

ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων



Πτυχιακή Εργασία

**«Ανάπτυξη Διαδικτυακής Πύλης Εκπαιδευτικού
Περιεχομένου στο Liferay»**

Περράκη Κωνσταντίνα

Καπετανάκη Ερατώ Στεφανία

Επιβλέπων Καθηγητής : Ακουμιανάκης Δημοσθένης

Ηράκλειο Νοέμβριος 2010

i. Περιεχόμενα

i.	Περιεχόμενα.....	2
ii.	Πίνακας εικονών	4
	Ευχαριστίες.....	6
	Εισαγωγή	8
1	Εισαγωγή και Ανάπτυξη Web Περιεχομένου	10
1.1	CMS.....	10
1.1.1	Joomla!.....	11
1.1.2	Drupal.....	13
1.1.3	Liferay.....	15
1.1.4	Συμπεράσματα για το CMS.....	17
2	Ανάπτυξη templates.....	20
2.1	Γλώσσες Προγραμματισμού	20
2.1.1	HTML	20
2.1.2	CSS	21
2.1.3	XML	27
2.1.4	XSL, XSLT, XSL-FO.....	29
2.1.5	Velocity Template	32
2.1.6	JAVASCRIPT	33
2.1.7	jQuery	36
3	Liferay CMS.....	41
3.1	Portlet.....	43
3.1.1	Portlet Container.....	44
3.2	Δημιουργία template στο Liferay	45
4	Λεπτομέρειες υλοποίησης - Ειδικά templates.....	49
4.1	Πρώτη Επαφή Με Templates	49

4.2	Ειδικά Templates εκπαιδευτικού περιεχομένου.....	52
4.2.1	Body and Galleria Aino.....	52
4.2.2	Image Gallery Rotator.....	56
4.2.3	Tabs Menu.....	59
4.2.4	Original Photo Gallery.....	61
5	Επίλογος.....	66
5.1	Συμπεράσματα.....	66
5.2	Μελλοντικά Έργα.....	67
6	Αναφορές.....	68

ii. Πίνακας εικονών

Εικόνα 1: Χαρακτηριστικά στοιχεία εικόνας	27
Εικόνα 2: Μετατροπή εγγράφου XML σε XHTML.....	31
Εικόνα 3: Add Application CMS	43
Εικόνα 4: Portlet container	44
Εικόνα 5: Δημιουργία Structure	46
Εικόνα 6: Προσθήκη στοιχείων στο Structure	46
Εικόνα 7: Δημιουργία Template	47
Εικόνα 8: Launch Editor Template	48
Εικόνα 9: Structure	49
Εικόνα 10: Σκαρίφημα εικόνας (center, left, right)	50
Εικόνα 11: Εικόνα στα δεξιά	51
Εικόνα 12: Εικόνα στα αριστερά	51
Εικόνα 13: Εικόνα στο κέντρο	51
Εικόνα 14: Structure	52
Εικόνα 15: Σκαρίφημα προτύπου	52
Εικόνα 16: Όψη Template	53
Εικόνα 17: Structure.....	54
Εικόνα 18: Σκαρίφημα προτύπου	55
Εικόνα 19: Όψη Template	55
Εικόνα 20: Structure	56
Εικόνα 21: Σκαρίφημα προτύπου	57
Εικόνα 22: Όψη Template χωρίς block.....	58
Εικόνα 23: Όψη Template με block	58
Εικόνα 24: Structure	59

Εικόνα 25: Σκαρίφημα προτύπου	60
Εικόνα 26: Όψη Template (1)	60
Εικόνα 27: Όψη Template (2)	61
Εικόνα 28: Structure	62
Εικόνα 29: Σκαρίφημα κεντρικού προτύπου	63
Εικόνα 30: Σκαρίφημα προτύπου pop up.....	63
Εικόνα 31: Όψη Template (1)	64
Εικόνα 32: Όψη Template (2)	64
Εικόνα 33: Όψη Template (3)	65

Ευχαριστίες

Αρχικά θα θέλαμε να ευχαριστήσουμε τον κ. Δημοσθένη Ακουμιανάκη για την καθοδήγηση του ως επιβλέπων καθηγητής στην πτυχιακή μας εργασία και την ευκαιρία που μας έδωσε να ενταχθούμε στο ερευνητικό εργαστήριο του iSTLab. Επίσης, χωρίς τις πολύτιμες και καίριες συμβουλές του Ανάργυρου Ακρίβου και Χαβιαρά Κώστα, δεν θα μπορούσαμε να έχουμε το σημερινό αποτέλεσμα. Αξιζεί να προσθέσουμε ότι χωρίς τις συμβουλές, την καθοδήγηση, την οργάνωση και υπερβολική υπομονή του Μηλολιδάκη Γιάννη δεν θα είχαμε βγάλει εις πέρας αυτήν την πτυχιακή. Τέλος, ευχαριστούμε τα μέλη του εργαστηρίου για το καθημερινό τους ενδιαφέρον και την καλή συνεργασία που αποκτήσαμε.

Περίληψη

Στην παρούσα πτυχιακή ασχοληθήκαμε με την ανάπτυξη προτύπων (templates) με σκοπό τη δημιουργία διαδικτυακής πύλης εκπαιδευτικού περιεχομένου. Συγκεκριμένα αναλύονται πρότυπα μερικά από τα οποία χρησιμοποιούνται στον ιστοχώρο του iSTLab. Τα πρότυπα αυτά αναπτύχθηκαν με την χρήση του Liferay CMS το οποίο βασίζεται στη γλώσσα προγραμματισμού java. Στο θεωρητικό μέρος της πτυχιακής αυτής γίνεται μια βασική αναφορά στις τεχνολογίες που χρησιμοποιήθηκαν για την κατασκευή των προτύπων οι οποίες είναι οι HTML, JavaScript, XSL, jQuery. Σκοπός αυτής της αναφοράς είναι εκτός από το πρακτικό μέρος, όπου είναι εμφανής η χρήση των τεχνολογιών αυτών, να είναι κατανοητές από τους αναγνώστες αυτής της πτυχιακής και οι θεωρητικές έννοιες που τις αντιπροσωπεύουν (π.χ css, portal, portlet container κτλ).

Abstract

This thesis dealt with the development of templates whose content is educational. It analyzed some of the templates used on the website of iSTLab. The templates were developed using the Liferay CMS which is based on the programming language java. The technologies used are HTML, JavaScript, XSL and jQuery. In the theoretical part of this thesis all the technologies used for manufacturing templates are described in detail. The aim of this report, besides the practical part in which the usage of the technologies is evident, is for all readers to understand the details (ex. css, portal, portlet container etc).

Λέξεις κλειδιά

[CMS](#), [Liferay CMS](#), [HTML](#), [XML](#), [XSL-XSLT](#), [Velocity Template](#), [CSS](#), [Java-JavaScript](#), [jQuery](#), [Portlet](#), [Portlet Container](#)

Εισαγωγή

Η αποτελεσματική υλοποίηση κάθε εκπαιδευτικού προγράμματος, ανεξάρτητα από την τελική μορφή υλοποίησής του (face to face, blended learning, e-learning) εξαρτάται σε πολύ μεγάλο βαθμό από την ποιότητα και την αποτελεσματική χρήση του εκπαιδευτικού υλικού. Με στόχο τη βέλτιστη επίτευξη των προσδοκώμενων αποτελεσμάτων, η πτυχιακή εργασία αυτή αποτελείται από πέντε κεφάλαια.

Ξεκινώντας κάνουμε μια μικρή περιγραφή για CMS. Πιο συγκεκριμένα αναλύουμε και συγκρίνουμε τα 3 state of the art συστήματα, την Joomla!, το Drupal και το Liferay. Η Joomla! είναι ένα CMS αρκετά εύκολο στην χρήση, δεν απαιτεί άδεια χρήσης και είναι αρκετά αξιόπιστη για την ανάπτυξη σοβαρών δικτυακών τόπων. Το Drupal αντιθέτως, αποτελεί ένα ελαφρώς πιο πολύπλοκο CMS. Είναι επίσης ελεύθερο λογισμικό, αρκετά ευέλικτο και έχει αποκτήσει με τον καιρό τη δική του κοινότητα που το υποστηρίζει. Τέλος, το Liferay, το οποίο αποτελεί και το επίκεντρο της πτυχιακής αυτής, είναι το κυρίαρχο προϊόν δημιουργίας portal σήμερα.

Συνεχίζοντας στο δεύτερο κεφάλαιο περιγράφουμε τις πιο διαδεδομένες γλώσσες προγραμματισμού οι οποίες χρησιμοποιήθηκαν για την παραγωγή του τελικού αποτελέσματος αυτής της πτυχιακής. Πιο συγκεκριμένα θα ασχοληθούμε με την HTML, μια γλώσσα σήμανσης και το CSS, μια γλώσσα βοηθητική όσον αφορά την προσαρμοστικότητα του στυλ ενός εγγράφου. Επίσης κάνουμε εκτενή αναφορά στην XML καθώς και στις γλώσσες XSL, XSLT και XSL-FO που επιτρέπουν τον μετασχηματισμό ενός XML αρχείου σε διάφορες μορφές. Θα δώσουμε έμφαση στην VM, μια όχι και τόσο γνωστή γλώσσα αλλά αρκετά χρήσιμη για την δημιουργία θεμάτων (Visual themes) και τον μετασχηματισμό XML δομών. Τέλος θα αναφερθούμε στις γλώσσες Java, JavaScript και στην βιβλιοθήκη διαδραστικών αντικειμένων jQuery, η οποία πρόκειται για μια βιβλιοθήκη JavaScript ανοικτού κώδικα.

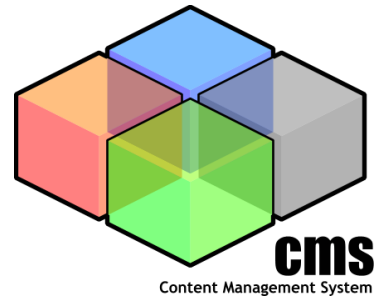
Ακολουθεί αναλυτική αναφορά στο Liferay στο τρίτο κεφάλαιο. Θα περιγράψουμε αναλυτικά τα χαρακτηριστικά του όπως την δημοσιοποίηση web περιεχομένου, την δημιουργία image gallery, τον ευέλικτο μηχανισμό μοτίβων που υποστηρίζει και το δομημένο περιεχόμενο. Παρακάτω γίνεται αναφορά στη δομή ενός portlet καταλήγοντας στην δημιουργία ενός template στο Liferay και πως φιλοξενείται σε ένα portlet περιγράφοντας αναλυτικά το κάθε στάδιο. Θα καταγράψουμε τρεις διαδικασίες που ακολουθούνται τυφλά, τη δημιουργία structure, τη συσχέτισή του με ένα template παρουσίασης και έπειτα τη δημιουργία ενός προτύπου που κάνει χρήση των δυο πρώτων.

Στο τέταρτο κεφάλαιο θα παρουσιάσουμε με λεπτομέρεια τις δικές μας υλοποιήσεις που βασίζονται στα εργαλεία και τις τεχνικές που αναφέραμε στα προηγούμενα κεφάλαια. Το κεφάλαιο χωρίζεται σε 3 μέρη, την πρώτη επαφή που είχαμε με templates, την τροποποίηση ήδη υλοποιημένων templates, την ενσωμάτωση τους στο Liferay και τέλος την κατασκευή ενός εξ'ολοκλήρου νέου template και την ενσωμάτωση και εκείνου στο Liferay.

1 Εισαγωγή και Ανάπτυξη Web Περιεχομένου

1.1 CMS

Ο όρος CMS (*Content Management Systems, Συστήματα Διαχείρισης Περιεχομένου*) αναφέρεται στις εφαρμογές που επιτρέπουν στο χρήστη να διαχειρίζεται το δικτυακό του περιεχόμενο, όπως κείμενα, εικόνες, πίνακες κ.λπ., με εύκολο τρόπο. Οι εφαρμογές διαχείρισης περιεχομένου καθιστούν δυνατή την αλλαγή του περιεχομένου χωρίς να είναι απαραίτητες ειδικές γνώσεις σχετικές με τη δημιουργία ιστοσελίδων ή γραφικών, λόγω του ότι τα κείμενα γράφονται μέσω κάποιων online html editors, ειδικών δηλαδή κειμενογράφων, παρόμοιων με το MS Word, που επιτρέπουν τη μορφοποίηση των κειμένων.



Οι αλλαγές του περιεχομένου μπορούν να γίνουν από οποιονδήποτε υπολογιστή που είναι συνδεδεμένος στο Διαδίκτυο, χωρίς να χρειάζεται να έχει εγκατεστημένα ειδικά προγράμματα επεξεργασίας ιστοσελίδων, γραφικών κ.λπ. πέραν του συστήματος το οποίο χρησιμοποιεί ο χρήστης. Μέσω ενός browser, ο χρήστης μπορεί να συντάξει ένα κείμενο και να ενημερώσει άμεσα το δικτυακό του τόπο.

Τα CMS διαφοροποιούνται μεταξύ τους σε αρκετά σημεία, επειδή όμως όλα έχουν κοινό στόχο θα πρέπει οπωσδήποτε να υποστηρίζουν κάποιες βασικές λειτουργίες. Έτσι, διακρίνονται κάποια υποσυστήματα τα οποία είναι βασικά και θα πρέπει να τα διαθέτει οποιοδήποτε σοβαρό CMS. Αυτά είναι:

- Σύστημα σύνταξης (authoring)
- Σύστημα διαχείρισης (Management)
- Σύστημα αυτοματοποίησης κύκλου εργασιών (workflow automation)
- Σύστημα έκδοσης

Στο πλαίσιο της παρούσας πτυχιακής θα αναφερθούμε στα συστήματα διαχείρισης περιεχομένου Joomla, Drupal και Liferay.

1.1.1 Joomla!

Είναι το πρώτο CMS που θα εξετάσουμε.

Παρέχει στο χρήστη τη δυνατότητα να δημοσιεύσει στο διαδίκτυο μια προσωπική ιστοσελίδα, αλλά και



έναν ολόκληρο εταιρικό δικτυακό τόπο. Πρόκειται για εφαρμογή ανοικτού κώδικα. Η χρήση της Joomla! είναι απολύτως δωρεάν. Είναι ελεύθερη για χρήση, εύκολη στην τροποποίηση και δεν απαιτείται κάποια άδεια χρήσης.

Όσον αφορά τη χρήση της είναι αρκετά απλή. Η Joomla! εγκαθίσταται σε έναν κεντρικό υπολογιστή, τον web server. Ο χρήστης έχει πρόσβαση στο περιβάλλον διαχείρισης μέσω ενός browser, όπως είναι ο Firefox ή ο Internet Explorer.

Ενσωματώνοντας ορισμένα πρόσθετα εργαλεία/επεκτάσεις, τα οποία διατίθενται δωρεάν, διευρύνονται οι δυνατότητες και λειτουργίες της Joomla!, κάνοντάς την μια εφαρμογή αξιόπιστη για την ανάπτυξη σοβαρών δικτυακών τόπων.

Παρακάτω αναφέρονται επιγραμματικά πλεονεκτήματα και μειονεκτήματα:

Πλεονεκτήματα της Joomla:

- Απλή εγκατάσταση
- Έτοιμα εμπορικά θέματα
- Επιλογή παρόχου - Οι περισσότερες εταιρείες διαφημιστικών ιστοσελίδων στην Ελλάδα χρησιμοποιούν Joomla

Μειονεκτήματα της Joomla:

- Αδυναμία πραγματικής ενσωμάτωσης διαδικτυακών εφαρμογών
- Αδύνατη υποστήριξη SEO (Search Engine Optimization)
- Φτωχή διαχείριση πολυγλωσσικότητας
- Πρωτόγονη διαχείριση αδειών πρόσβασης
- Περιορισμένη κατηγοριοποίηση/ταξινόμηση περιεχομένου
- Απουσία Enterprise δυνατοτήτων (αναθεώρηση περιεχομένου, caching, NBS κτλ)

Παρακάτω παρουσιάζεται ένας συγκεντρωτικός πίνακας των χαρακτηριστικών του συγκεκριμένου CMS.

Προγραμματιστής (-ες)	Η Ομάδα Έργου Joomla
Development status	Active
Γράφει σε	PHP
Λειτουργικό σύστημα	Cross-platform
Μέγεθος	6,7 MB (συμπιεσμένο) 17,7 MB (χωρίς συμπίεση)
Τύπος	Σύστημα Διαχείρισης Περιεχομένου
Άδεια	GNU General Public License
Ιστοσελίδα	http://www.joomla.org/

1.1.2 Drupal

Το επόμενο CMS στη λίστα της εξέτασης είναι το Drupal. Αποτελεί ανοικτό/ελεύθερο λογισμικό, γραμμένο στη γλώσσα προγραμματισμού PHP. Όπως πολλά σύγχρονα CMS, επιτρέπει στο διαχειριστή συστήματος:



- να οργανώνει το περιεχόμενο
- να προσαρμόζει την παρουσίαση
- να αυτοματοποιεί διαχειριστικές εργασίες
- να διαχειρίζεται τους επισκέπτες του ιστοτόπου και αυτούς που συνεισφέρουν

Παρόλο που υπάρχει μια πολύπλοκη προγραμματιστική διεπαφή, οι περισσότερες εργασίες μπορούν να γίνουν με λίγο ή και καθόλου προγραμματισμό.

Το Drupal μπορεί να εκτελεστεί σε διάφορες πλατφόρμες, συμπεριλαμβανομένων των λειτουργικών συστημάτων Windows, Mac OS X, Linux, FreeBSD, ή οποιασδήποτε πλατφόρμας που υποστηρίζει είτε το διακομιστή ιστοσελίδων Apache HTTP Server, είτε το Internet Information Services, καθώς επίσης και τη γλώσσα προγραμματισμού PHP. Απαιτεί μια βάση δεδομένων όπως η MySQL και η PostgreSQL για την αποθήκευση του περιεχομένου και των ρυθμίσεών του.

Παρακάτω αναφέρονται επιγραμματικά πλεονεκτήματα:

Πλεονεκτήματα του Drupal:

- Ελεύθερο λογισμικό - δεν χρειάζεται άδεια για εγκατάσταση ή χρήση
- Μηδενικά ετήσια τέλη χρήσης
- Απεριόριστη επεκτασιμότητα και ευελιξία
- Μεγάλη κοινότητα που υποστηρίζει και αναπτύσσει το Drupal

- Μείωση του χρόνου ανάπτυξης σε σχέση με ιδιωτικούς φορείς παροχής και συστήματα
- Βελτίωση της ασφάλειας μέσω συνεχής ενημέρωσης εκδόσεων
- Τεράστια ποικιλία εφαρμογών για εμπορικούς, διαφημιστικούς και κοινωνικούς ιστοτόπους

Παρακάτω παρουσιάζεται ένας συγκεντρωτικός πίνακας των χαρακτηριστικών του συγκεκριμένο CMS.

Προγραμματιστής (-ες)	Dries Buytaert
Development status	Active
Γράφει σε	PHP
Λειτουργικό σύστημα	Cross-platform
Πρώτη έκδοση	Ιανουάριος 2001
Τύπος	Content management framework , Content management system , Community, Blog software
Άδεια	GNU General Public License Version 2
Ιστοσελίδα	http://drupal.org

1.1.3 Liferay

Τελευταίο CMS στη σύγκριση αλλά και το επίκεντρο της πτυχιακής μας αποτελεί το Liferay Enterprise Portal. Είναι το κυρίαρχο προϊόν



δημιουργίας δικτυακών πύλων σήμερα και πρόκειται για ένα ελεύθερο και ανοικτού κώδικα CMS γραμμένο σε Java και διανέμεται υπό την [GNU Lesser General Public License](#). Το Liferay είναι μια πύλη που είναι σε εξέλιξη από το 2000 από τον [Chief Software Architect](#) Brian Chan.

Το Liferay Portal χρησιμοποιώντας τεχνολογίες αιχμής (Java, J2EE, Web 2.0) παρέχει ένα σύνολο καινοτόμων χαρακτηριστικών που βελτιώνουν τη δημιουργία μιας επιτυχημένης δικτυακής πύλης και είναι κατάλληλο τόσο για δημόσια έργα όσο και για ιδιωτικά.

Βασικά Χαρακτηριστικά

- Συμβατό με τους περισσότερους application servers, βάσεις δεδομένων και λειτουργικά συστήματα, με πάνω από 700 δυνατούς συνδυασμούς ανάπτυξης
- Ενσωματωμένο πρόγραμμα διαχείρισης εγγράφων (CMS)
- Προηγμένη ευχρηστία με πάνω από 60 προεγκατεστημένα portlets
- Χαρακτηρίζεται ως μια από τις πιο ασφαλείς πλατφόρμες ανάπτυξης portals

Το Liferay χρησιμοποιεί ένα πλούσιο σύνολο διεπαφών χρήστη κατασκευασμένο με τεχνολογία Ajax για την παροχή ευκολιών όπως επανατοποθέτηση των portlets με απλό drag-and-drop, δυναμικό φορτωτή portlets και διαμορφωτή στυλ που επιτρέπει την παραμετροποίηση χρωμάτων, γραμματοσειρών και συνδέσμων χωρίς την επεξεργασία stylesheets ή HTML. Οι αλλαγές και οι ρυθμίσεις πραγματοποιούνται με ένα κλικ μέσω μιας αρχιτεκτονικής θεμάτων που επιτρέπει στους διαχειριστές της πύλης να θέτουν σε λειτουργία νέες γραφικές διεπαφές που θα μπορούν να επιλεγούν από τους τελικούς χρήστες χωρίς να απαιτείται τροποποίηση του κεντρικού κώδικα. Μια

σειρά τέτοιων θεμάτων μπορεί να προστίθεται άμεσα και εύκολα μέσω του ενσωματωμένου διαχειριστή ενημερώσεων λογισμικού.

Επιπλέον τα portlets είναι άμεσα διαθέσιμα με τη φόρτωσή τους έτσι ώστε ο χρήστης να μην χρειάζεται να αναμένει τη φόρτωση μιας ολόκληρης ιστοσελίδας, πράγμα πολύ πρακτικό.

Παρακάτω παρουσιάζεται ένας συγκεντρωτικός πίνακας των χαρακτηριστικών του συγκεκριμένου CMS.

Προγραμματιστής (-ες)	Liferay, Inc
Development status	Active
Γράφει σε	Java
Λειτουργικό σύστημα	Cross-platform
Μέγεθος	6,7 MB (συμπιεσμένο) 17,7 MB (χωρίς συμπίεση)
Τύπος	Σύστημα Διαχείρισης Περιεχομένου
Άδεια	LGPL License
Ιστοσελίδα	www.liferay.com

1.1.4 Συμπεράσματα για το CMS

Το τελευταίο διάστημα υπάρχει μεγάλος ανταγωνισμός στα συστήματα διαχείρισης περιεχομένου ανοιχτού κώδικα, με πολλές αξιολογες προτάσεις. Η επιλογή δεν είναι εύκολη, κάθε εφαρμογή έχει πλεονεκτήματα και μειονεκτήματα, ενώ μεγάλο ρόλο στην τελική απόφαση παίζουν οι απαιτούμενες προδιαγραφές των υπό κατασκευή websites. Έτσι, κρίναμε αναγκαίο να κάνουμε μία σύγκριση μεταξύ των 3 διαφορετικών συστημάτων όσον αφορά τα τεχνικά τους χαρακτηριστικά.

Το Drupal είναι κατάλληλο για σύνθετες εγκαταστάσεις διαχείρισης περιεχομένου και κάθετες εφαρμογές. Η εξατομικευμένη φύση των θεματικών παραλλαγών του και η σύνθετη και ευέλικτη αρχιτεκτονική του, απαιτεί εξειδικευμένους σχεδιαστές και προγραμματιστές. Το Drupal, μετά από επαγγελματική προσαρμογή, παραμένει το πιο εργονομικό και εύχρηστο σύστημα για τον τελικό χρήστη. Η Joomla είναι κατάλληλη για απλούστερες εγκαταστάσεις με έτοιμες θεματικές παραλλαγές. Η απλή φύση της Joomla την έχει καταστήσει πολύ δημοφιλή στις ελληνικές εταιρείες κατασκευής διαφημιστικών ιστοσελίδων. Οι εγκαταστάσεις Joomla δεν απαιτούν γνώσεις σχεδίασης ή προγραμματισμού.

System Requirements	Drupal	Joomla!	Liferay
Application Server	Apache	CGI	J2EE
Database	MySQL PostgreSQL	MySQL	All types
License	Open Source	Open Source	Open Source
Programming Language	PHP	PHP	Java
Web Server	Apache	Apache	Apache
Drag-N-Drop Content	Free Add On	Όχι	Ναι
FTP Support	Limited	Ναι	Limited
Metadata	Ναι	Ναι	Ναι

Όπως φαίνεται από την παραπάνω σύγκριση, τα συστήματα διαχείρισης περιεχομένου (CMS) είναι συστήματα με πολλές και ποικίλες δυνατότητες. Σύμφωνα λοιπόν με τις ανάγκες των εφαρμογών τις οποίες υλοποιούμε στα πλαίσια αυτής της πτυχιακής, το πιο κατάλληλο δείχνει να είναι το Liferay κυρίως λόγω του ότι χρησιμοποιεί ως γλώσσα προγραμματισμού την Java.

Πρόκειται για ένα αυτοματοποιημένο εργαλείο διαχείρισης περιεχομένου που χρησιμοποιείται για την δημιουργία και την ανάπτυξη δικτυακών πύλων, καθώς και εσωτερικών ή διαδικτυακών κάθετων εφαρμογών, που απαιτούν την ύπαρξη αναβαθμισμένων συστημάτων διαχείρισης πολύμορφου περιεχομένου. Η επιλογή του στηρίζεται στο γεγονός ότι επιτρέπει στους ανεξάρτητους χρήστες του να το χρησιμοποιούν ελεύθερα, με την χρήση ενός απλού browser, χωρίς να είναι απαραίτητη η εγκατάσταση και λειτουργία κάποιου επιπρόσθετου λογισμικού.

Το λογισμικό αυτό είναι κατάλληλα σχεδιασμένο και υλοποιημένο ώστε να προσφέρει προηγμένους μηχανισμούς διαχείρισης περιεχομένου και να επιτρέπει την ανάπτυξη ηλεκτρονικών διαδικτυακών πύλων. Το σύστημα Liferay προσφέρει ανεπτυγμένες και σύγχρονες υπηρεσίες υποστήριξης.

2 Ανάπτυξη templates

2.1 Γλώσσες Προγραμματισμού

2.1.1 HTML

Τα αρχικά **HTML** προέρχονται από τις λέξεις **HyperText Markup Language**. Η HTML είναι μια γλώσσα σήμανσης (*markup language*), δηλαδή ένας ειδικός τρόπος γραφής κειμένου και όχι μια γλώσσα προγραμματισμού όπως συχνά θεωρείται. Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου. Επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Αξίζει να σημειωθεί ότι η HTML είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας.



Ο browser παίρνει τις πληροφορίες από τον Web server, τις μορφοποιεί και τις εμφανίζει κατάλληλα. Διαφορετικοί browsers μπορεί να μορφοποιούν και να εμφανίζουν το ίδιο αρχείο με διαφορετικό τρόπο, πράγμα που γίνεται συχνά, ανάλογα με τις δυνατότητες του συστήματος στο οποίο τρέχουν και τις επιλογές διαμόρφωσης του προγράμματος του browser.

Η HTML ορίζει ένα σύνολο κοινών στυλ για τις Web σελίδες όπως:

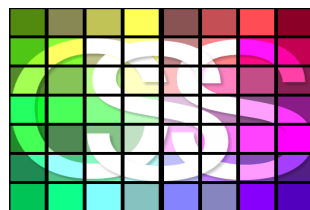
- τίτλοι (titles)
- επικεφαλίδες (headings)
- παράγραφοι (paragraphs)
- λίστες (lists)
- πίνακες (tables).

Ορίζει επίσης στολ χαρακτήρων, όπως η έντονη γραφή (boldface) και οι ενότητες κώδικα. Κάθε στοιχείο έχει ένα όνομα και περιέχεται μέσα στα σύμβολα < >, που αποκαλούνται tags (ετικέτες). Όταν γράφουμε μια Web σελίδα με την HTML, στην ουσία δίνουμε τίτλους στα διάφορα στοιχεία της σελίδας μ' αυτά τα tags. Όταν διαβάζουμε μια σελίδα γραμμένη με την HTML σ' έναν browser, εκείνος «διαβάζει» τα tags της HTML και μορφοποιεί το κείμενο και τις εικόνες στην οθόνη.

Όσον αφορά τη δομή της HTML, οι σελίδες που γράφουμε είναι απλά αρχεία κειμένου σε μορφή ASCII.

2.1.2 CSS

Το CSS προέρχεται από τα ακρωνύμια των **Cascading Style Sheets** και είναι μια απλή γλώσσα που μας βοηθάει να ορίσουμε με σαφή τρόπο και ιδιαίτερη ευελιξία πως θα εμφανίζονται τα διάφορα στοιχεία στην ιστοσελίδα μας. Για



παράδειγμα γράφοντας τις σελίδες μας με HTML κώδικα, μπορούμε να ορίσουμε κάποια χαρακτηριστικά των στοιχείων της σελίδας μας όπως το χρώμα και το μέγεθος του κειμένου αλλά και άλλων όπως πίνακες, links, λίστες κτλ. Για να αλλάξουμε το χρώμα κάποιου κειμένου ή το χρώμα ενός πίνακα, θα πρέπει να βρούμε το χρώμα αυτό μέσα στον κώδικα και να το αλλάξουμε. Όταν έχουμε να διαχειριστούμε μια μόνο σελίδα, η διαδικασία αυτή μπορεί να είναι εύκολη, αλλά ένα site αποτελείται από δεκάδες σελίδες τις οποίες χρειάζεται να διαχειριζόμαστε με ευκολία και ταχύτητα. Έτσι με την χρήση CSS μπορούν εύκολα και οργανωμένα σε στολ να ορισθούν τα χρώματα και τα μεγέθη και έπειτα να εφαρμοσθούν τα στολ αυτά στα στοιχεία των σελίδων του site μας χωρίς να αλλάζει η δομή του κώδικα.

Ένα άλλο σημαντικό πλεονέκτημα της CSS στις σελίδες είναι ο «καθαρότερος» κώδικας, χωρίς πολλές ιδιότητες στις ετικέτες οι οποίες τον κάνουν δυσανάγνωστο. Επίσης καθώς το αρχείο, μέσα στο οποίο ορίζονται τα στολ, "διαβάζεται" από τον browser μόνο μια φορά και έπειτα αποθηκεύεται στην cache memory, μειώνεται έτσι το

μέγεθος της πληροφορίας που γίνεται download από τους browsers και γίνεται γρηγορότερη η πλοήγηση.

Γενικότερα η χρήση CSS σε σχέση με την HTML έδωσε την δυνατότητα για :

- Πολύ μεγάλη ευελιξία, καθώς το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την HTML.
- Καθώς η εμφάνιση ενός site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS και κάθε αλλαγή στο στυλ μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, χωρίς να χρειάζεται να επεμβαίνουμε στον κώδικα κάθε σελίδας ξεχωριστά η συντήρηση των ιστοσελίδων γίνεται με μεγαλύτερη ταχύτητα και ευελιξία.
- Παράγεται αρχείο μικρότερου μεγέθους, δεδομένου ότι ο κάθε κανόνας μορφοποίησης που θα χρησιμοποιηθεί γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.
- Εφόσον υπάρχει διαχωρισμός μεταξύ του περιεχομένου και της μορφοποίησης του, οι μηχανές αναζήτησης δεν «χάνονται» ανάμεσα τους, αλλά έχουν πρόσβαση στο καθαρό περιεχόμενο, οπότε είναι ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν.
- Το αρχείο CSS μπορούμε να το ενσωματώσουμε εντός του κώδικα ή να τα χρησιμοποιούμε σαν εξωτερικό αρχείο. Στην περίπτωση του εξωτερικού αρχείου, ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site το αποθηκεύει στην μνήμη (cache), και κάθε φορά που ο χρήστης του κατεβάζει κάποια άλλη σελίδα του site μας δεν χρειάζεται να το κατεβάσει ξανά και ξανά. Αυτό είναι πολύ σημαντικό καθώς οι σελίδες είναι «γρηγορότερες».

Παρακάτω θα αναφερθούμε και στους κανόνες σύνταξης. Η CSS αποτελείται από κανόνες οι οποίοι αφορούν το **τι** θα μορφοποιηθεί και τους επιλογείς (CSS selectors) οι οποίοι αφορούν το **πως** θα μορφοποιηθεί αυτό.

Η σύνταξη είναι η εξής:

επιλογέας

```
{ ιδιότητα1: τιμή1;  
  ιδιότητα2: τιμή2;  
  ιδιότητα3: τιμή3; }
```

Μέσω των CSS selectors επιλέγονται τα στοιχεία που θα μορφοποιήσουμε κάθε φορά. Ουσιαστικά αποτελούν έναν αριθμό «κανόνων» ώστε να καταλάβει ο browser που θέλουμε να εφαρμοστούν τα όσα γράφουμε.

Εδώ θα εξετάσουμε τους πιο βασικούς selectors:

➤ **.όνομα_κλάσης**

Όταν ο selector στην αρχή του περιλαμβάνει τελεία (.). Το χρησιμοποιούμε σε στοιχεία στα οποία θέλουμε να έχουν όλα την ίδια μορφοποίηση, ενώ θα εμφανίζονται όσες φορές θέλουμε στην σελίδα μας. Για παράδειγμα, ο παρακάτω κανόνας CSS:

```
.container {  
  overflow: hidden;  
  width: 2000px;  
  margin: 0 auto;  
}
```

➤ #όνομα_id

Όταν ο selector στην αρχή του περιλαμβάνει ένα χαρακτήρα δίσωσης (#). Πρόκειται για τον selector που χρησιμοποιείται συχνότερα απ' όλους. Κάθε document μπορεί να έχει μόνο μία και μοναδική τιμή για κάθε attribute. Κανένα element δεν μπορεί να έχει την ίδια id τιμή με κάποιο άλλο, κάνοντας το με αυτόν τον τρόπο μοναδικό. Μπορούμε λοιπόν να το επιλέξουμε γράφοντας απλά την τιμή που έχει στο id του με μία δίσωση (#) μπροστά, προσθέτοντας εάν θέλουμε, και τον τύπο του element πριν από την δίσωση.

```
#theTabs{  
  
    float: left;  
  
    list-style-type: none;  
  
    border-left: 1px solid #999;  
  
    width: 80%; }
```

Ψευδό κλάσεις και ψευδό στοιχεία:

➤ στοιχείο: hover

Από τις πιο συχνά χρησιμοποιούμενες. Αφορά τα στοιχεία τύπου <στοιχείο> τη στιγμή που ο χρήστης έχει το δείκτη του ποντικιού πάνω σε κάποιο από αυτά (χωρίς να πατάει κάποιο πλήκτρο). Μπορεί να μας βοηθήσει να δημιουργήσουμε διάφορα όμορφα εφέ, τα οποία παλιότερα ήταν εφικτά μόνο με JavaScript.

```
.image_thumb ul li.hover {  
  
    list-style-type: none;  
  
    background: #ddd;  
  
    cursor: pointer; }
```


Μπορούμε να εισάγουμε CSS στη σελίδα μας με τρεις διαφορετικούς τρόπους, αναλόγως την περίπτωση. Οι τρόποι αυτοί, κατά σειρά φθίνουσα προτεραιότητας, είναι:

➤ **Σε μια γραμμή CSS**

Αν επιθυμούμε να μορφοποιήσουμε ένα στοιχείο μόνο, και δεν πρόκειται να χρειαστούμε αυτό το είδος μορφοποίησης για τίποτε άλλο στο site, μπορούμε να γράψουμε «χόμα» CSS properties μέσα στο attribute style, το οποίο το διαθέτει σχεδόν κάθε στοιχείο HTML. Για παράδειγμα, για να κάνουμε μια συγκεκριμένη παράγραφο κόκκινη, μπορούμε να προσθέσουμε `<p style="color: red">this is red text</p>`.

➤ **CSS για μια συγκεκριμένη σελίδα**

Πολλές φορές, μπορεί να θέλουμε να εφαρμόσουμε κάποιους κανόνες CSS μόνο για μια συγκεκριμένη σελίδα και όχι για όλο το site. Ένας τρόπος να το κάνουμε αυτό, για να μην δημιουργήσουμε ξεχωριστό αρχείο CSS είναι να εισάγουμε εντός των tags `<head>...</head>` της σελίδας τους κανόνες CSS μέσα σε `<style>` tags (με την html attribute type τους σε text/ css). Για παράδειγμα, για να κάνουμε το χρώμα γραμμάτων μιας συγκεκριμένης σελίδας γκρι, θα γράφαμε στο head της:

```
<style type= "text/css"> body { color:gray; } </style>
```

➤ **Εξωτερικό αρχείο CSS**

Η πιο «σωστή» χρήση του CSS, είναι η χρήση εξωτερικού αρχείου CSS. Για να τη χρησιμοποιήσουμε, γράφουμε τους κανόνες CSS μας σε ένα αρχείο .css (πχ main.css) και στο head της κάθε σελίδας του site μας γράφουμε:

```
<link href="main.css" type="text/css" />
```

➤ Ιδιότητες που αφορούν τη **μορφοποίηση κειμένου**:

- color
- font size
- font family
- font style
- font weight
- text decoration
- text align
- font

➤ Ιδιότητες που αφορούν το **φόντο**:

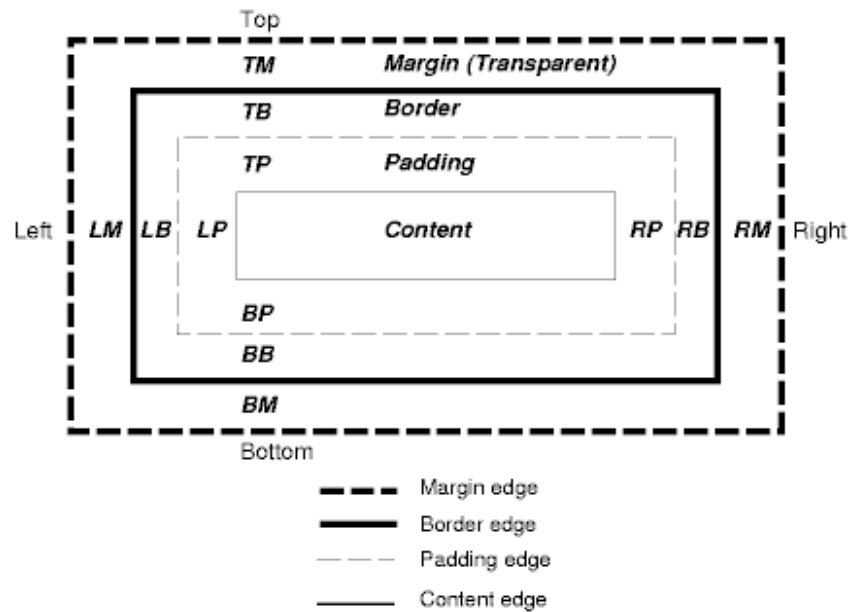
- background color
- background image
- background repeat
- background position

➤ Ιδιότητες που αφορούν το **περίγραμμα**:

- border color
- border width
- border style

➤ Διάφορες χρήσιμες ιδιότητες:

- padding
- margin
- width, height
- float



Εικόνα 1: Χαρακτηριστικά στοιχεία εικόνας

2.1.3 XML

Στις σημερινές απαιτήσεις των χρηστών από το διαδίκτυο συμπεριλαμβάνεται και η φύση των διαθέσιμων εγγράφων. Πρέπει να είναι εύκολα προσβάσιμα, μεταφέριμα και ευέλικτα. Πρέπει επίσης να είναι ανεξάρτητα οποιουδήποτε συστήματος και περιεχομένου. Η γλώσσα XML αναπτύχθηκε από μια Ομάδα Εργασίας της XML κάτω από την καλή κηδεμονία του διεθνούς οργανισμού World Wide Web Consortium (W3C) το 1996. Έλυσε πολλά από τα προβλήματα που αντιμετώπισαν οι σχεδιαστές του web και είναι υπεύθυνη για την XHTML, μια ανασχεδιασμένη HTML.

<?xml?>

Η XML σχεδιάστηκε με σκοπό να ικανοποιήσει πολλές ανάγκες δίνοντας στα έγγραφα ένα μεγαλύτερο επίπεδο προσαρμοστικότητας στο στυλ και τη δομή από αυτό που υπήρχε παλαιότερα στην HTML. Δεν πρόκειται για μια γλώσσα προγραμματισμού οπότε και δεν χρειάζεται, να είναι κάποιος προγραμματιστής για να την χρησιμοποιήσει ή να τη μάθει.

Στην πραγματικότητα, η XML είναι markup γλώσσα για έγγραφα που περιέχουν δομημένες πληροφορίες. Markup γλώσσα είναι ένας μηχανισμός που καθορίζει δομές σε ένα έγγραφο. Για να είμαστε ακριβείς είναι κάτι περισσότερο από μια markup γλώσσα, είναι metalanguage, δηλαδή μια γλώσσα που χρησιμοποιείται για να καθορίσει νέες markup γλώσσες. Η XML συμπληρώνει και δεν αντικαθιστά την HTML και ενώ η HTML χρησιμοποιείται στη διατύπωση και την εμφάνιση των δεδομένων, η XML αναπαριστά τη συναφή έννοια των δεδομένων.

Η XML αποτελεί ένα σύνολο δεδομένων (data objects) που ονομάζονται XML έγγραφα και περιγράφει τη συμπεριφορά των προγραμμάτων που τα επεξεργάζονται. Τα έγγραφα που είναι γραμμένα σε XML αποτελούνται από «περιεχόμενο» και «σήμανση». Ένα λογισμικό μοντέλο που καλείται επεξεργαστής XML χρησιμοποιείται για να διαβάσει XML έγγραφα και παρέχει πρόσβαση στο περιεχόμενο και τη δομή τους. Η σήμανση μπορεί να είναι:

- Στοιχεία (elements)
- Αναφορές οντοτήτων (entity references)
- Σχόλια (comments), τα οποία αρχίζουν με `<!--` και τελειώνουν με `-->`
- Εντολές επεξεργασίας (processing instructions), οι οποίες αρχίζουν με `<?` και τελειώνουν με `?>`, π.χ. `<?name data ?>`
- Τμήματα CDATA (CDATA sections)
- Ορισμός τύπου εγγράφου (document type declaration)

Οι βασικοί λόγοι για τους οποίους η XML χρησιμοποιείται με μεγάλη επιτυχία είναι οι εξής:

- Είναι απλό κείμενο, το οποίο μπορεί με ευκολία να διαβαστεί από ανθρώπους που δεν έχουν προγραμματιστική εμπειρία και γνώσεις.
- Ασχολείται μόνο με τη σημασία των δεδομένων, και όχι με την επεξεργασία τους.

- Είναι εύκολο να επεξεργαστούμε αρχεία XML, είτε με χρήση βιβλιοθηκών που έχουν δημιουργηθεί για τον σκοπό αυτό, είτε με χρήση της ίδιας της XML (XSL).

Υπάρχουν φυσικά και κάποιες περιπτώσεις όπου η χρήση XML δεν ενδείκνυται για χρήση:

- Η κωδικοποίηση σε XML παράγει μεγάλα αρχεία, αφού όλα τα δεδομένα βρίσκονται μέσα σε ετικέτες.
- Η κωδικοποίηση των δεδομένων σε ένα απλό κείμενο είναι πολύ χρονοβόρα, και πολλές φορές απαγορευτική για επιστημονικές εφαρμογές ή για εφαρμογές όπου η απόδοση είναι ένας πολύ σημαντικός παράγοντας.

2.1.4 XSL, XSLT, XSL-FO

Για τον τρόπο παρουσίασης ενός XML εγγράφου μέσα από μια εφαρμογή, δηλαδή το στυλ και τη μορφοποίησή του, υπάρχουν τεχνολογίες που καθιστούν την επεξεργασία και παρουσίαση αυτή πιο εύκολη. Οι τεχνολογίες αυτές είναι η XSLT και η XSLFO



(XSL Formatting Objects), οι οποίες μαζί με την σύσταση xpath (XML Path Language) υπάγονται στην σύσταση XSL (XML Stylesheet Language) του W3C. Η XSL είναι ιδιαίτερα χρήσιμη είτε όταν είναι απαραίτητο να δημοσιεύσουμε μεγάλο όγκο εγγράφων, είτε να αναδιοργανώσουμε κάποια έγγραφα XML έτσι ώστε να δημιουργήσουμε πίνακες περιεχομένων ή άλλες χρηστικές δομές αναπαράστασης των πληροφοριών. Η σύσταση XSLT αφορά στον μετασχηματισμό (transformation) ενός εγγράφου XML, ενώ η σύσταση XSLFO στην μορφοποίησή του (formatting) για παρουσίαση.

Την ίδια λειτουργία με την XSLFO επιτελεί και η τεχνολογία CSS (Cascading Style Sheets) ωστόσο το CSS είναι πιο δημοφιλής στους κατασκευαστές λογισμικού και παροχών περιεχομένου. Παρακάτω ακολουθεί μια ανάλυση των τριών βασικών τεχνολογιών με τις οποίες επιτυγχάνεται ο μετασχηματισμός και η μορφοποίηση των εγγράφων XML.

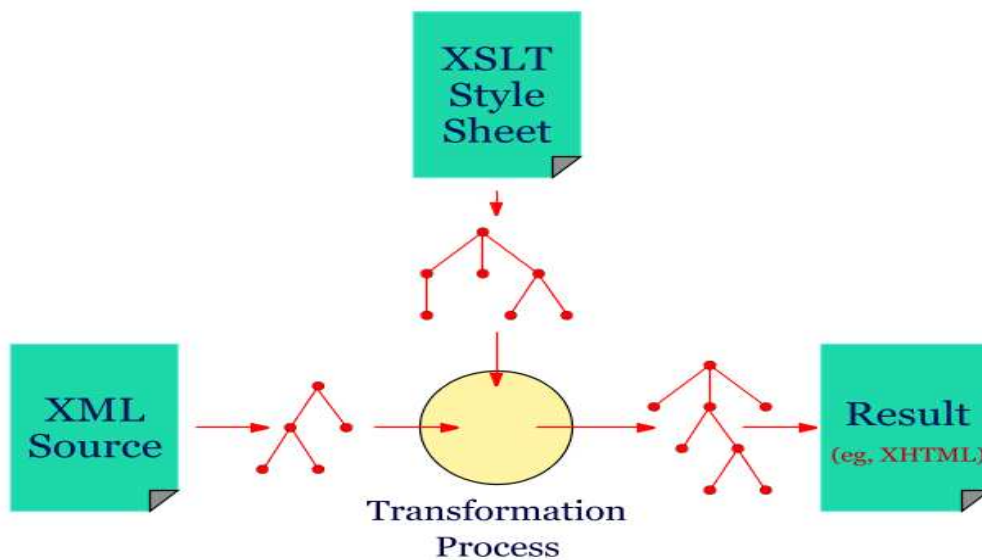
Τα αρχεία XSLT ονομάζονται και XSLT stylesheets και έχουν σύνταξη παρόμοια με αυτή της XML. Κάθε XSLT stylesheet είναι ένα καλά δομημένο XML έγγραφο και βοηθούν πολύ σε κάθε προσπάθεια δημοσίευσης περιεχομένου που υποστηρίζει XML. Ένα χαρακτηριστικό παράδειγμα είναι η δημιουργία της μορφοποίησης ενός website από περιεχόμενο μιας βάσης δεδομένων βασισμένης σε XML. Στην περίπτωση αυτή ένα αρχείο με οδηγίες μορφοποίησης XSLT βοηθά στην μετατροπή του εγγράφου .xml σε ένα έγγραφο .html. Η συνεισφορά του XSLT επεκτείνεται όταν θέλουμε να έχουμε πολλές διαφορετικές εκδόσεις του περιεχομένου ενός website όπως για παράδειγμα αρχεία .html, αρχεία απλού κειμένου .txt, αρχεία για εμφάνιση από φορητές συσκευές μέσω WAP (WML), κ.α.

Το δομικό στοιχείο σε ένα XSLT stylesheet είναι το element `<xsl:stylesheet>`. Διαθέτει σαν υποστοιχεία μια σειρά από elements `<xsl:template>...<xsl:template>`. Μέσα σε αυτά τα `<xsl:template>` elements εκφράζονται οι μετασχηματισμοί που πρέπει να γίνουν στο έγγραφο XML. Κάθε template περιέχει ένα attribute με όνομα `match`. Το `match` είναι μια διαδρομή για το στοιχείο του XML εγγράφου στο οποίο αναφέρεται το template. Παράλληλα με το `match`, κάθε template υπάρχει μια δήλωση `<xsl:apply-templates/>` η οποία είναι μια κλήση που πρέπει να γίνει στο stylesheet κατά την ιεραρχική σάρωση των στοιχείων του πηγαίου XML εγγράφου από τον αναλυτή.

Τα βασικά σύμβολα που χρησιμοποιούνται για την σύνταξη των διαδρομών είναι τα σύμβολα

- / (στοιχείο ρίζα και άμεσος απόγονος)
- // (όλοι οι απόγονοι)
- * (όλοι οι άμεσοι απόγονοι)
- | (επιλογή ενός εκ των δύο στοιχείων που βρίσκονται εκατέρωθεν του συμβόλου).

Οι διαδρομές αντιστοιχίζονται τόσο σε elements όσο και σε attributes του πηγαίου XML εγγράφου. Η διαδρομή που αναφέρεται σε ένα στοιχείο url με attribute το protocol που έχει τιμή mailto γράφεται ως `match="url[@protocol='mailto']"`, δηλαδή το attribute στο οποίο γίνεται η αναφορά μπαίνει σε αγκύλες και πριν από το όνομα του τοποθετείται το σύμβολο.



Εικόνα 2: Μετατροπή εγγράφου XML σε XHTML

2.1.5 Velocity Template

Το Velocity
Template γλώσσας



(VTL), είναι λογισμικό ανοικτού κώδικα. Η σύνταξη και μορφοποίηση ενός template με τη χρήση της VM είναι κατανοητή και παράλληλα γρήγορη σε εφαρμογή. Ακόμη και για την ανάπτυξη μιας ιστοσελίδας από άτομα με μικρή ή καθόλου εμπειρία προγραμματισμού, όπως υπήρξαμε εμείς, δίνει την δυνατότητα σύντομα να είναι σε θέση να χρησιμοποιεί VTL για να ενσωματώσει το δυναμικό περιεχόμενο σε μια ιστοσελίδα.

Η VTL χρησιμοποιεί αναφορές για να ενσωματώσει το δυναμικό περιεχόμενο σε μια ιστοσελίδα, και μια μεταβλητή είναι ένα είδος αναφοράς. Μεταβλητές είναι ένα είδος αναφοράς που μπορεί να αναφέρεται σε κάτι που καθορίζεται στον κώδικα Java, ή ότι μπορεί να πάρει την αξία της από μια VTL δήλωση στην ιστοσελίδα την ίδια. Εδώ είναι ένα παράδειγμα μιας δήλωσης VTL που θα μπορούσε να ενταχθεί σε ένα έγγραφο HTML:

```
#set ($a = "Velocity")
```

Η δήλωση αυτή, όπως και όλες οι δηλώσεις VTL, αρχίζει με το χαρακτήρα # και περιέχει μια οδηγία: σύνολο. Ο χαρακτήρας # ακολουθείται από μια εντολή.

Πρόκειται για ένα εύκολο στη χρήση γενικό εργαλείο templating. Είναι χρήσιμη σε κάθε τομέα εφαρμογή Java η οποία απαιτεί τη μορφοποίηση δεδομένων.

- Προσαρμόζεται σε πολλούς τομείς εφαρμογής
- Προσφέρει έναν απλό, σαφή τρόπο σύνταξη για τον σχεδιαστή προτύπου
- Προσφέρει ένα απλό μοντέλο προγραμματισμού.

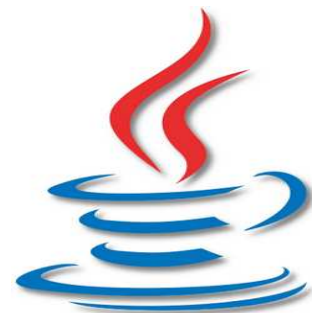
- Επειδή τα πρότυπα της και ο κωδικός είναι ξεχωριστά, μπορούν να αναπτυχθούν και να διατηρηθούν αυτοτελώς
- Ενσωματώνεται εύκολα σε οποιοδήποτε περιβάλλον Java εφαρμογής, ιδιαίτερα στα servlets
- Επιτρέπει την δημιουργία πρότυπων για να έχει πρόσβαση σε οποιαδήποτε δημόσια μέθοδος αντικείμενα δεδομένων στο πλαίσιο

Η Velocity χρησιμοποιείται επιτυχώς σε:

- Σε Servlet που βασίζονται στο Web εφαρμογών
- Σε Java και SQL παραγωγή κώδικα
- Σε XML επεξεργασία και μεταποίηση
- Σε επεξεργασία κειμένου, όπως η παραγωγή αρχείο RTF

2.1.6 JAVASCRIPT

Η Java είναι μια γλώσσα προγραμματισμού, γνωστή σε όλους. Ο προγραμματιστής μπορεί να δημιουργήσει μία και μόνο έκδοση του προγράμματος, που να τρέχει μέσα από ένα μεταφραστή της Java, ένα πρόγραμμα όπως το HotJava, NetBeans ή Netscape Navigator. Έτσι, ένα και μόνο πρόγραμμα Java μπορεί να τρέξει σε πολλά και διαφορετικά λειτουργικά συστήματα, εφόσον ο χρήστης έχει έναν διερμηνευτή που να τρέχει στο σύστημά του. Η JavaScript αποτελεί το πρώτο βήμα στην απλοποίηση της Java. Παρόλο που η Netscape ξεκίνησε την ανάπτυξη της JavaScript αυτόνομα, γρήγορα συνεργάστηκαν με τη Sun, συμφωνώντας να κάνουν τη JavaScript ένα υποσύνολο της Java.



Η JavaScript μοιάζει κατά κάποιο τρόπο με τη Java αλλά είναι πολύ πιο απλή στη χρήση. Παρακάτω στον πίνακα παρουσιάζονται αναφορικά τα βασικά χαρακτηριστικά της.

JavaScript
Σχετικά εύκολη στη χρήση
Δεν χρειάζεστε τίποτε άλλο από Πληροφορίες για το πως να γράψετε σενάρια.
Τα προγράμματα ενσωματώνονται στη σελίδα με τη μορφή σεναρίου.
Η JavaScript τρέχει πολύ γρήγορα.
Η JavaScript είναι πιο κατάλληλη για σχετικά απλές εφαρμογές.
Η JavaScript είναι βασισμένη στο αντικείμενο.
Η JavaScript είναι πιο χαλαρή στη χρήση μεταβλητών.
Η JavaScript χρησιμοποιεί δυναμική σύνδεση. Οι αναφορές στα αντικείμενα ελέγχονται όταν το σενάριο εκτελείται.

Η JavaScript είναι μια γλώσσα συγγραφής σεναρίων που αναπτύχθηκε από την εταιρεία Netscape ώστε να μπορούν οι συγγραφείς ιστοσελίδων να σχεδιάζουν διαδραστικές ιστοσελίδες. Στην ουσία είναι μια επέκταση της γλώσσας HTML καθώς περιέχει τις γνωστές εντολές if, switch, for κ.ά. των γλωσσών προγραμματισμού που δεν περιέχονται στην απλή HTML. Ο κώδικας της JavaScript ενσωματώνεται μέσα στον κώδικα της HTML με τα ειδικά tags:

<script language=javascript> ... </script>

Αυτό σημαίνει ότι η επεξεργασία του κώδικα JavaScript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στον server, αλλά στο πρόγραμμα

περιήγησης των επισκεπτών. Συγκεκριμένα, η JavaScript δεν έχει καμία απαίτηση από πλευράς δυνατοτήτων του server για να εκτελεστεί αλλά βασίζεται στις δυνατότητες του browser. Η JavaScript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Επίσης μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Παρόλα αυτά, οι δυνατότητές της είναι σημαντικά μικρότερες από αυτές της PHP και δεν παρέχει συνδεσιμότητα με βάσεις δεδομένων.

Ο κώδικας JavaScript μιας σελίδας περιλαμβάνεται από τις ετικέτες της HTML `<script language="javascript">` και `</script>` και την απεικόνιση κειμένου αναλαμβάνει η εντολή `document.write`. Για παράδειγμα, ο ακόλουθος κώδικας JavaScript:

```
<script language="javascript">  
document.write('Hello world!');  
</script>
```

Θα εμφανίσει το εξής κείμενο: **Hello world!**


Αν ο κώδικας JavaScript περιέχει περισσότερες από μία εντολές, αυτές θα πρέπει να διαχωριστούν μεταξύ τους με το χαρακτήρα του ελληνικού ερωτηματικού '!'. Βεβαίως οφείλουμε να πούμε πως πλέον η χρήση του χαρακτήρα αυτού για την τελευταία εντολή δεν είναι απαραίτητη μιας και η διαχώριση των εντολών στους νεότερους browsers δεν είναι απαραίτητη.

Γενικότερα με την JavaScript μπορούμε να κάνουμε τα εξής:

- Πολυμερή έγγραφα με πλαίσια
- Επαναφόρτωση μέρους του παραθύρου
- Δημιουργία εγγράφων με αλληλεπίδραση
- Περισσότερο έλεγχο στην αλληλεπίδραση με το χρήστη
- Έγγραφα με μνήμη
- Ζωντανά έγγραφα

- Μηνύματα που ολισθαίνουν
- Ρολόγια
- Χρονικό μηχανισμό αντίστροφης μέτρησης
- Έγγραφα με αυτόματη ενημέρωση

2.1.7 jQuery

Η jQuery πρωτοεμφανίστηκε τον Ιανουάριο του 2006 στο BarCamp από τον  John Resig. Πρόκειται για μια βιβλιοθήκη (framework) JavaScript ανοιχτού κώδικα, υπό τις άδειες MIT License και την GNU General Public License, που χρησιμοποιείται από προγραμματιστές για τη ταχεία ανάπτυξη ιστοσελίδων και διαδικτυακών εφαρμογών που χρειάζονται μεγάλο γνώμονα ευχρηστίας και διαδραστικότητας (interactivity).

Αναρωτηθήκατε αρχικά γιατί η jQuery ονομάστηκε έτσι; Αναγνωρίζοντας το J σαν συντομογραφία της JavaScript αναζητήσαμε την έννοια query. Η έννοια query είναι γνωστότερη στην περιοχή των βάσεων δεδομένων όπου περιγράφει μία ερώτηση, μια αναζήτηση αν θέλετε, του χρήστη ως προς τη βάση δεδομένων με σκοπό να του επιστραφεί το σύνολο των εγγραφών, των στοιχείων δηλαδή της βάσης δεδομένων, που ικανοποιούν το ερώτημα αυτό. Τον ρόλο των queries στην JQuery βιβλιοθήκη παίζουν δύο ομάδες συναρτήσεων που ονομάζονται Selectors και Filters. Παρακάτω γίνεται εκτεταμένη αναφορά και ανάλυση αυτών των εννοιών με στόχο την κατανόηση τους.

Χρησιμοποιώντας την jQuery μπορούμε να κάνουμε πάρα πολλά πράγματα όπως:

- Επιλογή HTML στοιχείων
- Διαμόρφωση HTML στοιχείων
- Διαμόρφωση CSS στοιχείων
- Διεργασίες HTML γεγονότων
- Εφέ JavaScript και animations

- Διαμόρφωση του HTML DOM (Document Object Module)
- Χρήση AJAX

Για να ενσωματώσουμε την βιβλιοθήκη σε ένα html αρχείο, προσθέτουμε στο <head> μέρος της σελίδας μας τον εξής κώδικα:

```
<script type="text/javascript" src="jquery.js"></script>
```

Μπορούμε επίσης να συνδέσουμε το αρχείο με την online έκδοση από την Google με τον εξής τρόπο:

```
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.0/jquery.min.js"> </script>
```

Η σύνταξη είναι πολύ απλή

\$(επιλογή).action()

- Το δολλάριο (\$) είναι το σύμβολο που υποδηλώνει την jQuery
- Ο (επιλογέας) είναι το ερώτημα (query) για το HTML στοιχείο που θέλουμε
- Η δράση (action) είναι ότι θα συμβεί στο στοιχείο που επιλέξαμε

Παραδείγματα

- `$(this).hide()` - Κρύβει το επιλεγμένο αντικείμενο
- `$("p").hide()` - Κρύβει όλες τις παραγράφους
- `$("p.test").show()` - Εμφανίζει όλες τις παραγράφους με class="test"
- `$("#test").show()` - Κρύβει όλα τα στοιχεία με id="test"

Η function Document Ready

Για να διασφαλίσουμε ότι η σελίδα μας έχει φορτώσει πλήρως πριν τρέξουν οι jQuery εντολές μας, χρησιμοποιούμε το εξής function:

```
$(document).ready(function(){  
  
    --εντολές jQuery--  
  
});
```

Έτσι αποφεύγουμε προβλήματα όπως το να προσπαθούμε να κάνουμε επιλογή ενός στοιχείου που δεν έχει φορτώσει ακόμα, ή να πάρουμε το μέγεθος μιας εικόνας που δεν έχει εμφανιστεί πλήρως.

Επιλογέας (selector)

Παραπάνω είδαμε πως στην σύνταξη της jQuery υπάρχει ο επιλογέας, που μας επιτρέπει να ψάξουμε και να βρούμε HTML στοιχεία στην σελίδα μας. Ας δούμε λοιπόν τι ακριβώς σημαίνει αυτό.

Επιλογέας Στοιχείων

Η jQuery μπορεί και χρησιμοποιεί τα ονόματα των CSS στυλ για να επιλέξει HTML στοιχεία. Έτσι:

- Το \$("p") επιλέγει όλα τα στοιχεία <p>
- Το \$("p.intro") επιλέγει όλα τα στοιχεία <p> με class="intro".
- Το \$("p#demo") επιλέγει τα στοιχεία <p> που έχουν id="demo".

Επιλογέας Ιδιοτήτων

Η jQuery μπορεί επίσης να επιλέξει στοιχεία με συγκεκριμένες ιδιότητες. Έτσι:

- Το `$("[href]")` επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα href στην σελίδα μας.
- Το `$("[href='#']")` επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα href και την τιμή value="#".
- Το `$("[href!='#']")` επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα href και επίσης την ιδιότητα <>"#".
- Το `$("[href$='.jpg']")` επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα href και συμπεριέχουν το ".jpg".

Τα γεγονότα της jQuery (Events)

Η jQuery χρησιμοποιεί τα γεγονότα (συμβάντα-events) για να διαχειριστεί τα διάφορα στοιχεία που επιλέγουμε.

Τα βασικά γεγονότα στην jQuery

- `$(document).ready(function)` - Ο έλεγχος ολοκλήρωσης φόρτωσης μιας σελίδας
- `$(selector).click(function)` - Ο έλεγχος για το κλικ σε ένα στοιχείο
- `$(selector).dblclick(function)` - Ο έλεγχος για το διπλό κλικ σε ένα στοιχείο
- `$(selector).focus(function)` - Ο έλεγχος εστίασης σε ένα στοιχείο
- `$(selector).mouseover(function)` - Ο έλεγχος για το hover του ποντικιού σε ένα στοιχείο

Τα εφέ της jQuery αναλυτικά

- `$(selector).hide()` - Απόκρυψη επιλεγμένων στοιχείων
- `$(selector).show()` - Εμφάνιση επιλεγμένων στοιχείων
- `$(selector).toggle()` - Εμφάνιση/ Απόκρυψη επιλεγμένων στοιχείων
- `$(selector).slideDown()` - Slide-down
- `$(selector).slideUp()` - Slide-up
- `$(selector).slideToggle()` - Slide up/Slide down
- `$(selector).fadeIn()` - Εμφάνιση (Fade in)
- `$(selector).fadeOut()` - Απόκρυψη (Fade out)
- `$(selector).fadeTo()` - Απόκρυψη (Fade out) σε συγκεκριμένο opacity
- `$(selector).animation()` - Εφαρμογή animation σε επιλεγμένα στοιχεία

Μπορείτε να δείτε μερικά παραδείγματα από το W3Schools παρακάτω, πατώντας στα ονόματα του κάθε εφέ.

- [jQuery hide\(\)](#) - Ένα απλό εφέ απόκρυψης.
- [jQuery slideToggle\(\)](#) - Ένα απλό slide panel.
- [jQuery fadeTo\(\)](#) - Εφέ ξεθωριάσματος.
- [jQuery animate\(\)](#) - Δημιουργία animation.

3 Liferay CMS

Το CMS το οποίο χρησιμοποιείται από το Liferay παρέχει ιδιαίτερη ευκολία στη δημιουργία, οργάνωση και παρουσίαση περιεχομένου στο διαδίκτυο. Όπως προαναφέραμε χρησιμοποιεί τη Java ως γλώσσα προγραμματισμού. Μπορεί να «τρέξει» σε κάθε λειτουργικό σύστημα και με οποιοδήποτε application server και η ΒΔ που χρησιμοποιείται αφήνεται στην κρίση του χρήστη.



Χαρακτηριστικά

Τα χαρακτηριστικά του CMS Συστήματος Διαχείρισης Περιεχομένου του Liferay παρατίθενται παρακάτω:

- **Δημοσιοποίηση web (Web publishing)**

Το σύστημα δημοσιοποίησης web του Liferay μπορεί να χρησιμοποιηθεί ώστε να δημιουργηθούν ιστοσελίδες γρήγορα, με τη βοήθεια επαναχρησιμοποιούμενου περιεχομένου, ευέλικτων μοτίβων διάταξης και δυναμικά παραγόμενης ταξινόμησης του ιστοχώρου. Τα μοτίβα μπορούν να γραφτούν χρησιμοποιώντας Velocity (VM) ή XSLT.

- **Ευέλικτος μηχανισμός μοτίβων (Flexible Templating Mechanism, XSL/VM)**

Τα μοτίβα που χρησιμοποιούνται για τη δημιουργία των άρθρων μπορούν να παραχθούν είτε με XSL είτε με Velocity (VM), δίνοντας έτσι ευελιξία στο χρήστη στη σχεδίαση.

- **Βιβλιοθήκη εγγράφων (Document Library)**

Η Βιβλιοθήκη Εγγράφων παρέχει μια κεντρική αποθήκευση με εκδόσεις και υπηρεσίες βιβλιοθήκης και στηρίζεται σε JCR-170 μεταγλώττιση (Java Content Repository- Jackrabbit) η οποία αποθηκεύει μια ποικιλία εγγράφων με διαφορετικούς τύπους (π.χ. .PDF, .DOC, .XSL, κ.α.).

- **Έκθεση εικόνων (Image Gallery)**

Όπως και η Βιβλιοθήκη Εγγράφων έτσι και η Έκθεση Εικόνων αποθηκεύει εικόνες που χρησιμοποιούνται στο portal.

- **Δημοσιοποίηση του Portal και Staging (Portal Publishing and Staging)**

Το **Publishing** δίνει τη δυνατότητα να γίνεται επεξεργασία περιεχομένου σε μια «ζωντανή» σελίδα του ιστοχώρου χωρίς να επηρεάζεται αυτό που βλέπει ο χρήστης μέχρι να ολοκληρώσει αυτό που δημιουργεί. Το **Staging** επιτρέπει να υπάρχουν πολλαπλά αντίγραφα τις ίδιας σελίδας και να γίνεται επισκόπηση εκ των προτέρων χωρίς να εμποδίζεται η λειτουργία των «ζωντανών» σελίδων.

- **Υποστήριξη sitemap πρωτοκόλλου (Sitemap Protocol Support)**

Οι νέες σελίδες του ιστοχώρου είναι ανιχνεύσιμες από τις βασικές μηχανές αναζήτησης, καθώς το Liferay υποστηρίζει αυτόματη ενημέρωση της sitemap πληροφορίας επιτρέποντας το web-crawling.

- **Meta Tags**

Το σύστημα ετικετών του Liferay επιτρέπει στον χρήστη να βάζει ετικέτες στο web περιεχόμενο και στα έγγραφα και να δημοσιοποιεί δυναμικά το περιεχόμενο μέσω των ετικετών.

- **Δομημένο Περιεχόμενο (Structured Content)**

Ακόμα και άτομα που δεν έχουν τεχνικές γνώσεις μπορούν να δημιουργήσουν web περιεχόμενο χρησιμοποιώντας ένα από τα WYSIWG (What You See Is What You Get) προγράμματα, αποφεύγοντας έτσι πολύπλοκο κώδικα. Το Liferay διαχωρίζει τη διαδικασία δημιουργίας περιεχομένου σε τρεις περιοχές: δομές, μοτίβα και άρθρα.



Εικόνα 3: Add Application CMS

3.1 Portlet

Το portlet είναι ένα web component που στηρίζεται στην τεχνολογία Java. Το διαχειρίζεται ένας portlet container, ο οποίος επεξεργάζεται αιτήσεις και παράγει δυναμικό περιεχόμενο. Τα portlet χρησιμοποιούνται από τα portal ως pluggable συστατικά διεπαφής χρήστη και παράγουν ένα στρώμα παρουσίασης στα Information Systems.

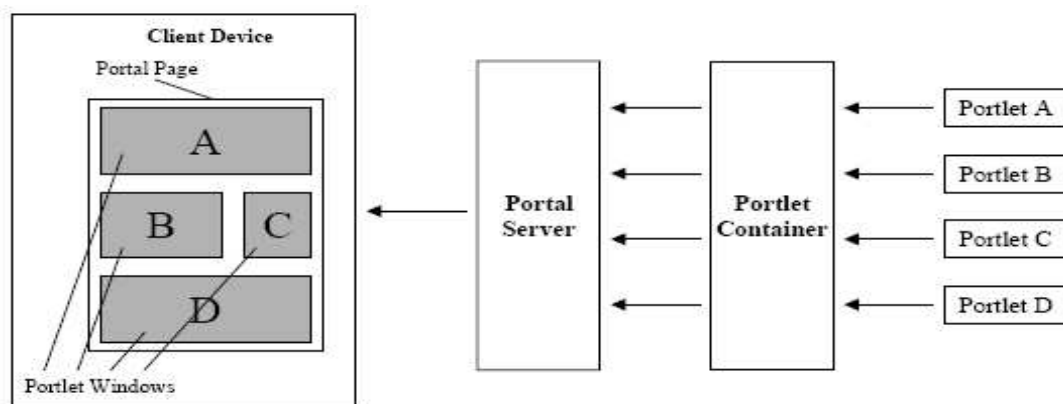
Οι web clients αλληλεπιδρούν με τα portlet μέσω μιας διαδικασίας αίτησης/απάντησης που υλοποιείται από το portal. Κανονικά, οι χρήστες αλληλεπιδρούν με το περιεχόμενο που παράγεται από τα portlet. Το περιεχόμενο αυτό από ένα portlet μπορεί να ποικίλει ανάλογα με τα δικαιώματα πρόσβασης (user configuration) που έχει ένας χρήστης στο συγκεκριμένο portlet.

Ο portlet container αναλαμβάνει να φορτώσει και να εκκινήσει το portlet. Οι δύο αυτές διαδικασίες μπορούν να γίνουν όταν ο portlet container ξεκινήσει την portlet εφαρμογή ή να αναβληθούν ωσότου θεωρηθεί ότι το portlet χρειάζεται για να υπηρετήσει ένα αίτημα.

Αφού γίνει η εκκίνηση, ο portlet container πρέπει να αρχικοποιήσει το portlet για να μπορεί να χειριστεί τα αιτήματα.

3.1.1 Portlet Container

Ο Portlet Container περιέχει portlet, διαχειρίζεται τον κύκλο ζωής τους και παρέχει το απαιτούμενο περιβάλλον εκτέλεσης τους. Επίσης, λαμβάνει αιτήσεις από το portal για να εφαρμόσει request στα portlet τα οποία φιλοξενεί. Δεν είναι υπεύθυνος να αθροίζει το περιεχόμενο που παράγεται από τα portlet, αυτό είναι ευθύνη του portal. Ένα portal και ένας portlet container μπορούν να αποτελέσουν ένα ενιαίο component μιας συλλογής εφαρμογών ή να αποτελέσουν δύο διαφορετικά component μιας portal εφαρμογής.



Εικόνα 4: Portlet container

3.2 Δημιουργία template στο Liferay

Σκοπός ενός template είναι να δώσει τη δυνατότητα στους χρήστες να δημιουργήσουν πολλά και διαφορετικά πρότυπα για μια δομή (structure), έτσι ώστε να υπάρχει μια ποικιλία ως προς τον τρόπο με τον οποίο θα παρουσιάζεται η διάταξη ενός προτύπου. Τα πρότυπα αυτά μπορούν να αναπτυχθούν είτε με XSL, είτε με CSS, είτε με VM δίνοντας έτσι ευελιξία ως προς τον τρόπο με τον οποίο θα γίνει η σχεδίαση.

Ακολουθούμε τρεις διαδικασίες για τη δημιουργία ενός προτύπου:

- **Δημιουργία δομής (structure):**

Καθορίζει τον τύπο και τον αριθμό των στοιχείων που θα λάβουν μέρος σε ένα άρθρο (πχ, κείμενο, εικόνες, κλπ...)

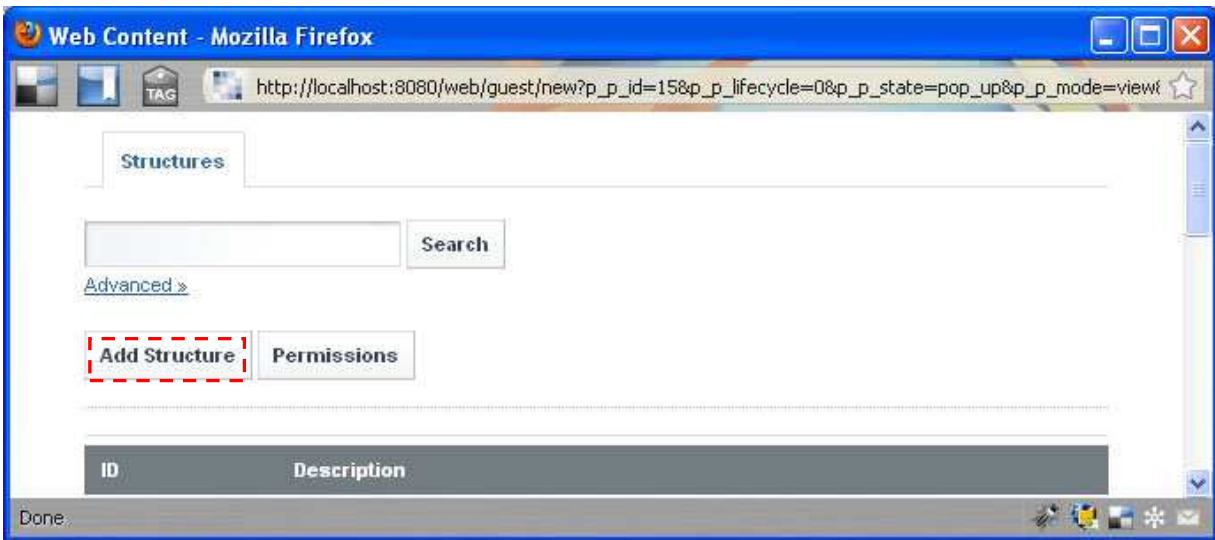
- **Δημιουργία προτύπου (template):**

Ο σκοπός ενός προτύπου (template) είναι να δώσει τη δυνατότητα στους χρήστες να δημιουργούν πολλά διαφορετικά πρότυπα για μια δομή, έτσι ώστε να υπάρχει ποικιλία όσον αφορά την διάταξη (παρουσίαση) του άρθρου.

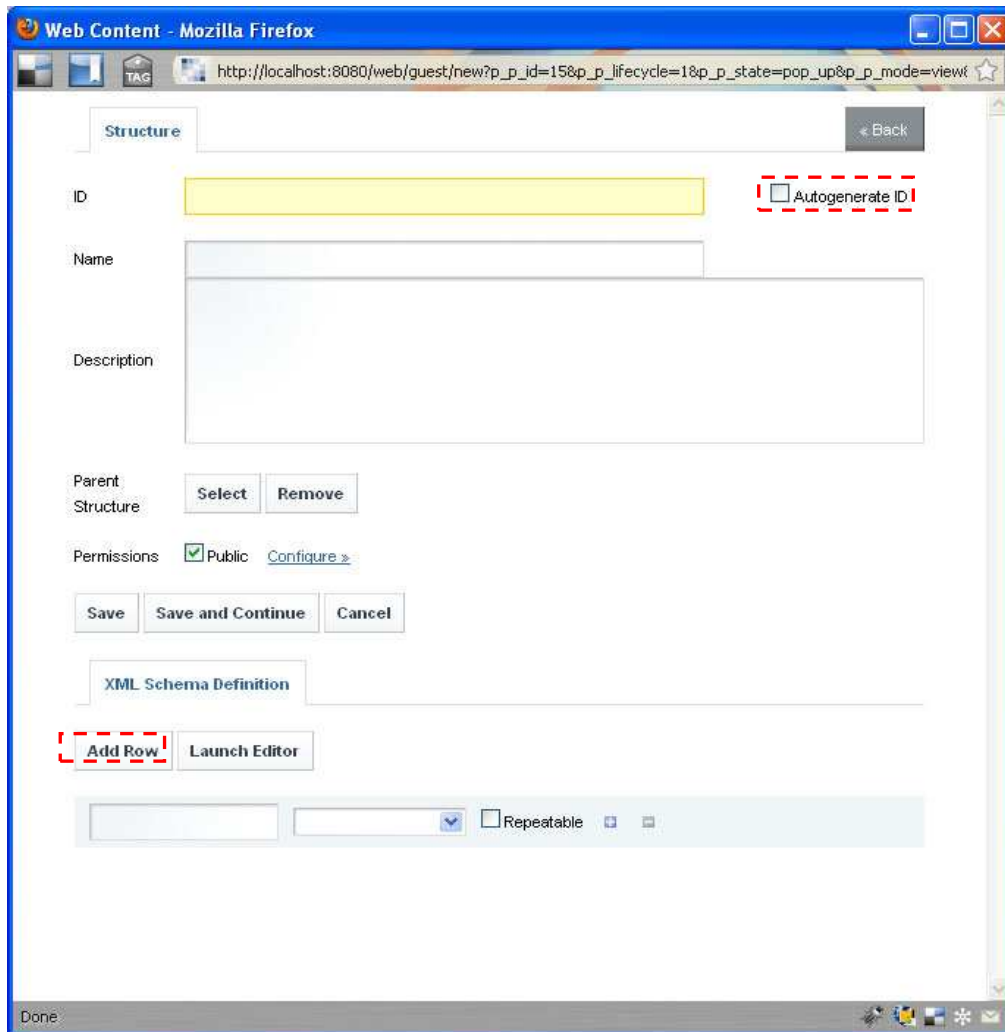
- **Δημιουργία άρθρου (article):**

Σε αυτήν τη φάση δημιουργείται το άρθρο, με βάση τη δομή και το πρότυπο που καθορίστηκε στα προηγούμενα βήματα.

Αρχικά θα πρέπει να δημιουργηθεί η δομή (structure) (Εικόνα 5), επιλέγοντας να ορισθεί αυτόματα ένα ID (Εικόνα 6) για αυτήν και δίνουμε ένα σαφές όνομα και μια περιγραφή όπου σε συνδυασμό με το ID θα ορίσουν μοναδικά το structure. Έπειτα θα ορίσουμε τον αριθμό και τον τύπο των στοιχείων (Εικόνα 6) που θα αποτελούν το structure.

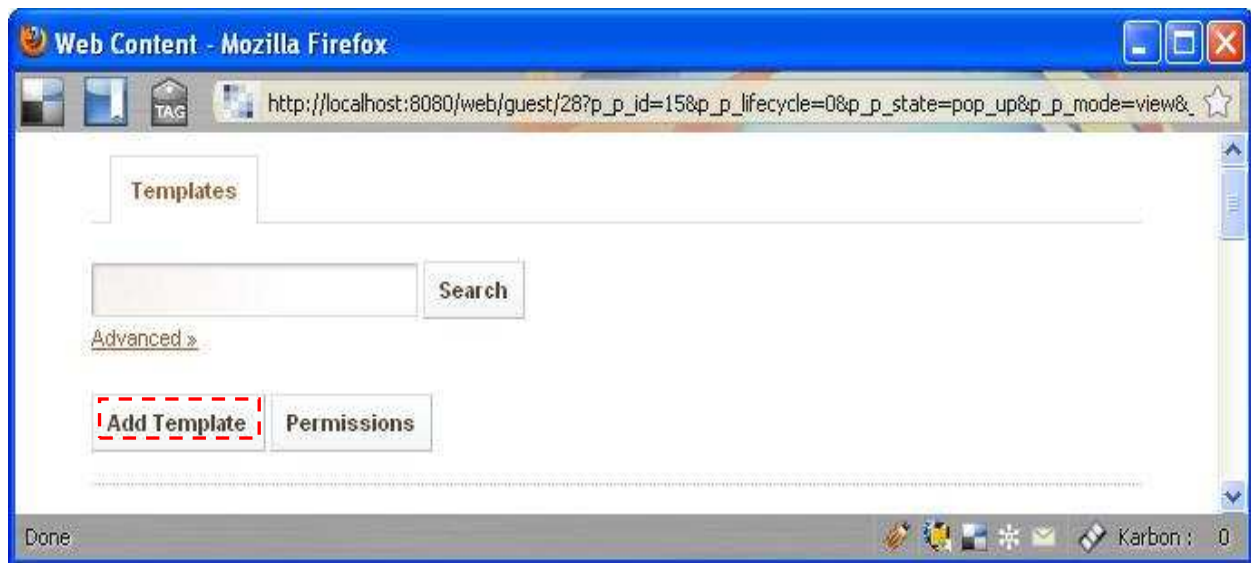


Εικόνα 5: Δημιουργία Structure

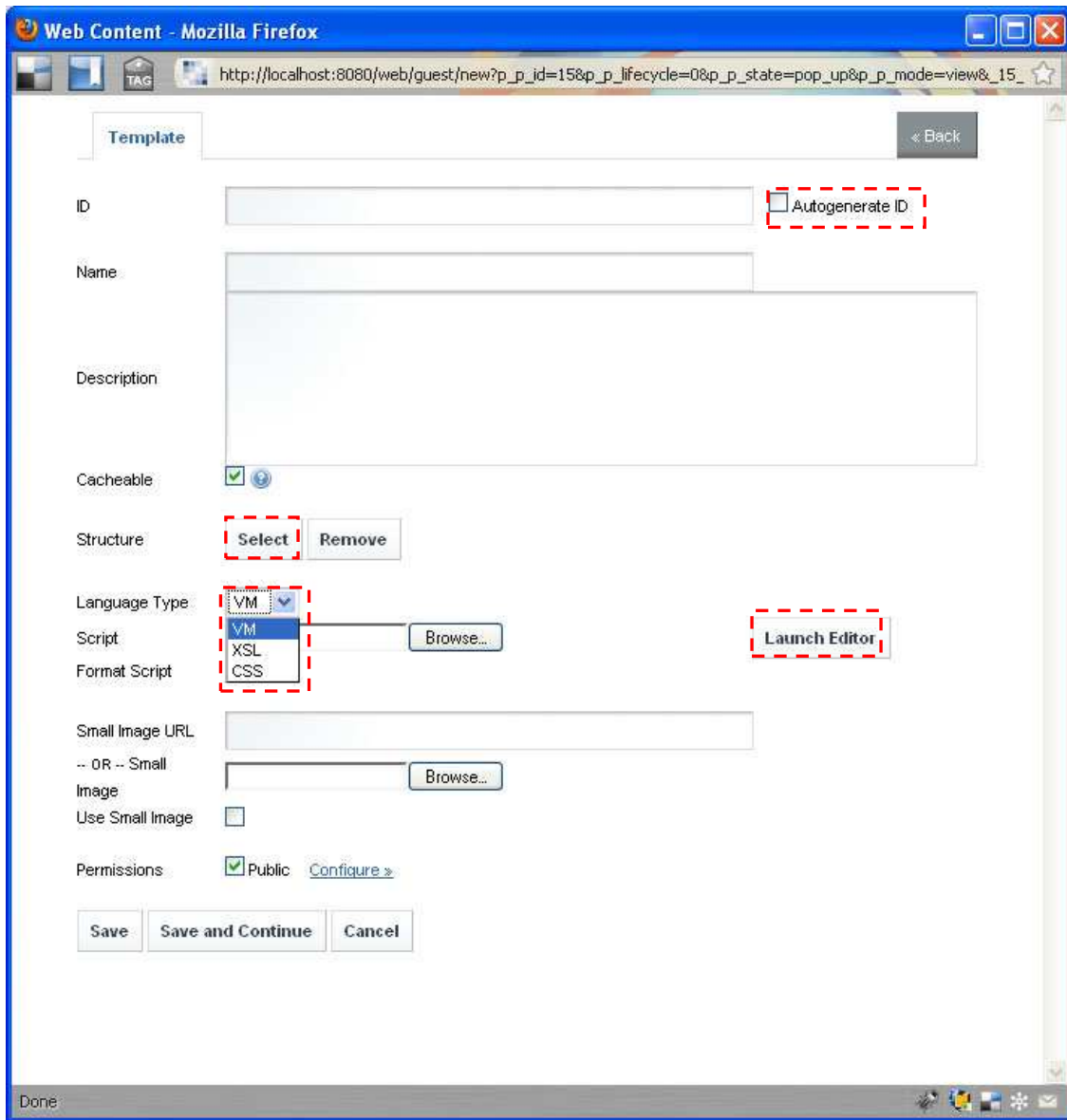


Εικόνα 6: Προσθήκη στοιχείων στο Structure

Αφού δημιουργηθεί η δομή το επόμενο βήμα είναι να δημιουργηθεί το template επιλέγοντας να ορισθεί αυτόματα ένα ID (Εικόνα 8), και για αυτό και δίνουμε ένα σαφές όνομα και μια περιγραφή όπου σε συνδυασμό με το ID θα ορίσουν μοναδικά το template. Εδώ θα πρέπει να βρούμε και να επιλέξουμε το structure μέσω της επιλογής select που δημιουργήσαμε νωρίτερα έτσι ώστε να γίνει η σύνδεση με το template αυτό. Έπειτα επιλέγουμε την γλώσσα, την οποία θα χρησιμοποιήσουμε και πατάμε Launch Editor .



Εικόνα 7: Δημιουργία Template



Εικόνα 8: Launch Editor Template

4 Λεπτομέρειες υλοποίησης - Ειδικά templates

4.1 Πρώτη Επαφή Με Templates

Σε αυτήν την πτυχιακή το πρώτο template το οποίο δημιουργήσαμε έγινε σε γλώσσα XSL. Περιέχει έναν τίτλο (title), το κυρίως κείμενο και μια εικόνα που στοιχίζεται αρχικά στο κέντρο (center), έπειτα δεξιά (right) και στο τέλος αριστερά (left).

The image shows two screenshots from a web application. The top screenshot is titled "Structure" and displays the configuration for a template with ID 10212. It includes fields for Name (containing "image left"), Description, Parent Structure (with "Select" and "Remove" buttons), URL, and WebDAV URL. Below these fields are "Save", "Save and Continue", and "Cancel" buttons. The bottom screenshot is titled "XML Schema Definition" and shows a table with three rows: "title" (Text type), "image" (Image type), and "body" (Text Area (HTML) type). Each row has a "Repeatable" checkbox and navigation icons.

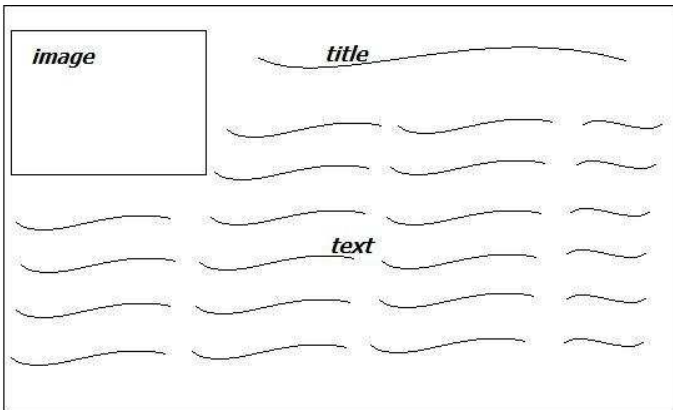
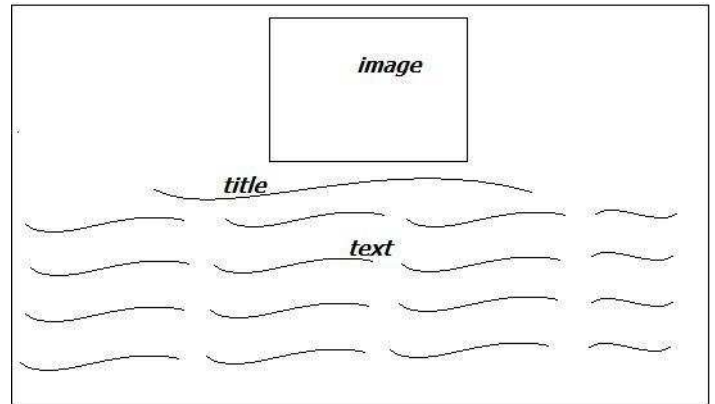
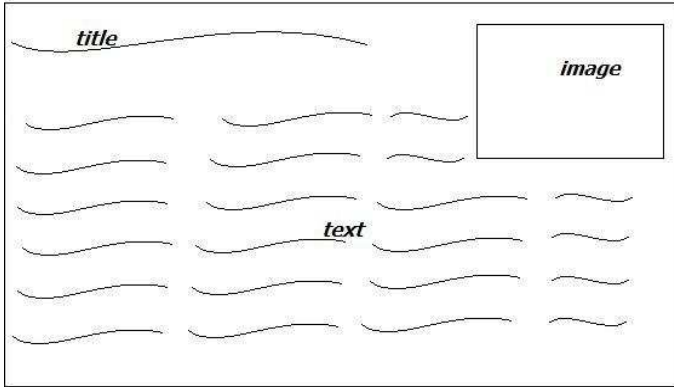
Structure			
ID	10212		
Name	image left		
Description			
Parent Structure	<input type="button" value="Select"/> <input type="button" value="Remove"/>		
URL	http://localhost:8080/c/journal/get_structure?groupId=16&structureId=10212		
WebDAV URL	http://localhost:8080/tunnel-web/secure/webdav/liferay.com/guest/journal		
<input type="button" value="Save"/> <input type="button" value="Save and Continue"/> <input type="button" value="Cancel"/>			

XML Schema Definition			
<input type="button" value="Add Row"/> <input type="button" value="Launch Editor"/> <input type="button" value="Download"/>			
title	Text	<input type="checkbox"/> Repeatable	<input type="button" value="+"/> <input type="button" value="-"/> <input type="button" value="↑"/> <input type="button" value="↓"/>
image	Image	<input type="checkbox"/> Repeatable	<input type="button" value="+"/> <input type="button" value="-"/> <input type="button" value="↑"/> <input type="button" value="↓"/>
body	Text Area (HTML)	<input type="checkbox"/> Repeatable	<input type="button" value="+"/> <input type="button" value="-"/> <input type="button" value="↑"/> <input type="button" value="↓"/>

Εικόνα 9: Structure

Η παραπάνω εικόνα δείχνει τα 3 στοιχεία που απαρτίζουν το template. Ο τίτλος (title) τύπου Text, η εικόνα (image) τύπου Image και το κυρίως «σώμα» τύπου Text Area (HTML).

Παρακάτω παρουσιάζονται το σκαρίφημα που δημιουργήθηκε για να έχουμε το ανάλογο αποτέλεσμα και οι τρεις εναλλαγές τις εικόνας την οποία χρησιμοποιήσαμε σε αυτό.



Εικόνα 10: Σκαρίφημα εικόνας (center, left, right)

Καλώς Ήρθατε στο Δικτυακό Τόπο του Τ.Ε.Ι. Κρήτης

Το ΤΕΙ Κρήτης είναι ένα Ίδρυμα σε δυναμική πορεία ανάπτυξης με άριστες προοπτικές απασχόλησης των αποφοίτων του και συμβολή στην αναπτυξιακή διαδικασία του τόπου. Εκτός από το κεντρικό συγκρότημα στο Ηράκλειο, διαθέτει παραρτήματα σε όλους τους νομούς, σε άλλες πέντε πόλεις της Κρήτης.

Το ΤΕΙ Κρήτης είναι από τα λίγα Ίδρύματα της χώρας που έχει ήδη συμμετάσχει προαιρετικά σε πρόγραμμα αξιολόγησης της ποιότητάς του.

Το νέο θεσμικό πλαίσιο για την Ανώτατη Εκπαίδευση δημιουργεί νέες απαιτήσεις, νέες προκλήσεις και νέες προοπτικές. Στόχος μας είναι σύντομα να καταστεί ένα «επώνυμο» Ίδρυμα στο χώρο της Ανώτατης Εκπαίδευσης.

Το ΤΕΙ Κρήτης είναι από τα πλέον δημοφιλή ιδρύματα μεταξύ των υποψηφίων. Αυτό το επιτυγχάνει με

- ο την ποιότητα των σπουδών και της έρευνας,
- ο τη φροντίδα για καλή ποιότητα ζωής των φοιτητών του
- ο τα αντικείμενα των τμημάτων του που είναι σύγχρονα και καλύπτουν ευρύ φάσμα των ενδιαφερόντων τους.

Άλλωστε η Κρήση, με τον πολιτισμό, την ιστορία της, το φυσικό περιβάλλον και τους φιλόξετους κατοίκους της είναι εξαιρετικά ελκυστική επιλογή!

Καλή πλοήγηση!



Εικόνα 11: Εικόνα στα δεξιά

Καλώς Ήρθατε στο Δικτυακό Τόπο του Τ.Ε.Ι. Κρήτης



Το ΤΕΙ Κρήτης είναι ένα Ίδρυμα σε δυναμική πορεία ανάπτυξης με άριστες προοπτικές απασχόλησης των αποφοίτων του και συμβολή στην αναπτυξιακή διαδικασία του τόπου. Εκτός από το κεντρικό συγκρότημα στο Ηράκλειο, διαθέτει παραρτήματα σε όλους τους νομούς, σε άλλες πέντε πόλεις της Κρήτης.

Το ΤΕΙ Κρήτης είναι από τα λίγα Ίδρύματα της χώρας που έχει ήδη συμμετάσχει προαιρετικά σε πρόγραμμα αξιολόγησης της ποιότητάς του.

Το νέο θεσμικό πλαίσιο για την Ανώτατη Εκπαίδευση δημιουργεί νέες απαιτήσεις, νέες προκλήσεις και νέες προοπτικές. Στόχος μας είναι σύντομα να καταστεί ένα «επώνυμο» Ίδρυμα στο χώρο της Ανώτατης Εκπαίδευσης.

Το ΤΕΙ Κρήτης είναι από τα πλέον δημοφιλή ιδρύματα μεταξύ των υποψηφίων. Αυτό το επιτυγχάνει με

- ο την ποιότητα των σπουδών και της έρευνας,
- ο τη φροντίδα για καλή ποιότητα ζωής των φοιτητών του
- ο τα αντικείμενα των τμημάτων του που είναι σύγχρονα και καλύπτουν ευρύ φάσμα των ενδιαφερόντων τους.

Άλλωστε η Κρήση, με τον πολιτισμό, την ιστορία της, το φυσικό περιβάλλον και τους φιλόξετους κατοίκους της είναι εξαιρετικά ελκυστική επιλογή!

Καλή πλοήγηση!

Εικόνα 12: Εικόνα στα αριστερά

Καλώς Ήρθατε στο Δικτυακό Τόπο του Τ.Ε.Ι. Κρήτης



Το ΤΕΙ Κρήτης είναι ένα Ίδρυμα σε δυναμική πορεία ανάπτυξης με άριστες προοπτικές απασχόλησης των αποφοίτων του και συμβολή στην αναπτυξιακή διαδικασία του τόπου. Εκτός από το κεντρικό συγκρότημα στο Ηράκλειο, διαθέτει παραρτήματα σε όλους τους νομούς, σε άλλες πέντε πόλεις της Κρήτης.

Το ΤΕΙ Κρήτης είναι από τα λίγα Ίδρύματα της χώρας που έχει ήδη συμμετάσχει προαιρετικά σε πρόγραμμα αξιολόγησης της ποιότητάς του.

Το νέο θεσμικό πλαίσιο για την Ανώτατη Εκπαίδευση δημιουργεί νέες απαιτήσεις, νέες προκλήσεις και νέες προοπτικές. Στόχος μας είναι σύντομα να καταστεί ένα «επώνυμο» Ίδρυμα στο χώρο της Ανώτατης Εκπαίδευσης.

Το ΤΕΙ Κρήτης είναι από τα πλέον δημοφιλή ιδρύματα μεταξύ των υποψηφίων. Αυτό το επιτυγχάνει με

- ο την ποιότητα των σπουδών και της έρευνας,
- ο τη φροντίδα για καλή ποιότητα ζωής των φοιτητών του
- ο τα αντικείμενα των τμημάτων του που είναι σύγχρονα και καλύπτουν ευρύ φάσμα των ενδιαφερόντων τους.

Άλλωστε η Κρήση, με τον πολιτισμό, την ιστορία της, το φυσικό περιβάλλον και τους φιλόξετους κατοίκους της είναι εξαιρετικά ελκυστική επιλογή!

Καλή πλοήγηση!

Εικόνα 13: Εικόνα στο κέντρο

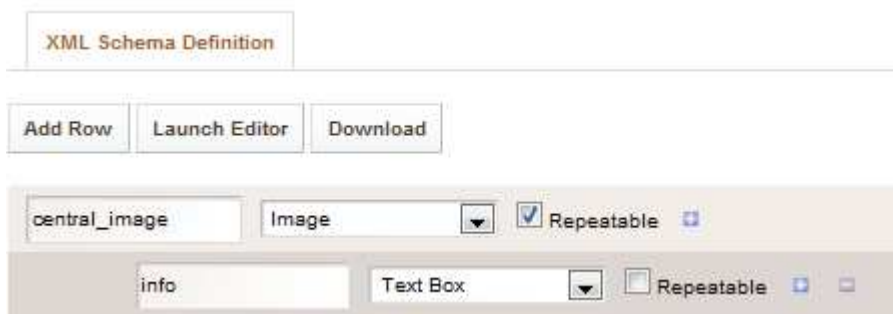
4.2 Ειδικά Templates εκπαιδευτικού περιεχομένου

4.2.1 Body and Galleria Aino

4.2.1.1 Όψη

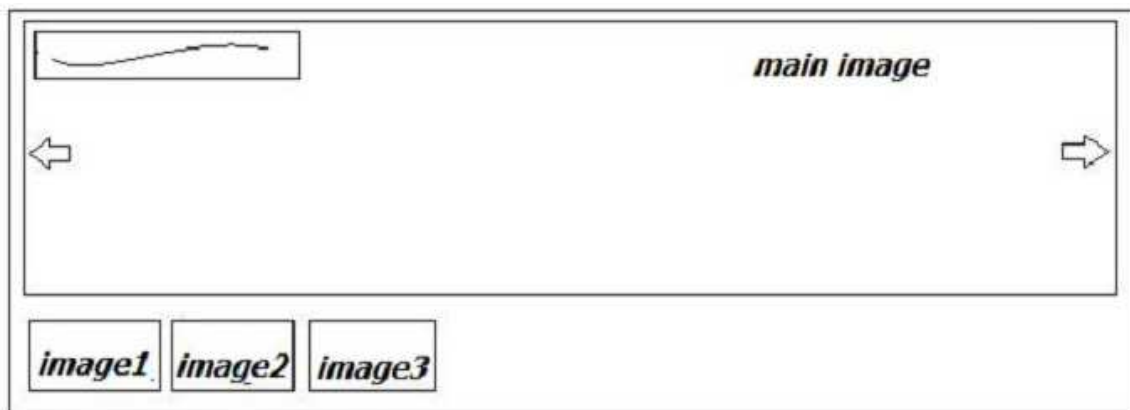
Το template που παρουσιάζεται παρακάτω είναι μια δομή η οποία αποτελείται από δύο περιοχές. Αρχικά δημιουργήθηκε το Galleria Aino αλλά για τη χρήση του στο iSTLab προστέθηκε και το επιπλέον τμήμα που αποτελείται από το Body.

Παρακάτω παρουσιάζεται η αρχική όψη που είχε το πρότυπο πριν την προσθήκη του Body.

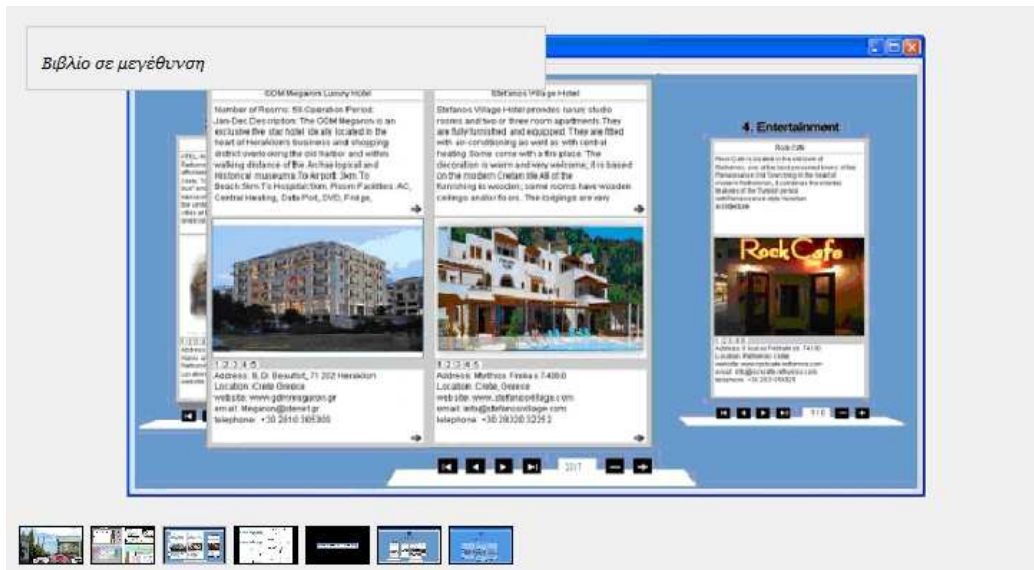


Εικόνα 14: Structure

Η παραπάνω εικόνα δείχνει τα 2 στοιχεία τα οποία απαρτίζουν το template. Το *central_image* τύπου *Image*, το οποίο και είναι επαναλαμβανόμενο, και σαν «παιδί» έχει το *info*, τύπου *Text Box*, για να υπάρχει περιγραφή της εικόνας.



Εικόνα 15: Σκαρίφημα προτύπου



Εικόνα 16: Όψη Template

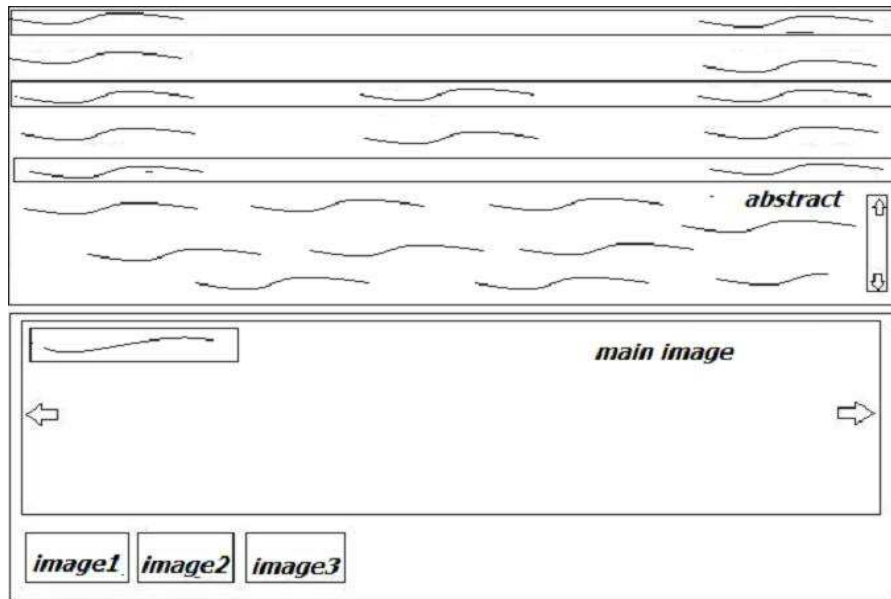
Ακολούθησε η προσθήκη του Body. Εκεί περιλαμβάνεται το κυρίως κείμενο που περιγράφει συνοπτικά την πτυχιακή στην οποία αναφέρεται (ονοματεπώνυμο σπουδαστή, τίτλος κτλ). Η δεύτερη περιοχή περιλαμβάνει ένα open source image gallery, γραμμένο σε JavaScript η οποία περιέχει μια κεντρική εικόνα στην οποία και παρουσιάζεται με γραφικό τρόπο το αντικείμενο της πτυχιακής και στο κάτω μέρος ένα σύνολο από μικρές εικόνες στις οποίες κάνοντας click, εμφανίζονται σαν κεντρική. Παρατηρώντας την κεντρική εικόνα πάνω αριστερά υπάρχει εικονίδιο με το οποίο μπορεί ο χρήστης να δει μια σύντομη περιγραφή και της ίδιας της εικόνας.

Παρακάτω παρουσιάζεται η όψη που θα έχει το πρότυπο αυτό και το Structure.



Εικόνα 17: Structure

Η παραπάνω εικόνα απεικονίζει τα στοιχεία από τα οποία απαρτίζεται το template. Υπάρχει η σειρά (row) main_title η οποία είναι τύπου Text και έχει ως «παιδιά» τα: date, st_name, supervisor, link_ethesis και abstract όλα τύπου Text. Επίσης υπάρχει και η σειρά (row) central_image τύπου Image και είναι επαναλαμβανόμενη (Repeatable). Αυτό δηλώνει ότι έχει τη δυνατότητα επαναχρησιμοποίησης με ακριβώς την ίδια μορφή μέσα στο template. Η σειρά αυτή έχει σαν «παιδί» την info τύπου Text Box. Παρακάτω ακολουθεί το σκαρίφημα του template.



Εικόνα 18: Σκαριφίμα Προτύπου

4.2.1.2 Εφαρμογή

ΤΙΤΛΟΣ ΠΤΥΧΙΑΚΗΣ:		ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΟΥΣΙΑΣΗΣ:
Προγραμματισμός διεπαφών χρήστη-υπολογιστή με βιβλιοθήκες 3D Μελέτη Περίπτωσης Βιβλιοθήκης Java 3D και Εφαρμογή στο eKaNES		2008
ΟΝΟΜΑΤΕΠΩΝΥΜΟ:	ΕΠΙΒΛΕΠΩΝ:	LIBRARY TEI OF CRETE:
Πλεμνός Ανάργυρος	Ακουμιανάκης Δημοσθέης	ETHESIS Full Copy

ΠΕΡΙΛΗΨΗ:

Ο σκοπός της παρούσα πτυχιακής εργασίας είναι να αναφέρει μια πρωτότυπη διεπαφή (αποκαλείται 3DPV) με την χρήση της γραφικής βιβλιοθήκης Java 3D, η οποία υλοποιήθηκε στα πλαίσια του eKaNES και αφορά την αναπαράσταση των τουριστικών πακέτων ως τρισδιάστατα γραφικά σχήματα. Η διεπαφή μας χρησιμοποιεί δυο επίπεδα σχεδιασμού, τα οποία μετατρέπουν τα δεδομένα σε τυποποιημένα δεδομένα (πρώτο επίπεδο) και στην συνέχεια τα τυποποιημένα δεδομένα εκκινούνται σε τρισδιάστατα γραφικά σχήματα (δεύτερο επίπεδο). Η διαφορά της διεπαφής μας με άλλα έργα είναι ότι η εφαρμογή χαρακτηρίζεται με την χρήση μεταφοράς, η οποία χαρακτηρίζει την αντιστοίχιση μεταξύ δεδομένων ενός πεδίου αναφοράς σε σύμβολα ενός πεδίου στόχου. Εκτός από την χρήση της μεταφοράς, η αναπαράσταση των δεδομένων γίνεται και με την χρήση των διαδραστικών αντικείμενων του Prefuse. Με αυτόν τον τρόπο, ο χρήστης εξάγει περισσότερες πληροφορίες διαν η αναπαράσταση των δεδομένων παρουσιάζεται με πολλούς τρόπους και επίσης, γίνονται πιο κατανητά τα δεδομένα διαν εκκινούνται με γραφικό τρόπο χρησιμοποιώντας βιβλιοθήκες όπως η Java 3D και το Prefuse. Τέλος, η εφαρμογή μας περιέχει

Εικόνα 19: Οψη Template

Το template αυτό χρησιμοποιήθηκε για την παρουσίαση των πτυχιακών εργασιών που διεκπεραιώθηκαν στο εργαστήριο του iSTLab.

4.2.2 Image Gallery Rotator

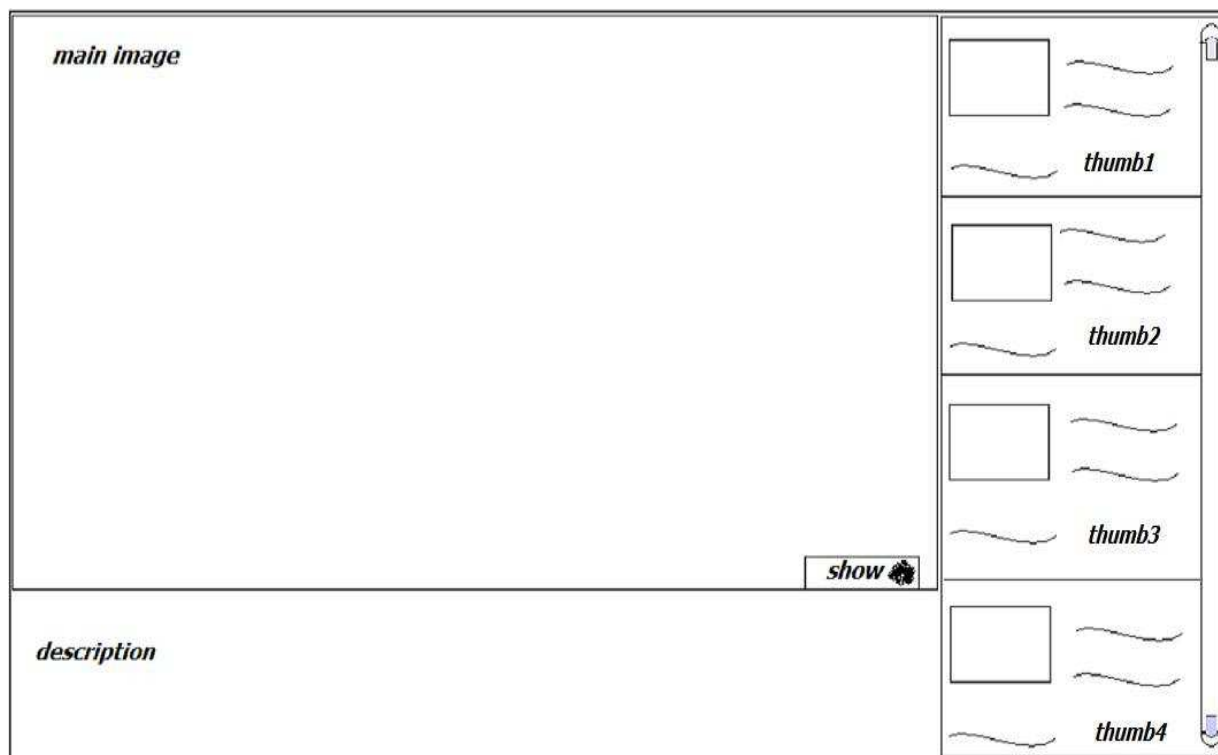
4.2.2.1 Όψη

Το gallery αποτελείται από μια κεντρική εικόνα και ένα menu στα δεξιά. Το menu αποτελείται από μικρότερες εικόνες (thumbs). Κάθε εικόνα χαρακτηρίζεται από ένα μικρο κείμενο με το όνομα του σπουδαστή που έχει διεκπεραιώσει την πτυχιακή και ένα link το οποίο παραπέμπει στο ψηφιακό τεκμήριο στο e Thesis. Στο πλαίσιο της κεντρικής εικόνας υπάρχει η δυνατότητα εμφάνισης και απόκρυψης ενός πλαισίου κειμένου (block) το οποίο περιλαμβάνει μια μικρή περίληψη για την πτυχιακή, με τη χρήση ενός κουμπιού Hide/ Show. Η παρακάτω εικόνα δείχνει τα στοιχεία από τα οποία αποτελείται το template.

The screenshot displays the XML Schema Definition (XSD) editor interface. At the top, there are three buttons: 'Save', 'Save and Continue', and 'Cancel'. Below them is a tab labeled 'XML Schema Definition'. Underneath the tab are three buttons: 'Add Row', 'Launch Editor', and 'Download'. The main area shows a tree structure of elements. The root element is 'central_image', which is of type 'Text' and has a 'Repeatable' checkbox that is unchecked. It contains six child elements: 'image' (Image type, Repeatable unchecked), 'image_thumb' (Image type, Repeatable unchecked), 'descr' (Text Area (HTML) type, Repeatable unchecked), 'txt' (Text Area (HTML) type, Repeatable unchecked), 'link' (Text Area (HTML) type, Repeatable unchecked), and 'rep_image' (Text type, Repeatable checked). The 'rep_image' element also contains six child elements: 'image' (Image type, Repeatable unchecked), 'image_thumb' (Image type, Repeatable unchecked), 'descr' (Text Area (HTML) type, Repeatable unchecked), 'txt' (Text Area (HTML) type, Repeatable unchecked), and 'link' (Text Area (HTML) type, Repeatable unchecked). Each element has a set of control icons: a blue square, a green up arrow, and a green down arrow.

Εικόνα 20: Structure

Έχει 2 σειρές, την *central_image* και την *rep_image*. Οι ονομασίες δίνονται έτσι ώστε να είναι κατανοητές ως προς την επαναληψιμότητά τους. «Παιδιά» και των δύο σειρών και «αδέρφια» μεταξύ τους είναι το *image*, *image_thumb*, *descr*, *txt* και *link*. Παρακάτω ακολουθεί το σκαρίφημα του *template*.



Εικόνα 21: Σκαρίφημα προτύπου

4.2.2.2 Εφαρμογή

Η αξιοποίηση του συγκεκριμένου template παρουσιάζεται παρακάτω:



Εικόνα 22: Όψη Template χωρίς block



Εικόνα 23: Όψη Template με block

4.2.3 Tabs Menu

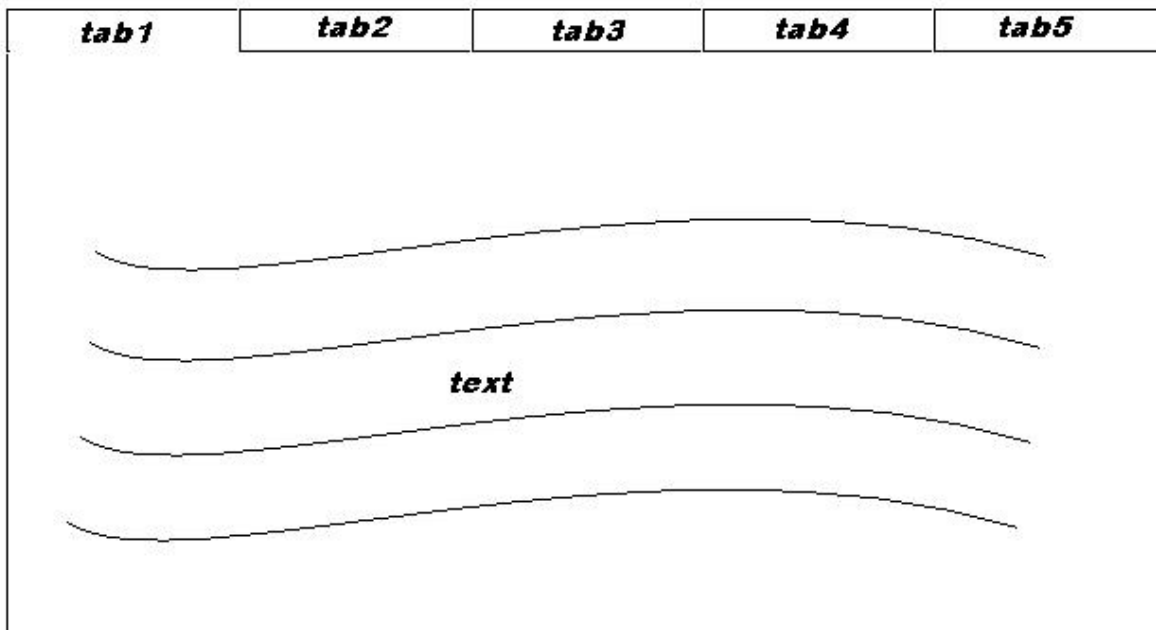
4.2.3.1 Όψη

Το template αποτελείται από 5 καρτέλες (tabs) των οποίων το περιεχόμενο είναι διαφορετικό αλλά μέρος ενός κοινού θέματος (π.χ ενός μαθήματος). Η παρακάτω εικόνα δείχνει τα στοιχεία από τα οποία αποτελείται το template.

The image shows a user interface for editing a template structure. At the top, there are three buttons: 'Save', 'Save and Continue', and 'Cancel'. Below these is a section titled 'XML Schema Definition'. Underneath, there are three more buttons: 'Add Row', 'Launch Editor', and 'Download'. The main part of the interface is a table with five rows, each representing a tab in the menu. The first row is labeled 'tabs_menu' and has a dropdown menu set to 'Text' and a 'Repeatable' checkbox. The subsequent rows are labeled 'id', 'about', 'schedule', 'material', and 'tools'. Each of these rows has a dropdown menu set to 'Text Area (HTML)', a 'Repeatable' checkbox, and two green arrows (one pointing up and one pointing down) for moving the row up or down.

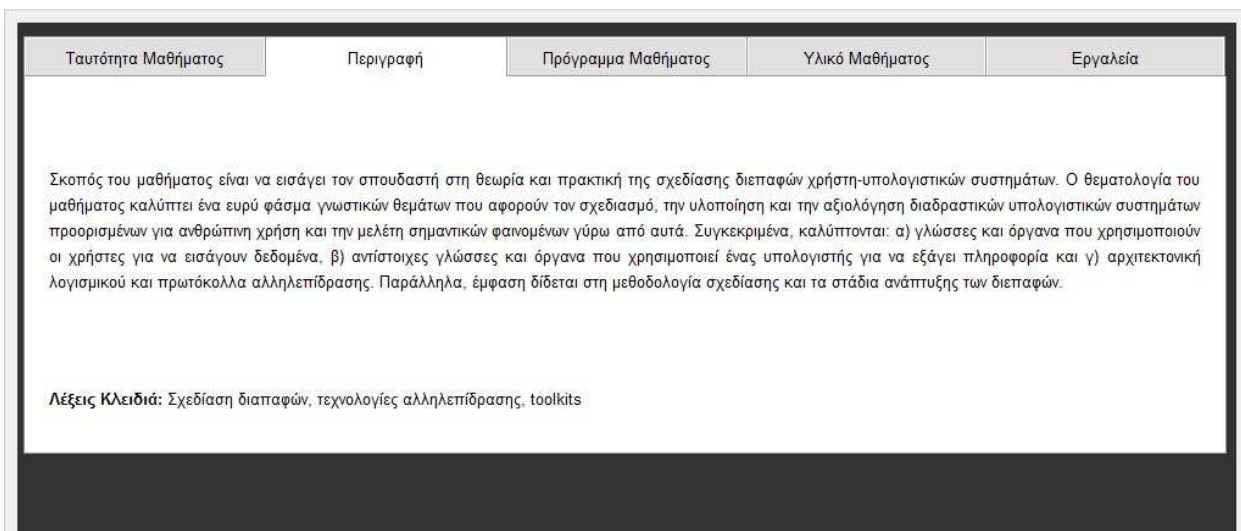
Εικόνα 24: Structure

Η σειρά είναι tabs_menu και είναι τύπου Text. Έχει σαν «παιδιά» τα id, about, schedule, material, tools και είναι όλα τύπου Text Area (HTML) και το σκαρίφημα του είναι αυτό που ακολουθεί.



Εικόνα 25: Σκαρίφημα προτύπου

4.2.3.2 Εφαρμογή



Εικόνα 26: Όψη Template (1)

Ταυτότητα Μαθήματος	Περιγραφή	Πρόγραμμα Μαθήματος	Υλικό Μαθήματος	Εργαλεία
				Notes-Use case diagram.pdf 148.45 KB
				Perigramma.pdf 49.11 KB
				Product Report.doc 30.5 KB
				proposals.mdl 199.83 KB
				Slides για use case modelling 202.59 KB
				UMLnotes.pdf 2.33 MB
				use_case_template-GR.doc 39.5 KB
				Από διαγράμματα ακολουθίας σε πρωτότυπα διεκπερών.pdf 61.06 KB
				ATM.mdl 172.97 KB

Εικόνα 27: Όψη Template (2)

4.2.4 Original Photo Gallery

4.2.4.1 Όψη

Το template αποτελείται από μια κεντρική εικόνα (main image) όπου στις τέσσερις γωνίες της έχουν τοποθετηθεί εικόνες μικρότερου μεγέθους που αντιπροσωπεύουν το περιεχόμενο του template (π.χ ενός projects του iSTLab). Περνώντας με το mouse πάνω από την μικρή εικόνα εμφανίζεται το σύμβολο (i), στο οποίο κάνοντας click, εμφανίζεται ένα pop up παράθυρο το οποίο περιέχει μια εικόνα και ένα κείμενο όπου είναι η περιγραφή του εκάστοτε project. Υπάρχει η δυνατότητα να γίνεται κλείσιμο του παραθύρου μέσω ενός ειδικού κουμπιού (X) και η επανάληψη της ενέργειας μπορεί να γίνει όσες φορές επιθυμεί ο χρήστης. Η παρακάτω εικόνα δείχνει τα στοιχεία από τα οποία αποτελείται το template.

Structure

ID 19309

Name telikovm

Description telikovm

Parent Structure

URL http://localhost:8080/c/journal/get_structure?groupId=16&structureId=19309

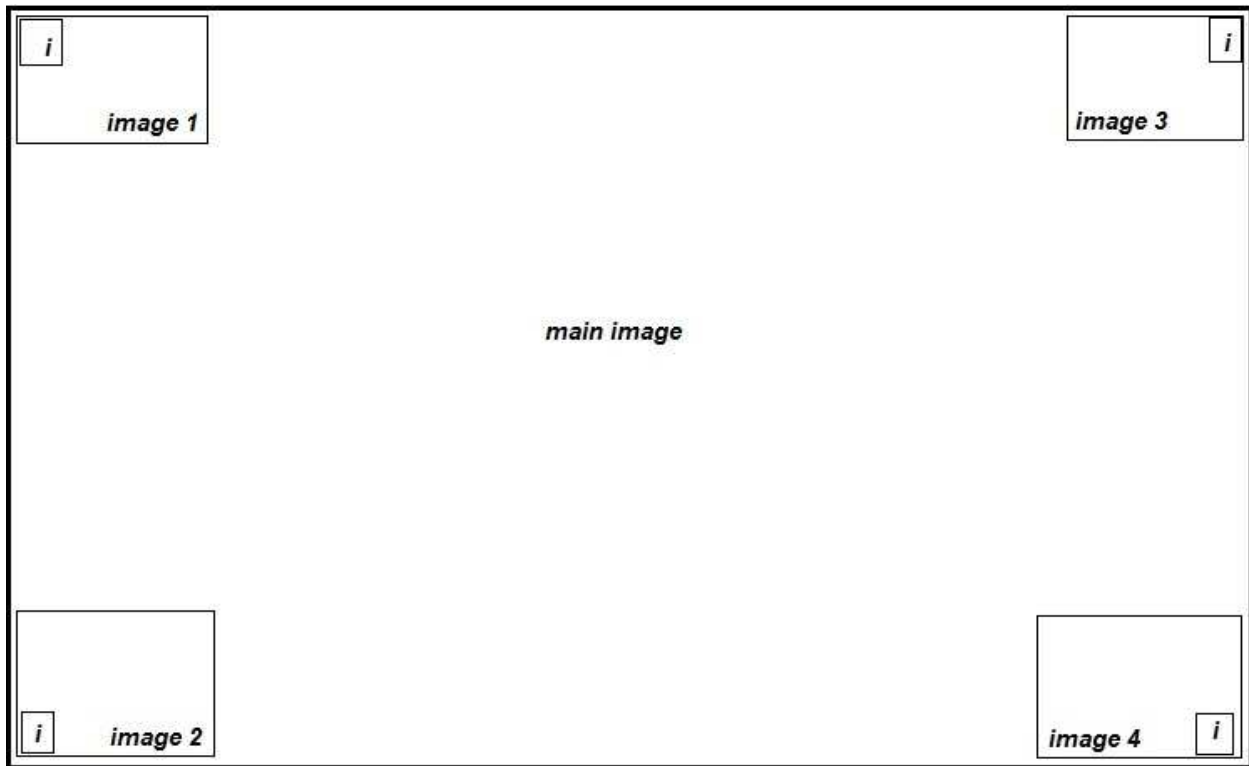
WebDAV URL http://localhost:8080/tunnel-web/secure/webdav/liferay.com/guest/journal

XML Schema Definition

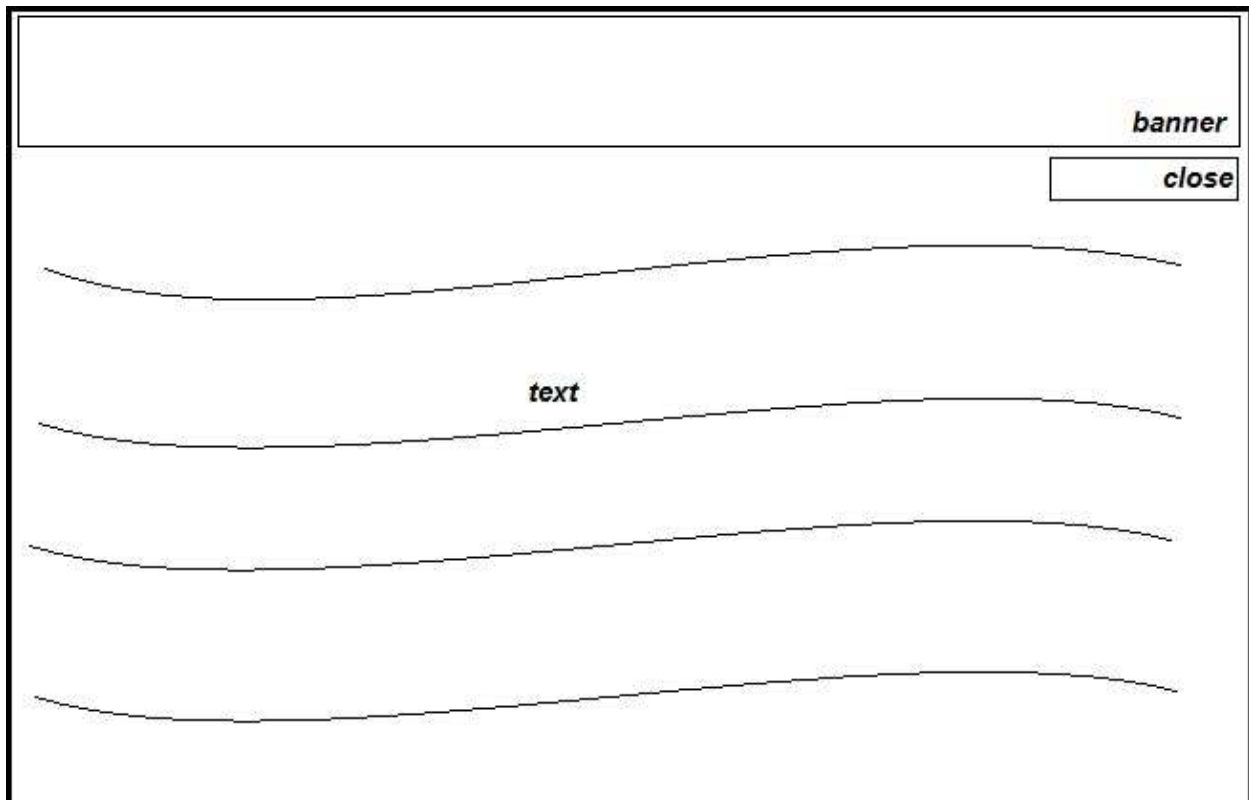
gallery	Text	<input type="checkbox"/> Repeatable		
central_image	Image	<input type="checkbox"/> Repeatable	<input type="button" value="Add"/>	<input type="button" value="Remove"/>
rep_image	Image	<input type="checkbox"/> Repeatable	<input type="button" value="Add"/>	<input type="button" value="Remove"/>

Εικόνα 28: Structure

Η σειρά είναι gallery και είναι τύπου Text. Έχει σαν «παιδιά» τα central_image, rep_image και είναι όλα τύπου image. Παρακάτω απεικονίζεται και το σκαρίφημα του template που μόλις περιγράψαμε.



Εικόνα 29: Σκαρίφημα κεντρικού προτύπου



Εικόνα 30: Σκαρίφημα προτύπου pop up

4.2.4.2 Εφαρμογή

Η εφαρμογή του template που περιγράφεται παρουσιάζεται παρακάτω.



Εικόνα 31: Όψη Template (1)



Εικόνα 32: Όψη Template (2)



Εικόνα 33: Όψη Template (3)

5 Επίλογος

5.1 Συμπεράσματα

Σκοπός αυτής της πτυχιακής ήταν η Ανάπτυξη Διαδικτυακής Πύλης Εκπαιδευτικού Περιεχομένου στο Liferay CMS. Αρχικά, έγινε έρευνα στο διαδίκτυο για να βρεθεί υλικό για το ανοικτού κώδικα Liferay CMS που χρησιμοποιήθηκε, το οποίο βασίζεται στη γλώσσα προγραμματισμού java και εκτελείται με οποιονδήποτε application server και με οποιαδήποτε Βάση Δεδομένων. Κατά την ανάπτυξη της διαδικτυακής πύλης προέκυψε η ανάγκη για την δημιουργία portlets τα οποία ενσωματώσαμε στο Liferay CMS. Για τη δημιουργία αυτών χρησιμοποιήσαμε τις HTML, JavaScript, XML, VM και jQuery τεχνολογίες. Επίσης στα πλαίσια αυτής της εργασίας αναπτύχθηκαν ειδικά πρότυπα (templates), τα οποία διαμορφώθηκαν ανάλογα ώστε να μπορούν να χρησιμοποιηθούν από τους χρήστες ανάλογα με τις ανάγκες των εφαρμογών τους.

Το template [“Body and Galleria Aino”](#), χρησιμοποιείται σήμερα στον ιστοχώρο το εργαστηρίου iSTLab, περιγράφοντας πτυχιακές εργασίες που διεκπεραιώθηκαν στο εργαστήριο από φοιτητές του τμήματος Εφαρμοσμένης Πληροφορικής & Πολυμέσων (π.χ [rogdakis-thesis](#)). Μέσω αυτού ο χρήστης μπορεί να δει πληροφορίες που αφορούν τον ίδιο τον φοιτητή, την ημερομηνία διεκπεραίωσης της πτυχιακής, μια μικρή περίληψη του αντικειμένου καθώς και φωτογραφικό υλικό οργανωμένο σε ένα image gallery.

Το template [“Image Gallery Rotator”](#), δημιουργήθηκε με σκοπό να παρουσιάσει επίσης το σύνολο των πτυχιακών εργασιών που διεκπεραιώθηκαν στο εργαστήριο του iSTLab με ένα διαφορετικό γραφικό τρόπο. Δίνοντας στο χρήστη τη δυνατότητα, να επιλέξει από το menu την εκάστοτε πτυχιακή για την οποία θα έχει πρόσβαση στην περίληψη της και στο φωτογραφικό υλικό που την αντιπροσωπεύει.

Το template [“Tabs Menu”](#) , υλοποιήθηκε προκειμένου να περιέχει την περιγραφή ενός θεωρητικού μαθήματος, δίνοντας πληροφορίες για αυτό σε διαφορετικά tabs π.χ ταυτότητα, περιγραφή, εβδομαδιαίο πρόγραμμα, υλικό μαθήματος, καθώς επίσης και εργαλεία χρήσης.

Τέλος το template [“Original Photo Gallery”](#) , υλοποιήθηκε εξ'ολοκλήρου από εμάς, με σκοπό την παρουσίαση των projects που έχουν διεκπαιρωθεί από το ακαδημαϊκό προσωπικό του εργαστηρίου iSTLab. Σε αυτό παρουσιάζεται μια συνοπτική περιγραφή για κάθε ένα από τα project μέσω ενός pop up παραθύρου όπου εμφανίζεται κάνοντας click στην εικόνα που αντιπροσωπεύει το κάθε ένα.

5.2 Μελλοντικά Έργα

Αργότερα κάθε ένα από αυτά τα template θα μπορούσε να επεκταθεί προσθέτοντας περισσότερη αλληλεπίδραση με τους χρήστες. Θα ήταν πολύ χρήσιμο αν δινόταν η δυνατότητα στους χρήστες να μπορούν να προσθέτουν σχόλια για το εκάστοτε θέμα που διαπραγματεύεται το template και τα σχόλια αυτά να αποθηκεύονταν σε μια ΒΔ. Θα μπορούσαν επίσης να δοθούν επιπλέον λειτουργίες που θα επέτρεπαν στους χρήστες εκτός από την ανάγνωση και την οπτική επαφή με το περιεχόμενο να επεμβαίνουν και να αλλάζουν την διάταξη των gallery, το χρώμα των γραμμών αλλά και την ροή των γεγονότων π.χ να έχουν την δυνατότητα να επιλέξουν αν θα γίνεται εμφάνιση του κειμένου ή της εικόνας μιας πτυχιακής κτλ.

Σίγουρα η εμπειρία της επεξεργασίας ήδη υπάρχόντων template αλλά και η δημιουργίας ενός εξ' ολοκλήρου δικού μας, ήταν μια εμπειρία η οποία μας γέμισε συναισθήματα. Αρχικά απορίας για το πώς λειτουργούν και τι κρύβεται πίσω από τα ελκυστικά εφέ, έπειτα προβληματισμού καθώς ήταν δύσκολη η κατανόηση εννοιών με τις οποίες ερχόμασταν σε επαφή για πρώτη φορά και τελικά ικανοποίησης για τη δημιουργία του τελικού αποτελέσματος. Ήρθαμε σε επαφή με γλώσσες προγραμματισμού και προγράμματα με τα οποία δυστυχώς δεν είχαμε την απαιτούμενη οικειότητα.

6 Αναφορές

- ❖ <http://en.wikipedia.org>
- ❖ <http://www.w3schools.com>
- ❖ <http://tomcat.apache.org/>
- ❖ <http://www.joomla.org>
- ❖ <http://drupal.org/>
- ❖ <http://www.liferay.com>
- ❖ <http://wiki.liferay.com>
- ❖ <http://velocity.apache.org/>
- ❖ <http://www.jquery.com>
- ❖ <http://greetuts.net/jquery-explaine/>
- ❖ <http://nefeli.lib.teicrete.gr/>
- ❖ <http://galleria.aino.se/>
- ❖ <http://www.sohtanaka.com/web-design/examples/image-rotator/#>
- ❖ Πτυχιακή εργασία: [Καττιμέρη Χριστίνα\(2009\) Παρουσίαση ηλεκτρονικού βιβλίου σε τρισδιάστατο περιβάλλον](#)
- ❖ Πτυχιακή εργασία: [Ρομπογιαννάκη Εμμανουέλα\(2007\) Ανάπτυξη διαδικτυακού κόμβου λειτουργίας τομέων εκπαιδευτικών μονάδων](#)