεί το Αίτημα στην οθόνη σας:

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:GR

State or Province Name (full name) [Some-State]:Lakonia

Locality Name (eg, city) []:Sparti

Organization Name (eg, company) [Internet Widgits Pty Ltd]:Some organization

Organizational Unit Name (eg, section) []:Some unit

Common Name (eg, YOUR name) []:www.example.com <1>

Email Address []:

Please enter the following 'extra' attributes to be sent with your certificate request

A challenge password []:

An optional company name []:

-----BEGIN CERTIFICATE REQUEST-----

MIIcVzCCAacCAQAwejELMAkGA1UEBhMCR1IxEDA0BgNVBAGTB0xha29uaWExDzAN

BgNVBActBlNwYXJ0aTEaMBgGA1UEChMRU29tZSBvcmdhbml6YXRpb24xEjAQBgNV

BAstCVNvbWUgdW5pdDEYMBYGA1UEAxMPd3d3LmV4YW1wbGUuY29tMIIBIjANBgkq

hkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAtAR3RZavJpfC/hAdzb+/O2fgdSEPIVJw

nAS6IYnEKLGP7qmaPGQZK06tl6FAIAepU8i9bXfcnX4k9JvIzsEBAm454SyTVcGo
jxQ4buadLQyLSNiK9nEMbMexkwYZkYkOHGb70mBcw1FT3Jhr1sqCRMPHujE4UttS
dhc5bNo2IZK34c9GipnOQ6dz3IVa/21JGgPbm6b0wuv4i9R8YQs6d0EIUIIya2VS
6Kd3gB+3U87URqneYUdL6gYqzVGV/Fc1R+HsoBl8P3VWYbKDHaLXXxnIC84av4nM
c+20C8gUuUAWXdd2/uhNnU4IVTc8HwjB9LxLORJvGSMno/H8vy9VUwIDAQABoAAw
DQYJKoZIhvcNAQEFBQADggEBAEyke2DL8Pay015JLeC6elzC8uBuPtO5gYHdlmIQ
IE2MnFzjWWvWF8Y5eCUtjv+ccmL3Yj6YvtjoEN5VyWLUdVABrrnv+F+yXRTRPD+3
sm9Rd/XWSrv+Gu5KPZ2RtuligTwljMZ7MLGkH8HvZzhNnb0rKzILVvybbVns5kTp
sjyYBdS0CSAzsrMyYItUjLg/alhnFA/c3rG9B+nrQrbf8+K+jgV+l2z3cTuXLiw
dSdgj9+KosvH8K+gwtAcB1FQ3lamYr1hcGWzOg9seF122JfpWxmXqW8D5lPntrDs
TXwM9KO+aSDtF4ei4Y52wniB9spK5K+3bbWr3MBL6WqZ1KE=

-----END CERTIFICATE REQUEST-----

Το κομμάτι μεταξύ -----BEGIN CERTIFICATE REQUEST----- και -----END CERTIFICATE REQUEST--
--- θα πρέπει να το αντιγράψετε και να το επικολλήσετε στο πεδίο «Certificate Request» της
Εφαρμογής.

Microsoft Windows

Από το μενού «Start» επιλέξτε Administrative Tools → Internet Services Manager.

Επιλέξτε, κάτω από την ενότητα Web Sites, την ιστοσελίδα για την οποία θέλετε να εκδώσετε
πιστοποιητικό.

Κάνετε δεξί κλικ στην ιστοσελίδα και επιλέξτε Properties.

Στην καρτέλα Directory Security επιλέξτε Server Certificate.

Επιλέξτε Create new certificate και πατήστε Next.

Επιλέξτε Prepare the request now, but send it later και πατήστε Next.

Επιλέξτε ένα όνομα για το πιστοποιητικό, ώστε να μπορείτε να το εντοπίζετε εύκολα στον κατάλογο των πιστοποιητικών. Το όνομα αυτό είναι για δική σας διευκόλυνση και μόνο.

Συνιστούμε να επιλέξετε μέγεθος κλειδιού (bit length) 2048 bits.

Η τιμή του πεδίου Organization δε λαμβάνεται υπόψη από το σύστημα. Συμπληρώστε το πεδίο Organizational Unit και πατήστε Next.

Εισάγετε το πλήρες όνομα του εξυπηρετητή (Fully Qualified Domain Name - fqdn) στο πεδίο Common name. Αυτό πρέπει να είναι το όνομα με το οποίο συνδέονται οι χρήστες σας στον εξυπηρετητή, π.χ. www.example.com. Αν ο εξυπηρετητής σας έχει περισσότερα από ένα ονόματα στην ίδια διεύθυνση IP, τότε το σύστημα θα σας δώσει τη δυνατότητα καθορισμού των επιπλέον ονομάτων που θέλετε να περιλαμβάνονται στο πιστοποιητικό, μετά την αποστολή του CSR. Συμπληρώστε το όνομα και πατήστε Next.

Αφήστε τις επιλογές της οθόνης αυτής στις τρέχουσες τιμές τους, δε λαμβάνονται υπόψη από το σύστημα. Πατήστε Next.

Επιλέξτε μια τοποθεσία για την αποθήκευση του Αιτήματος Υπογραφής Πιστοποιητικού (CSR). Τα περιεχόμενα του αρχείου αυτού θα αποσταλούν στο σύστημα αργότερα. Πατήστε Next.

Επαληθεύστε τις πληροφορίες που εισαγάγατε και γυρίστε πίσω να διορθώσετε τυχόν λάθη. Θυμηθείτε ότι το Common Name πρέπει να αντιστοιχεί στο όνομα του εξυπηρετητή όπως το χρησιμοποιούν οι χρήστες των εφαρμογών σας. Οι πληροφορίες τοποθεσίας καθώς και το πεδίο Organization αγνοούνται από το σύστημα και αντικαθίστανται από αυτές του Οικείου Φορέα σας. Όταν ολοκληρώσετε τυχόν διορθώσεις πατήστε Next.

Παράρτημα Γ

Επιστολές ανάθεσης έργου

Γ.1 Επιστολή αίτησης άδειας υλοποίησης έργου

Προς Επιστημονικό Υπεύθυνο
του Κέντρου Έλεγχου και Διαχείρισης Δικτύων
(Κ.Ε.Ε.Δ.) του ΤΕΙ Κρήτης

Αγαπητέ κ. Βασιλάκη

Όπως έχουμε ήδη συζητήσει σε προηγούμενες συναντήσεις μας, για επιλογή υποψήφιων θεμάτων για την διπλωματική μου διατριβή, στα πλαίσια του μεταπτυχιακού προγράμματος "Πληροφοριακά Συστήματα" του Ανοικτού Πανεπιστημίου Κύπρου, θα ήθελα να σας ενημερώσω ότι εγκρίθηκε η πρόταση για «Υλοποίηση Προσωποποιημένων Πυλών για Εκπαιδευτικά Ιδρύματα».

Για το λόγο αυτό θα ήθελα και γραπτώς με τη μορφή μιας επιστολής να μου δώσετε την άδεια για την εφαρμογή της στο ΤΕΙ Κρήτης όπου έχω την χαρά να εργάζομαι.

Ευχαριστώ εκ των προτέρων

Γιώργος Γεωργιλιάς
Τμήμα Πληροφορικής Κεντρικής Διοίκησης
τηλ. 2810 379815, ggeorge@staff.teicrete.gr

Γ.2 Επιστολή έγκρισης ανάθεσης έργου

Προς κάθε ενδιαφερόμενο,

Με τη παρούσα επιστολή θέλω να δηλώσω, ότι το Κέντρο Έλεγχου και Διαχείρισης Δικτύων του ΤΕΙ Κρήτης, ενδιαφέρεται για την υλοποίηση μιας προσωποποιημένης εκπαιδευτικής πύλης, όπου οι χρήστες, μετά από την ταυτοποίησή τους, θα έχουν πρόσβαση στις διαδικτυακές υπηρεσίες του Ιδρύματος.

Για το λόγο αυτό δίνουμε την έγκρισή μας για υλοποίηση της πύλης στον κ. Γεώργιο Γεωργιλιά, οποίος είναι φοιτητής στο Ανοικτό Πανεπιστήμιο Κύπρου στα πλαίσια της μεταπτυχιακής διατριβής του.

Κώστας Βασιλάκης



Καθηγητής Πληροφορικής
Επιστημονικός Υπεύθυνος
Κέντρου Έλεγχου και Διαχείρισης
Δικτύων (Κ.Ε.Ε.Δ.) ΤΕΙ Κρήτης