ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ

ΚΡΗΤΗΣ

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων

Πτυχιακή Εργασία

«Ανάπτυξη διαδικτυακού κόμβου λειτουργίας τομέων εκπαιδευτικών μονάδων »

Ρομπογιαννάκη Εμμανουέλα

Επιβλέπων καθηγητής : Ακουμιανάκης Δημοσθένης

Πίνακας Περιεχομένων

1 ΕΙΣΑΓΩΓΗ 4

2 ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ 4

2.1 ΤΙ ΕΙΝΑΙ PORTAL 4
2.2 ΤΙ ΕΙΝΑΙ PORTLET 6
2.3 ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ ΕΝΑ PORTAL 7
2.4 ΤΙ ΕΙΝΑΙ CMS 8

3 ΑΝΑΠΤΥΞΗ TEMPLATES 10

3.1 ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ 10
3.1.1 HTML 10
3.1.2 XSLT 12
3.2 LIFERAY TEMPLATES 15
3.3 ΔΗΜΙΟΥΡΓΙΑ TEMPLATES ΣΤΟ LIFERAY 15

4 ΕΙΔΙΚΑ TEMPLATES 20

4.1 NEWSPAPER TEMPLATE 20
4.1.1 Όψη 20
4.1.2 Κώδικας 20
4.1.3 Εφαρμογή στο portal του eKoNES 25
4.2 BODY-AND-TABS TEMPLATE 26
4.2.1 Όψη 26
4.2.2 Κώδικας 26
4.2.3 Εφαρμογή στο portal του eKoNES 34
4.3 IMAGE GALLERY 34
4.3.1 Όψη 34
4.3.2 Κώδικας 35
4.3.3 Εφαρμογή στο Portal του Τομέα Τηλεπικοινωνιών & Πολυμέσων 41
4.4 SLIDER CONTENT 41
4.4.1 Όψη 41
4.4.2 Κώδικας 42
4.4.3 Εφαρμογή στο portal του Τομέα Τηλεπικοινωνιών & Πολυμέσων 46
4.5 BODY-AND-2-FADE-SLIDESHOWS TEMPLATE 47
4.5.1 Όψη 47
4.5.2 Κώδικας 47
4.5.3 Εφαρμογή στο portal του eKoNES 54
4.6 MOTION-IMAGES TEMPLATE 54
4.6.1 Όψη 54
4.6.2 Κώδικας 55
4.6.3 Εφαρμογή στο portal του eKoNES 58

5 ΣΕΝΑΡΙΑ ΧΡΗΣΗΣ ΤΟΥ ΤΟΜΕΑ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ & ΠΟΛΥΜΕΣΩΝ (ΤΤΠ) 59

5.1 ΕΓΓΡΑΦΗ ΧΡΗΣΤΗ 59
5.2 LOGIN ΧΡΗΣΤΗ 60
5.3 ΔΙΑΧΕΙΡΙΣΗ ΑΝΑΚΟΙΝΩΣΕΩΝ 61
5.4 ΕΠΙΣΚΟΠΗΣΗ ΔΗΜΟΣΙΟΥ ΠΕΡΙΕΧΟΜΕΝΟΥ 65
5.5 ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΦΙΛ ΧΡΗΣΤΗ 65

6 ΣΥΜΠΕΡΑΣΜΑΤΑ 67

7 ΒΙΒΛΙΟΓΡΑΦΙΑ 67

Ευρετήριο Εικόνων

Εικονα 1: Δομη μιας Portal σελιδας 6
Εικονα 2: Λειτουργια ενος Portal 8
Εικονα 3: Λειτουργια ενος XSLT επεξεργαστη 13
Εικονα 4: Απλο αρθρο στο Lifery 16
Εικονα 5: Δημιουργια δομης (structure) 17
Εικονα 6: Εισαγωγη των στοιχειων του αρθρου 17
Εικονα 7: Δημιουργια template 18
Εικονα 8: Εισαγωγη κωδικα XSLT 19
Εικονα 9: Δημιουργια αρθρου 19
Εικονα 10: Όψη του προτυπου Newspaper 20
Εικονα 11: Newspaper template στο eKoNES 25
Εικονα 12: Όψη του προτυπου Body-and-tabs 26
Εικονα 13: Body-and-tabs template στο eKoNES 34
Εικονα 14: Όψη του προτυπου image gallery 34
Εικονα 15: Image Gallery στον Τομεα Τηλεπ/νιων & Πολυμεσων 41
Εικονα 16: Όψη του προτυπου Slider Content 42
Εικονα 17: Slider Content στον Τομεα Τηλεπ/νιων &Πολυμεσων 46
Εικονα 18: Όψη του προτυπου Body-and-2-fade-slideshows 47
Εικονα 19: Body-and-2-fade-slideshows template στο eKoNES 54
Εικονα 20: Όψη του προτυπου Motion-images 54
Εικονα 21: Motion-images template στο eKoNES 58
Εικονα 22:Εγγραφη χρηστη 59
Εικονα 23: Η διαδικασια εγγραφης 60
Εικονα 24: Αλλαγη κωδικου χρηστη 60
Εικονα 25: Εισοδος χρηστη 60
Εικονα 26: Όψη εγγεγραμενου χρηστη 61
Εικονα 27: Το Forum του Τομεα 61
Εικονα 28: Διαλογος αποστολης ανακοινωσης 63
Εικονα 29: Η διαδικασια reply 63
Εικονα 30: Σχολιασμος ανακοινωσης 64
Εικονα 31: Επισκοπηση μαθηματων 65
Εικονα 32: Επιλογη μενου "My account" για αλλαγη προφιλ 66
Εικονα 33:Αλλαγη προφιλ 66

1 Εισαγωγή

Στην παρούσα πτυχιακή ασχοληθήκαμε με την ανάπτυξη προτύπων (templates) για τον προσδιορισμό των τύπων της πληροφορίας που λαμβάνουν μέρος σε ένα δημοσίευμα και της οπτικής διάταξης του δημοσιεύματος.

Επίσης αναλύεται ο διαδικτυακός κόμβος του Τομέα Τηλεποικοινωνιών και Πολυμέσων ο οποίος αναπτύχθηκε με τη χρήση του ανοικτού κώδικα Σύστημα Διαχείρισης Περιεχομένου Liferay (Liferay CMS), το οποίο βασίζεται στη γλώσσα προγραμματισμού java και εκτελείται με οποιονδήποτε application server και με οποιαδήποτε Βάση Δεδομένων. Οι τεχνολογίες που χρησιμοποιήθηκαν είναι οι JSP, HTML, Javascript και XML. Στην πτυχιακή αυτη γίνεται μια σύνοψη βασικών τεχνολογιών και εργαλείων που αναπτύχθηκαν για να υποστηρίξουν την αρχική λειτουργία του διαδικτυακού κόμβου σε πιλοτικό επίπεδο.

2 Εισαγωγικές έννοιες

2.1 Τι είναι Portal

Portal είναι μια web εφαρμογή η οποία ενεργεί ως «πύλη» μεταξύ των χρηστών και μιας σειράς διαφορετικών, υψηλού επιπέδου, εφαρμογών. Παρέχει μια ενιαία λειτουργία μέσω μιας ιστοσελίδας και η βασική ιδέα είναι να συλλεχθούν πληροφορίες από διαφορετικές πηγές ώστε να δημιουργηθεί ένα ενιαίο σημείο πρόσβασης στις πληροφορίες αυτές.

Τα πρώτα Portal ήταν μηχανές αναζήτησης. Ο όρος όμως "μηχανή αναζήτησης" ήταν ανεπαρκής για να περιγράψει το μεγάλο εύρος των προσφορών. Έναντι των αρχικών μηχανών αναζήτησης Διαδικτύου, τα σημερινά Portal προσφέρουν μια πιο δομημένη διεπαφή. Παρουσιάζουν πληροφορίες από διαφορετικές πηγές με έναν ενοποιημένο τρόπο και εκτός από τα πρότυπα μηχανών αναζήτησης, προσφέρουν κι άλλες υπηρεσίες όπως το ηλεκτρονικό ταχυδρομείο, ειδήσεις, και διάφορα άλλα χαρακτηριστικά γνωρίσματα. Η τεχνολογία των Portal πλέον έχει ωριμάσει σημαντικά δεδομένου ότι έχει χρησιμοποιηθεί για να «χτίσει» μια σειρά από σημαντικές εφαρμογές όπως: εξειδικευμένα, επιχειρηματικά, εκπαιδευτικά Portal κλπ…

Η σημαντικότερες λειτουργίες ενός Portal είναι οι εξής:

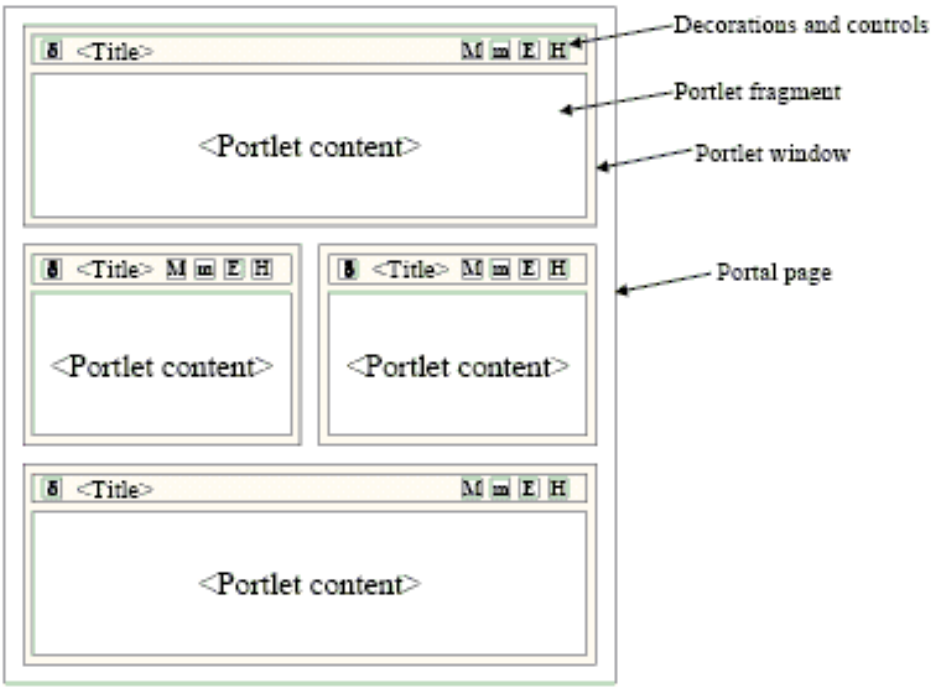
- **Search and navigation:** Αυτή η λειτουργία αποτελεί τη βάση για τα περισσότερα «επιτυχημένα» Portal, με την έννοια ότι ένα portal πρέπει να προσφέρει στους χρήστες του μια αποδοτική αναζήτηση ενός περιεχομένου. Δηλαδή πρέπει να παρουσιάζει αυτόματα την πληροφορία και τις υπηρεσίες που θα χρειαζόταν ένας χρήστης την κατάλληλη στιγμή, σύμφωνα με το προφίλ του.
- **Portlet container:** Ένας portlet container ελέγχει το κύκλο ζωής του portlet και του παρέχει παρέχει το απαιτούμενο περιβάλλον εκτέλεσης. Επίσης είναι υπεύθυνος για την αρχικοποίηση και την καταστροφή του portlet καθώς και για την μεταφορά των αιτήσεων των χρηστών σε αυτό.
- **Content aggregator:** Μια από τις κύριες εργασίες ενός Portal είναι να αθροίζει το περιεχόμενο που παράγεται από τα portlet.
- **Common services:** Μια από τις κύριες "δυνάμεις" ενός Portal είναι το σύνολο των κοινών υπηρεσιών που παρέχει. Μερικές από αυτές τις υπηρεσίες είναι οι εξής:
    - **Single sign on:** Επιτρέπει στους χρήστες την πρόσβαση σε όλες τις εφαρμογές κάνοντας login μόνο μια φορά χωρίς να είναι απαραίτητο να κάνουν log in σε κάθε εφαρμογή χωριστά.
    - **Personalization:** Επιτρέπει στο χρήστη να επιλέγει την πληροφορία και τις υπηρεσίες που τον ενδιαφέρουν και να προσαρμόζει την παρουσίασή τους σύμφωνα με τις απαιτήσεις του. Η εξατομίκευση είναι ζωτικής σημασίας για την παράδοση των σωστών πληροφοριών στους χρήστες: κάθε χρήστης παίρνει μόνο τις πληροφορίες που προσαρμόζονται συγκεκριμένα στις ανάγκες του/της. Η εξατομίκευση πρέπει να βασιστεί στους ρόλους των χρηστών.

Δομή μιας Portal σελίδας

Κάθε Portal σελίδα αποτελείται από ένα ή περισσότερα portlet windows. Κάθε portlet αποτελείται από δυο μέρη:

Το πρώτο μέρος είναι το *decoration* το οποίο αποφασίζει πώς ο τίτλος ,τα κουμπιά ελέγχου και τα άλλα στοιχειά του portlet θα εμφανιστούν. Το δεύτερο

μέρος είναι το *portlet fragment* το οποίο είναι το περιεχόμενο του portlet. Στη συνέχεια το portal συναθροίζει τα portlet windows σε ένα πλήρες έγγραφο, την portal σελίδα Η γενική δομή μιας Portal σελίδας φαίνεται στην παρακάτω εικόνα:



Εικόνα 1: Δομή μιας Portal σελίδας

Η εμφάνιση του portal όπως τα χρώματα, τα logo καθώς και οι εικόνες για τα στοιχεία ελέγχου, καθορίζονται από τον portal server. Με την αλλαγή όμως κάποιον standard .jsp και .css αρχείων μπορούμε να αλλάξουμε εντελώς την εμφάνιση του portal.

2.2 Τι είναι Portlet

Portlet είναι ένα web component το οποίο δημιουργείται σε ένα portlet container και παράγει δυναμικό περιεχόμενο. Τα portlet συναθροίζονται σε μια portal σελίδα και το περιεχόμενο που παράγουν ονομάζεται fragment. Fragment είναι ένα κομμάτι από markup κώδικα (π.χ. HTML, XMTHL) το οποίο εμμένει σε συγκεκριμένους κανόνες και μπορεί να αθροιστεί μαζί με άλλα fragment για να αποτελέσουν ένα ολοκληρωμένο έγγραφο. Το περιεχόμενο που παράγεται από ένα

portlet μπορεί να ποικίλει ανάλογα με τα δικαιώματα πρόσβασης (user configuration) που έχει ένας χρήστης στο συγκεκριμένο portlet. Οι web clients αλληλεπιδρούν με τα portlet μέσω μιας διαδικασίας αίτησης/απάντησης που υλοποιείται από το portal.

Τα portlets δεν είναι άμεσα προσπελάσιμα μέσω μιας url. Δεν υπάρχουν δηλαδή url που να οδηγούν σε portlets ,υπάρχουν όμως url που οδηγούν σε σελίδες που περιέχουν portlets. Επίσης δεν μπορούν να παράγουν ένα αυθαίρετο περιεχόμενο δεδομένου ότι το περιεχόμενο που παράγεται από ένα portlet πρόκειται να είναι μέρος μιας Portal σελίδας. Επομένως εάν ένας portal server ζητήσει ένα html/text περιεχόμενο τότε όλα τα portlet πρέπει να παράγουν ένα html/text περιεχόμενο.

2.3 Πως λειτουργεί ένα Portal

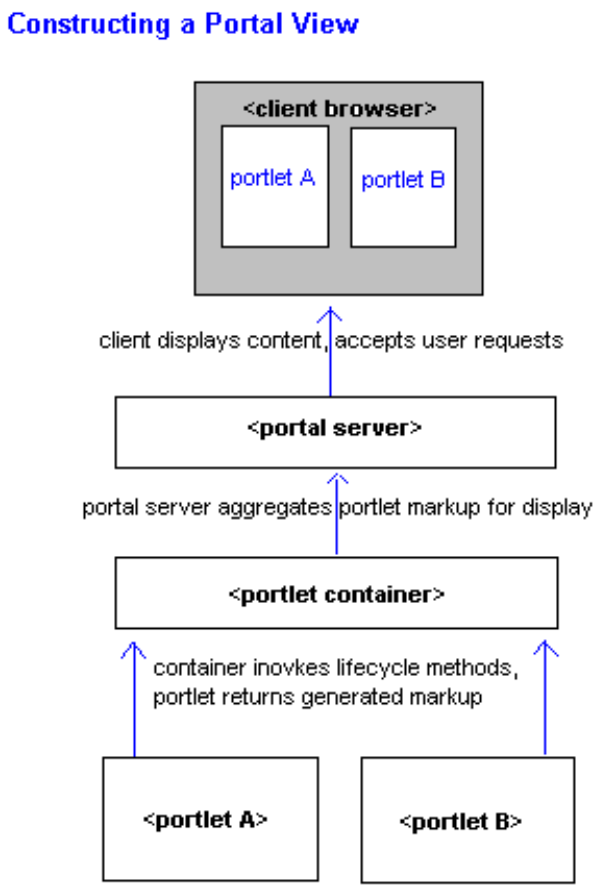
Τι είναι Portlet Container

Ο Portlet Container βρίσκεται ανάμεσα στο Portal και στα portlets και αυτό που κάνει είναι να προωθεί αιτήσεις στα portlet. Είναι αρμόδιος για την λειτουργία των portlet και τους παρέχει το απαιτούμενο περιβάλλον εκτέλεσης. Επίσης διαχειρίζεται τον κύκλο ζωής τους.

Ένα παράδειγμα που εξηγεί την λειτουργία ενός portal και που δείχνει τους ρόλους του καθενός από αυτά (Portlet, Portal και Portlet Container) είναι το εξής:

- Ο χρήστης στέλνει ένα αίτημα σε μια Portal σελίδα.
- Αυτό το αίτημα φτάνει στο Portal το οποίο εξετάζει το portlet που εκτελείται από την λίστα των portlets, ώστε να ικανοποιήσει αυτό το αίτημα.
- Εάν το αίτημα δεν περιέχει ενέργεια που να στοχεύει σε αυτό το portlet τότε το Portal επικαλείται το ζητούμενο portlet μέσω του Portlet Container ο οποίος «τρέχει» στη συνέχεια το ζητούμενο portlet και επιστρέφει την απάντηση στο portal.

Η ακολουθία γεγονότων που λαμβάνουν χώρα σε ένα portal φαίνεται στην παρακάτω εικόνα.



Εικόνα 2: Λειτουργία ενός Portal

2.4 Τι είναι CMS

Το **CMS (content management system- Σύστημα διαχείρισης περιεχομένου)** είναι ένα σύστημα λογισμικού που χρησιμοποιείται για την δημιουργία, οργάνωση και παρουσίαση περιεχομένου στο διαδίκτυο. Το περιεχόμενο αυτό είναι αρχεία υπολογιστών, εικόνες, αρχεία ήχου, ηλεκτρονικά έγγραφα κλπ... Η ιδέα πίσω από ένα CMS είναι να καταστήσει αυτά τα αρχεία διαθέσιμα μεταξύ των ατόμων ενός γραφείου καθώς και μεταξύ ατόμων σε όλο το διαδίκτυο. Εταιρείες χρησιμοποιούν τα CMS για να διαμοιράζονται αρχεία με ευκολία, καθώς τα περισσότερα συστήματα χρησιμοποιούν λογισμικό βασισμένο σε servers. Πολλά CMS περιλαμβάνουν ένα χαρακτηριστικό για Web Content και μερικά έχουν ένα χαρακτηριστικό για «διαδικασία ροής εργασίας» ("workflow process").

Workflow process

Η «ροή εργασίας» είναι η ιδέα της μετακίνησης ενός ηλεκτρονικού εγγράφου είτε για επικύρωση είτε για προσθήκη περιεχομένου. Μερικά CMS διευκολύνουν αυτή τη διαδικασία με ειδοποίηση email και αυτοματοποιημένης δρομολόγησης. Αυτή είναι ιδανικά μια συνεργατική δημιουργία εγγράφων. Ένα CMS διευκολύνει την οργάνωση, τον έλεγχο και τη δημοσιοποίηση ενός μεγάλου αριθμού εγγράφων και άλλου περιεχομένου όπως εικόνες και πηγές πολυμέσων.

Τα χαρακτηριστικά του Συστήματος Διαχείρισης Περιεχομένου του Liferay που χρησιμοποιήθηκε σαυτην την εργασία, παρατίθενται παρακάτω:

- Δημοσιοποίηση περιεχομένου (Web publishing): Το σύστημα δημοσιοποίησης περιεχομένου του Liferay Journal μπορεί να χρησιμοποιηθεί ώστε να δημιουργηθούν ιστοσελίδες γρήγορα, με τη βοήθεια επαναχρησιμοποιούμενου περιγραμμάτων, ευέλικτων μοτίβων διάταξης και δυναμικά παραγόμενης ταξινόμησης του ιστοχώρου. Τα μοτίβα μπορούν να γραφτούν χρησιμοποιώντας Velocity 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 στον προγραμματιστή στη σχεδίαση των ιστοχώρων του.
- Βιβλιοθήκη εγγράφων (Document Library): Η Βιβλιοθήκη Εγγράφων παρέχει μια κεντρική αποθήκευση με εκδόσεις και υπηρεσίες βιβλιοθήκης (check in/check out) και στηρίζεται σε JCR-170 μεταγλώττιση (Java Content Repository- Jackrabbit) η οποία αποθηκεύει και δίνει ένα μοναδικό URL σε ποικιλία εγγράφων (π.χ. .PDF, .DOC, .XSL, κ.α.).
- Έκθεση εικόνων (Image Gallery): Όπως και η Βιβλιοθήκη Εγγράφων έτσι και η Έκθεση Εικόνων παρέχει αποθήκευση σε εικόνες που χρησιμοποιούνται στο portal και τους δίνει ένα μοναδικό URL.
- Δημοσιοποίηση του Portal και Staging (Portal Publishing and Staging): Δίνεται η δυνατότητα να γίνεται επεξεργασία περιεχομένου σε μια live σελίδα του ιστοχώρου χωρίς να επηρεάζεται αυτό που βλέπει ο χρήστης ωσότου τελειώσει η διεργασία. Το Staging επιτρέπει να υπάρχουν ανά πάσα στιγμή

πολλαπλά αντίγραφα τις ίδιας σελίδας και να γίνεται προκαταρκτική επισκόπηση χωρίς να εμποδίζεται η λειτουργία των live 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.
- Ομοσπονδιακή αναζήτηση (Federated Search): Το Liferay αυξάνει την απόδοση του Amazon OpenSearch, καθώς επιστρέφει ομόσπονδα (federated) αποτελέσματα αναζήτησης από διάφορες πηγές περιεχομένου συμπεριλαμβανομένου των portlet του Liferay και των εξωτερικών ολοκληρωμένων εφαρμογών.
- Meta Tags: Το σύστημα ετικετών του Liferay επιτρέπει στον χρήστη να βάζει ετικέτες στο web περιεχόμενο, στα έγγραφα, στα νήματα του πίνακα ανακοινώσεων και να δημοσιοποιεί δυναμικά το περιεχόμενο μέσω των ετικετών.

3 Ανάπτυξη templates

3.1 Γλώσσες Προγραμματισμού

3.1.1 HTML

Τα αρχικά **HTML** προέρχονται από τις λέξεις **HyperText Markup Language**. Η html στην ουσία δεν είναι μια γλώσσα προγραμματισμού. Είναι μια περιγραφική γλώσσα, δηλαδή ένας ειδικός τρόπος γραφής κειμένου. Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου. Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Ο browser αναγνωρίζει αυτόν τον τρόπο γραφής και εκτελεί τις εντολές που περιέχονται σε αυτόν. Αξίζει να σημειωθεί ότι η html είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας. Η html χρησιμοποιεί τις ειδικές ετικέτες (τα tags) για να δώσει τις απαραίτητες οδηγίες στον browser. Τα tags είναι εντολές που συνήθως ορίζουν την αρχή ή το το τέλος μιας λειτουργίας. Τα

tags βρίσκονται πάντα μεταξύ των συμβόλων < και > Π.χ. <BODY>. Οι οδηγίες είναι case insensitive, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία. Ένα αρχείο HTML πρέπει να έχει κατάληξη htm ή html.
Η βασική δομή μιας σελίδας HTML είναι η εξής:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
 <html>
 <head> <!-- Το head περιέχει πληροφορίες που αφορά το αρχείο και δεν εμφανίζεται στην οθόνη. -->
</head>
 <body>
<!-- η HTML που θα εμφανιστεί -->
</body>
</html>
```

Η πρώτη γραμμή <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN"> δεν είναι τεχνικά απαραίτητη αλλά είναι ένας κώδικας που λέει στον browser για ποια έκδοση της HTML είναι γραμμένη η συγκεκριμένη σελίδα. Όλη η HTML περιέχεται μέσα στην <html>...</html> ετικέτα. Μέσα σε αυτή περιέχονται η <head>...</head> και μετά η <body>...</body>. Κάποιες από τις πιο βασικές HTML ετικέτες φαίνονται στον παρακρατώ πίνακα:

Structure Tags	
<html>...</html>	Περιλαμβάνει όλο το HTML έγγραφο
<head>...</head>	Περιλαμβάνει την επικεφαλίδα του HTML εγγράφου
<body>...</body>	Περιλαμβάνει το «ορατό» κομμάτι του HTML εγγράφου
Title and Headings	
<title>...</title>	Καθορίζεται ο τίτλος της σελίδας
...	Επικεφαλίδα πρώτου επιπέδου
...	Επικεφαλίδα δεύτερου επιπέδου
...	Επικεφαλίδα τρίτου επιπέδου
...	Επικεφαλίδα τέταρτου επιπέδου
...	Επικεφαλίδα πέμπτου επιπέδου
...	Επικεφαλίδα έκτου επιπέδου
Paragraphs	
...	Φτιάχνει παράγραφο
 …</br>	Αλλάζει γραμμή
Links	
<a>...	Δημιουργεί ένα σύνδεσμο σε ένα άλλο έγγραφο ή αρχείο
Lists	

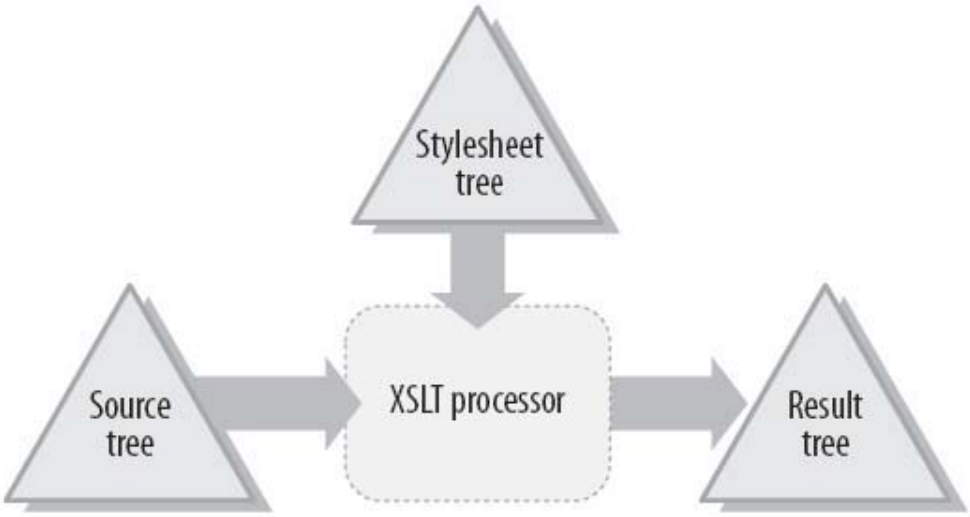
...	Δημιουργεί μια αριθμημένη λίστα
...	Δημιουργεί μια μη αριθμημένη λίστα
...	Σε συνδυασμό με τα <OL>…</OL> ή
Character Formating	
…	Κάνει το κείμενο ένα μέγεθος μικρότερο.
…	Δίνει έμφαση στο κείμενο κάνοντάς το bold.
_…	Κάνει το κείμενο subscript, δηλαδή $_{\text{this}}$.
[…]	Κάνει το κείμενο superscript, δηλαδή ^{this}.
<tt> … </tt>	Κάνει το κείμενο fixed-width και mono-spaced γραμματοσειράς, όπως μια παλιά δακτυλογραφική μηχανή.
<u> … </u>	Υπογραμμίζει το κείμενο
…	Κάνει το κείμενο πλάγιο
Multimedia	
<img>…</img>	Τοποθετεί μια εικόνα στη σελίδα.
<embed>…</embed>	Προσθέτει ένα πολυμεσικό στοιχείο κατευθείαν στη σελίδα, επιτρέποντας στον browser να το εκτελέσει με ένα plug-in.
<script>...</script>	Προσθέτει ένα script, συνήθως JavaScript, στη σελίδα.<Noscript>…</noscript>: Εσωκλείεται οτιδήποτε επιθυμεί να παρουσιάζεται στους browsers που δεν υποστηρίζει script.
Tables	
<table>…</table>	Τοποθετεί έναν πίνακα στη σελίδα.
<tr>…</tr>	Ξεκινά μια νέα γραμμή πίνακα με κελιά
<td>…</td>	Περιέχει ένα κελί του πίνακα με περιεχόμενα.
Forms	
<form>…</form>	Ξεκινά μια περιοχή φόρμας.
<input>	Επιτρέπει να προστεθούν διάφορα πεδία εισόδου, όπως text-boxes, checkboxes, radio buttons, submit και reset buttons ανάλογα με την τιμή της παραμέτρου type.
<textarea> … </textarea>	Προσθέτει μια περιοχή με πολλαπλές γραμμές εισόδου.
<select> … </select>	Τοποθετεί ένα άδειο drop-down box. Μπορούν να προστεθούν επιλογές με τις ετικέτες <option>… </option>

3.1.2 XSLT

Extensible Stylesheet Language Transformations (XSLT) είναι μια γλώσσα βασισμένη στην XML η οποία χρησιμοποιείται για να μετατρέψει ένα XML έγγραφο σε ένα άλλο XML έγγραφο με διαφορετική δομή, σε ένα HTML έγγραφο ή σε έναν άλλο τύπο εγγράφου που να αναγνωρίζεται από ένα browser. Το αρχικό έγγραφο δεν αλλάζει, αλλά δημιουργείται ένα νέο έγγραφο το οποίο είναι βασισμένο στο περιεχόμενο ενός υπάρχοντος εγγράφου. Η XSLT σχεδιάστηκε για να

χρησιμοποιείται σαν μέρος της XSL, και είναι μια γλώσσα για το σχεδιασμό stylesheet.

Ένας XSLT επεξεργαστής, για να μετατρέψει ένα XML έγγραφο, το αναλύει σε μια δομή τριών δέντρων: το δέντρο πηγής (source tree), το stylesheet δέντρο (stylesheet tree), και το δέντρο αποτελέσματος (result tree). Η παρακάτω εικόνα παρουσιάζει τη σχέση μεταξύ αυτών των τριών στοιχειών. Τα δέντρα stylesheet και πηγής «τροφοδοτούν» τον επεξεργαστή XSLT, ο οποίος παράγει έπειτα το δέντρο αποτελέσματος.



Εικόνα 3: Λειτουργία ενός XSLT επεξεργαστή

Το στοιχείο το οποίο αποτελεί τη ρίζα σε ένα XSLT stylesheet είναι το <xsl:stylesheet> ή <xsl:transform>. Μπορούν να χρησιμοποιηθούν και τα δύο, αφού συμπεριφέρνονται ακριβώς με τον ίδιο τρόπο. Ο σωστός τρόπος για να δηλωθεί ένα XSLT stylesheet είναι:

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

ή

```
<xsl:transform version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

XSLT Στοιχεία (Elements)

Στοιχεία (Elements)	Περιγραφή	Παράδειγμα
xsl:apply-templates	Εφαρμόζει ένα πρότυπο στο τρέχον στοιχείο	<xsl:apply-templates/>

xsl:attribute	Προσθέτει ιδιότητες στα στοιχεία	<xsl:attribute name="attributename" namespace="uri"> ….. </xsl:attribute>
xsl:attribute-set	Δημιουργεί ένα set από ιδιότητες	<xsl:attribute-set name="name" use-attribute-sets="name-list"> ….. </xsl:attribute-set>
xsl:choose	Εκφράζει πολλαπλές επιλογές	<xsl:choose> … </xsl:choose>
xsl:comment	Δημιουργεί ένα σχόλιο στο δέντρο αποτελέσματος	<xsl:comment> ….. </xsl:comment>
xsl:copy	Δημιουργεί ένα αντίγραφο του τρέχοντος κόμβου, χωρίς να αντιγράφει τα παιδιά του κόμβου και τις ιδιότητες.	<xsl:copy use-attribute-sets="name-list"> ….. </xsl:copy>
xsl:copy-of	Δημιουργεί ένα αντίγραφο του τρέχοντος κόμβου , αντιγράφοντας και τα παιδιά του κόμβου και τις ιδιότητες του.	<xsl:copy-of select="expression"/>
xsl:for-each	Δημιουργεί ένα loop μέσα σε ένα κόμβο	<xsl:for-each select="expression"> …. </xsl:for-each>
xsl:if	Περιλαμβάνει ένα template το οποίο εφαρμόζεται μόνο εάν μια συγκεκριμένη κατάσταση είναι αληθής.	<xsl:if test="expression"> ….. </xsl:if>
xsl:import	Εισάγει τα περιεχόμενα ενός stylesheet σε ένα άλλο	<xsl:import href="URI"/>
xsl:include	Περιλαμβάνει τα περιεχόμενα ενός stylesheet σε ένα άλλο	<xsl:include href="URI"/>
xsl:otherwise	Καθορίζει τη default ενέργεια για το <xsl:choose> στοιχείο . Αυτή η ενέργεια εκτελείται μόνο όταν καμία από τις προϋποθέσεις του <xsl:when> στοιχείου δεν ικανοποιείται.	<xsl:otherwise> ….. </xsl:otherwise>
xsl:output	Καθορίζει το format του εγγράφου παραγωγής	<xsl:output method="xml |html | text | name" version="string" encoding="string" omit-xml-declaration= "yes | no" standalone= "yes | no" doctype-public="string" doctype-system="string" cdata-section-elements="namelist" indent="yes | no" media-type="string"/>
xsl:param	Δηλώνει μια local ή global παράμετρο	<xsl:param name="name" select="expression"> …… </xsl:param>
xsl:template	Καθορίζει τα πρότυπα επεξεργασίας .Το “match” καθορίζει πότε το πρότυπο πρέπει να χρησιμοποιηθεί και το “name” δίνει στο πρότυπο ένα όνομα	<xsl:template name="name" match="pattern" > ….. </xsl:template>

xsl:value-of	Χρησιμοποιείται για να επιλέξει την τιμή ενός στοιχείου XML και να το εμφανίσει στην έξοδο	<xsl:value-of select="expression" />
xsl:variable	Δηλώνει μια local ή global μεταβλητή	<xsl:variable name="name" select="expression"> …… </xsl:variable>
xsl:when	Καθορίζει μια ενέργεια για το <xsl:choose> στοιχείο, αξιολογώντας μια έκφραση, η οποία αν είναι αληθής τότε εκτελείται η ενέργεια αυτή	<xsl:when test="boolean-expression"> ….. </xsl:when>

3.2 Liferay templates

Το Liferay Journal διαχωρίζει τη δημιουργία περιεχομένου του σχεδιαστή από αυτή του συγγραφέα. 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 επιτρέπει στους χρήστες να δημιουργήσουν, να επεξεργαστούν και να δημοσιεύσουν άρθρα χρησιμοποιώντας ειδικά templates που καθορίζουν την παρουσίαση του άρθρου.

Τα πρότυπα (templates) που χρησιμοποιούνται στο Liferay, μπορούν να δημιουργηθούν είτε με XSL είτε με Velocity (VM) δίνοντας έτσι ευελιξία στο προγραμματιστή κατά την σχεδίαση των ιστοχώρων του. Στην πτυχιακή αυτη, τα templates αναπτύχθηκαν με την χρήση της XSLT.

3.3 Δημιουργία templates στο Liferay

Το Liferay Journal διαχωρίζει τη δημιουργία ενός άρθρου σε τρεις διαδικασίες:

- **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):** Καθορίζει την διάταξη (παρουσίαση) των στοιχείων που ορίστηκαν κατά την δημιουργία της δομής.
- **Δημιουργία άρθρου (article):** Σε αυτήν τη φάση δημιουργείται το άρθρο, με βάση τη δομή και το πρότυπο που καθορίστηκε στα προηγούμενα βήματα.

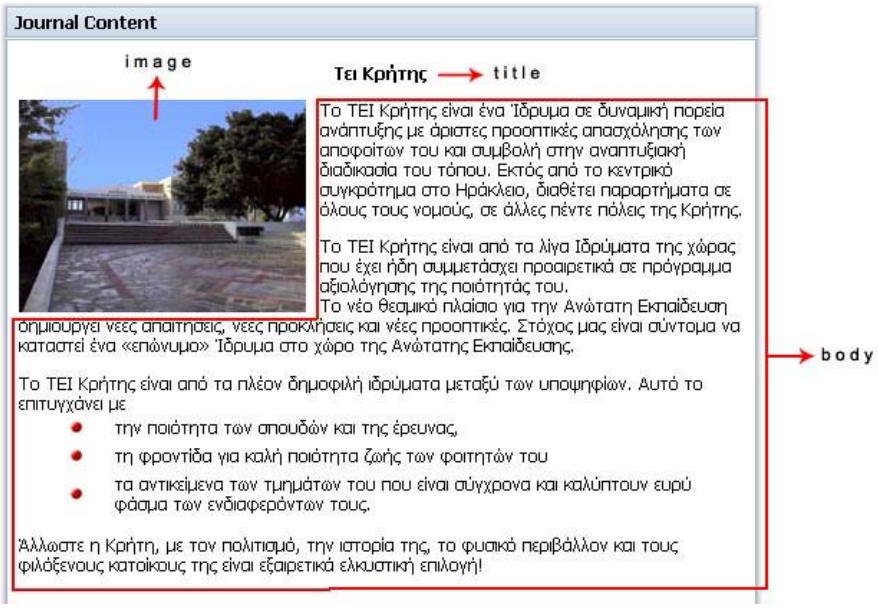
Ο σκοπός της δομής (structure) είναι να ενοποιεί ένα σύνολο από όμοια άρθρα, με τον ίδιο αριθμό και τύπο στοιχείων. Για παράδειγμα εάν θέλουμε σε ένα site να εισάγουμε πολλές βιογραφίες, η κάθε μια από αυτές απαιτεί έναν τίτλο, μια εικόνα, καθώς και μια περιοχή για το κείμενο της βιογραφίας. Επομένως, η δομή που θα δημιουργηθεί θα πρέπει να περιέχει όλα αυτά τα στοιχεία και κάθε ένα από αυτά θα ονομάζεται αναλόγως. Αυτό επιτρέπει στο χρήστη να δημιουργήσει άρθρα από βιογραφίες με διαφορετικό περιεχόμενο κάθε φορά χωρίς να χρειάζεται να δημιουργεί τη δομή εκ νέου για κάθε άρθρο.

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.

Όλα τα στοιχεία (κείμενα, εικόνες κλπ..) εισάγονται όταν ο χρήστης προσθέτει ή επεξεργάζεται ένα άρθρο. Την στιγμή αυτή επίσης, ο χρήστης επιλέγει τη διάταξη των στοιχείων του άρθρου σύμφωνα με τα templates που συνδέονται με τη δομή. Εφόσον υπάρχουν διαφορετικά template συνδεδεμένα σε μια δομή, ο χρήστης μπορεί να επιλέξει το template που διαμορφώνει την παρουσίαση του άρθρου με τον τρόπο που επιθυμεί.

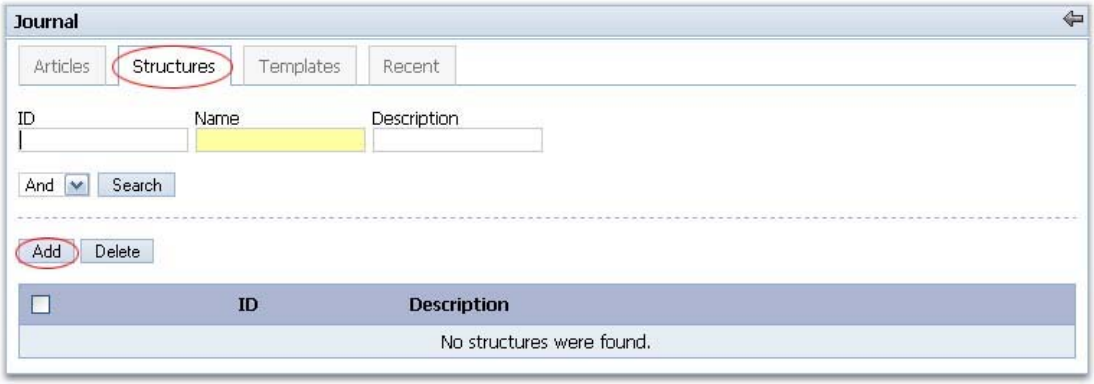
Δημιουργία ενός απλού άρθρου στο Liferay.

Το άρθρο το οποίο θα δημιουργήσουμε θα περιέχει ένα τίτλο (title), μια εικόνα (image) και το κυρίως κείμενο (body) όπως φαίνεται στην εικονα.



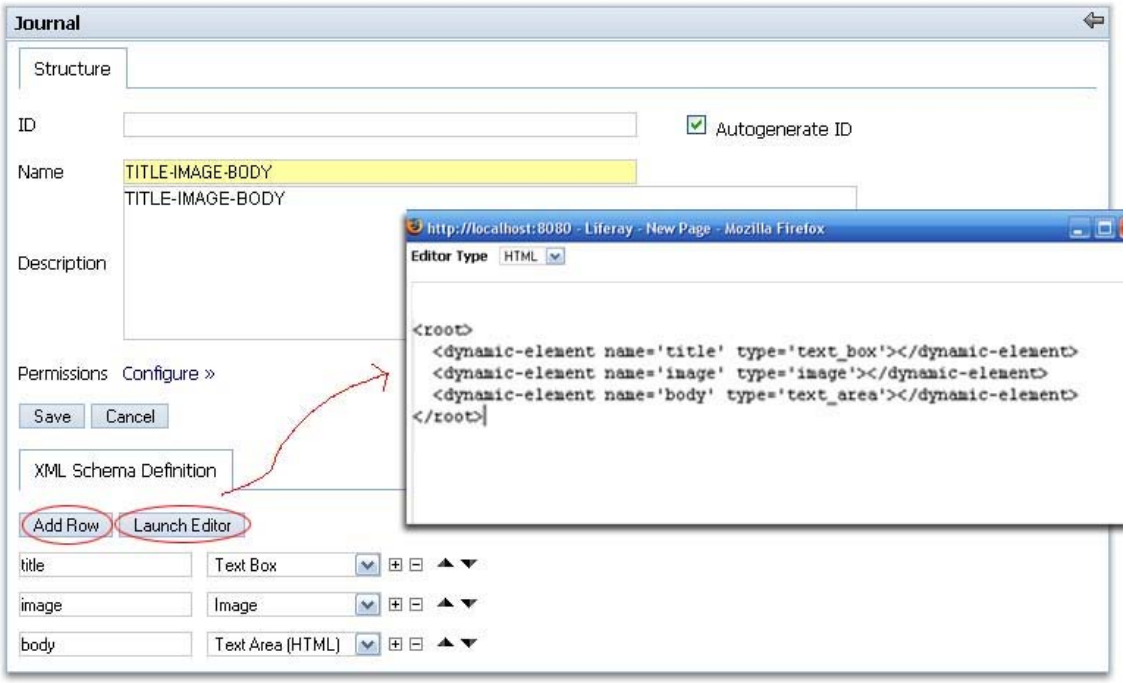
Εικόνα 4: Απλό άρθρο στο Lifery

Αρχικά όπως αναφέραμε παραπάνω, θα δημιουργήσουμε τη δομή (structure) του άρθρου. Για να προσθέσουμε ενα structure επιλέγουμε απο το Journal Portlet του Liferay την ετικέτα Structures.



Εικόνα 5: Δημιουργία δομής (structure)

Πατώντας "Add" εμφανίζεται η παρακάτω σελίδα δημιουργίας δομών.



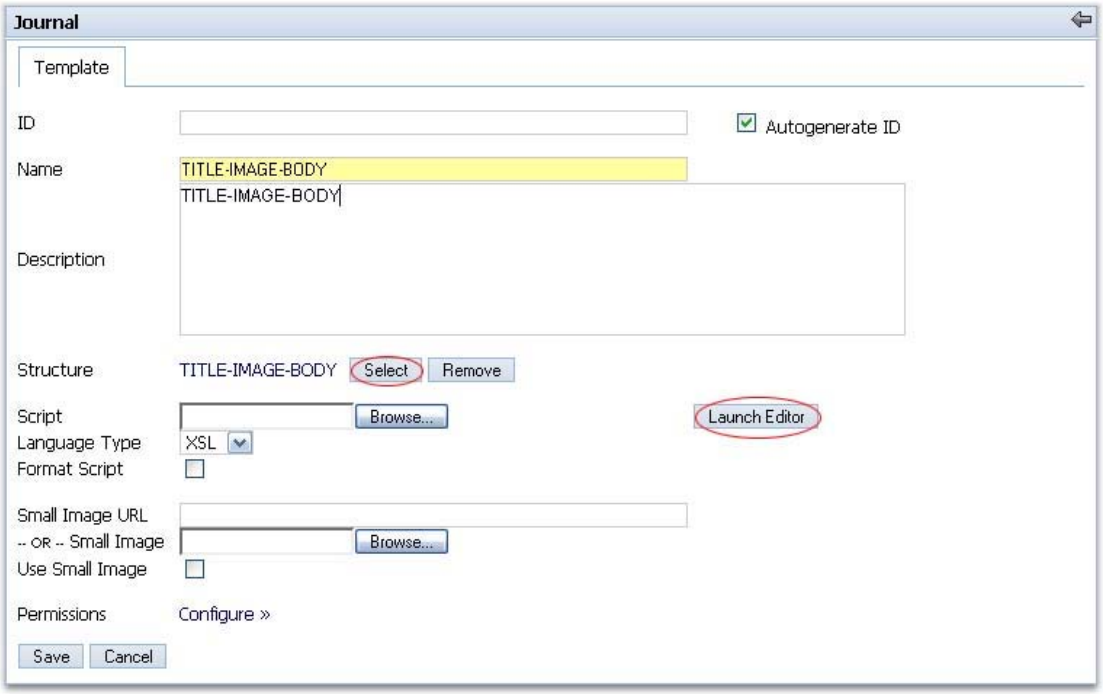
Εικόνα 6: Εισαγωγή των στοιχείων του άρθρου

Στη σελίδα αυτη καταχωρούμε τα στοιχεία του structure. Δίνουμε σε αυτό ενα ID ή επιλέγουμε Auto generate ώστε να δημιουργηθεί αυτόματα. Δίνουμε ένα σαφές όνομα και μια περιγραφή στα πεδία name και description αντίστοιχα. Αυτά τα πεδία χρησιμοποιούνται μαζι με το ID για τον προσδιορισμό του structure και ειναι

μοναδικά. Μετά την εισαγωγή αυτων των στοιχείων, καθορίζουμε τον αριθμό και τον τύπο στοιχείων που θα περιέχει το structure, μέσω της επιλογής “Add Row”.

Πατώντας “Add Row” εμφανίζεται κάθε φορά μια γραμμή με δυο νέα πεδία στα οποία εισάγουμε το όνομα του στοιχείου και τον τύπο του (οπως φαίνεται στην παραπάνω εικόνα το πρώτο στοιχείο ειναι ο τίτλος (title) και είναι τύπου Text Box). Αυτη η διαδικασία μπορέι να γίνει και μέσω της επιλογής “Launch Editor” χρησιμοποιώντας XML ή HTML. Και οι δυο μέθοδοι παράγουν το ίδιο αποτέλεσμα. Στη συνέχεια αποθηκεύουμε τη δομή επιλέγοντας “Save”.

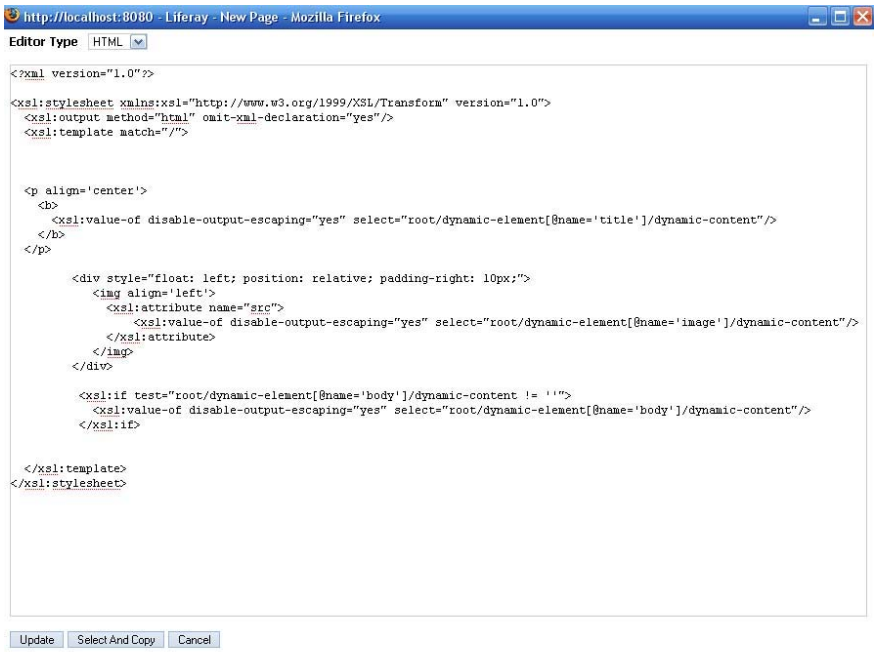
Αφού δημιουργήσουμε τη δομή, το επόμενο βήμα ειναι η δημιουργία του template. Απο το Journal Portlet επιλέγουμε την ετικέτα “Templates”. Πατώντας “Add” εμφανίζεται η σελίδα δημιουργίας templates.



Εικόνα 7: Δημιουργία template

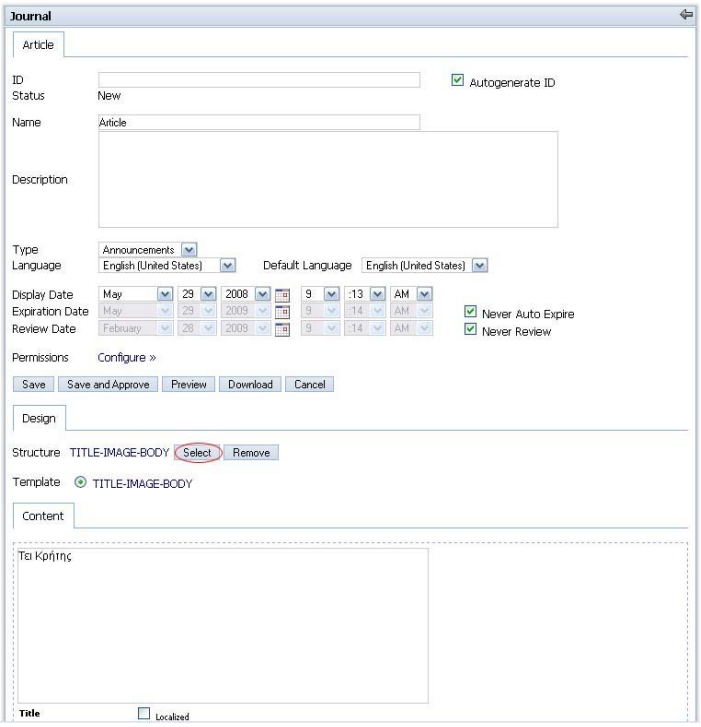
Εισάγουμε τα στοιχεία του template με τον ίδιο τρόπο που περιγράψαμε προηγουμένως στη δημιουργία του structure. Επιλέγουμε “Select” και βρίσκουμε το structure που φτιάξαμε προηγουμένως έτσι ώστε να το “σνδέσουμε” με το template.

Επιλέγουμε “Launch Editor” για να δημιουργήσουμε το template σε XSLT, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 8: Εισαγωγή κώδικα XSLT

Τέλος, για να δημιουργήσουμε το άρθρο επιλέγουμε απο το Journal Portlet την ετικέτα "Articles" και στη συνέχεια "Add". Στην σελίδα που εμφανίζεται πατώντας "Select" επιλέγουμε το structure και το template με τα οποία αυτό το άρθρο θα συνδεθεί. Στη συνέχεια εμφανίζονται τα πεδία περιεχομένου στα οποία εισάγουμε τον τίτλο, την εικόνα και το κυρίως κείμενο του άρθρου. Αποθηκεύουμε το άρθρο επιλέγοντας "Save and Approve".



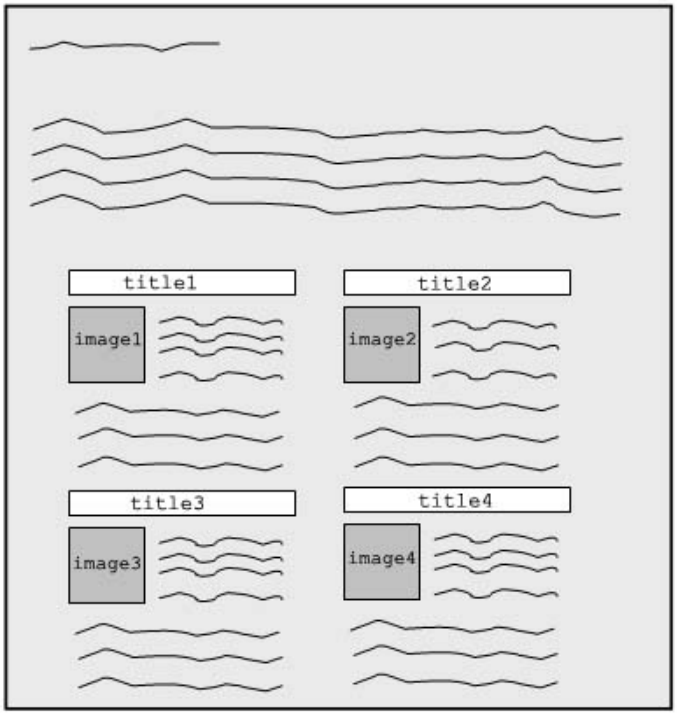
Εικόνα 9: Δημιουργία άρθρου

4 Ειδικά templates

4.1 Newspaper template

4.1.1 Όψη

Το Newspaper template παρουσιάζει ένα άρθο με μια δομή παρόμοια με αυτη μιας εφημερίδας. Αποτελέιται απο μια περιοχή για το κυρίως κείμενο και απο τέσσερα μικρότερα τμήματα απο τα οποια το καθένα αποτελείται απο ένα τίτλο, μια εικόνα και ενα μικρο κείμενο. Το template αυτο μπορεί να χρησιμοποιηθεί για την παρουσίαση των νέων ενος site ή κάποιων γεγονότων. Το πρότυπο αυτου του template παρουσιάζεται στην παρακάτω εικόνα.



Εικόνα 10: Όψη του προτύπου Newspaper

4.1.2 Κώδικας

Η δομή του συγκεκριμλενου περιγράμμαοτος ορίζεται ως εξής:

```
<root>
  <dynamic-element name='textarea' type='text_area'></dynamic-element>
  <dynamic-element name='title1' type='text'></dynamic-element>
  <dynamic-element name='image1' type='image'></dynamic-element>
  <dynamic-element name='caption1' type='text_box'></dynamic-element>
```

```
  <dynamic-element name='textarea1' type='text_area'></dynamic-element>
  <dynamic-element name='title2' type='text'></dynamic-element>
  <dynamic-element name='image2' type='image'></dynamic-element>
  <dynamic-element name='caption2' type='text_box'></dynamic-element>
  <dynamic-element name='textarea2' type='text_area'></dynamic-element>
  <dynamic-element name='title3' type='text'></dynamic-element>
  <dynamic-element name='image3' type='image'></dynamic-element>
  <dynamic-element name='caption3' type='text_box'></dynamic-element>
  <dynamic-element name='textarea3' type='text_area'></dynamic-element>
  <dynamic-element name='title4' type='text'></dynamic-element>
  <dynamic-element name='image4' type='image'></dynamic-element>
  <dynamic-element name='caption4' type='text_box'></dynamic-element>
  <dynamic-element name='textarea4' type='text_area'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="html" omit-xml-declaration="yes"/>
  <xsl:template match="/">

     <xsl:if test="root/dynamic-element[@name='textarea']/dynamic-content != ''">
          <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='textarea']/dynamic-content"/>
        </xsl:if>

<br /><br />

 <table width="90%" border="0" cellpadding="3" cellspacing="3" align="center" >


      <tr bgcolor="#9BC2E1">
        <td width="50%">
          <div align="center">
           <xsl:if test="root/dynamic-element[@name='title1']/dynamic-content != ''">
            <b><xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='title1']/dynamic-content"/></b>
           </xsl:if>
          </div>
        </td>


        <td width="50%">
           <div align="center">
             <xsl:if test="root/dynamic-element[@name='title2']/dynamic-content != ''">
               <b><xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='title2']/dynamic-content"/></b>
             </xsl:if>
```

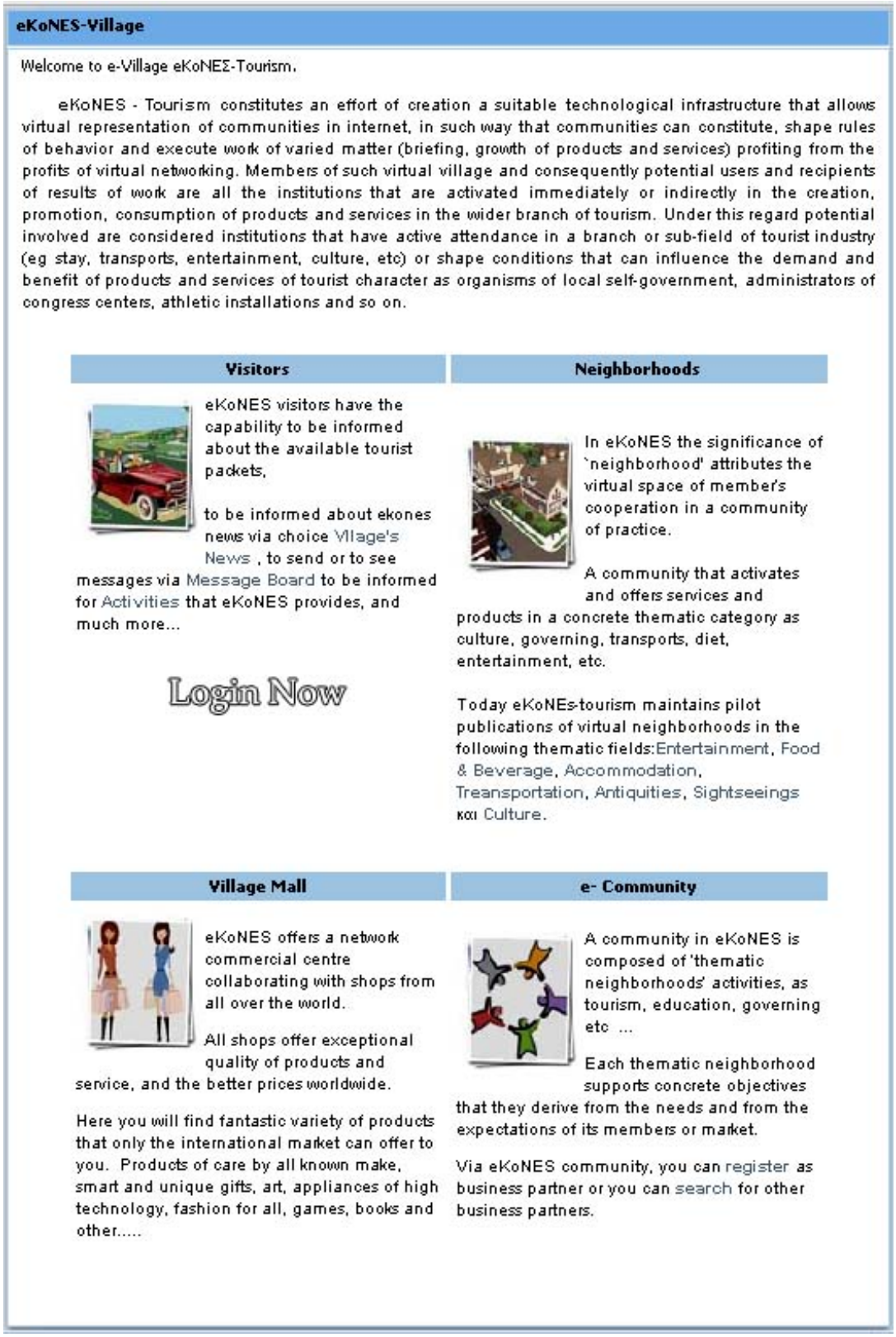
```
            </div>
          </td>
                
                
        </tr>
                
       <tr bgcolor="#FFFFFF">
         <td width="50%">
           <table width="35%" border="0" align="left">
             <tr>
               <td  valign="top">
             <xsl:if test="root/dynamic-element[@name='image1']/dynamic-content != ''">
                 <img  width="100%"  >
                   <xsl:attribute name="src">
                     <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image1']/dynamic-content"/>
                   </xsl:attribute>
                 </img>
                 
             </xsl:if>
               </td>
             </tr>
             <tr>
               <td align="center">
                 <b>
                   <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='caption1']/dynamic-content"/>
                 </b>
              </td>
             </tr>
           </table>
         <xsl:if test="root/dynamic-element[@name='textarea1']/dynamic-content != ''">
           <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='textarea1']/dynamic-content"/>
         </xsl:if>
                 
         </td>
                 
         <td width="50%">
           <table width="35%" border="0" align="left">
             <tr>
               <td  valign="top">
             <xsl:if test="root/dynamic-element[@name='image2']/dynamic-content != ''">
                 <img  width="100%"  >
                   <xsl:attribute name="src">
                     <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image2']/dynamic-content"/>
                   </xsl:attribute>
                 </img>
             </xsl:if>
               </td>
             </tr>
             <tr>
```

```
<td align="center">
<b>
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='caption2']/dynamic-content"/>
</b>
</td>
</tr>
</table>
<xsl:if test="root/dynamic-element[@name='textarea2']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='textarea2']/dynamic-content"/>
</xsl:if>
</td>
</tr>

<tr bgcolor="#9BC2E1">
<td width="50%">
<div align="center">
<xsl:if test="root/dynamic-element[@name='title3']/dynamic-content != ''">
<b> <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='title3']/dynamic-content"/> </b>
</xsl:if>
</div>
</td>

<td width="50%">
<div align="center">
<xsl:if test="root/dynamic-element[@name='title4']/dynamic-content != ''">
<b><xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='title4']/dynamic-content"/> </b>
</xsl:if>
</div>
</td>

</tr>

<tr bgcolor="#FFFFFF">
<td width="50%">
<table width="35%" border="0" align="left">
<tr>
<td  valign="top">
<xsl:if test="root/dynamic-element[@name='image3']/dynamic-content != ''">
<img  width="100%"  >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='image3']/dynamic-content"/>
</xsl:attribute>
</img>
</xsl:if>
</td>
```

```
            </tr>
            <tr>
              <td align="center">
                <b>
                  <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='caption3']/dynamic-content"/>
                </b>
              </td>
            </tr>
          </table>
        <xsl:if test="root/dynamic-element[@name='textarea3']/dynamic-content != ''">
          <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='textarea3']/dynamic-content"/>
        </xsl:if>
        </td>

        <td width="50%">
                  <table width="35%" border="0" align="left">
            <tr>
              <td  valign="top">
            <xsl:if test="root/dynamic-element[@name='image4']/dynamic-content != ''">
                <img  width="100%"  >
                  <xsl:attribute name="src">
                    <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image4']/dynamic-content"/>
                  </xsl:attribute>
                </img>
              </xsl:if>
              </td>
            </tr>
            <tr>
              <td align="center">
                <b>
                  <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='caption4']/dynamic-content"/>
                </b>
              </td>
            </tr>
          </table>
        <xsl:if test="root/dynamic-element[@name='textarea4']/dynamic-content != ''">
          <xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='textarea4']/dynamic-content"/>
        </xsl:if>
        </td>
      </tr>

</table>
  </xsl:template>
</xsl:stylesheet>
```

4.1.3 Εφαρμογή στο portal του eKoNES

Η αξιοποίηση του συγκεκριμλενου template για τις ανάγκες του έργου eKoNES παρουσιάζεται παρακάτω.



eKoNES-Village

Welcome to e-Village eKoNES-Tourism.

eKoNES - Tourism constitutes an effort of creation a suitable technological infrastructure that allows virtual representation of communities in internet, in such way that communities can constitute, shape rules of behavior and execute work of varied matter (briefing, growth of products and services) profiting from the profits of virtual networking. Members of such virtual village and consequently potential users and recipients of results of work are all the institutions that are activated immediately or indirectly in the creation, promotion, consumption of products and services in the wider branch of tourism. Under this regard potential involved are considered institutions that have active attendance in a branch or sub-field of tourist industry (eg stay, transports, entertainment, culture, etc) or shape conditions that can influence the demand and benefit of products and services of tourist character as organisms of local self-government, administrators of congress centers, athletic installations and so on.

Visitors

eKoNES visitors have the capability to be informed about the available tourist packets,

to be informed about ekones news via choice Village's News , to send or to see messages via Message Board to be informed for Activities that eKoNES provides, and much more...

Login Now

Neighborhoods

In eKoNES the significance of `neighborhood' attributes the virtual space of member's cooperation in a community of practice.

A community that activates and offers services and products in a concrete thematic category as culture, governing, transports, diet, entertainment, etc.

Today eKoNEs-tourism maintains pilot publications of virtual neighborhoods in the following thematic fields:Entertainment, Food & Beverage, Accommodation, Treansportation, Antiquities, Sightseeings και Culture.

Village Mall

eKoNES offers a network commercial centre collaborating with shops from all over the world.

All shops offer exceptional quality of products and service, and the better prices worldwide.

Here you will find fantastic variety of products that only the international market can offer to you. Products of care by all known make, smart and unique gifts, art, appliances of high technology, fashion for all, games, books and other.....

e- Community

A community in eKoNES is composed of 'thematic neighborhoods' activities, as tourism, education, governing etc ...

Each thematic neighborhood supports concrete objectives that they derive from the needs and from the expectations of its members or market.

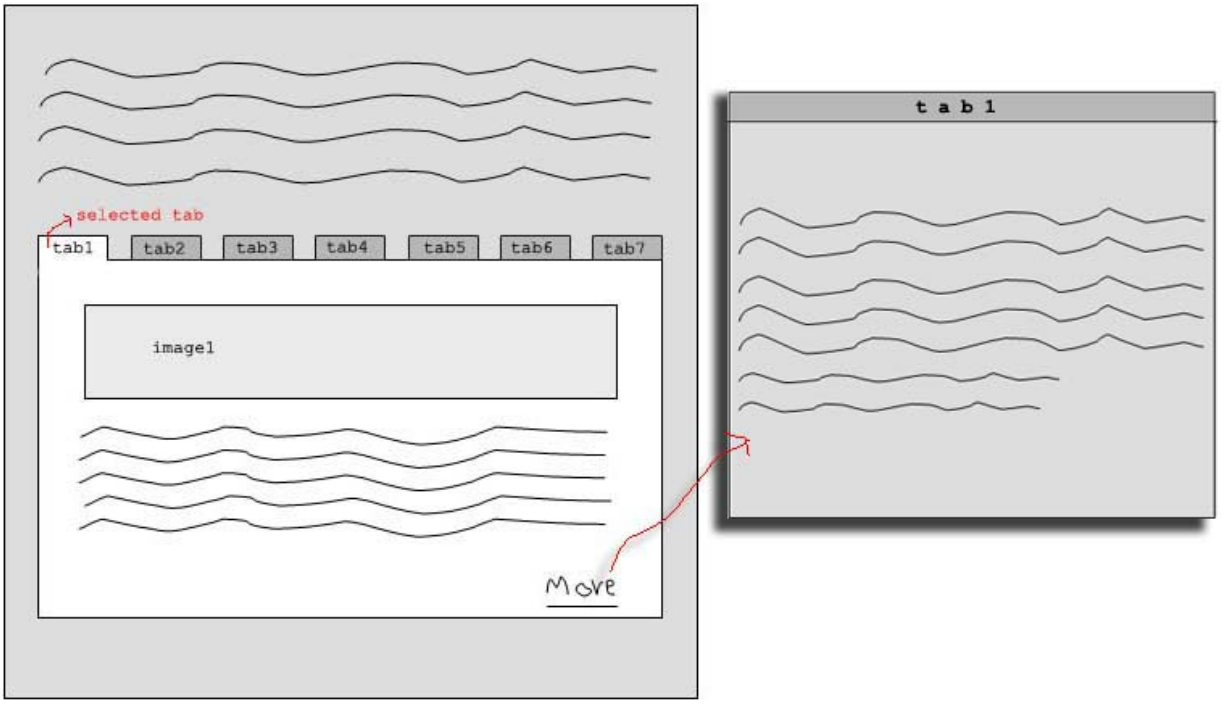
Via eKoNES community, you can register as business partner or you can search for other business partners.

Εικόνα 11: Newspaper template στο eKoNES

4.2 Body-and-tabs template

4.2.1 Όψη

Το template αυτο αποτελεί ένα προκατασκευασμένο συστατικό που συναθροίζει επτά (7) καρτέλες (tabs) οι οποίες με την ενεργοποίηση τους εμφανίζουν συγκεκριμένο κείμενο. Αποτελείται απο μια περιοχή για το κυρίως κείμενο και απο μια άλλη που περιέχει τα tabs.



Εικόνα 12: Όψη του προτύπου Body-and-tabs

4.2.2 Κώδικας

Structure

```
<root>
  <dynamic-element name='textarea' type='text_area'></dynamic-element>
  <dynamic-element name='tab1' type='text_box'></dynamic-element>
  <dynamic-element name='body1' type='text_area'></dynamic-element>
  <dynamic-element name='tab2' type='text_box'></dynamic-element>
  <dynamic-element name='body2' type='text_area'></dynamic-element>
  <dynamic-element name='tab3' type='text_box'></dynamic-element>
  <dynamic-element name='body3' type='text_area'></dynamic-element>
  <dynamic-element name='tab4' type='text_box'></dynamic-element>
  <dynamic-element name='body4' type='text_area'></dynamic-element>
  <dynamic-element name='tab5' type='text_box'></dynamic-element>
  <dynamic-element name='body5' type='text_area'></dynamic-element>
  <dynamic-element name='tab6' type='text_box'></dynamic-element>
```

```
  <dynamic-element name='body6' type='text_area'></dynamic-element>
  <dynamic-element name='tab7' type='text_box'></dynamic-element>
  <dynamic-element name='body7' type='text_area'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="html" omit-xml-declaration="yes"/>
  <xsl:template match="/">

  <HTML><HEAD>

<script language="javascript">
<![CDATA[

function ddtabcontent(tabinterfaceid){
        this.tabinterfaceid=tabinterfaceid
        this.tabs=document.getElementById(tabinterfaceid).getElementsByTagName("a
")
        this.enabletabpersistence=true
        this.hottabspositions=[]
        this.currentTabIndex=0
        this.subcontentids=[]
        this.revcontentids=[]
        this.selectedClassTarget="link"
}

ddtabcontent.getCookie=function(Name){
        var re=new RegExp(Name+"=[^;]+", "i");
        if (document.cookie.match(re))
                return document.cookie.match(re)[0].split("=")[1]
        return ""
}

ddtabcontent.setCookie=function(name, value){
        document.cookie = name+"="+value+";path=/"
}

ddtabcontent.prototype={

        expandit:function(tabid_or_position){
                this.cancelautorun()
                var tabref=""
                try{
                        if (typeof tabid_or_position=="string" &&
document.getElementById(tabid_or_position).getAttribute("rel"))
                                tabref=document.getElementById(tabid_or_position)
                        else if (parseInt(tabid_or_position)!=NaN &&
this.tabs[tabid_or_position].getAttribute("rel"))
```

```
                                        tabref=this.tabs[tabid_or_position]
                        }
                        catch(err){alert("Invalid Tab ID or position entered!")}
                        if (tabref!="")
                                this.expandtab(tabref)
                },

                cycleit:function(dir, autorun){
                        if (dir=="next"){
                                var
currentTabIndex=(this.currentTabIndex<this.hottabspositions.length-1)?
this.currentTabIndex+1 : 0
                        }
                        else if (dir=="prev"){
                                var currentTabIndex=(this.currentTabIndex>0)?
this.currentTabIndex-1 : this.hottabspositions.length-1
                        }
                        if (typeof autorun=="undefined")
                                this.cancelautorun()
                        this.expandtab(this.tabs[this.hottabspositions[currentTabIndex]])
                },

                setpersist:function(bool){
                                this.enabletabpersistence=bool
                },

                setselectedClassTarget:function(objstr){
                        this.selectedClassTarget=objstr || "link"
                },

                getselectedClassTarget:function(tabref){
                        return (this.selectedClassTarget==("linkparent".toLowerCase()))?
tabref.parentNode : tabref
                },

                expandtab:function(tabref){
                        var subcontentid=tabref.getAttribute("rel")
                        //Get "rev" attr as a string of IDs in the format ",john,george,trey,etc,"
to easily search through
                        var associatedrevids=(tabref.getAttribute("rev"))?
","+tabref.getAttribute("rev").replace(/\s+/, "")+"," : ""
                        this.expandsubcontent(subcontentid)
                        this.expandrevcontent(associatedrevids)
                        for (var i=0; i<this.tabs.length; i++){

                this.getselectedClassTarget(this.tabs[i]).className=(this.tabs[i].getAttribute("re
l")==subcontentid)? "selected" : ""
                        }
                        if (this.enabletabpersistence)
                                ddtabcontent.setCookie(this.tabinterfaceid, tabref.tabposition)
                        this.setcurrenttabindex(tabref.tabposition)
                },

```

```
expandsubcontent:function(subcontentid){
        for (var i=0; i<this.subcontentids.length; i++){
                var subcontent=document.getElementById(this.subcontentids[i])
                subcontent.style.display=(subcontent.id==subcontentid)?
"block" : "none"
        }
},

expandrevcontent:function(associatedrevids){
        var allrevids=this.revcontentids
        for (var i=0; i<allrevids.length; i++){


        document.getElementById(allrevids[i]).style.display=(associatedrevids.indexOf(
","+allrevids[i]+",")!=-1)? "block" : "none"
        }
},

setcurrenttabindex:function(tabposition){
        for (var i=0; i<this.hottabspositions.length; i++){
                if (tabposition==this.hottabspositions[i]){
                        this.currentTabIndex=i
                        break
                }
        }
},

autorun:function(){
        this.cycleit('next', true)
},

cancelautorun:function(){
        if (typeof this.autoruntimer!="undefined")
                clearInterval(this.autoruntimer)
},

init:function(automodeperiod){
        var persistedtab=ddtabcontent.getCookie(this.tabinterfaceid)
        var persisterror=true
        this.automodeperiod=automodeperiod || 0
        for (var i=0; i<this.tabs.length; i++){
                this.tabs[i].tabposition=i //remember position of tab relative to its
peers
                if (this.tabs[i].getAttribute("rel")){
                        var tabinstance=this
                        this.hottabspositions[this.hottabspositions.length]=i


        this.subcontentids[this.subcontentids.length]=this.tabs[i].getAttribute("rel")
                        this.tabs[i].onclick=function(){
                                tabinstance.expandtab(this)
                                tabinstance.cancelautorun()
```

```
                                        return false
                                    }
                                    if (this.tabs[i].getAttribute("rev")){ //if "rev" attr
defined, store each value within "rev" as an array element

        this.revcontentids=this.revcontentids.concat(this.tabs[i].getAttribute("rev").split
(/\s*,\s*/))
                                    }
                                    if (this.enabletabpersistence &&
parseInt(persistedtab)==i || !this.enabletabpersistence &&

this.getselectedClassTarget(this.tabs[i]).className=="selected"){
                                        this.expandtab(this.tabs[i])
                                        persisterror=false
                                    }
                                }
                            }
                            if (persisterror) //if an error has occured while trying to retrieve
persisted tab (based on its position within its peers)
                                this.expandtab(this.tabs[this.hottabspositions[0]]) //Just select
first tab that contains a "rel" attr
                            if (parseInt(this.automodeperiod)>500 &&
this.hottabspositions.length>1){
                                this.autoruntimer=setInterval(function(){tabinstance.autorun()},
this.automodeperiod)
                            }
                        }

}
]]>

</script>
<style type="text/css">

.shadetabs{
padding: 3px 0;
margin-left: 0;
margin-top: 1px;
margin-bottom: 0;
font: bold 8.5px Verdana;
list-style-type: none;
text-align: left; /*set to left, center, or right to align the menu as desired*/
}

.shadetabs li{
display: inline;
margin: 0;
}

.shadetabs li a{
text-decoration: none;
position: relative;
z-index: 1;
```

```
padding: 3px 7px;
margin-right: 3px;
border: 1px solid #778;
color: #2d2b2b;
background-color: #6CAAE7;
}

.shadetabs li a:visited{
color: #2d2b2b;
}

.shadetabs li a:hover{
text-decoration: underline;
color: #2d2b2b;
}

.shadetabs li a.selected{ /*selected main tab style */
position: relative;
top: 1px;
}

.shadetabs li a.selected{ /*selected main tab style */
background-color: white;
border-bottom-color: transparent;
}

.shadetabs li a.selected:hover{ /*selected main tab style */
text-decoration: none;
}

.tabcontent{
display:none;
width:450px;
}

@media print {
.tabcontent {
display:block !important;
}
}


</style>
</HEAD>
<BODY>

    <xsl:if test="root/dynamic-element[@name='textarea']/dynamic-content != """>
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='textarea']/dynamic-content"/>
      </xsl:if>

<br /><br />
<ul id="countrytabs" class="shadetabs">
```

```
<li><a href="#" rel="country1" class="selected"> <xsl:if test="root/dynamic-element[@name='tab1']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab1']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country2"><xsl:if test="root/dynamic-element[@name='tab2']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab2']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country3"><xsl:if test="root/dynamic-element[@name='tab3']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab3']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country4"><xsl:if test="root/dynamic-element[@name='tab4']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab4']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country5"><xsl:if test="root/dynamic-element[@name='tab5']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab5']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country6"><xsl:if test="root/dynamic-element[@name='tab6']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab6']/dynamic-content"/>
      </xsl:if></a></li>
<li><a href="#" rel="country7"><xsl:if test="root/dynamic-element[@name='tab7']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab7']/dynamic-content"/>
      </xsl:if></a></li>

</ul>

<div style="border:1px solid gray; width:500px; height:290px; margin-bottom: 1em; padding: 10px; background-color:white">

<div id="country1" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body1']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body1']/dynamic-content"/>
      </xsl:if>
</div>

<div id="country2" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body2']/dynamic-content != "">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body2']/dynamic-content"/>
      </xsl:if>
```

```
</div>

<div id="country3" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body3']/dynamic-content != ''">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body3']/dynamic-content"/>
      </xsl:if>
</div>
<div id="country4" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body4']/dynamic-content != ''">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body4']/dynamic-content"/>
      </xsl:if>
</div>
<div id="country5" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body5']/dynamic-content != ''">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body5']/dynamic-content"/>
      </xsl:if>
</div>
<div id="country6" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body6']/dynamic-content != ''">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body6']/dynamic-content"/>
      </xsl:if>
</div>
<div id="country7" class="tabcontent">
<xsl:if test="root/dynamic-element[@name='body7']/dynamic-content != ''">
        <xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body7']/dynamic-content"/>
      </xsl:if>
</div>


</div>

<script language="javascript">

var countries=new ddtabcontent("countrytabs")
countries.setpersist(true)
countries.setselectedClassTarget("link") //"link" or "linkparent"
countries.init()

</script>
</BODY></HTML>
  </xsl:template>
</xsl:stylesheet>
```

4.2.3 Εφαρμογή στο portal του eKoNES

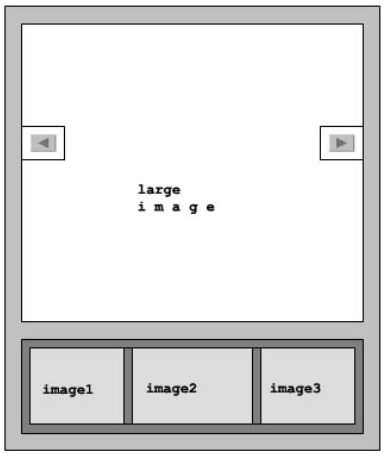
Η αξιοποίηση του συγκεκριμλενου template για τις ανάγκες του έργου eKoNEΣ παρουσιάζεται παρακάτω.

Εικόνα 13: Body-and-tabs template στο eKoNES

4.3 Image Gallery

4.3.1 Όψη

Το template αυτο εξυπηρετεί την οπτική παράσταση ενός θέματος μέσω εικόνων καθώς και την γρήγορη εναλλαγή από εικόνα σε εικόνα. Ο χρήστης μπορεί να χρησιμοποιήσει μια συσκευή κατάδειξης όπως το ποντίκι και να επιλέξει μια εικόνα με απευθείας χειρισμό ή να επιλέξει τα κουμπιά κατεύθυνσης του πληκτρολογίου για να μετακινηθεί από μια εικόνα σε μια άλλη. Στο κάτω μέρος υπάρχουν οι μικρογραφίες των εικόνων .



Εικόνα 14: Όψη του προτύπου image gallery

4.3.2 Κώδικας

Structure

```
<root>
  <dynamic-element name='image1' type='image'></dynamic-element>
  <dynamic-element name='image2' type='image'></dynamic-element>
  <dynamic-element name='image3' type='image'></dynamic-element>
  <dynamic-element name='image4' type='image'></dynamic-element>
  <dynamic-element name='image5' type='image'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html" omit-xml-declaration="yes"/>
<xsl:template match="/">


<html>
<head>


<style type="text/css">
* {margin:0; padding:0}

#gallery {width:486px; margin:40px auto; padding:10px; border:6px solid #8D8E8F; background:#D1D2D3}
#images {border:2px solid #9ac1c9; height:225px; background:#eef5f6; margin-bottom:20px}
#image {position:relative; width:472px; height:375px; padding:5px; border:2px solid #20364C; background:#FFF url(/html/js/fullsize/images/loading.gif) center center no-repeat}
#image img {position:absolute; top:5px; left:5px; width:472px; height:375px}
#thumbwrapper {margin-top:10px; padding:5px 0; width:482px; height:100px; border:2px solid #20364C; background:#FFF}
#thumbarea {position:relative; overflow:hidden; height:100px; width:482px; }
#thumbs {position:absolute; list-style:none; margin-left:5px; height:100px; width:5000px; border-right:5px solid #FFF}
#thumbs li {float:left; margin-right:5px; cursor:pointer}
.imgnav {position:absolute; height:375px; width:20%; z-index:100; height:375px; width:20%; z-index:100; outline:none; cursor:pointer}
#previmg {left:0; background:url(/html/js/fullsize/images/left.gif) left center no-repeat; border-left:5px solid #FFF}
#previmg:hover {opacity:1; filter:alpha(opacity=100)}
#nextimg {right:0; background:url(/html/js/fullsize/images/right.gif) right center no-repeat; border-right:5px solid #FFF}
#nextimg:hover {opacity:1; filter:alpha(opacity=100)}
</style>
```

```
</head>
<body onload="slideShow()">

<div id="gallery">
<div id="imagearea">
<div id="image">
<a href="javascript:imgNav(-1)" class="imgnav " id="previmg"></a>
<a href="javascript:imgNav(1)" class="imgnav " id="nextimg"></a>
</div>
</div>
<div id="thumbwrapper">
<div id="thumbarea">
<ul id="thumbs">
<li value="1"><img name="image1"  width="179" height="100"><xsl:attribute
name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image1']/dynamic-content"/>
</xsl:attribute>
</img></li>
<li value="2"><img name="image1"  width="179" height="100"><xsl:attribute
name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image2']/dynamic-content"/>
</xsl:attribute>
</img></li>
<li value="3"><img name="image1"  width="179" height="100"><xsl:attribute
name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image3']/dynamic-content"/>
</xsl:attribute>
</img></li>
<li value="4"><img name="image1"  width="179" height="100"><xsl:attribute
name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image4']/dynamic-content"/>
</xsl:attribute>
</img></li>
<li value="5"><img name="image1"  width="179" height="100"><xsl:attribute
name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image5']/dynamic-content"/>
</xsl:attribute>
</img></li>
</ul>
</div>
</div>
</div>

<script language="javascript">
<![CDATA[
var imgid = 'image';
```

```
var imgdir = '/html/js/fullsize';
var imgext = '.jpg';
var thumbid = 'thumbs';
var auto = true;
var autodelay = 5;
]]>
</script>
<script language="javascript" >
<![CDATA[
var bxs,bxe,fxs,fxe,ys,ye,ta,ia,ie,st,ss,ft,fs,xp,yp,ci,t,tar,tarl;
ta = document.getElementById(thumbid);
ia = document.getElementById(imgid);
t = ta.getElementsByTagName('li')
ie = document.all ? true : false;
st = 3;
ss = 3;
ft = 10;
fs = 5;
xp,yp = 0;
document.onmousemove = getPos;

function slideShow(){
var taw = ta.parentNode.offsetWidth;
var taa = taw / 4;
bxs = leftPos(ta);
bxe = bxs + taa;
fxe = bxs + taw;
fxs = fxe - taa;
ys = topPos(ta);
ye = ys + ta.offsetHeight;
var len = t.length;
tar = [];
for(i=0; i < len; i++){
var id = t[i].value;
tar[i] = id;
t[i].onclick = new Function("getImg('" + id + "')");
if(i == 0) {
getImg(id);
}
}
tarl = tar.length;
}

function scrlThumbs(d){
clearInterval(ta.timer);
var l;
if(d == -1){
l = 0;
}else{
l = t[tarl-1].offsetLeft - (ta.parentNode.offsetWidth - t[tarl-1].offsetWidth) + 10;
}
```

```
ta.timer = setInterval(function(){scrlMv(d,l)},st);
}

function scrlMv(d,l){
ta.style.left = ta.style.left || '0px';
var left = ta.style.left.replace('px','');
if(d == 1){
if(l - Math.abs(left) <= ss){
cnclScrl(ta.id);
ta.style.left = '-' + l + 'px';
}else{
ta.style.left = left - ss + 'px';
}
}else{
if(Math.abs(left) - l <= ss){
cnclScrl(ta.id);
ta.style.left = l + 'px';
}else{
ta.style.left = parseInt(left) + ss + 'px';
}
}
}

function cnclScrl(){clearTimeout(ta.timer)}

function getImg(id){
if(auto){clearTimeout(ia.timer)}
if(ci != null){
var ts = ia.getElementsByTagName('img');
var tsl = ts.length;
var x = 0;
for(x; x < tsl; x++){
if(ci.id != id){
var o = ts[x];
clearInterval(o.timer);
o.timer = setInterval(function(){fdOut(o)},fs);
}
}
}
var i;
if(!document.getElementById(id)){
i = document.createElement('img');
ia.appendChild(i);
i.id = id;
i.av = 0;
i.style.opacity = 0;
i.style.filter = 'alpha(opacity=0)';
i.src = imgdir + '/' + id + imgext;

}else{
i = document.getElementById(id);
```

```
clearInterval(i.timer);
}
i.timer = setInterval(function(){fdIn(i)},fs);
}

function imgNav(d){
var curr = 0;
for(key in tar){
if(tar[key] == ci.id){
curr = key;
}
}
if(tar[parseInt(curr) + d]){
getImg(tar[parseInt(curr) + d]);
}else{
if(d == 1){
getImg(tar[0]);
}else{
getImg(tar[tarl - 1]);
}
}
}

function autoSlide(){
ia.timer = setInterval(function(){imgNav(1)}, autodelay * 1000);
}

function fdIn(i){
if(i.complete){
i.av = i.av + fs;
i.style.opacity = i.av / 100;
i.style.filter = 'alpha(opacity=' + i.av + ')';
}
if(i.av >= 100){
if(auto){autoSlide()}
clearInterval(i.timer);
ci = i;
}
}

function fdOut(i){
i.av = i.av - fs;
i.style.opacity = i.av / 100;
i.style.filter = 'alpha(opacity=' + i.av + ')';
if(i.av <= 0){
clearInterval(i.timer);
if(i.parentNode){i.parentNode.removeChild(i)}
}
}

function getPos(e){
```

```
if(ie){
xp = event.clientX + document.body.scrollLeft;
yp = event.clientY + document.body.scrollTop;
}else{
xp = e.pageX;
yp = e.pageY;
}
if(xp < 0){xp = 0}
if(yp < 0){yp = 0}
if(xp > bxs && xp < bxe && yp > ys && yp < ye){
scrlThumbs(-1);
}else if(xp > fxs && xp < fxe && yp > ys && yp < ye){
scrlThumbs(1);
}else{
cnclScrl();
}
}

function leftPos(t){
var left = 0;
if(t.offsetParent){
while(1){
left += t.offsetLeft;
if(!t.offsetParent){break}
t = t.offsetParent;
}
}else if(t.x){
left += t.x;
}
return left;
}

function topPos(t){
var top = 0;
if(t.offsetParent){
while(1){
top += t.offsetTop;
if(!t.offsetParent){break}
t = t.offsetParent;
}
}else if(t.y){
top += t.y;
}
return top;
}

]]>
</script>

</body>
</html>
```

```
</xsl:template>
</xsl:stylesheet>
```

4.3.3 Εφαρμογή στο Portal του Τομέα Τηλεπικοινωνιών & Πολυμέσων

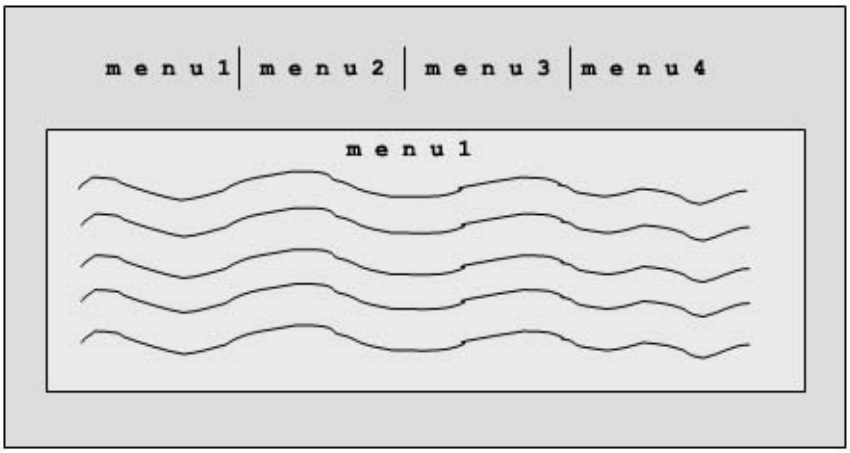
Η αξιοποίηση του συγκεκριμένου template για τις ανάγκες του Τομέα Τηλεπικοινωνιών & Πολυμέσων παρουσιάζεται παρακάτω.

Εικόνα 15: Image Gallery στον Τομέα Τηλεπ/νιων & Πολυμέσων

4.4 Slider Content

4.4.1 Όψη

Ο στόχος του template Slider Content είναι να προσφέρει μια εμφωλευμένη δομή επιλογής και επισκόπησης περιεχομένου. Η δομή επιλογής βρίσκεται στην κορυφή και αποτελείται από τους τίτλους που αντιστοιχούν σε περιοχές κειμένου. Επιλέγοντας από τη δομή επιλογής, η περιοχή κειμένου ενημερώνεται εμφανίζοντας το αντίστοιχο κείμενο.



Εικόνα 16: Όψη του προτύπου Slider Content

4.4.2 Κώδικας

Structure

```
<root>
  <dynamic-element name='tab1' type='text_box'></dynamic-element>
  <dynamic-element name='content1' type='text_area'></dynamic-element>
  <dynamic-element name='tab2' type='text_box'></dynamic-element>
  <dynamic-element name='content2' type='text_area'></dynamic-element>
  <dynamic-element name='tab3' type='text_box'></dynamic-element>
  <dynamic-element name='content3' type='text_area'></dynamic-element>
  <dynamic-element name='tab4' type='text_box'></dynamic-element>
  <dynamic-element name='content4' type='text_area'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html" omit-xml-declaration="yes"/>
<xsl:template match="/">

<html>
<head>

<style type="text/css">

#wrapper {width:508px; margin:0 auto}
h2 {font-size:14px; margin:15px 0}
.link {cursor:pointer; color:#666}
.link:hover {color:#000}
.slider {position:relative; overflow:hidden; border:2px solid #9ac1c9; height:120px;
background:#eef5f6; margin-bottom:20px}
```

```
.slidercontent {position:absolute}
.slidercontent .upper {border-bottom:1px solid #9ac1c9}
.section {padding:10px}
.newsslider {position:relative; overflow:hidden; border:2px solid #9ac1c9; height:95px;
width:300px; background:#eef5f6; margin-bottom:20px}
.newsslidercontent {position:absolute}
.newsslidercontent .upper {border-bottom:1px solid #9ac1c9}
.newssection {padding:10px; height:75px}
</style>

<script type="text/javascript" >
<![CDATA[
var SLIDETIMER = 3;
var SLIDESPEED = 3;
var SCROLLTIMER = 3;
var SCROLLSPEED = 3;
var STARTINGOPACITY = 40;


function slideContent(id,prefix,timer) {
var div = document.getElementById(id);
var slider = div.parentNode;
clearInterval(slider.timer);
slider.section = parseInt(id.replace(/\D/g,''));
slider.target = div.offsetTop;
slider.style.top = slider.style.top || '0px';
slider.current = slider.style.top.replace('px','');
slider.direction = (Math.abs(slider.current) > slider.target) ? 1 : -1;
slider.style.opacity = STARTINGOPACITY * .01;
slider.style.filter = 'alpha(opacity=' + STARTINGOPACITY + ')';
slider.timer = setInterval( function() { slideAnimate(slider,prefix,timer) },
SLIDETIMER);
}

function slideAnimate(slider,prefix,timer) {
var curr = Math.abs(slider.current);
var tar = Math.abs(slider.target);
var dir = slider.direction;
if((tar - curr <= SLIDESPEED && dir == -1) || (curr - tar <= SLIDESPEED && dir ==
1)) {
slider.style.top = (slider.target * -1) + 'px';
slider.style.opacity = 1;
slider.style.filter = 'alpha(opacity=100)';
clearInterval(slider.timer);
if(slider.autoscroll) {
setTimeout( function() { autoScroll(slider.id,prefix,timer) }, timer * 1000);
}
} else {
var pos = (dir == 1) ? parseInt(slider.current) + SLIDESPEED : slider.current -
SLIDESPEED;
slider.current = pos;
```

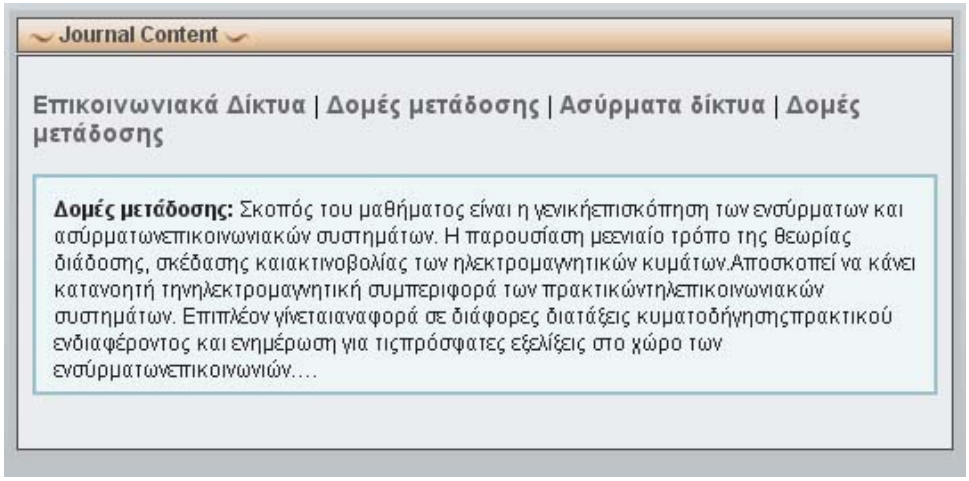
```
slider.style.top = pos + 'px';
}
}


function scrollContent(id,dir) {
var div = document.getElementById(id);
clearInterval(div.timer);
var sections = div.getElementsByTagName('div');
var length = sections.length;
var limit;
if(dir == -1) {
limit = 0;
} else {
if(length > 1) {
limit = sections[length-1].offsetTop;
} else {
limit = sections[length-1].offsetHeight - div.parentNode.offsetHeight + 20;
}
}
div.style.opacity = STARTINGOPACITY * .01;
div.style.filter = 'alpha(opacity=' + STARTINGOPACITY + ')';
div.timer = setInterval( function() { scrollAnimate(div,dir,limit) }, SCROLLTIMER);
}

function scrollAnimate(div,dir,limit) {
div.style.top = div.style.top || '0px';
var top = div.style.top.replace('px','');
if(dir == 1) {
if(limit - Math.abs(top) <= SCROLLSPEED) {
cancelScroll(div.id);
div.style.top = '-' + limit + 'px';
} else {
div.style.top = top - SCROLLSPEED + 'px';
}
} else {
if(Math.abs(top) - limit <= SCROLLSPEED) {
cancelScroll(div.id);
div.style.top = limit + 'px';
} else {
div.style.top = parseInt(top) + SCROLLSPEED + 'px';
}
}
}


function cancelScroll(id) {
var div = document.getElementById(id);
div.style.opacity = 1;
div.style.filter = 'alpha(opacity=100)';
clearTimeout(div.timer);
```

```
}

function autoScroll(id,prefix,timer,restart) {
var div = document.getElementById(id);
div.autoscroll = (!div.autoscroll && !restart) ? false : true;
if(div.autoscroll) {
var sections = div.getElementsByTagName('div');
var length = sections.length;
div.section = (div.section && div.section < length) ? div.section + 1 : 1;
slideContent(prefix + '-' + div.section,prefix,timer);
}
}

function cancelAutoScroll(id) {
var div = document.getElementById(id);
div.autoscroll = false;
}

]]>
</script>
</head>
<body onload="autoScroll('newsslider','newssection',5,true)">
<div id="wrapper">

<h2><span class="link" onmousedown="slideContent('section-1')"><xsl:if test="root/dynamic-element[@name='tab1']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab1']/dynamic-content"/>
</xsl:if></span> | <span class="link" onmousedown="slideContent('section-2')"><xsl:if test="root/dynamic-element[@name='tab2']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab2']/dynamic-content"/>
</xsl:if></span> | <span class="link" onmousedown="slideContent('section-3')"><xsl:if test="root/dynamic-element[@name='tab3']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab3']/dynamic-content"/>
</xsl:if></span>  | <span class="link" onmousedown="slideContent('section-4')"><xsl:if test="root/dynamic-element[@name='tab4']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='tab4']/dynamic-content"/>
</xsl:if></span></h2>
<div class="slider">
<div class="slidercontent" id="slider">
<div id="section-1" class="section upper">
<xsl:if test="root/dynamic-element[@name='content1']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='content1']/dynamic-content"/>
```

```
</xsl:if>
</div>
<div id="section-2" class="section upper">
<xsl:if test="root/dynamic-element[@name='content2']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='content2']/dynamic-content"/>
</xsl:if>
</div>
<div id="section-3" class="section">
<xsl:if test="root/dynamic-element[@name='content3']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='content3']/dynamic-content"/>
</xsl:if>
</div>
<div id="section-4" class="section">
<xsl:if test="root/dynamic-element[@name='content4']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='content4']/dynamic-content"/>
</xsl:if>
</div>
</div>
</div>

</div>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

4.4.3 Εφαρμογή στο portal του Τομέα Τηλεπικοινωνιών & Πολυμέσων

Η αξιοποίηση του συγκεκριμλενου template για τις ανάγκες του Τομέα Τηλεπικοινωνιών & Πολυμέσων παρουσιάζεται παρακάτω.

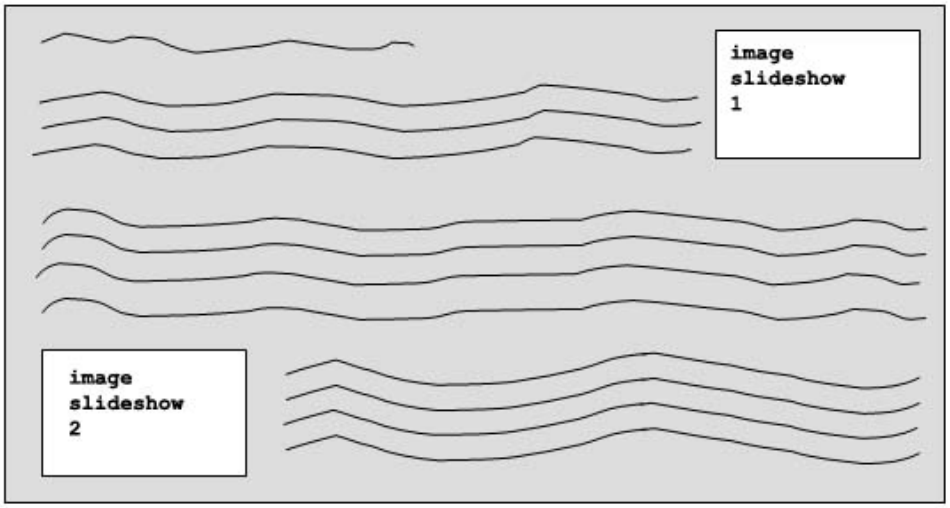


Εικόνα 17: Slider Content στον Τομέα Τηλεπ/νιων &Πολυμέσων

4.5 Body-and-2-fade-slideshows template

4.5.1 Όψη

Το template αυτο περιέχει τρείς περιοχές για κείμενο και δυο slideshows (επάνω δεξιά και κάτω αριστερά). Αυτα τα slideshows χρησιμοποιούν ενα fade in - fade out εφέ για την εναλλαγή των εικόνων η οποία γίνεται αυτόματα.



Εικόνα 18: Όψη του προτύπου Body-and-2-fade-slideshows

4.5.2 Κώδικας

Structure

```
<root>
  <dynamic-element name='title' type='text_box'></dynamic-element>
  <dynamic-element name='body1' type='text_area'></dynamic-element>
  <dynamic-element name='body2' type='text_area'></dynamic-element>
  <dynamic-element name='body3' type='text_area'></dynamic-element>
  <dynamic-element name='image4' type='image'></dynamic-element>
  <dynamic-element name='image5' type='image'></dynamic-element>
  <dynamic-element name='image6' type='image'></dynamic-element>
  <dynamic-element name='image7' type='image'></dynamic-element>
  <dynamic-element name='image8' type='image'></dynamic-element>
  <dynamic-element name='image1' type='image'></dynamic-element>
  <dynamic-element name='image2' type='image'></dynamic-element>
  <dynamic-element name='image3' type='image'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html" omit-xml-declaration="yes"/>
<xsl:template match="/">

<html>
<head>

<style type="text/css">

.contentContainer{
padding-left:10px;
padding-right:10px;
padding-bottom:10px;
}
.contentContainer p,.contentContainer h1{
margin-top:0px;
}

.imageSlideshowHolder,.imageSlideshowHolder_left{
margin:5px;     /* "Air" */
float:right; /* Floating gallery at the right side of other web page content */
width:151px;    /* Image width */
height:100px;   /* Image height */

position:relative;
}
.leftFloatingDiv{
float:left;
width:160px;
height:110px;
}

.imageSlideshowHolder img{
position:absolute;
left:0px;
top:0px;
}
</style>
<SCRIPT type="text/javascript">
<![CDATA[
var slideshow2_noFading = false;
var slideshow2_timeBetweenSlides = 1000;
var slideshow2_fadingSpeed = 10;

var slideshow2_stats = new Array();

var slideshow2_slideIndex = new Array();
var slideshow2_slideIndexNext = new Array();
```

```
var slideshow2_imageDivs = new Array();
var slideshow2_currentOpacity = new Array();
var slideshow2_imagesInGallery = new Array();
var Opera = navigator.userAgent.indexOf('Opera')>=0?true:false;
function createParentDivs(imageIndex,divId)
{
if(imageIndex==slideshow2_imagesInGallery[divId]){
showGallery(divId);
}else{
var imgObj = document.getElementById(divId + '_' + imageIndex);
if(Opera)imgObj.style.position = 'static';
if(!slideshow2_imageDivs[divId])slideshow2_imageDivs[divId] = new Array();
slideshow2_imageDivs[divId][slideshow2_imageDivs[divId].length] =  imgObj;

imgObj.style.visibility = 'hidden';
imageIndex++;
createParentDivs(imageIndex,divId);
}
}

function showGallery(divId)
{
if(slideshow2_slideIndex[divId]==-1)slideshow2_slideIndex[divId]=0; else
slideshow2_slideIndex[divId]++;
if(slideshow2_slideIndex[divId]==slideshow2_imageDivs[divId].length)slideshow2_slideI
ndex[divId]=0;
slideshow2_slideIndexNext[divId] = slideshow2_slideIndex[divId]+1;
if(slideshow2_slideIndexNext[divId]==slideshow2_imageDivs[divId].length)slideshow2_
slideIndexNext[divId] = 0;


slideshow2_currentOpacity[divId]=100;


slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.visibility = 'visible';
if(Opera)slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.display =
'inline';
if(navigator.userAgent.indexOf('Opera')<0){
slideshow2_imageDivs[divId][slideshow2_slideIndexNext[divId]].style.visibility =
'visible';
}

if(document.all){
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.filter =
'alpha(opacity=100)';
slideshow2_imageDivs[divId][slideshow2_slideIndexNext[divId]].style.filter =
'alpha(opacity=1)';
}else{
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.opacity = 0.99;
slideshow2_imageDivs[divId][slideshow2_slideIndexNext[divId]].style.opacity = 0.01;
}
```

```
setTimeout('revealImage("' + divId + '")',slideshow2_timeBetweenSlides);
}

function revealImage(divId)
{

if(slideshow2_noFading){
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.visibility = 'hidden';
if(Opera)slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.display =
'none';
showGallery(divId);
return;
}
slideshow2_currentOpacity[divId]--;
if(document.all){
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.filter =
'alpha(opacity='+slideshow2_currentOpacity[divId]+')';
slideshow2_imageDivs[divId][slideshow2_slideIndexNext[divId]].style.filter =
'alpha(opacity='+(100-slideshow2_currentOpacity[divId])+')';
}else{
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.opacity =
Math.max(0.01,slideshow2_currentOpacity[divId]/100);
slideshow2_imageDivs[divId][slideshow2_slideIndexNext[divId]].style.opacity =
Math.min(0.99,(1 - (slideshow2_currentOpacity[divId]/100)));
}
if(slideshow2_currentOpacity[divId]>0){
setTimeout('revealImage("' + divId + '")',slideshow2_fadingSpeed);
}else{
slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.visibility = 'hidden';
if(Opera)slideshow2_imageDivs[divId][slideshow2_slideIndex[divId]].style.display =
'none';
showGallery(divId);
}
}

function initImageGallery(divId)
{
var slideshow2_galleryContainer = document.getElementById(divId);


slideshow2_slideIndex[divId] = -1;
slideshow2_slideIndexNext[divId] = false;

var galleryImgArray = slideshow2_galleryContainer.getElementsByTagName('IMG');
for(var no=0;no<galleryImgArray.length;no++){
galleryImgArray[no].id = divId + '_' + no;
}

slideshow2_imagesInGallery[divId] = galleryImgArray.length;
```

```
createParentDivs(0,divId);

}
]]>
</SCRIPT>
</head>
<body>

<div id="mainContainer">

<div class="contentContainer">
<div class="imageSlideshowHolder" id="imageSlideshowHolder">
<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image1']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image2']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image3']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image4']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image5']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image6']/dynamic-content"/>
```

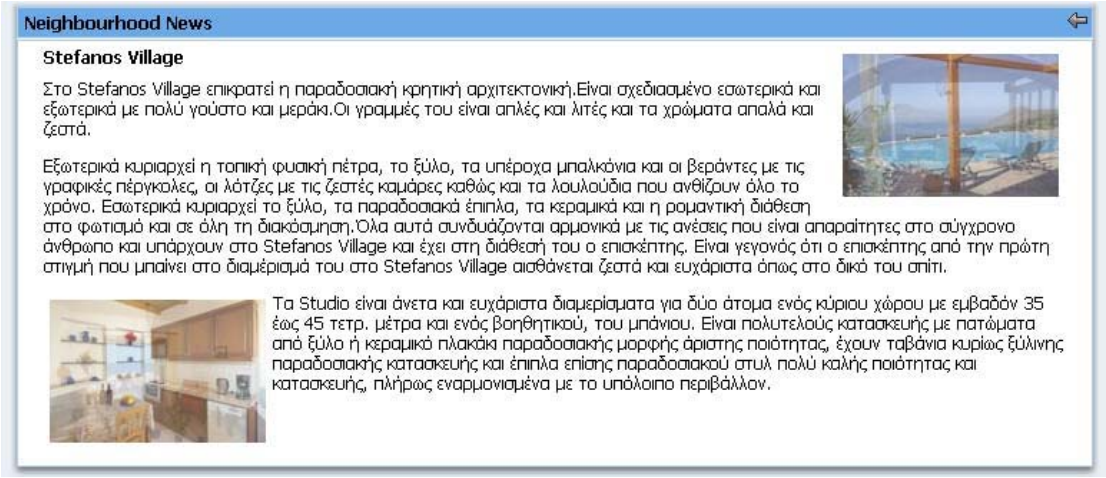
```
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image7']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image8']/dynamic-content"/>
</xsl:attribute>
</img>
</div>

<h1><xsl:if test="root/dynamic-element[@name='title']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='title']/dynamic-content"/>
</xsl:if></h1>

<p><xsl:if test="root/dynamic-element[@name='body1']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='body1']/dynamic-content"/>
</xsl:if></p>

<p><xsl:if test="root/dynamic-element[@name='body2']/dynamic-content != ''">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='body2']/dynamic-content"/>
</xsl:if></p>

<div class="leftFloatingDiv">
<div class="imageSlideshowHolder" id="imageSlideshowHolder2">
<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image8']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image7']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
```

```
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image6']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image5']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image4']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image3']/dynamic-content"/>
</xsl:attribute>
</img>

<img>
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image2']/dynamic-content"/>
</xsl:attribute>
</img>

<img >
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image1']/dynamic-content"/>
</xsl:attribute>
</img>
</div>
</div>

<p><xsl:if test="root/dynamic-element[@name='body3']/dynamic-content != ""><
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='body3']/dynamic-content"/>
</xsl:if></p>
</div>
</div>

<SCRIPT type="text/javascript">
initImageGallery('imageSlideshowHolder');
```

initImageGallery('imageSlideshowHolder2');
</SCRIPT>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

4.5.3 Εφαρμογή στο portal του eKoNES

Η αξιοποίηση του συγκεκριμλενου template για τις ανάγκες του έργου eKoNES παρουσιάζεται παρακάτω.

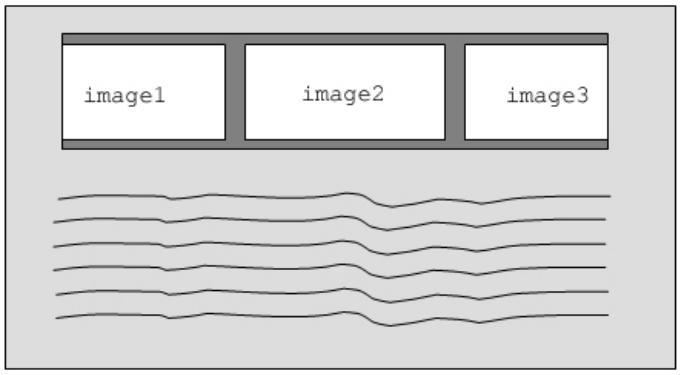


Εικόνα 19: Body-and-2-fade-slideshows template στο eKoNES

4.6 Motion-images template

4.6.1 Όψη

Το template αυτό χρησιμοποιείται για να παρουσιάσει ένα άρθρο το οποίο περιέχει ένα κυρίως κείμενο και εικόνες. Για την παρουσίαση των εικόνων χρησιμοποιείται ένα slideshow με αυτόματη εναλλαγή εικόνων.



Εικόνα 20: Όψη του προτύπου Motion-images

4.6.2 Κώδικας

Structure

```
<root>
  <dynamic-element name='image1' type='image'></dynamic-element>
  <dynamic-element name='image2' type='image'></dynamic-element>
  <dynamic-element name='image3' type='image'></dynamic-element>
  <dynamic-element name='image4' type='image'></dynamic-element>
  <dynamic-element name='image5' type='image'></dynamic-element>
  <dynamic-element name='body' type='text_area'></dynamic-element>
</root>
```

Template

```
<?xml version="1.0"?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html" omit-xml-declaration="yes"/>
<xsl:template match="/">
<center>
<script language="javascript">

var sliderwidth="400px"

var sliderheight="150px"

var slidespeed=1

slidebgcolor="black"

var leftrightslide=new Array()
var finalslide=''
leftrightslide[0]='<img width="105">
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image1']/dynamic-content"/>
</xsl:attribute>
</img> ';
leftrightslide[1]='<img width="105">
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image2']/dynamic-content"/>
</xsl:attribute>
</img>';
leftrightslide[2]='<img width="105">
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image3']/dynamic-content"/>
```

```
</xsl:attribute>
</img>';
leftrightslide[3]='<img width="105">
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image4']/dynamic-content"/>
</xsl:attribute>
</img>';
leftrightslide[4]='<img width="105">
<xsl:attribute name="src">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-
element[@name='image5']/dynamic-content"/>
</xsl:attribute>
</img>';

var imagegap=" "

var slideshowgap=5

var copyspeed=slidespeed
leftrightslide='<nobr>'+leftrightslide.join(imagegap)+'</nobr>'
var iedom=document.all||document.getElementById
if (iedom)
document.write('<span id="temp" style="visibility:hidden;position:absolute;top:-
100px;left:-9000px">'+leftrightslide+'</span>')
var actualwidth=''
var cross_slide, ns_slide
<![CDATA[
function fillup(){
if (iedom){
cross_slide=document.getElementById? document.getElementById("test2") :
document.all.test2
cross_slide2=document.getElementById? document.getElementById("test3") :
document.all.test3
cross_slide.innerHTML=cross_slide2.innerHTML=leftrightslide
actualwidth=document.all? cross_slide.offsetWidth :
document.getElementById("temp").offsetWidth
cross_slide2.style.left=actualwidth+slideshowgap+"px"
}
else if (document.layers){
ns_slide=document.ns_slidemenu.document.ns_slidemenu2
ns_slide2=document.ns_slidemenu.document.ns_slidemenu3
ns_slide.document.write(leftrightslide)
ns_slide.document.close()
actualwidth=ns_slide.document.width
ns_slide2.left=actualwidth+slideshowgap
ns_slide2.document.write(leftrightslide)
ns_slide2.document.close()
}
```

```
lefttime=setInterval("slideleft()",30)
}
window.onload=fillup

function slideleft(){
if (iedom){
if (parseInt(cross_slide.style.left)>(actualwidth*(-1)+8))
cross_slide.style.left=parseInt(cross_slide.style.left)-copyspeed+"px"
else
cross_slide.style.left=parseInt(cross_slide2.style.left)+actualwidth+slideshowgap+"px"

if (parseInt(cross_slide2.style.left)>(actualwidth*(-1)+8))
cross_slide2.style.left=parseInt(cross_slide2.style.left)-copyspeed+"px"
else
cross_slide2.style.left=parseInt(cross_slide.style.left)+actualwidth+slideshowgap+"px"

}
else if (document.layers){
if (ns_slide.left>(actualwidth*(-1)+8))
ns_slide.left-=copyspeed
else
ns_slide.left=ns_slide2.left+actualwidth+slideshowgap

if (ns_slide2.left>(actualwidth*(-1)+8))
ns_slide2.left-=copyspeed
else
ns_slide2.left=ns_slide.left+actualwidth+slideshowgap
}
}


if (iedom||document.layers){
with (document){
document.write('<table border="0" cellspacing="0" cellpadding="0"><td>')
if (iedom){
write('<div
style="position:relative;width:'+sliderwidth+';height:'+sliderheight+';overflow:hidden"
>')
write('<div
style="position:absolute;width:'+sliderwidth+';height:'+sliderheight+';background-
color:'+slidebgcolor+'" onMouseover="copyspeed=0"
onMouseout="copyspeed=slidespeed">')
write('<div id="test2" style="position:absolute;left:0px;top:0px"></div>')
write('<div id="test3" style="position:absolute;left:-1000px;top:0px"></div>')
write('</div></div>')
}
else if (document.layers){
write('<ilayer width='+sliderwidth+' height='+sliderheight+' name="ns_slidemenu"
bgColor='+slidebgcolor+'>')
write('<layer name="ns_slidemenu2" left=0 top=0 onMouseover="copyspeed=0"
onMouseout="copyspeed=slidespeed"></layer>')
```

```
write('<layer name="ns_slidemenu3" left=0 top=0 onMouseover="copyspeed=0" onMouseout="copyspeed=slidespeed"></layer>')
write('</ilayer>')
}
document.write('</td></table>')
}
}

]]>
</script>
</center>
<br />
<xsl:if test="root/dynamic-element[@name='body']/dynamic-content != "">
<xsl:value-of disable-output-escaping="yes" select="root/dynamic-element[@name='body']/dynamic-content"/>
</xsl:if>
</xsl:template>
</xsl:stylesheet>
```

4.6.3 Εφαρμογή στο portal του eKoNES

Η αξιοποίηση του συγκεκριμλενου template για τις ανάγκες του έργου eKoNES παρουσιάζεται παρακάτω.

Εικόνα 21: Motion-images template στο eKoNES

5 Σενάρια χρήσης του Τομέα Τηλεπικοινωνιών & Πολυμέσων (ΤΤΠ)

5.1 Εγγραφή χρήστη

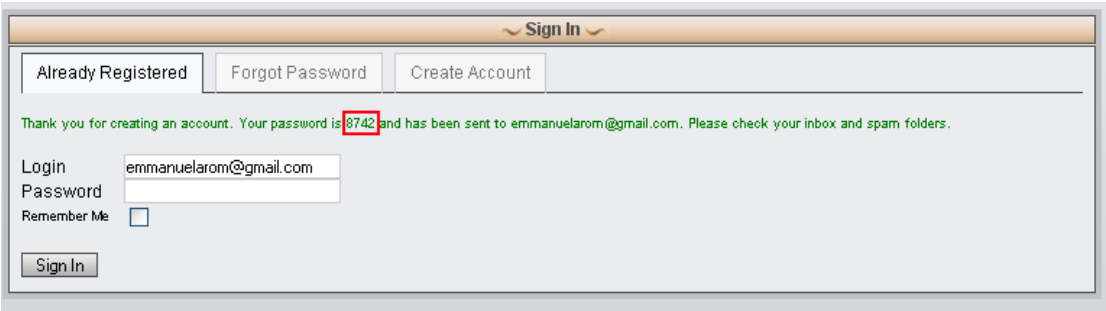
Ο χρήστης του διαδικτυακού κόμβου του Τομέα Τηλεπικοινωνιών & Πολυμέσων (ΤΤΠ) εγγράφεται στο portal μέσω μιας τυποποιημένης διαδικασίας που περιγράφεται ακολούθως.

Η πρώτη εγγραφή απαιτεί την εισαγωγή των υποχρεωτικών προσωπικών στοιχείων του χρήστη. Τα στοιχεία αυτα ειναι το όνομα, το επίθετο, το email και ειναι υποχρεωτικά. Σε περίπτωση που ο χρήστης δεν συμπληρώσει κάποιο υποχρεωτικό πεδίο, όπως το όνομα, το σύστημα τον ενημερώνει με μήνυμα λάθους και τον προτρέπει να συμπληρώσει σωστά το πεδίο για να προχωρήσει στην αποθήκευση των στοιχείων.

Στη συνέχεια εισάγωνται τα προαιρετικά προσωπικά στοιχεία όπως middle name, nickname, birthdate, κ.λ.π. Έκτος απο τα υποχρεωτικά στοιχεία και τα προαιρετικά, ο χρήστης πρέπει να πληκτρολογήσει έναν τετραψήφιο αριθμό που παρέχεται δυναμικά απο το σύστημα και χρησιμοποιείται για την επιβεβαίωση της εγγραφής. Αφου συμπληρώσει υα στοιχεία αυτα, ο χρήστης υποβάλει την φόρμα εγγραφής.

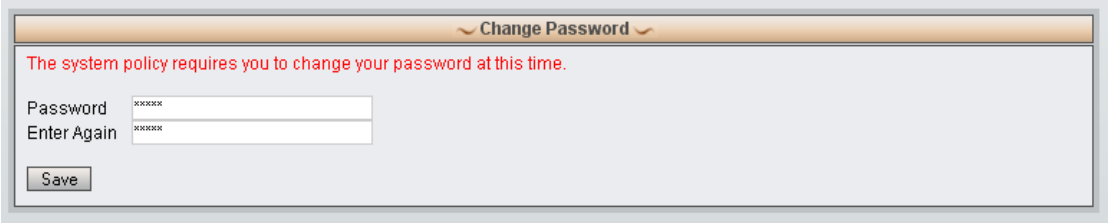
Εικόνα 22:Εγγραφή χρήστη

Στη συνέχεια, το σύστημα παρέχει στο χρήστη ένα δυναμικά παραγόμενο κωδικό τον οποίο εισάγει ο χρήστης στο αντίστοιχο πεδίο όπως φαίνεται στην εικόνα.



Εικόνα 23: Η διαδικασία εγγραφής

Αφού εισάγει ο χρήστης τον κωδικό αυτο, του ζητείται να εισάγει ενα δικό του κωδικό για την ολοκλήρωση της εγγραφης. Το σύστημα ζητά τον κωδικό αυτο δυο φορές για επιβεβαίωση.

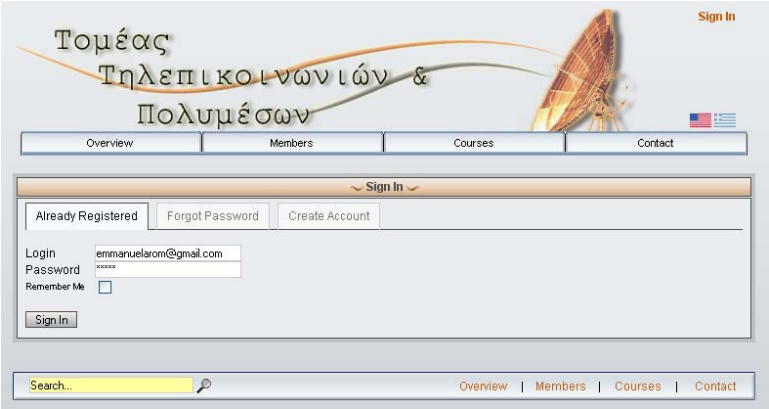


Εικόνα 24: Αλλαγή κωδικού χρήστη

5.2 Login χρήστη

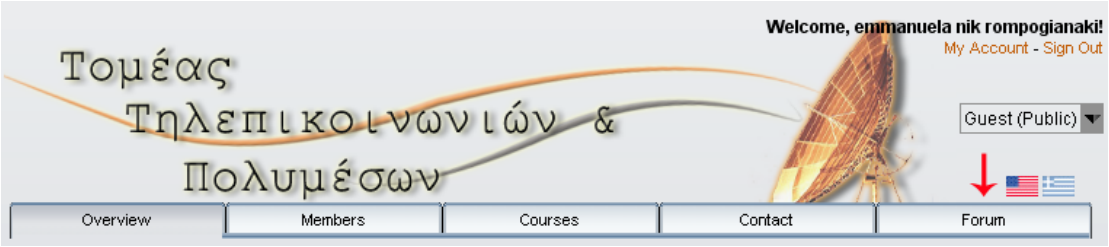
Η διαδικασία Login αφορά τους ήδη εγγεγραμμένους χρήστες του συστήματος, δηλαδή προϋποθέτει την επιτυχή εγγραφή ενός επισκέπτη με το σενάριο που περιγράφθηκε νωρίτερα. Για να κάνει ο χρήστης Login στο Portal του ΤΤΠ θα πρέπει να εισάγει τα στοιχεία που όρισε κατα τη διαδικασία της εγγραφής που περιγράψαμε νωρίτερα.

Τα στοιχεία αυτα ειναι το username και το password που αφορούν το email και τον κωδικό αντίστοιχα. Αφού εισάγει αυτά τα στοιχεία ο χρήστης υποβάλει το αίτημα εισαγωγής στο portal του ΤΤΠ.



Εικόνα 25: Είσοδος χρήστη

Η επιτυχής εγγραφή ενός χρήστη του επιτρέπει πρόσβαση σε προστατευόμενο περιεχόμενο και υπηρεσίες όπως ειναι ο πίνακας μηνυμάτων (forum) του τομέα που κατασκευάστηκε για να υποστηρίξει ασύγχρονη επικοινωνία μεταξύ των εγγεγραμμένων μελών.



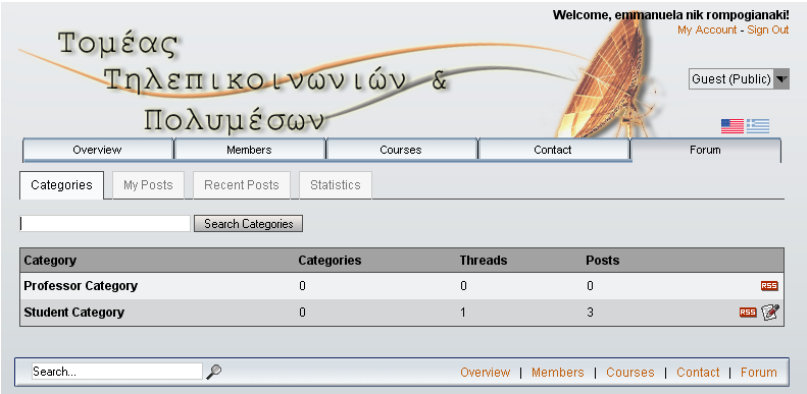
Εικόνα 26: Όψη εγγεγραμένου χρήστη

5.3 Διαχείριση Ανακοινώσεων

Η υπηρεσία διαχείρισης ασύγχρονων μηνυμάτων εμφανίζεται στον εξουσιοδοτημένο χρήστη στην κορυφή της μπάρας επιλογής (Εικόνα 27) .Ενεργοποιώντας (δηλ. επιλέγοντας) την υπηρεσία αυτή οι χρήστες μπορούν να εντοπίζουν, να διαβάζουν και να σχολιάζουν ανακοινώσεις που τους ενδιαφέρουν. Αυτό καθίσταται δυνατό με τη διαχείριση των ανακοινώσεων.

Τα συγκεκριμένα βήματα που πρέπει να εκτελέσει ένας χρήστης για να εισάγει μια νέα ανακοίνωσης είναι τα ακόλουθα:

- Επιλογή κατηγορίας ανακοίνωσης
- Εισαγωγή στοιχείων ανακοίνωσης
- Αποθήκευση ανακοίνωσης
- Εύρεση ανακοίνωσης
- Σχολιασμός ανακοίνωσης

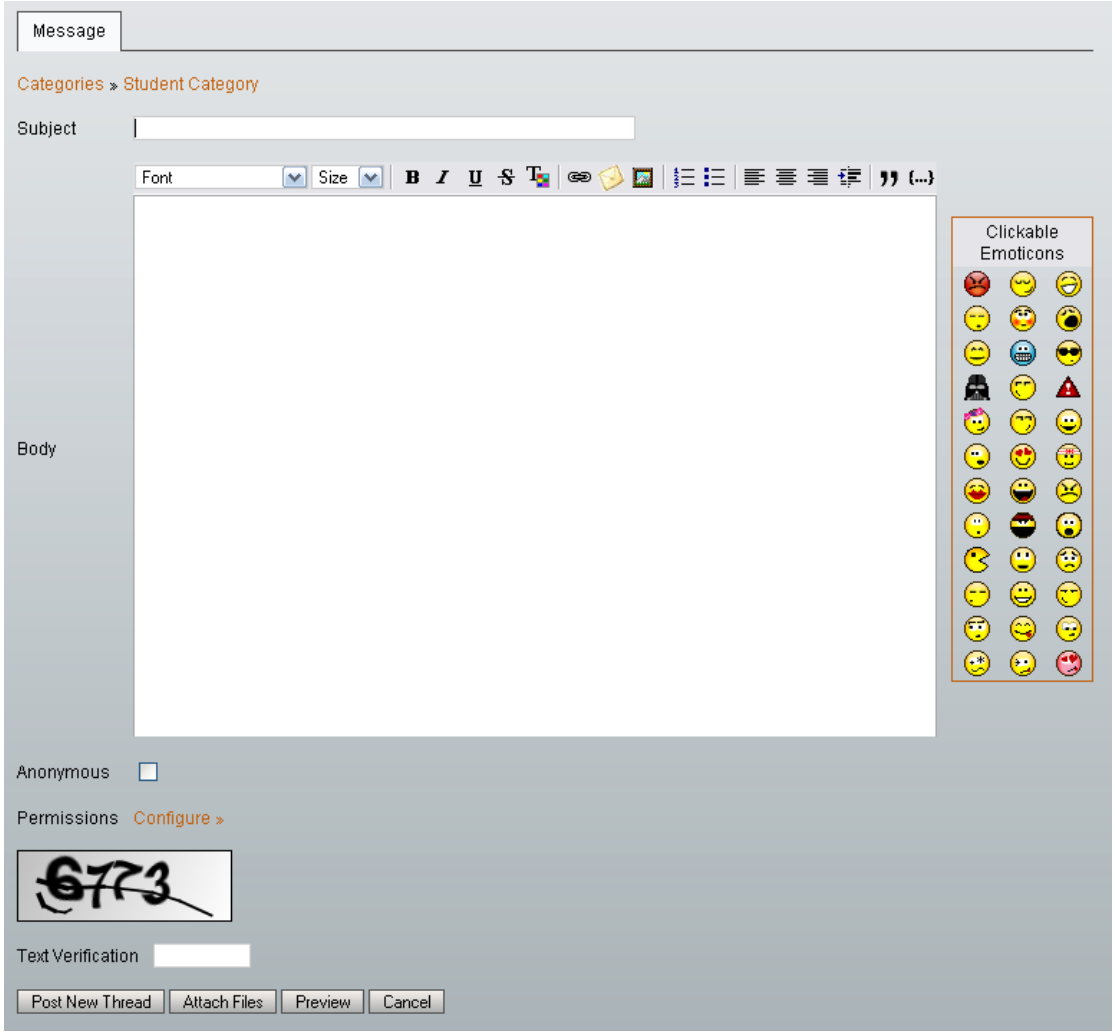


Εικόνα 27: Το Forum του Τομέα

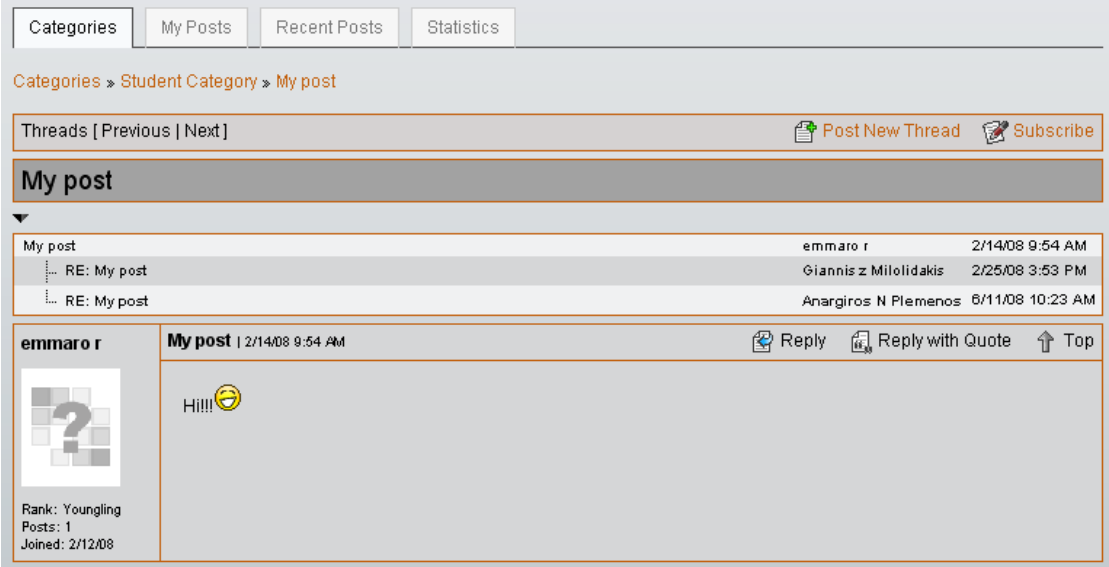
Ειδικότερα, ο χρήστης επιλέγει την κατηγορία που θέλει να τοποθετηθεί η ανακοίνωση και στη συνέχεια εισάγει τα στοιχεία της ανακοίνωσης, δηλαδή το όνομα, το περιεχόμενο και ίσως κάποιο συνημμένο έγγραφο. Η υποβολή της φόρμας ανακοίνωσης γίνεται πατώντας το αντίστοιχο κουμπί εντολής. Αν ο χρήστης επιθυμεί να σχολιάσει μια υπάρχουσα ανακοίνωση τότε πρέπει πρώτα να εντοπίσει τη συγκεκριμένη ανακοίνωση και στη συνέχεια να εισάγει το σχολιασμό. Η εύρεση μιας ανακοίνωσης μπορεί να γίνει μέσω πλοήγησης είτε με βάσει κάποιας λέξης - κλειδιού. Η επιλογή "Reply" επιτρέπει το σχολιασμό της ανακοίνωσης.

Ενδεικτικά, ας υποθέσουμε ένα νέο χρήστη ο οποίος αποφασίζει να δημιουργήσει ένα νέο νήμα συζήτησης. Αυτό επιτυγχάνεται μέσω της επιλογής «Post New Thread». Για την εισαγωγή δικής του ανακοίνωσης, εμφανίζεται μια φόρμα συμπλήρωσης στοιχείων της ανακοίνωσης(όπως φαίνεται στην Εικόνα 28). Αρχικά, βλέπουμε το tab «Messages» και έτσι ξέρουμε ότι πρόκειται να εισάγουμε ανακοίνωση, ενώ με το breadcrumb σε ποια κατηγορία θα ανήκει η ανακοίνωση. Έπειτα, ακολουθεί ένα textfield για τον τίτλο του thread, ένας WYSIWYG editor για το κυρίως μέρος της ανακοίνωσης (με textarea για την υποδοχή του περιεχομένου), ένα checkbox για εμφάνιση / απόκρυψη του ονόματος του poster, ένα link για τον καθορισμό των permissions της ανακοίνωσης, ένα textfield για την εισαγωγή του verification text, και τέλος κουμπιά ελέγχου για αποθήκευση της ανακοίνωσης, για προεσκόπηση της ανακοίνωσης και για ακύρωση της διαδικασίας.

Στη συνέχεια βλέπουμε τη διεπαφή (Εικόνα 29) η οποία παρουσιάζει πληροφορία για το post και για τον poster και παρέχει επίσης κουμπιά για διαδικασίες reply και post νέας ανακοίνωσης. Συγκεκριμένα, στο πάνω μέρος βλέπουμε το tab των κατηγοριών και ακριβώς από κάτω το breadcrumb με το path της ανακοίνωσης. Έπειτα δίνεται η δυνατότητα πλοήγησης ανάμεσα στα post με συνδέσμους «Previous» και «Next», και η δυνατότητα για «Post New Thread» και «Subscribe» με τους αντίστοιχους συνδέσμους. Ακολουθεί ένα text με το θέμα του post, ένα textarea με πληροφορίες για τον poster (ονοματεπώνυμο, φωτογραφία, κατάταξη, αριθμό post που έχει γράψει και ημερομηνία εγγραφής στο portal), ένα text με πληροφορία του post (τίτλος και ημερομηνία που έγινε posted), δύο link για «Reply» και «Reply with quote», ένα link για μετάβαση στην κορυφή της σελίδας και τέλος ένα textarea που παρουσιάζει το περιεχόμενο της ανακοίνωσης.



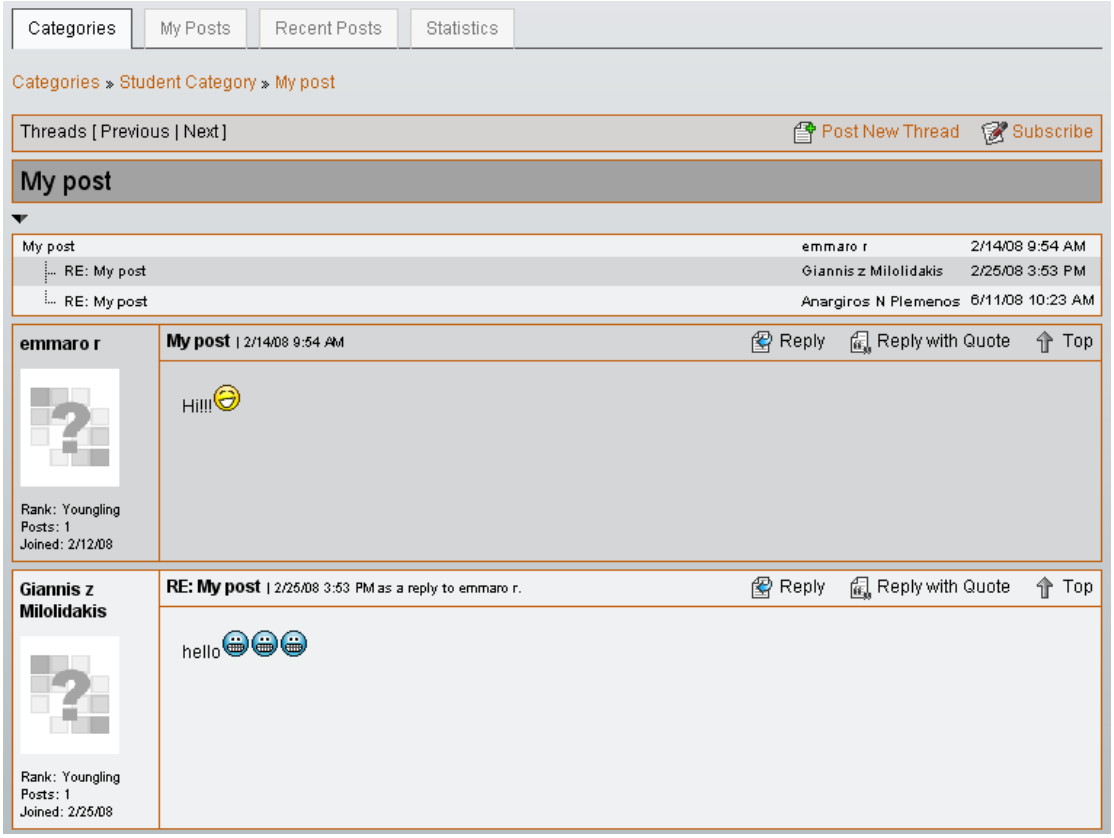
Εικόνα 28: Διάλογος αποστολής ανακοίνωσης



Εικόνα 29: Η διαδικασία reply

Στη συνέχεια του σεναρίου δίνεται η δυνατότητα κάποιος να σχολιάσει την ανακοίνωση που εισήχθη νωρίτερα. Η διαδικασία επιτυγχάνεται ακολουθώντας το σύνδεσμο “Reply” που φαίνεται στο παραπάνω σχήμα (Εικόνα 29). Η διαδικασία εισαγωγής ανακοίνωσης είναι ίδια με αυτή που περιγράφηκε νωρίτερα. Έτσι, όταν έχει αποθηκευτεί η απαντητική ανακοίνωση βλέπουμε μια διεπαφή όπως η παρακάτω (Εικόνα 30). Μερικά στοιχεία είναι ίδια με την προηγούμενη διεπαφή και έτσι δε θα τα σχολιάσουμε. Τα νέα στοιχεία αφορούν ένα tree με το post και τα αντίστοιχα reply (παρουσιάζεται επίσης το όνομα του poster και η ημερομηνία που έγινε το post) και τρία κουμπιά ελέγχου του post (τα οποία μόνο ο poster του συγκεκριμένου post έχει δικαίωμα να βλέπει):

- για edit ( )
- για καθορισμό των δικαιωμάτων ( ) και
- για διαγραφή ( )

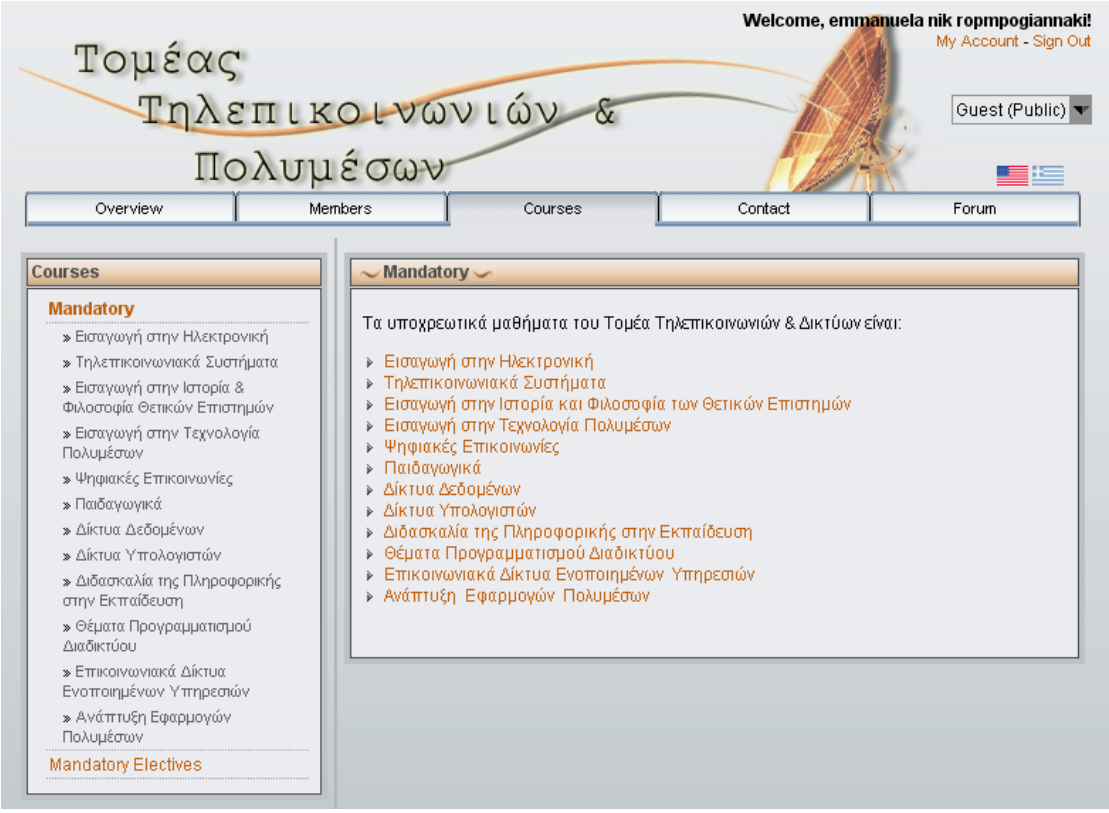


Εικόνα 30: Σχολιασμός ανακοίνωσης

5.4 Επισκόπηση Δημόσιου Περιεχομένου

Το δημόσιο περιεχόμενο του ΤΠΠ περιλαμβάνει γενικές πληροφορίες για τον Τομέα όπως ανακοινώση, σύνοψη δραστηριοτήτων και δράσεων καθώς και αναλυτική παρουσίαση των μαθημάτων του Τομέα. Ενδεικτικά η Εικόνα 31 παρουσιάζει τα μαθήματα του συγκεκριμένου Τομέα. Τα μαθήματα αυτά μπορούν επίσης να παρουσιαστούν ομαδοποιημένα (ως κύκλοι μαθημάτων) χρησιμοποιώντας ένα από τα templates που αναπτύξαμε σε προηγούμενο κεφάλαιο.

Για να κάνει ο χρήστης επισκόπηση των μαθημάτων πρεπει να επιλέξει το μενόυ «Courses». Στην συνέχεια εμφανίζονται τα μαθήματα ανα κατηγορίες (υποχρεωτικα, υποχρεωτικά επιλογής).

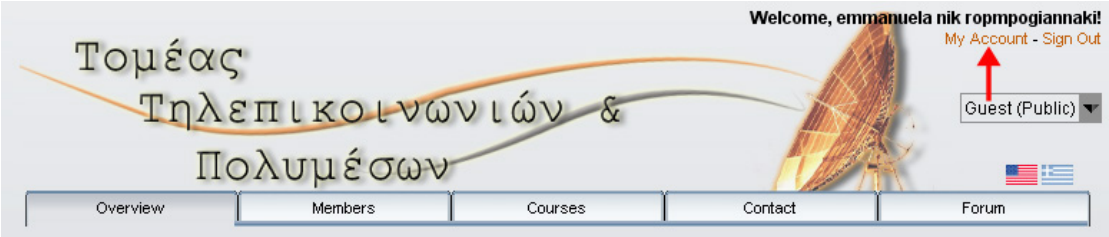


Εικόνα 31: Επισκόπηση μαθημάτων

5.5 Διαχείριση προφίλ χρήστη

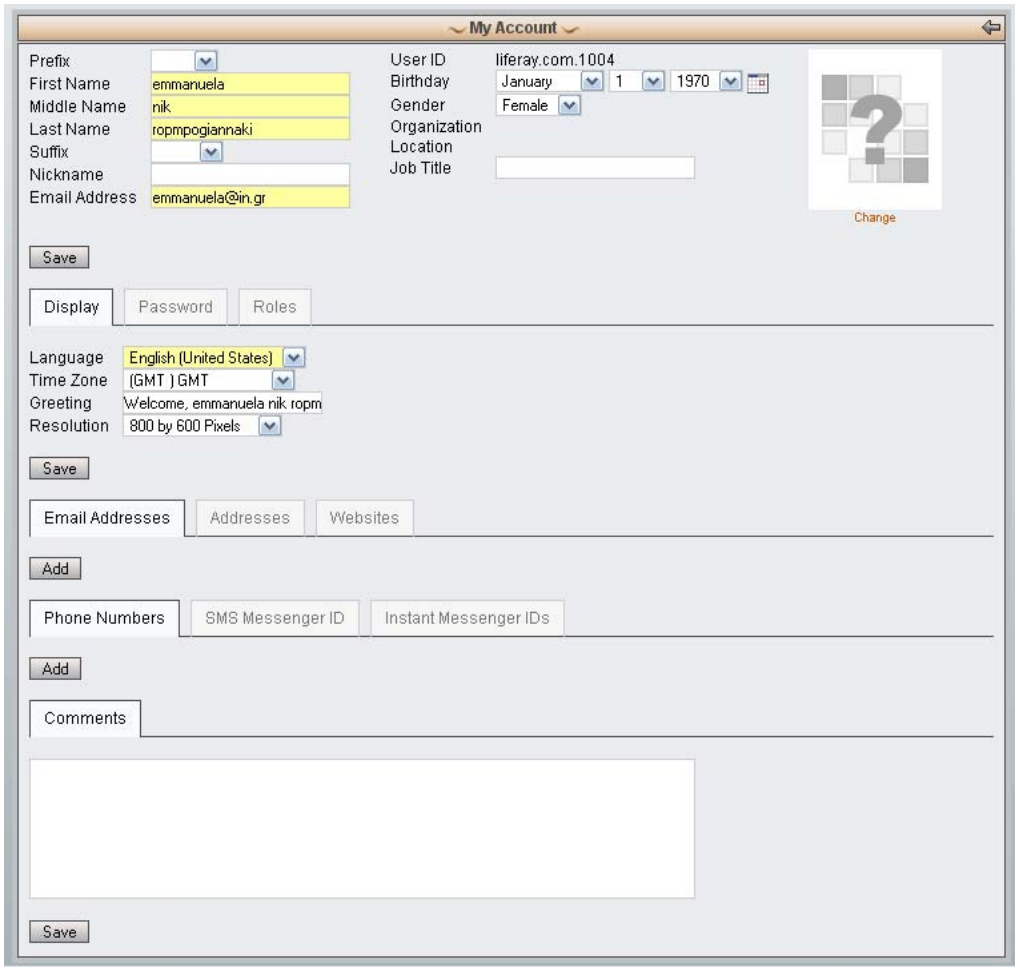
Στη διαχείριση του προφίλ ένας χρήστης μπορεί να αλλάξει τα υποχρεωτικά στοιχεία του όπως e-mail και κωδικό, αλλά μπορεί να διορθώσει ή να συμπληρώσει νέα στοιχεία, όπως Job Title, Phone Numbers, κ.λ.π.

Για να αλλάξει ενας χρήστης το προφίλ του αρχικά επιλέγει από το μενού χρήστη την επιλογή "My account"



Εικόνα 32: Επιλογή μενου "My account" για αλλαγή προφίλ

Στη συνέχεια εμφανίζεται η παρακάτω σελίδα στην οποία μπορεί ο χρήστης να κάνει αλλαγή των υποχρεωτικών στοιχείων (όπως το όνομα και επώνυμό του, το e-mail, τον κωδικό, κ.λ.π.), αλλαγή των προαιρετικών στοιχείων (όπως το Job Title, e-mail Addresses, κ.λ.π. καθώς να κάνει upload μια εικόνα η οποία τον χαρακτηρίζει). Αφού έχει αλλάξει τα υποχρεωτικά και ενδεχομένως και τα προαιρετικά του στοιχεία, ο χρήστης υποβάλει τις αλλαγές.



Εικόνα 33:Αλλαγή προφίλ

6 Συμπεράσματα

Σκοπός αυτή της πτυχιακής εργασίας ηταν η ανάπτυξη ενός διαδικτυακού κόμβου λειτουργίας τομέων εκπαιδευτικών μονάδων. Για το σκοπό αυτό χρησιμοποιήθηκε το ανοικτού κώδικα Liferay CMS το οποίο βασίζεται στη γλώσσα προγραμματισμού java και εκτελείται με οποιονδήποτε application server και με οποιαδήποτε Βάση Δεδομένων. Ετσι αναπτύχθηκε το Portal του Τομέα Τηλεπικοινωνιών και Πολυμέσων το οποίο δίνει τη δυνατότητα στο χρήστη να εγγραφεί και να έχει πρόσβαση σε θέματα που αφορούν τον Τομέα, όπως είναι τα μαθήματα του τομέα (υποχρεωτικά, υποχρεωτικά επιλογής) αλλα και σε ανακοινώσεις που δημοσιεύονται από το διαχειριστή. Επίσης στα πλαίσια αυτης της εργασίας αναπτύχθηκαν ειδικά πρότυπα (templates) ώστε να μπορούν να χρησιμοποιηθούν από τους χρήστες ανάλογα με τις ανάγκες των άρθρων που δημοσιεύουν .

7 Βιβλιογραφία

Πτυχιακή Εργασία «Εικονικές κοινότητες πρακτικής: Παρουσίαση και διάχυση πληροφοριών στο διαδίκτυο με την χρήση Portals.»
Μαρκοπούλου Μαρία

Πηγές από το διαδίκτυο:
http://www.sapdesignguild.org/
http://en.wikipedia.org
http://www.liferay.com
http://www.w3schools.com/
http://wiki.liferay.com