

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ
ΜΑΚΕΔΟΝΙΑΣ**

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη και υλοποίηση ενός συστήματος διαχείρισης δημόσιων
οθονών για το Πανεπιστήμιο Δυτικής Μακεδονίας

Design and Implementation of a Public Display Management
System for the University of Western Macedonia

Τσολάκης Πέτρος

Επιβλέπων Καθηγητής: Δασυγένης Μηνάς

Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής Υπολογιστών

Κοζάνη, Φεβρουάριος 2016

Περίληψη

Στην παρούσα διπλωματική εργασία, αναπτύχθηκε ένα σύστημα διαχείρισης δημόσιων οθονών.

Το σύστημα αυτό, αποτελείται από έναν ιστότοπο και πολλαπλούς σταθμούς βάσεις, που έχει συνδεθεί στο καθένα μια οθόνη και έχει πρόσβαση στο διαδίκτυο. Ο ιστότοπος περιλαμβάνει ένα σχεδιαστικό εργαλείο, με το οποίο μπορούμε να δημιουργήσουμε, χωρίς ιδιαίτερες γνώσεις, ένα περιβάλλον που αποτελείται διαφορετικά στοιχεία. Το περιβάλλον αυτό, είναι δομημένο ως μια σελίδα της HTML. Τα στοιχεία του περιβάλλοντος, είναι διάφοροι τύποι δεδομένων όπως κείμενο, εικόνα και βίντεο.

Έπειτα, μπορούμε να το προωθήσουμε το περιβάλλον αυτό σε ένα σταθμό βάσης. Ο ρόλος του σταθμού βάσης είναι να το προβάλλει σε έναν φυλλομετρητή σελίδων, διαμορφώνοντας το, ανάλογα με τα στοιχεία που συμπεριλήφθηκαν. Οι σταθμοί βάσεις αποτελούνται από έναν υπολογιστή, μικρών απαιτήσεων σε υλικό.

Οι χρήστες και οι σταθμοί βάσεις ανήκουν σε ομάδες διαχείρισης. Κάθε χρήστης μπορεί να αλληλεπιδράσει, μόνο με τους σταθμούς βάσεις της ομάδας του. Με αυτόν τον τρόπο, ελέγχονται οι σταθμοί βάσεις και ρυθμίζονται ποια περιβάλλοντα θα προβάλλονται σε αυτούς.

Λέξεις κλειδιά: Δημόσιες οθόνες, Προγραμματισμός διαδικτύου, Σύστημα διαχείρισης, Βάση δεδομένων, Τεχνολογίες προγραμματισμού πελάτη, Τεχνολογίες προγραμματισμού διακομιστή.

Abstract

In this thesis, a management system of public displays was designed.

This system consists of a website and multiple base stations which has connected a screen and has access to the internet. The website includes a design tool, with which we can create, without special knowledge, a enviroment which consists of various elements. This enviroment, is structured like a HTML webpage. The elements of the enviroment, are various data types like text, image and video.

Then we can push this enviroment to a base station. The role of the base station is to display it to a web browser, formatting it, according to the elements that have included. The base stations consists of a computer, with minimum requirements in hardware.

Users and base stations are divided into management groups. Each user can only interact, with the base stations of his group. In that way, the base stations are manipulated and are set which enviroments are going to be displayed.

Keywords: Public displays, Web development, Management System, Database, Client-side scripting, Server-side scripting.

Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο "Ανάπτυξη και υλοποίηση ενός συστήματος διαχείρισης δημόσιων οθονών για το Πανεπιστήμιο Δυτικής Μακεδονίας" καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Δασυγένη Μηνά, αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Copyright (C) Τσολάκης Πέτρος, Δασυγένης Μηνάς, 2016, Κοζάνη

Περιεχόμενα

1	Εισαγωγή	13
	Σύνοψη κεφαλαίου	16
2	Θεωρητικό υπόβαθρο	19
	2.1 Προγραμματισμός Διαδικτύου	19
	2.1.1 HTML	20
	2.1.2 CSS	22
	2.1.3 JavaScript	23
	2.1.4 PHP	24
	2.1.5 AJAX	25
	2.1.6 MySQL	25
	2.2 Βοηθητικά εργαλεία λογισμικού	26
	2.2.1 LAMP	26
	2.2.2 Geany	26
	2.2.3 phpMyAdmin	26
	2.2.4 GIMP	27
	Σύνοψη κεφαλαίου	27
3	Υλικό μέρος συστήματος	29
	3.1 BeagleBoard-xM	29
	3.2 Debian	31
	3.3 Samsung SyncMaster T200	32
	Σύνοψη κεφαλαίου	33
4	Λογισμικό μέρος συστήματος	35
	4.1 Περιγραφή συστήματος	36
	4.2 Απαιτήσεις συστήματος	37
	4.3 Σχεδιασμός βάσης δεδομένων	39

4.3.1	Πίνακας <i>users_information</i>	39
4.3.2	Πίνακας <i>layouts</i>	40
4.3.3	Πίνακας <i>contents</i>	41
4.3.4	Πίνακας <i>screens</i>	42
4.3.5	Πίνακας <i>content_scheduler</i>	43
4.3.6	Πίνακας <i>announcements</i>	44
4.3.7	Πίνακας <i>groups</i>	45
4.3.8	Πίνακας <i>users_privileges</i>	45
4.3.9	Πίνακας <i>screens_groups</i>	46
4.3.10	Διάγραμμα οντοτήτων-συσχετίσεων	46
4.4	Λειτουργίες ιστοτόπου	48
4.4.1	Η αρχική σελίδα	48
4.4.2	Η κεντρική σελίδα ελέγχου	49
4.4.3	Λειτουργίες υπερχρήστη	51
4.4.4	Λειτουργίες χρηστών	58
4.5	Ασφάλεια συστήματος	69
4.6	Ανάλυση κώδικα	70
4.6.1	Σύνδεση χρήστη	70
4.6.2	Φιλτράρισμα δεδομένων	71
4.6.3	Περιβάλλον σχεδίασης	71
4.6.4	Λειτουργία οθόνης	72
4.6.5	Ενημέρωση οθόνης	73
4.7	Μετρικά στοιχεία κώδικα	74
4.8	Δοκιμές	75
	Σύνοψη κεφαλαίου	77
5	Συμπεράσματα και μελλοντικές επεκτάσεις	79
5.1	Σύνοψη και συμπεράσματα	79
5.2	Προβλήματα κατά την ανάπτυξη	80
5.3	Μελλοντικές επεκτάσεις	81
	Σύνοψη κεφαλαίου	82
	Παράρτημα	83
	Εγκατάσταση αρχείων στο διακομιστή	83
	Εγκατάσταση λειτουργικού στο σταθμό βάσης	84
	Ρυθμίσεις λειτουργικού στο σταθμό βάσης	85

Κατάλογος σχημάτων

2.1	Κανόνας της CSS	22
2.2	Συνάρτηση γραμμένη σε JavaScript	24
2.3	Συνάρτηση γραμμένη σε jQuery	24
3.1	Το BeagleBoard-xM	31
3.2	Η Samsung SyncMaster T200	32
4.1	Αρχιτεκτονική συστήματος	37
4.2	Το διάγραμμα οντοτήτων-συσχετίσεων της βάσης δεδομένων	47
4.3	Η σελίδα σύνδεσης	49
4.4	Η κεντρική σελίδα ελέγχου του υπερχρήστη	50
4.5	Οι επιπλέον πληροφορίες του μενού User	50
4.6	Η φόρμα εγγραφής χρήστη	51
4.7	Λανθασμένη εισαγωγή τιμής	52
4.8	Έγκυρες τιμές εισαγωγής	53
4.9	Επιτυχής εγγραφή χρήστη	54
4.10	Μήνυμα στο νέο χρήστη	54
4.11	Αναζήτηση χρήστη	55
4.12	Δημιουργία οθόνης	56
4.13	Αναζήτηση οθόνης	56
4.14	Διαχείριση ομάδας	57
4.15	Διάγραμμα use cases του υπερχρήστη	58
4.16	Ενεργές οθόνες	59
4.17	Έκτακτα περιεχόμενα	59
4.18	Η περιοχή σχεδίασης διατάξεων	60
4.19	Οι αποθηκευμένες διατάξεις	61
4.20	Επιλογές δεδομένων περιεχομένου	63
4.21	Ρυθμίσεις στοιχείων <div>	64

4.22	Σχεδίαση περιεχομένου	65
4.23	Προβαλλόμενο περιεχόμενο σε οθόνη	66
4.24	Μενού εναλλαγής περιεχομένων	67
4.25	Επιλογή οθόνης	68
4.26	Διάγραμμα use cases των χρηστών	69
4.27	Σύνδεση του χρήστη στο σύστημα - Κώδικας	71
4.28	Φιλτράρισμα δεδομένων - Κώδικας	71
4.29	Σχεδίαση ενός στοιχείου <div> - Κώδικας	72
4.30	Η λειτουργία της οθόνης - Κώδικας	73
4.31	Ενημέρωση οθόνης - Κώδικας	74
4.32	BeagleBoard-xM και οθόνη	76

Κατάλογος πινάκων

2.1	Οι ετικέτες της HTML	21
2.2	Οι ιδιότητες της HTML	22
3.2	Τεχνικά χαρακτηριστικά του BeagleBoard-xM	30
3.3	Τεχνικά χαρακτηριστικά της οθόνης	32
4.1	Πίνακας users_information	39
4.2	Πίνακας layouts	40
4.3	Πίνακας contents	41
4.4	Πίνακας screens	42
4.5	Πίνακας content_scheduler	44
4.6	Πίνακας announcements	44
4.7	Πίνακας groups	45
4.8	Πίνακας users_privileges	45
4.9	Πίνακας screens_groups	46
4.10	Πίνακας μετρικών στοιχείων κώδικα	74

Κεφάλαιο 1

Εισαγωγή

Στη σημερινή εποχή, η διάδοση της πληροφορίας έχει γίνει άμεση και πολλοί περιορισμοί του παρελθόντος έχουν καταργηθεί. Καταλυτικό ρόλο σε αυτόν τον τομέα είναι η ανάπτυξη του Διαδικτύου, όπου με τις υψηλές ταχύτητες και τις επιλογές στο περιεχόμενο που προσφέρει, έχει αντικαταστήσει σε μεγάλο βαθμό τα παραδοσιακά μέσα ενημέρωσης και ψυχαγωγίας. Ακόμη, η εξάπλωση των οπτικοακουστικών μέσων διάδοσης των πληροφοριών έχει εισβάλει στη ζωή των ανθρώπων, κυρίως μέσω των έξυπνων κινητών τηλεφώνων (smartphones) και των δημόσιων οθονών. Στην παρούσα διπλωματική εργασία ασχοληθήκαμε με τη δημιουργία ενός δικτύου δημόσιων οθονών και την απομακρυσμένη διαχείρισή τους.

Δημόσιες οθόνες ονομάζονται αυτές οι οποίες τοποθετούνται σε διάφορους χώρους με σκοπό την παροχή πληροφοριών στο κοινό. Οι πληροφορίες θα πρέπει να επιλέγονται κατάλληλα ώστε να ανταποκρίνονται στις απαιτήσεις του μέρους και του κοινού. Το περιεχόμενο των οθονών μπορεί να είναι από ένα απλό κείμενο συνδυασμένο με κάποιες εικόνες, μέχρι δυναμικό περιεχόμενο που ανανεώνεται με βάση την ώρα της μέρας. Το υλικό μπορεί να αφορά την ενημέρωση των ανθρώπων για ένα γεγονός, την εύρεση νέων πληροφοριών, την προειδοποίηση για μία έκτακτη εξέλιξη, την ψυχαγωγία και τη διαφήμιση.

Τα πλεονεκτήματα αυτού του μέσου προβολής είναι σημαντικά. Πρόκειται για έναν αποτελεσματικό και γρήγορο τρόπο διάδοσης της πληροφορίας για τη διεπαφή μεταξύ των υπευθύνων του χώρου και των παρευρισκομένων. Το περιεχόμενο μπορεί να αλλάξει οποιαδήποτε στιγμή προσφέροντας νέες πληροφορίες. Ο χρήστης προσελκύεται από αυτήν τη μορφή επικοινωνίας και νιώθει πιο κοντά το χώρο αυτό, ειδικά αν έχει και άμεση διεπαφή. Εξοικονομούνται πόροι για τις υπόλοιπες ανάγκες της οντότητας, αφού το κόστος συντήρησης και λειτουργίας είναι σχετικά χαμηλό.

Στη χώρα μας, η χρήση των δημόσιων οθονών δεν είναι αρκετά διαδεδομένη. Περιορίζεται σε στατικό περιεχόμενο που σπάνια ανανεώνεται. Έτσι δεν μπορούν οι πολίτες να δείξουν το ανάλογο ενδιαφέρον. Εν αντιθέσει με το εξωτερικό που η ανάπτυξη του είναι ραγδαία και εισέρχονται σε όλο και μεγαλύτερα κομμάτια της κοινωνίας όπως είναι οι επιχειρήσεις, τα μουσεία, τα μέσα μαζικής συγκοινωνίας, τα πανεπιστήμια και αλλού.

Ένα δίκτυο δημόσιων οθονών αποτελείται από τις οθόνες οι οποίες προβάλλουν το περιεχόμενο στους χρήστες. Η οθόνη, η οποία έχει πρόσβαση στο διαδίκτυο, αντλεί τις πληροφορίες αυτές από έναν κεντρικό διακομιστή (server). Το προβαλλόμενο περιεχόμενο καθορίζεται από τους διαχειριστές του συστήματος οι οποίοι είναι υπεύθυνοι για την επιλογή και ενημέρωση του υλικού των οθονών.

Ο τομέας των δημόσιων οθονών είναι αρκετά διευρυμένος με πολλές εφαρμογές να έχουν αναπτυχθεί για αυτόν. Στη συνέχεια, θα αναφέρουμε μερικές από αυτές.

Στο πανεπιστήμιο Sunway της Μαλαισίας, αναπτύχθηκε μια εφαρμογή [1] στο ήδη υπάρχον δίκτυο δημόσιων οθονών, ώστε οι προβαλλόμενες πληροφορίες να μεταφράζονται σε μια άλλη γλώσσα. Οι φοιτητές του πανεπιστημίου, μπορούν μέσω της τεχνολογίας Bluetooth, ενός γραπτού μηνύματος κειμένου (SMS) ή ενός αιτήματος HTTP, να επιλέξουν τι περιεχόμενο θέλουν να δουν σε μια δημόσια οθόνη και σε ποια γλώσσα να μεταφραστεί.

Στη δημοσίευση "Timely and keyword-based dynamic content selection for public displays" [2], αναπτύχθηκε ένα σύστημα για δυναμική δημιουργία και επιλογή περιεχομένων σε μια δημόσια οθόνη. Συγκεκριμένα, το σύστημα δέχεται κάποιες λέξεις κλειδιά από το χρήστη, οι οποίες έχουν διαφορετική βαρύτητα η κάθε μία και με βάση αυτές, αναζητεί στο διαδίκτυο τις ανάλογες αναφορές. Έπειτα, αξιολογεί την χρονική στιγμή που εμφανίστηκαν και το είδος του περιεχομένου που βρέθηκε και με αυτόν τον τρόπο, ανανεώνει τη δημόσια οθόνη.

Η εφαρμογή UniDisplay [3], παρουσιάζει ένα τρόπο εμφάνισης κειμένου και εικόνων σε μια δημόσια οθόνη από τους χρήστες. Το σύστημα εγκαταστάθηκε σε ένα πανεπιστήμιο και αρχικά ρώτησαν τους φοιτητές του πανεπιστημίου, τι είδους περιεχόμενο θα ήθελαν να προβάλλεται, με ποιο τρόπο να καταχωρούνταν από τους χρήστες και πόσο ανεκτικοί είναι στην επίβλεψη του περιεχομένου τους. Έπειτα, δημιούργησαν το UniDisplay, το οποίο προβάλλει τα μηνύματα κάνοντας προηγουμένως ένα έλεγχο επί αυτών με βάση κάποια συγκεκριμένα στοιχεία.

Στο πανεπιστήμιο Oulu της Φινλανδίας, αναπτύχθηκε ένα σύστημα [4] για την εκτέλεση μιας εφαρμογής μέσω δημόσιας οθόνης. Συγκεκριμένα, εγκαταστάθηκαν στην οθόνη μια σειρά από εφαρμογές με δύο τρόπους εκτέλεσης. Ένας τρόπος είναι ένα μενού συντομεύσεων και ο άλλος τρόπος από τον κατάλογο όλων των

εφαρμογών. Με αυτήν την διαδικασία, μέτρησαν πως ο τρόπος εκτέλεσης μιας εφαρμογής επιδρά ώστε, ο χρήστης να αναζητήσει και τις άλλες εφαρμογές και να αλληλεπιδράσει με την οθόνη .

Στη δημοσίευση "Touch to Play - Exploring Touch-Based Mobile Interaction with Public Displays" [5] αναπτύχθηκε μια εφαρμογή για την επικοινωνία μιας δημόσιας οθόνης με ένα κινητό τηλέφωνο και την αλληλεπίδραση της, με το χρήστη. Ειδικότερα, στη δημόσια οθόνη εγκαταστάθηκε ένα παιχνίδι και η εικόνα του παιχνιδιού προβάλλονταν σε μια επιφάνεια μέσω ενός προβολέα (projector). Έπειτα, κάνοντας χρήση της τεχνολογία NFC, ο χρήστης με το κινητό του τηλέφωνο μπορούσε να επιδράσει με την οθόνη και να συνεχίσει να παίζει το παιχνίδι που προβάλλονταν.

Στο πανεπιστήμιο Columbia των ΗΠΑ, αναπτύχθηκε μια συσκευή [6] που μπορεί να φορεθεί (wearable device) και η οποία καθοδηγεί το χρήστη για μια τοποθεσία που θέλει να πάει. Συγκεκριμένα, ο χρήστης φοράει την συσκευή, κοιτάζει μια δημόσια οθόνη με ένα χάρτη και στη συνέχεια η συσκευή ανιχνεύει το μάτι του για να δει σε ποιο σημείο του χάρτη εστιάζει, και μέσω ενός ακουστικού του δίνει τις κατάλληλες οδηγίες για να το προσεγγίσει. Σε περίπτωση που η συσκευή μπορεί να επικοινωνήσει με τη δημόσια οθόνη, στο χάρτη τοποθετούνται ενδείξεις και πληροφορίες σχετικά με τον προορισμό του .

Πέρα από τις ερευνητικές εφαρμογές που αναφέρθηκαν, υπάρχουν και εμπορικά λογισμικά με τα οποία μπορεί να σχεδιαστεί το περιεχόμενο μιας δημόσιας οθόνης.

Το λογισμικό Xibo [7] είναι ένα ανοιχτού κώδικα πρόγραμμα, με το οποίο μπορούμε να σχεδιάσουμε και να προσθέσουμε περιεχόμενο σε μια δημόσια οθόνη. Με το Xibo ο χρήστης μπορεί να σχεδιάσει μια επιφάνεια και στη συνέχεια να προσθέσει περιεχόμενο όπως κείμενο, εικόνα, βίντεο, ενημερώσεις RSS και να το προωθήσει σε μια οθόνη. Στην οθόνη απαιτείται να τρέχει η ειδική εφαρμογή προβολής περιεχομένου.

Η εταιρία Scala [8], εξειδικεύεται στην ανάπτυξη εφαρμογών για δημόσιες οθόνες. Ανάλογα με τον τύπο οθόνης που χρειάζεται ένας χρήστης, υπάρχει και η αντίστοιχη κατηγορία εφαρμογών που μπορεί να χρησιμοποιήσει. Υποστηρίζονται οθόνες για σκοπούς όπως, εύρεση πληροφοριών, διαφήμισης, οικονομικών στοιχείων και αλληλεπίδρασης μεταξύ των χρηστών. Τις δημόσιες οθόνες της Scala, εμπιστεύονται μεγάλες εταιρίες του εξωτερικού.

Η εταιρία Rise Vision [9], αναπτύσσει ένα λογισμικό διαχείρισης δημόσιων οθονών. Ο χρήστης μπορεί να χρησιμοποιήσει κάποια από τις ήδη υπάρχοντες διατάξεις και να προσθέσει σε αυτήν το περιεχόμενο που επιθυμεί. Κάποιες από τις διατάξεις και τα περιεχόμενα που μπορεί να προσθέσει είναι δωρεάν. Το λο-

γισμικό επιτρέπει στο χρήστη να διαχειριστεί πολλαπλές οθόνες με υποστήριξη χρονοπρογραμματισμού, για την προβολή των περιεχομένων.

Στην παρούσα διπλωματική εργασία, δεν αναπτύχθηκε ένα σύστημα που επιδρά με τις δημόσιες οθόνες. Το σύστημα που αναπτύξαμε, αφορά τη σχεδίαση και την προβολή του περιεχομένου σε συνδυασμό με τη διαχείριση των οθονών. Το σύστημα μας είναι ενός ευρύτερου σκοπού που αποσκοπεί στην προβολή περιεχομένου και όχι κάποιου ειδικού, που σκοπεύει σε συγκεκριμένες ενέργειες. Έτσι, αυτή η διπλωματική εργασία έχει περισσότερα κοινά στοιχεία με τις αντίστοιχες εμπορικές εφαρμογές που παρουσιάσαμε σε σχέση με τις ερευνητικές δημοσιεύσεις. Σε σύγκριση με το εμπορικό λογισμικό, το σύστημα μας είναι απολύτως δωρεάν, δεν απαιτεί εξειδικευμένο λογισμικό για τη σχεδίαση και παρουσίαση του περιεχομένου και μπορεί να εκτελεστεί στους περισσότερους ηλεκτρονικούς υπολογιστές.

Το παρόν κείμενο, είναι δομημένο από πέντε κεφάλαια που καλύπτουν όλο το πεδίο των γνώσεων και των απαιτήσεων που χρειάστηκαν για τη δημιουργία της εργασίας.

Στο παρόν κεφάλαιο, παρουσιάστηκε το αντικείμενο που πραγματεύεται η εργασία και η γενική της δομή.

Στο 2^ο κεφάλαιο, γίνεται αναφορά στις τεχνολογίες που χρησιμοποιήθηκαν για τον προγραμματισμό του συστήματος. Γίνεται κατανοητό, πως η χρήση της κάθε τεχνολογίας βοήθησε στην υλοποίηση της εργασίας και τι σκοπό εξυπηρέτησε.

Στο 3^ο κεφάλαιο, παρουσιάζεται ο εξοπλισμός που χρησιμοποιήθηκε στην εργασία. Υπάρχει μια σύντομη περιγραφή του κάθε κομματιού και ο ρόλος που έχει στο σύστημα.

Στο 4^ο κεφάλαιο, αναλύεται πως κατασκευάστηκε το σύστημα. Εξηγούνται οι απαιτήσεις του συστήματος, πως αυτές διαμόρφωσαν το τελικό αποτέλεσμα και πως μπορεί ο χρήστης, να επιδράσει με το περιβάλλον της εργασίας. Ακόμη δίνονται μερικά κομμάτια κώδικα για την καλύτερη κατανόηση του συστήματος, μαζί με κάποια στατιστικά στοιχεία σχετικά με τον κώδικα.

Στο 5^ο κεφάλαιο, υπάρχουν τα συμπεράσματα που εξάχθηκαν μετά την ολοκλήρωση και τη δοκιμή του συστήματος, και με βάση αυτά, ορίστηκαν οι μελλοντικές επεκτάσεις του συστήματος.

Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο της διπλωματικής εργασίας, αναφερθήκαμε στα κίνητρα που μας οδήγησαν στην ανάπτυξη αυτού του συστήματος, περιγράφηκε το

γενικότερο μοντέλο, δόθηκαν οι πρώτες κατευθύνσεις για το πως λειτουργεί το υπάρχον σύστημα, παρουσιάστηκαν κάποιες υπάρχοντες εφαρμογές πάνω στις δημόσιες οθόνες και αναφέρθηκε η δομή των κεφαλαίων που ακολουθούν.

Κεφάλαιο 2

Θεωρητικό υπόβαθρο

Σκοπός του κεφαλαίου αυτού είναι η καταγραφή των εργαλείων και τεχνικών που χρησιμοποιήθηκαν για την ανάπτυξη της παρούσας διπλωματικής εργασίας. Αναλύεται τι είναι ο προγραμματισμός διαδικτύου, το υλικό μέρος του συστήματος και τα βοηθητικά εργαλεία λογισμικού που συντέλεσαν στην υλοποίησης της εργασίας.

2.1 Προγραμματισμός Διαδικτύου

Με τον όρο προγραμματισμό διαδικτύου [10] εννοούμε το σύνολο των εργασιών για την ανάπτυξη ενός ιστοτόπου που οι χρήστες έχουν πρόσβαση μέσω του διαδικτύου. Ο προγραμματισμός διαδικτύου περιλαμβάνει το σχεδιαστικό κομμάτι του ιστοτόπου και το παρασκηνιακό κομμάτι, που αποτελείται από τα προγραμματιστικά σενάρια (scripts). Τα προηγούμενα συνοψίζονται σε δύο κατηγορίες τεχνολογιών: την τεχνολογία προγραμματισμού πελάτη (client-side scripting) και την τεχνολογία προγραμματισμού διακομιστή (server-side scripting).

Η τεχνολογία προγραμματισμού πελάτη [11] σχετίζεται με τα προγραμματιστικά σενάρια που εκτελούνται από τον φυλλομετρητή ιστοσελίδων (web browser) του χρήστη. Με αυτόν τον τρόπο το περιεχόμενο της ιστοσελίδας μεταβάλλεται χωρίς να υπάρχει ενδιάμεση επικοινωνία με το διακομιστή.

Η τεχνολογία προγραμματισμού διακομιστή [12] σχετίζεται με τα προγραμματιστικά σενάρια που εκτελούνται από τον διακομιστή. Για να επιτευχθεί αυτό ο φυλλομετρητής στέλνει ένα αίτημα στο διακομιστή, ο διακομιστής το επεξεργάζεται και του απαντά ανάλογα.

2.1.1 HTML

Η HTML (HyperText Markup Language) [13] είναι η γλώσσα σήμανσης (markup language) που χρησιμοποιείται για την ανάπτυξη ιστοσελίδων. Με την HTML ο προγραμματιστής μπορεί να περιγράψει με τον κώδικα του μια ιστοσελίδα και ο φυλλομετρητής θα την μετατρέψει και θα την προβάλλει σε ένα ανθρώπινο φιλικό οπτικό αποτέλεσμα. Τα αρχεία της HTML έχουν την κατάληξη .html. Ακόμη, κώδικας HTML μπορεί να τοποθετηθεί σε αρχεία της PHP. Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε η πέμπτη έκδοση της HTML.

Τα αρχεία της HTML αποτελούνται από διαφορετικά στοιχεία (elements). Το κάθε στοιχείο εξυπηρετεί διαφορετικό σκοπό και διαμορφώνει τον τρόπο προβολής του από το φυλλομετρητή. Τα στοιχεία αυτά δηλώνουν διάφορα είδη δεδομένων. Τα συνηθέστερα από αυτά είναι οι παράγραφοι, οι εικόνες, οι λίστες, οι σύνδεσμοι, τα κουμπιά και οι φόρμες. Τα περισσότερα από τα στοιχεία απαρτίζονται από δύο ετικέτες (tags), την αρχική και την τελική ετικέτα. Οι ετικέτες αυτές γράφονται μέσα στα σύμβολα της ανισότητας και έχουν την μορφή <tag> και </tag> που δηλώνουν αντίστοιχα την αρχική και τελική ετικέτα. Ανάμεσα στην αρχική και τελική ετικέτα ο προγραμματιστής προσθέτει το περιεχόμενο που θέλει να προβάλλει. Ακόμη κάθε tag μπορεί να έχει καμία ή περισσότερες ιδιότητες (attributes) που παρέχουν επιπλέον εξειδίκευση στο στοιχείο αυτό. Οι ιδιότητες γράφονται πάντα στην αρχική ετικέτα και είναι της μορφής attribute="value".

Τα στοιχεία της HTML μετατρέπονται σε περιεχόμενο προβολής από τον φυλλομετρητή. Ο φυλλομετρητής είναι ένα πρόγραμμα το οποίο «μεταφράζει» τα στοιχεία της HTML σε οντότητες, οπτικά αντιληπτές από τον χρήστη. Οι πιο δημοφιλείς φυλλομετρητές είναι ο Google Chrome και ο Mozilla Firefox.

Στη συνέχεια ακολουθούν δύο πίνακες με τα πιο συνηθισμένα στοιχεία και ιδιότητες που χρησιμοποιήθηκαν στην παρούσα διπλωματική εργασία.

ΣΤΟΙΧΕΙΑ	ΣΧΟΛΙΟ
<!DOCTYPE>	Πάντα πρώτο σε ένα αρχείο html. Ορίζει τον αριθμό έκδοσης της HTML του αρχείου.
<html></html>	Δηλώνει ότι είναι αρχείο HTML και την αρχή του. Εδώ τοποθετούνται όλες οι υπόλοιπες ετικέτες.
<head></head>	Σε αυτό τοποθετούνται ετικέτες που έχουν σχέση με τον τίτλο, τη μορφοποίηση, τα μεταδεδομένα (metadata) και προγραμματιστικά σενάρια
<meta></meta>	Περιλαμβάνει πληροφορίες σχετικά με το έγγραφο όπως την κωδικοποίηση του. Περιλαμβάνεται στο <head>.
<title></title>	Ο τίτλος της ιστοσελίδας. Περιλαμβάνεται στο <head>.
<style></style>	Η μορφοποίηση της ιστοσελίδας. Περιλαμβάνεται στο <head>.
<script></script>	Φορτώνει ένα εξωτερικό αρχείο που εκτελείται τοπικά στο χρήστη. Περιλαμβάνεται στο <head>.
<body></body>	Εδώ τοποθετείται το κυρίως περιεχόμενο όπως κείμενο και εικόνα.
<p></p>	Ορίζει μια παράγραφο.
<h1></h1>...<h6></h6>	Ορίζει μια επικεφαλίδα. Όσο μικρότερο νούμερο τόσο μεγαλύτερο μέγεθος έχει.
<table></table>	Ορίζει ένα πίνακα.
	Ορίζει μια εικόνα.
<a>	Ορίζει έναν υπερσύνδεσμο (hyperlink) προς μια άλλη ιστοσελίδα.
<div></div>	Ορίζει μια περιοχή του εγγράφου με κοινά χαρακτηριστικά.
<button></button>	Ορίζει ένα κουμπί που όταν πατηθεί εκτελεί μια ενέργεια (συνήθως ένα script).
<form></form>	Ορίζει μια φόρμα που ο χρήστης συμπληρώνει με δεδομένα.
<input>	Ορίζει ένα πεδίο εισαγωγής δεδομένων. Τα δεδομένα μπορεί να είναι πολλών διαφορετικών τύπων.
<select></select>	Ορίζει ένα πεδία επιλογής τιμής από λίστα.

Πίνακας 2.1: Οι ετικέτες της HTML

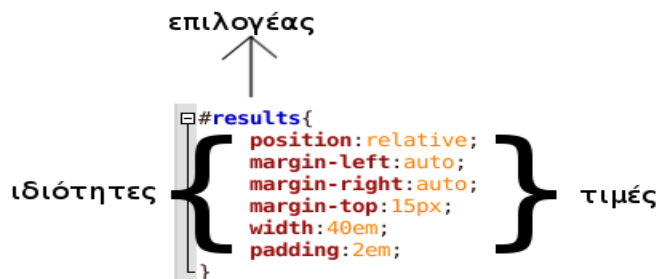
ΙΔΙΟΤΗΤΕΣ	ΣΧΟΛΙΟ
id	Χαρακτηρίζει μοναδικά το συγκεκριμένο στοιχείο. Δεν μπορεί να υπάρχει άλλο id με ίδιο όνομα.
class	Χαρακτηρίζει μια ομάδα στοιχείων για την μαζική διαχείρισή τους.
href	Δηλώνει μια διεύθυνση για ένα σύνδεσμο.
src	Δηλώνει μια διεύθυνση για μια εικόνα.
style	Μορφοποιεί το εκάστοτε στοιχείο.

Πίνακας 2.2: Οι ιδιότητες της HTML

2.1.2 CSS

Η CSS (Cascading Style Sheets) [14] είναι μια γλώσσα σήμανσης που περιγράφει πως εμφανίζονται τα διάφορα στοιχεία της HTML στην οθόνη. Συγκεκριμένα η CSS ελέγχει την εμφάνιση, την τοποθέτηση και την αισθητική μιας ιστοσελίδας. Χωρίς τη CSS η ιστοσελίδα φαίνεται εξωτερικά "φτωχή" για αυτό χρησιμοποιείται κατά κόρον για την καλύτερη εξυπηρέτηση του χρήστη. Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε η τρίτη έκδοση της CSS.

Η CSS γράφεται συνήθως σε εξωτερικά αρχεία που έχουν κατάληξη .css και συνδέονται με την ιστοσελίδα. Τα αρχεία αυτά δηλώνονται στο τμήμα <head> της HTML. Παρ'όλα αυτά ακόμη και αν δεν δηλωθεί ένα αρχείο .css ο κάθε φυλλομετρητής έχει εξορισμού κάποιες επιλογές διαμόρφωσης των στοιχείων. Σε κάθε αρχείο της CSS υπάρχουν ένας ή περισσότεροι κανόνες που ορίζουν την εμφάνιση των επιλεγμένων στοιχείων. Ένας κανόνας έχει την παρακάτω μορφή:



Σχήμα 2.1: Κανόνας της CSS

2.1.3 JavaScript

Η JavaScript [15] είναι μια γλώσσα προγραμματισμού σεναρίων (script language), που διερμηνεύεται (interpreted) από το φυλλομετρητή όταν η ιστοσελίδα έχει φορτωθεί. Αυτό σημαίνει ότι είναι μια γλώσσα που εκτελείται στην πλευρά του χρήστη (client-side) τρέχοντας τοπικά. Με αυτό τον τρόπο δεν επιβαρύνεται ο διακομιστής (server) με επιπλέον φόρτο εργασίας. Η JavaScript χρησιμοποιείται για τη δημιουργία δυναμικών περιβαλλόντων, όπου ανάλογα με τις επιλογές του χρήστη διαμορφώνεται το αντίστοιχο αποτέλεσμα.

Η JavaScript επιδρά πάνω στα διαφορετικά στοιχεία της HTML και αλλάζει τη λειτουργία τους και την εμφάνισή τους. Ακόμη μπορεί να επιδράσει με τρόπο που δεν είναι αντιληπτός στον χρήστη προσφέροντάς του μια προσωποποιημένη εμπειρία. Μεταξύ άλλων η JavaScript μπορεί να κάνει τα παρακάτω:

- Δημιουργία στοιχείων της HTML.
- Επικύρωση δεδομένων.
- Τροποποίηση της CSS.
- Επικοινωνία με τον διακομιστή.
- Αυτόματη αλλαγή των περιεχομένων της ιστοσελίδας.

Η JavaScript εκτελείται στο φυλλομετρητή του χρήστη και είναι συνήθως εκ των προτέρων ενεργοποιημένη. Ανάλογα με το φυλλομετρητή εκτελείται και η αντίστοιχη έκδοση της JavaScript με την τελευταία να είναι η 1.8. Συνήθως η JavaScript γράφεται σε αρχεία με κατάληξη .js τα οποία συνδέονται στην ιστοσελίδα στο τμήμα <head>.

jQuery

Η jQuery [16] είναι μια βιβλιοθήκη (library) της JavaScript. Με τον όρο βιβλιοθήκη εννοούμε ένα σύνολο συναρτήσεων όπου μπορεί ο προγραμματιστής να καλέσει χωρίς να τις αναπτύξει ο ίδιος και να πάρει το επιστρεφόμενο αποτέλεσμα. Η jQuery επιτρέπει στον προγραμματιστή να γράψει απλούστερα και γρηγορότερα ένα κομμάτι κώδικα αίροντας έτσι τους περιορισμούς της JavaScript. Επιπλέον συνοδεύεται από ένα πλήθος επεκτάσεων (plug-ins) γραμμένες από τους χρήστες, προσθέτοντας νέες συναρτήσεις στις ήδη υπάρχουσες. Στην παρούσα διπλωματική χρησιμοποιήθηκε η επέκταση jQuery UI (User Interface) [17] η οποία βοηθά

στην καλύτερα διεπαφή του χρήστη με την ιστοσελίδα. Η έκδοση της jQuery και της jQuery UI που χρησιμοποιήθηκε ήταν η 1.11.

Παρακάτω ακολουθούν δύο κομμάτια κώδικα που παράγουν το ίδιο αποτέλεσμα το πρώτο σε JavaScript και το δεύτερο σε jQuery.

```
function check_input() {
  var var1 = document.getElementById("id1").value;
  var var2 = document.getElementById("id2").value;

  if (var1 > 0 && var2 > var1) {
    document.getElementById("id1").style.backgroundColor="green";
    document.getElementById("id2").style.backgroundColor="green";
  }
}
```

Σχήμα 2.2: Συνάρτηση γραμμένη σε JavaScript

```
function check_input() {
  var var1 = $("#id1").val();
  var var2 = $("#id2").val();

  if (var1 > 0 && var2 > var1) {
    $("#id1, #id2").css("background-color", "green");
  }
}
```

Σχήμα 2.3: Συνάρτηση γραμμένη σε jQuery

2.1.4 PHP

Η PHP (PHP:Hypertext Preprocessor) [18] είναι μια γλώσσα προγραμματισμού σεναρίων που χρησιμοποιείται στην ανάπτυξη εφαρμογών διαδικτύου (web development) και μπορεί να ενσωματωθεί σε ένα αρχείο HTML. Η PHP είναι μια γλώσσα που εκτελείται στον διακομιστή (server-side) και το αποτέλεσμα της φαίνεται στο φυλλομετρητή του χρήστη. Επίσης η PHP μπορεί να τρέξει αυτόνομα για την εκτέλεση προγραμματιστικών σεναρίων. Τέλος, η PHP για να εκτελεστεί απαιτεί έναν εξυπηρετή ιστού (web server).

Μεταξύ άλλων οι λειτουργίες της PHP περιλαμβάνουν:

- Δυναμικός σχεδιασμός ιστοσελίδας.
- Συλλογή δεδομένων από φόρμες και επικύρωσή τους.
- Επικοινωνία με μια βάση δεδομένων.

- Εγκαθίδρυση ενός session ή cookie.
- Διαχείριση αρχείων.

Η κώδικας της PHP γράφεται μέσα στα αρχεία της HTML και περικλείεται μέσα στις ετικέτες `<?php ?>`. Ο φυλλομετρητής τότε κάνει ένα αίτημα στον διακομιστή και η απάντηση που λαμβάνει είναι σε μορφή HTML όπου μπορεί να την διαχειριστεί και να την προβάλει. Ο κώδικας της PHP τρέχει πάντα κατά τη διάρκεια της φόρτωσης της ιστοσελίδας. Τα αρχεία της PHP έχουν κατάληξη `.php` και είτε αποτελούν μέρος μια ιστοσελίδας είτε είναι αρχεία που δεν χρειάζονται την παρέμβαση του φυλλομετρητή για να τρέξουν. Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε η 5.5 έκδοση της PHP.

2.1.5 AJAX

Η AJAX (Asynchronous JavaScript and XML) [19] είναι μια τεχνική που επιτρέπει στο φυλλομετρητή να στείλει και να λάβει δεδομένα ασύγχρονα από τον διακομιστή. Αυτό σημαίνει ότι μπορεί να φορτωθεί ένα νέο στοιχείο χωρίς να ανανεωθεί ολόκληρη η ιστοσελίδα από την αρχή. Αυτό επιτρέπει στους προγραμματιστές να δημιουργούν δυναμικά περιβάλλοντα τα οποία διαμορφώνονται ανάλογα με τις εισόδους του χρήστη.

Η επικοινωνία μεταξύ του φυλλομετρητή και του διακομιστή γίνεται μέσω του αντικειμένου XMLHttpRequest. Η JavaScript διαχειρίζεται αυτό το αντικείμενο και όταν καλεστεί θα κάνει ένα αίτημα HTML στο διακομιστή. Ο διακομιστής θα λάβει αυτό το αίτημα, θα το επεξεργαστεί κατάλληλα χρησιμοποιώντας την PHP ή μια άλλη γλώσσα που εκτελείται σε αυτόν και θα απαντήσει με τα νέα δεδομένα που είναι της μορφής XML ή JSON. Τέλος ο φυλλομετρητής θα εμφανίσει την HTML και την CSS από τα δεδομένα που πήρε.

2.1.6 MySQL

Η MySQL [20] είναι ένα σύστημα διαχείρισης βάσης δεδομένων, όπου βασίζεται στην SQL για την επεξεργασία των δεδομένων στη βάση. Η MySQL εγκαθίσταται στο διακομιστή και διαχειρίζεται τα διάφορα αιτήματα που λαμβάνει. Τα δεδομένα αποθηκεύονται στη βάση σε μορφή εγγραφών σε πίνακες. Τα αιτήματα που λαμβάνει η MySQL είναι τα παρακάτω:

- Δημιουργία/Καταστροφή βάσης δεδομένων.

- Δημιουργία/Διαγραφή πίνακα σε βάση δεδομένων.
- Εισαγωγή/Ενημέρωση/Διαγραφή εγγραφής σε πίνακα.
- Αναζήτηση εγγραφής σε πίνακα.

Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε η έκδοση 5.5 της MySQL.

2.2 Βοηθητικά εργαλεία λογισμικού

Το λογισμικό που χρησιμοποιήθηκε, επιλέχθηκε βάση της εμπειρίας μας σε αυτό. Ως εκ τούτου, θα μπορούσαν να χρησιμοποιηθούν και άλλα προγράμματα με ανάλογα αποτελέσματα. Όλα τα παρακάτω προγράμματα είναι ανοιχτού κώδικα και διανέμονται δωρεάν.

2.2.1 LAMP

Το LAMP (Linux Apache MySQL PHP) [21] είναι μια πλατφόρμα ανάπτυξης διαδικτύου που βασίζεται στο λειτουργικό σύστημα Linux και περιλαμβάνει τον εξυπηρετή ιστού Apache, το σύστημα διαχείρισης βάσης δεδομένων MySQL και τον διερμηνευτή της γλώσσας PHP. Εγκαθιστώντας το LAMP μπορούμε να δημιουργήσουμε και να τρέξουμε δυναμικές ιστοσελίδες κάνοντας χρήση της PHP και της MySQL.

2.2.2 Geany

Το Geany [22] είναι ένας επεξεργαστής κειμένου (text editor). Στο Geany γράφτηκαν όλοι οι κώδικες της διπλωματικής εργασίας. Το Geany υποστηρίζει ένα πλήθος γλωσσών προγραμματισμού και οι βασικές του λειτουργίες περιλαμβάνουν το χρωματισμό λέξεων κλειδιών, την αυτόματη συμπλήρωση μεταβλητών και τελικών ετικετών της HTML, την σύμπτυξη μπλοκ κώδικα (code folding), λίστα συναρτήσεων και μεταβλητών. Το Geany μπορεί να παραμετροποιηθεί περισσότερο με την εγκατάσταση επεκτάσεων, που φτιάχνονται από τους χρήστες.

2.2.3 phpMyAdmin

Το phpMyAdmin [23] είναι ένα εργαλείο που μας επιτρέπει, τη διαχείριση μιας βάσης δεδομένων MySQL μέσω του ιστού. Συνήθως η σύνδεση σε μια βάση

δεδομένων γίνεται μέσω της κονσόλας. Αυτό μπορεί να αποτελέσει εμπόδιο για πολλούς χρήστες. Έτσι, το phpMyAdmin αίρει αυτόν τον περιορισμό παρέχοντας ένα φιλικό στο χρήστη περιβάλλον. Μέσω του phpMyAdmin ο διαχειριστής μπορεί να δει τις βάσεις δεδομένων του συστήματος, τους πίνακες και τις εγγραφές τους, καθώς και να επεξεργαστεί όλα αυτά τα στοιχεία αφού το εργαλείο αυτό αναλαμβάνει την μετατροπή των ενεργειών του χρήστη σε εντολές SQL.

2.2.4 GIMP

Το GIMP (GNU Image Manipulation Program) [24] είναι ένας επεξεργαστής εικόνας. Το GIMP χρησιμοποιήθηκε για το κόψιμο (crop) των εικόνων αυτού του κειμένου.

Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο, αναφερθήκαμε στις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος, κάποιες από τις βασικές τους δυνατότητες και πως αυτές μπορούν να εφαρμοστούν στην πράξη. Επίσης αναφερθήκαμε, στο εργαλείο λογισμικού που συντέλεσαν στην ανάπτυξη του συστήματος.

Κεφάλαιο 3

Υλικό μέρος συστήματος

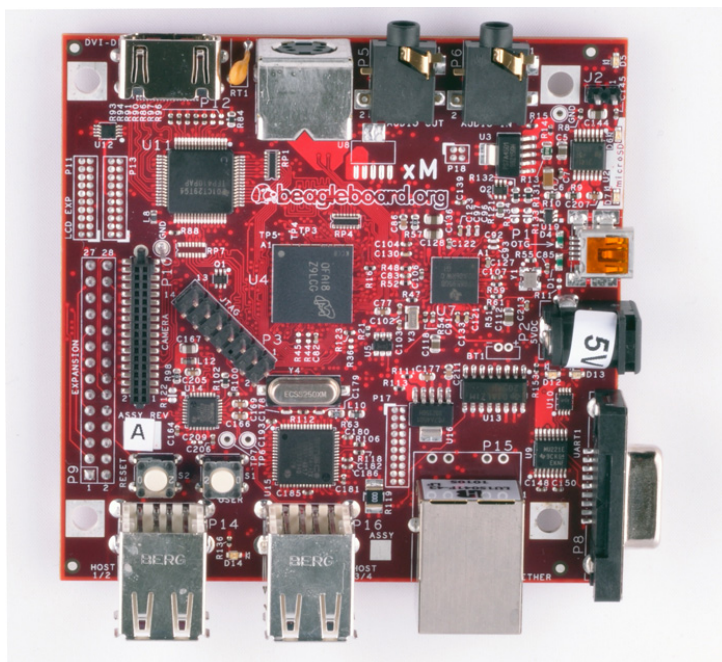
Σκοπός του κεφαλαίου αυτού, είναι η περιγραφή των υλικών κομματιών που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος διαχείρισης δημόσιων οθονών. Το υλικό αποτελείται από έναν υπολογιστή μονής πλακέτας (single-board computer) που λειτουργεί ως βάση και μια οθόνη. Στην παρούσα διπλωματική χρησιμοποιήθηκε το BeagleBoard-xM με λειτουργικό σύστημα Linux και η οθόνη Samsung SyncMaster T200. Θα μπορούσε να χρησιμοποιηθεί οποιοσδήποτε υπολογιστής που μπορεί να τρέξει ένα λειτουργικό σύστημα, είτε είναι μονής πλακέτας είτε επιτραπέζιος υπολογιστής. Το ίδιο ισχύει και για την οθόνη. Η επιλογή των εξαρτημάτων γίνεται με βάση τις ανάγκες των διαχειριστών και τις οικονομικές δυνατότητες. Σε ένα πραγματικό περιβάλλον θα μπορούσαν να τοποθετηθούν και να λειτουργήσουν παράλληλα, πολλές οθόνες η κάθε μία σε ξεχωριστή βάση.

3.1 BeagleBoard-xM

Το BeagleBoard-xM [25] [26] είναι ένας υπολογιστής μονής πλακέτας με όλα τα μέρη ενός επιτραπέζιου υπολογιστή ενσωματωμένα πάνω του. Το BeagleBoard-xM είναι μια χαμηλού κόστους πλακέτα που δεν περιλαμβάνει ανεμιστήρα (fan) και βασίζεται σε ένα χαμηλής κατανάλωσης επεξεργαστή ARM Cortex-A της Texas Instruments. Επιπρόσθετα, το BeagleBoard-xM μπορεί να τρέξει διαφορετικά λειτουργικά συστήματα, τα οποία εγκαθίστανται σε μια κάρτα microSD. Η τροφοδοσία του BeagleBoard-xM μπορεί να γίνει μέσω της θύρας USB OTG, αλλά για την υποστήριξη όλων των λειτουργιών χρειάζεται σύνδεση με σταθερό ρεύμα 5V. Στο BeagleBoard-xM συνδέθηκε η οθόνη που προβάλλει το περιεχόμενο της. Τέλος το κόστος του είναι 150€.

Επεξεργαστής	Texas Instruments Cortex A8 1GHz	
Μνήμη POP	Micron 4Gb MDDR SDRAM (512MB) 200MHz	
PMIC TPS65950	Ρυθμιστές ισχύος	
	Κωδικοποιητής και αποκωδικοποιητής ήχου	
	Επανεκκίνηση	
	USB OTG PHY	
Υποστήριξη απασφαλμάτωσης	14-ping JTAG	GPIO Pins
	UART	3 LED
PCB	(78.74 x 76.2 mm)	6 επίπεδα
Δείκτες	Τροφοδοσία, Σφάλμα τροφοδοσίας	2-User Controllable
	PMU	Τροφοδοσία USB
Θύρα HS USB 2.0 OTG	Mini AB USB connector	
	TPS65950 I/F	
Θύρες USB	SMSC LAN9514 Ethernet HUB	
	4 FS/LS/HS	Μέχρι 500mA για κάθε θύρα αν υπάρχει επαρκή τροφοδοσία
Ethernet	10/100	Από θύρα USB
Ήχος	3.5mm	
	Έξοδος L+R	Είσοδος L+R Stereo
SD/MMC	microSD	
Διεπαφή χρήστη	Κουμπί χρήστη	Κουμπί επανεκκίνησης
Βίντεο	DVI-D	S-Video
Κάμερα	Με σύνδεση	Υποστήριξη του Leopard Imaging
Τροφοδοσία	Τροφοδοσία USB	Τροφοδοσία σταθερού ρεύματος
Προστασία υπέρτασης	Απενεργοποίηση σε υπέρταση	
Κύρια επέκταση	Τροφοδοσία (5V & 1.8V)	UART
	McBSP	McSPI
	I ² C	GPIO
	MMC2	PWM
2 επεκτάσεις LCD	Πρόσβαση σε όλα τα σήματα LCD και στο I ² C	3.3V, 5V, 1.8V
Ήχος AUX	Σύνδεση 4 pin	McBSP2
Επέκταση AUX	MMC3	GPIO, ADC, HDQ

Πίνακας 3.2: Τεχνικά χαρακτηριστικά του BeagleBoard-xM



Σχήμα 3.1: Το BeagleBoard-xM

3.2 Debian

Το Debian [27] είναι ένα λειτουργικό σύστημα βασισμένο στον πυρήνα (kernel) του Linux. Το Debian δεν προορίζεται αποκλειστικά για υπολογιστές μονής πλακέτας, αλλά αποτελεί μια πλήρη λύση για έναν επιτραπέζιο υπολογιστή. Όλα τα σύγχρονα υλικά (hardware) υποστηρίζονται από αυτό και περιλαμβάνει χιλιάδες πακέτα έτοιμα για εγκατάσταση. Η εγκατάσταση λειτουργικού συστήματος είναι απαραίτητη, γιατί χρειάζεται για την εκτέλεση ενός φυλλομετρητή που θα προβάλει το περιεχόμενο της οθόνης. Αν και θα μπορούσε να χρησιμοποιηθεί κάποιο άλλο λειτουργικό σύστημα βασισμένο στο Linux, επιλέχθηκε το Debian γιατί θεωρείται σταθερότερο, έχει ένα μεγάλο αριθμό πακέτων τα οποία συντηρούνται εύκολα και υπάρχει καλή υποστήριξη από τη διαδικτυακή κοινότητα. Η έκδοση που εγκαταστάθηκε στο BeagleBoard-xM ήταν η 8^η έκδοση με το κωδικό όνομα "Jessie". Το Debian είναι λειτουργικό σύστημα ανοιχτού κώδικα και διανέμεται δωρεάν.

3.3 Samsung SyncMaster T200

Η οθόνη που χρησιμοποιήθηκε είναι η Samsung SyncMaster T200 [28]. Είναι μια οθόνη 20", που συνδέθηκε στο BeagleBoard-xM μέσω της θύρας HDMI και μέσω αυτής παρουσιάζεται το προβαλλόμενο περιεχόμενο.



Σχήμα 3.2: Η Samsung SyncMaster T200

Κατασκευαστής	Samsung
Μοντέλο	SyncMaster T200
Μέγεθος	20 ίντσες
Τεχνολογία	TFT
Μέγιστη ανάλυση	1680x1050
Χρώματα	16.7 εκατομμύρια
Φωτεινότητα	300 cd/m ²
Αντίθεση	1000:1
Τροφοδοσία	AC 100 - 240V

Πίνακας 3.3: Τεχνικά χαρακτηριστικά της οθόνης

Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο, αναφερθήκαμε στα υλικά κομμάτια που αποτελούν το σύστημα και που εμείς χρησιμοποιήσαμε. Περιγράψαμε τα τεχνικά τους χαρακτηριστικά και το ρόλος τους στο σύστημα.

Κεφάλαιο 4

Λογισμικό μέρος συστήματος

Σκοπός αυτού του κεφαλαίου είναι η περιγραφή του λογισμικού μέρους της διπλωματικής εργασίας. Το λογισμικό μέρος είναι το βασικό μέρος της εργασίας. Θα αναδειχθεί πως συνδέονται οι τεχνολογίες που αναφέρθηκαν στο Κεφάλαιο 2 και δημιουργούν το συγκεκριμένο ιστότοπο. Θα περιγραφτεί πως ο χρήστης μπορεί να δημιουργήσει το περιβάλλον που επιθυμεί, να το προωθήσει σε μια οθόνη και πως η οθόνη να το προβάλλει.

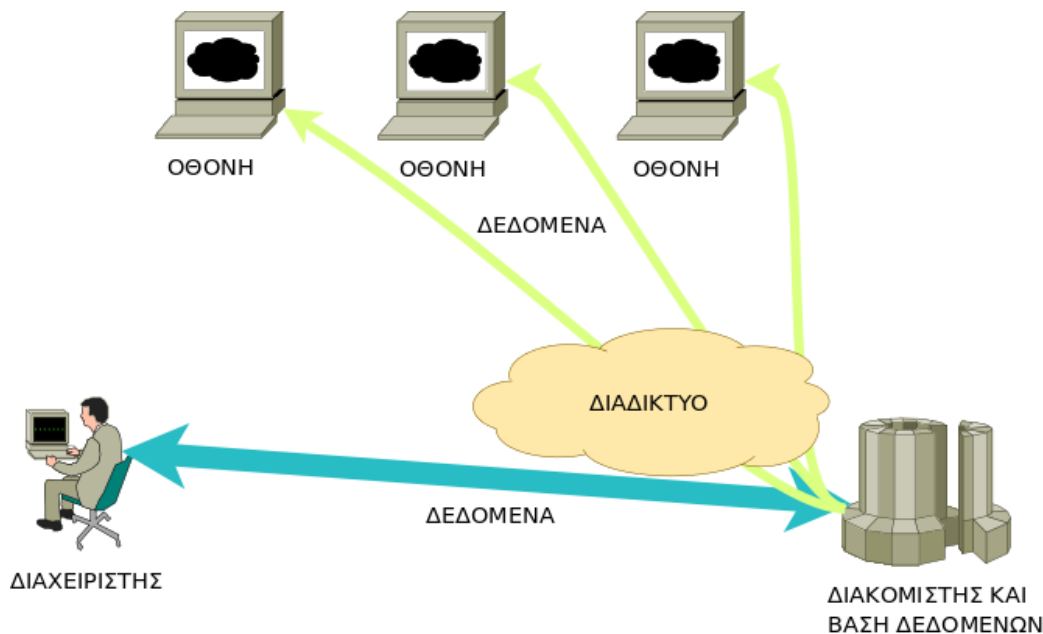
Οι ενότητες του κεφαλαίου είναι:

- Περιγραφή συστήματος
- Απαιτήσεις συστήματος
- Σχεδιασμός βάσης δεδομένων
- Λειτουργίες ιστοτόπου
- Ασφάλεια συστήματος
- Ανάλυση κώδικα
- Μετρικά στοιχεία κώδικα
- Δοκιμές

4.1 Περιγραφή συστήματος

Το σύστημα που αναπτύχθηκε, χρησιμοποιεί τις τεχνολογίες που αναφέραμε στο 2^ο Κεφάλαιο, ώστε να αναπτυχθεί μια εφαρμογή που προβάλλει δυναμικό περιεχόμενο στην οθόνη. Το σύστημα αποτελείται από χρήστες, οι οποίοι δημιουργούν και ελέγχουν τα περιεχόμενα και πολλαπλές οθόνες που μπορούν να προβάλλουν διαφορετικά περιεχόμενα. Συνοπτικά το σύστημα λειτουργεί ως εξής:

- Ο χρήστης ενεργοποιεί μια οθόνη. Αυτό σημαίνει ότι αυτή η οθόνη μπορεί να δεχτεί και να προβάλλει ένα περιεχόμενο.
- Ο χρήστης δημιουργεί και αποθηκεύει στη βάση δεδομένων του συστήματος ένα περιεχόμενο. Το κάθε περιεχόμενο αποτελείται από στοιχεία, τα οποία ανάλογα με τις ιδιότητες τους, διαμορφώνουν το τελικό αποτέλεσμα.
- Ο χρήστης μπορεί να προωθήσει ένα περιεχόμενο σε μια οθόνη.
- Ο διακομιστής αναλαμβάνει την προώθηση των περιεχομένων σε κάθε οθόνη. Αν αυτή είναι ενεργή, τότε το περιεχόμενο θα προβληθεί άμεσα.
- Η οθόνη, όταν ενεργοποιηθεί, δέχεται το περιεχόμενο, αναγνωρίζει τις ιδιότητες του και ανάλογα με αυτές, παρουσιάζει την τελική μορφή του.
- Η οθόνη έχει ορισμένο χρόνο λειτουργίας. Όταν θα πάψει να είναι ενεργή, θα σταματήσει η επικοινωνία με το διακομιστή, συνεπώς δεν μπορεί να αλλάξει το περιεχόμενό της.



Σχήμα 4.1: Αρχιτεκτονική συστήματος

4.2 Απαιτήσεις συστήματος

Βασιζόμενοι στην περιγραφή του συστήματος και έπειτα από προσπάθειες ώστε να είναι καλύτερη η εμπειρία του χρήστη, μπορούμε να εξάγουμε τις απαιτήσεις αυτού. Με τον όρο απαιτήσεις συστήματος εννοούμε τις προδιαγραφές που πρέπει να τηρεί ένα σύστημα ώστε να είναι λειτουργικό και διαχειρίσιμο. Για το παρόν σύστημα διαχείρισης δημόσιων οθονών οι απαιτήσεις είναι οι παρακάτω:

- Το σύστημα θα πρέπει να υποστηρίζει χρήστες. Ο κάθε χρήστης διαχειρίζεται τις δικές του οθόνες. Μόνο ο υπερχρήστης (root) μπορεί να προσθέσει νέους χρήστες. Για κάθε χρήστη αποθηκεύονται τα στοιχεία σύνδεσης του, μαζί με κάποια προσωπικά στοιχεία.
- Θα πρέπει να υπάρχουν ομάδες διαχείρισης. Μέλη της κάθε ομάδας μπορούν να είναι οι χρήστες και οι οθόνες. Μόνο ο υπερχρήστης μπορεί να προσθέσει νέες ομάδες.
- Θα πρέπει οι χρήστες να ανήκουν σε ομάδες. Ο κάθε χρήστης μπορεί να ανήκει σε μία ή περισσότερες ομάδες διαχείρισης που του επιτρέπει να ελέγχει

και να ανανεώνει το περιεχόμενο των οθονών αυτής της ομάδας. Ο υπερ-χρήστης αποφασίζει ποιοι χρήστες ανήκουν σε μια ομάδα. Ο υπερχρήστης είναι μέλος σε κάθε ομάδα.

- Θα πρέπει να υποστηρίζονται πολλαπλές οθόνες. Η κάθε οθόνη μπορεί να προβάλλει το δικό της περιεχόμενο που καθορίζεται από έναν χρήστη. Μόνο ο υπερχρήστης μπορεί να προσθέσει νέες οθόνες.
- Θα πρέπει οι οθόνες να ανήκουν σε ομάδες. Κάθε οθόνη μπορεί να ανήκει σε μία ή περισσότερες ομάδες διαχείρισης και ελέγχεται από τους χρήστες της ομάδας. Ο υπερχρήστης αποφασίζει ποιες οθόνες ανήκουν σε μια ομάδα.
- Θα πρέπει να μπορούν να δημιουργηθούν διατάξεις ιστοσελίδας (layouts). Αυτό πρέπει να γίνεται χωρίς τη γνώση HTML από τη μεριά του χρήστη. Αφού δημιουργηθούν, μετά αποθηκεύονται στη βάση δεδομένων. Οι διατάξεις θα πρέπει να είναι σε τέτοια μορφή ώστε να προσαρμόζονται στην ανάλυση κάθε οθόνης. Ο κάθε χρήστης μπορεί να δημιουργήσει πολλαπλές διατάξεις. Ο κάθε χρήστης μπορεί να επεξεργαστεί μόνο τις διατάξεις που δημιούργησε ο ίδιος. Ο υπερχρήστης μπορεί να ελέγξει όλες τις διατάξεις.
- Θα πρέπει σε κάθε διάταξη να μπορεί να προστεθεί περιεχόμενο. Το περιεχόμενο που μπορεί να προστεθεί προέρχεται από ένα ευρύ φάσμα επιλογών που περιλαμβάνει, μεταξύ άλλων, κείμενο, εικόνα, βίντεο και ενημερώσεις RSS. Ο κάθε χρήστης μπορεί να προσθέσει το δικό του περιεχόμενο και να το διαχειριστεί μόνο αυτός. Αφού το αποθηκεύσει στη βάση δεδομένων μπορεί να το προωθήσει σε μια ή περισσότερες οθόνες. Ο κάθε χρήστης μπορεί να δημιουργήσει πολλαπλά περιεχόμενα. Ο υπερχρήστης μπορεί να ανανεώσει μια οθόνη με όλα τα δημιουργηθέντα περιεχόμενα.
- Θα πρέπει το περιεχόμενο να αλλάζει δυναμικά. Αυτό σημαίνει ότι, ο χρήστης μπορεί να επιλέξει ποια περιεχόμενα θέλει να προβάλλονται και με ποια σειρά, και αυτά θα αλλάζουν σε τακτά χρονικά διαστήματα. Ακόμη, ένας παρευρισκόμενος στο χώρο της οθόνης θα μπορεί να διαλέξει ένα από αυτά τα περιεχόμενα οποιαδήποτε στιγμή και αυτό να προβληθεί άμεσα.
- Θα πρέπει να υπάρχουν έκτακτα περιεχόμενα. Δηλαδή μια οθόνη μπορεί να σταματάει την προβολή του περιεχομένου της, με σκοπό να προβάλλει ένα άλλο εκείνη τη χρονική στιγμή και για όσο διάστημα το ορίσει ο χρήστης. Μετά τα πέρασ του διαστήματος αυτού επανέρχεται το κανονικό περιεχόμενο.

- Θα πρέπει η κάθε οθόνη να είναι συνδεδεμένη σε μια βάση. Όπως εξηγήθηκε στο Κεφάλαιο 3, η βάση μπορεί να είναι ένας οποιοσδήποτε υπολογιστής ο οποίος έχει πρόσβαση στο διαδίκτυο, διαθέτει γραφικό περιβάλλον και μπορεί να τρέξει ένα φυλλομετρητή ιστοσελίδων. Στο φυλλομετρητή θα πρέπει να είναι ενεργοποιημένη η JavaScript και τα cookies.
- Θα πρέπει να υπάρχει ένα εύχρηστο περιβάλλον. Πιθανώς οι χρήστες να μην έχουν μεγάλη εξοικείωση με την τεχνολογία, έτσι το περιβάλλον θα πρέπει να τους παρέχει χρήσιμες απαντήσεις μετά από συγκεκριμένες ενέργειες, καθώς και μια φιλική εμπειρία περιήγησης στον ιστότοπο.

4.3 Σχεδιασμός βάσης δεδομένων

Με βάση λοιπόν τις απαιτήσεις του συστήματος, σχεδιάστηκε ανάλογα και η βάση δεδομένων. Όπως θα γίνει φανερό, στη βάση δεδομένων αποθηκεύονται όλες οι πληροφορίες που είναι αναγκαίες για τη λειτουργία του ιστοτόπου. Ο αριθμός των πινάκων που απαρτίζουν τη βάση δεδομένων είναι εννιά και ακολουθούν παρακάτω μαζί με τα πεδία τους.

4.3.1 Πίνακας *users_information*

Σε αυτόν τον πίνακα αποθηκεύονται οι πληροφορίες του χρήστη που έχουν σχέση με τα προσωπικά του στοιχεία.

Πεδίο	Τύπος
id	unsigned int(10)
username	varchar(15)
password	varchar(255)
password_plain	varchar(40)
fname	varchar(25)
lname	varchar(25)
email	varchar(50)

Πίνακας 4.1: Πίνακας *users_information*

- id: Μοναδικός αύξων αριθμός για κάθε χρήστη. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το id αυξάνεται αυτόματα κάθε φορά

που υπάρχει μια νέα εγγραφή στον πίνακα.

- `username`: Το όνομα χρήστη ενός μέλους. Είναι μοναδικό, μπορεί να περιλαμβάνει οποιουδήποτε χαρακτήρες αρκεί να μην ξεπερνούν το όριο των 15.
- `password`: Το συνθηματικό του χρήστη. Είναι αποθηκευμένο σε κατακερματισμένη (hashed) μορφή χρησιμοποιώντας τον εξ ορισμού αλγόριθμο κατακερματισμού της PHP, που το Φεβρουάριο του 2016 είναι ο `bcrypt` [29]. Το μέγεθος του πεδίου είναι 255 χαρακτήρες ώστε να καλύψει κάθε αλγόριθμο κατακερματισμού που μπορεί να προκύψει στο μέλλον.
- `password_plain`: Το συνθηματικό του χρήστη σε καθαρή μορφή. Κανονικά σε μια βάση δεδομένων απαγορεύεται η αποθήκευση ευαίσθητων πληροφοριών χωρίς κρυπτογράφηση. Αποθηκεύτηκε έτσι για τις ανάγκες της εργασίας για συχνή δοκιμή διάφορων σεναρίων.
- `fname`: Το μικρό όνομα του χρήστη. Αποτελείται από χαρακτήρες με μέγιστο μήκος το 25.
- `lname`: Το επώνυμο του χρήστη. Ίδιος τύπος πεδίου με το `fname`.
- `email`: Η ηλεκτρονική διεύθυνση ταχυδρομείου (e-mail) του χρήστη. Πριν εισαχθεί στη βάση δεδομένων ελέγχεται ότι είναι σε μια μορφή διεύθυνσης αποδεκτή από το σύστημα. Το μέγεθος του πεδίου δεν πρέπει να ξεπερνά τους 50 χαρακτήρες.

4.3.2 Πίνακας *layouts*

Σε αυτόν τον πίνακα αποθηκεύονται οι διατάξεις που δημιουργούνται από τους χρήστες.

Πεδίο	Τύπος
<code>id</code>	<code>unsigned int(10)</code>
<code>user_id</code>	<code>unsigned int(10)</code>
<code>layout_html</code>	<code>text</code>
<code>div_id</code>	<code>unsigned smallint(5)</code>
<code>preview</code>	<code>text</code>

Πίνακας 4.2: Πίνακας *layouts*

- **id**: Μοναδικός αύξων αριθμός για κάθε διάταξη. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το id αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.
- **user_id**: Ο αύξων αριθμός του χρήστη που δημιούργησε και του ανήκει αυτή η διάταξη.
- **layout_html**: Ο κώδικας HTML που απαιτείται για τη συγκεκριμένη διάταξη. Κάθε διάταξη αποτελείται από ένα ή περισσότερα κενά <div> της HTML. Είναι τύπου text που δηλώνει μια συμβολοσειρά (string) με μέγιστο μέγεθος 65535 χαρακτήρων.
- **div_id**: Αριθμός που δείχνει το τελευταίο <div> που δημιουργήθηκε σε αυτή τη διάταξη. Είναι ακέραιος αριθμός με μέγιστο όριο το 65535.
- **preview**: Μια μικρογραφία της συγκεκριμένης διάταξης. Ίδιος τύπος με το πεδίο *layout_html*.

4.3.3 Πίνακας *contents*

Σε αυτόν τον πίνακα αποθηκεύονται τα περιεχόμενα των διατάξεων που δημιουργούνται από τους χρήστες.

Πεδίο	Τύπος
id	unsigned int(10)
user_id	unsigned int(10)
name	varchar(256)
content_html	text
backcolor	varchar(7)
backimage_url	text
backimage_option	enum('stretched', 'tiled')

Πίνακας 4.3: Πίνακας contents

- **id**: Μοναδικός αύξων αριθμός για κάθε περιεχόμενο. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το id αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.

- `user_id`: Ο αύξων αριθμός του χρήστη που δημιούργησε και του ανήκει αυτό το περιεχόμενο.
- `name`: Το όνομα του συγκεκριμένου περιεχόμενου. Είναι τύπου `varchar`, δηλαδή αποτελεί μια συμβολοσειρά με μέγιστο μήκος 255 χαρακτήρων.
- `content_html`: Μια διάταξη συμπεριλαμβανομένης του περιεχομένου του κάθε `<div>` της HTML. Είναι τύπου `text` δηλαδή μια συμβολοσειρά με μέγιστο μέγεθος χαρακτήρων 65535.
- `backcolor`: Το χρώμα παρασκήνιου (`background`) για αυτό το περιεχόμενο. Είναι σε δεκαεξαδική μορφή ακολουθημένη από το σύμβολο της δέσμης, δηλαδή είναι της μορφής `#XXXXXX` όπου X ψηφία του δεκαεξαδικού συστήματος. Για αυτό το λόγο το πεδίο είναι τύπου `varchar` με μήκος ακριβώς 7 χαρακτήρων.
- `backimage_url`: Ένας σύνδεσμος που δείχνει σε μια εικόνα, για να τοποθετηθεί στο παρασκήνιο. Ίδιος τύπος με το πεδίο `content_html`.
- `backimage_option`: Ο τρόπος εμφάνισης της εικόνας παρασκήνιου, εφόσον υπάρχει. Αν έχει τιμή `"stretched"`, η εικόνα καλύπτει όλο το παρασκήνιο. Αν έχει τιμή `"tiled"`, η εικόνα επαναλαμβάνεται.

4.3.4 Πίνακας *screens*

Σε αυτόν τον πίνακα αποθηκεύονται οι πληροφορίες της οθόνης και το τρέχον περιεχόμενο που προβάλλεται.

Πεδίο	Τύπος
<code>id</code>	<code>unsigned int(10)</code>
<code>name</code>	<code>varchar(25)</code>
<code>description</code>	<code>varchar(300)</code>
<code>unique_id</code>	<code>varchar(10)</code>
<code>webid</code>	<code>varchar(10)</code>
<code>valid_time</code>	<code>int(11)</code>
<code>qrcode_id</code>	<code>varchar(15)</code>
<code>content_id</code>	<code>unsigned int(10)</code>

Πίνακας 4.4: Πίνακας *screens*

- **id:** Μοναδικός αύξων αριθμός για κάθε οθόνη. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το id αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.
- **name:** Το όνομα της συγκεκριμένης οθόνης. Είναι τύπου varchar, δηλαδή αποτελεί μια συμβολοσειρά με μέγιστο μήκος 25 χαρακτήρων.
- **description:** Μια περιγραφή της οθόνης. Αποτελεί μια συμβολοσειρά με μέγιστο μήκος 300 χαρακτήρες.
- **unique_id:** Ένα αλφαριθμητικό που χαρακτηρίζει μοναδικά μια οθόνη. Χρησιμοποιείται για την απευθείας ενεργοποίηση μιας οθόνης χωρίς τη σύνδεση του χρήστη. Αποτελείται από ακριβώς 10 χαρακτήρες.
- **webid:** Μοναδικό αλφαριθμητικό ακριβώς 10 χαρακτήρων, που αποτελεί το αναγνωριστικό κάθε οθόνης ώστε να μπορεί να ανανεωθεί το περιεχόμενό της. Ορίζεται όταν η οθόνη ενεργοποιείται από ένα χρήστη.
- **valid_time:** Χρονική στιγμή κατά την οποία η οθόνη σταματάει την επικοινωνία με το διακομιστή και έτσι δεν μπορεί να λάβει νέα δεδομένα. Η αποθήκευση γίνεται σε έναν ακέραιο αριθμό που δείχνει πόσα δευτερόλεπτα έχουν περάσει από την 1/1/1970 και ώρα 00:00:00 UTC. Αυτός ο τρόπος αναπαράστασης της ώρας είναι γνωστός και ως χρόνος Unix (Unix time) [30].
- **qrcode_id:** Μοναδικό αλφαριθμητικό ακριβώς 15 χαρακτήρων, που εισάγεται σε έναν κώδικα QR και όταν σαρωθεί από ένα θεατή της οθόνης τον οδηγεί σε ένα σύνδεσμο ώστε να αλλάξει το περιεχόμενο της οθόνης. Ανανέωνεται σε κάθε επιτυχή αλλαγή περιεχομένου.
- **content_id:** Ο αύξων αριθμός του περιεχομένου που προβάλλεται εκείνη τη δεδομένη στιγμή.

4.3.5 Πίνακας *content_scheduler*

Σε αυτόν τον πίνακα αποθηκεύονται τα περιεχόμενα που εναλλάσσονται ανά τακτά χρονικά διαστήματα για κάθε οθόνη.

Πεδίο	Τύπος
screen_id	unsigned int(10)
content_id	unsigned int(10)
queue	unsigned smallint(5)
refresh_rate	unsigned mediumint(9)

Πίνακας 4.5: Πίνακας content_scheduler

- screen_id: Ο αύξων αριθμός της οθόνης.
- content_id: Ο αύξων αριθμός του περιεχομένου.
- queue: Η σειρά που θα εμφανιστεί το συγκεκριμένο περιεχόμενο σε αυτήν την οθόνη. Είναι ένας ακέραιος αριθμός με μέγιστη τιμή το 65535.
- refresh_rate: Η χρονική διάρκεια που θα προβληθεί το συγκεκριμένο περιεχόμενο. Αποθηκεύεται σε δευτερόλεπτα και είναι ένας ακέραιος αριθμός με μέγιστη τιμή το 16777215.

4.3.6 Πίνακας *announcements*

Σε αυτόν τον πίνακα αποθηκεύονται τα περιεχόμενα που θέλει ο χρήστης να εμφανίζονται σε έκτακτες στιγμές παρακάμπτοντας τον πίνακα *content_scheduler*.

Πεδίο	Τύπος
screen_id	unsigned int(10)
content_id	unsigned int(10)
datetimefrom	int(10)
datetimeto	int(10)

Πίνακας 4.6: Πίνακας announcements

- screen_id: Ο αύξων αριθμός της οθόνης.
- content_id: Ο αύξων αριθμός του περιεχομένου.
- datetimefrom: Η χρονική στιγμή έναρξης της προβολής του περιεχομένου. Η μορφή του είναι σε χρόνο Unix.

- `datetimeo`: Η χρονική στιγμή λήξης της προβολής του περιεχομένου. Η μορφή του είναι σε χρόνο Unix.

4.3.7 Πίνακας *groups*

Σε αυτόν τον πίνακα αποθηκεύονται οι ομάδες διαχείρισης.

Πεδίο	Τύπος
id	unsigned int(10)
name	varchar(25)
description	varchar(300)

Πίνακας 4.7: Πίνακας *groups*

- `id`: Μοναδικός αύξων αριθμός για κάθε ομάδα διαχείρισης. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το `id` αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.
- `name`: Το όνομα της ομάδας διαχείρισης. Είναι μια συμβολοσειρά με μέγιστο αριθμό χαρακτήρων 25.
- `description`: Μια περιγραφή της ομάδας διαχείρισης. Είναι μια συμβολοσειρά με μέγιστο αριθμό χαρακτήρων 300.

4.3.8 Πίνακας *users_privileges*

Σε αυτόν τον πίνακα αποθηκεύονται σε ποιες ομάδες ανήκει ένας χρήστης.

Πεδίο	Τύπος
rule	unsigned int(10)
user_id	unsigned int(10)
group_id	unsigned int(10)

Πίνακας 4.8: Πίνακας *users_privileges*

- `rule`: Μοναδικός αύξων αριθμός για κάθε συνδυασμό χρήστη και ομάδα διαχείρισης. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το

rule αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.

- user_id: Ο αύξων αριθμός του χρήστη.
- group_id: Ο αύξων αριθμός της ομάδας διαχείρισης.

4.3.9 Πίνακας *screens_groups*

Σε αυτόν τον πίνακα αποθηκεύονται σε ποιες ομάδες ανήκει ένας χρήστης.

Πεδίο	Τύπος
rule	unsigned int(10)
screen_id	unsigned int(10)
group_id	unsigned int(10)

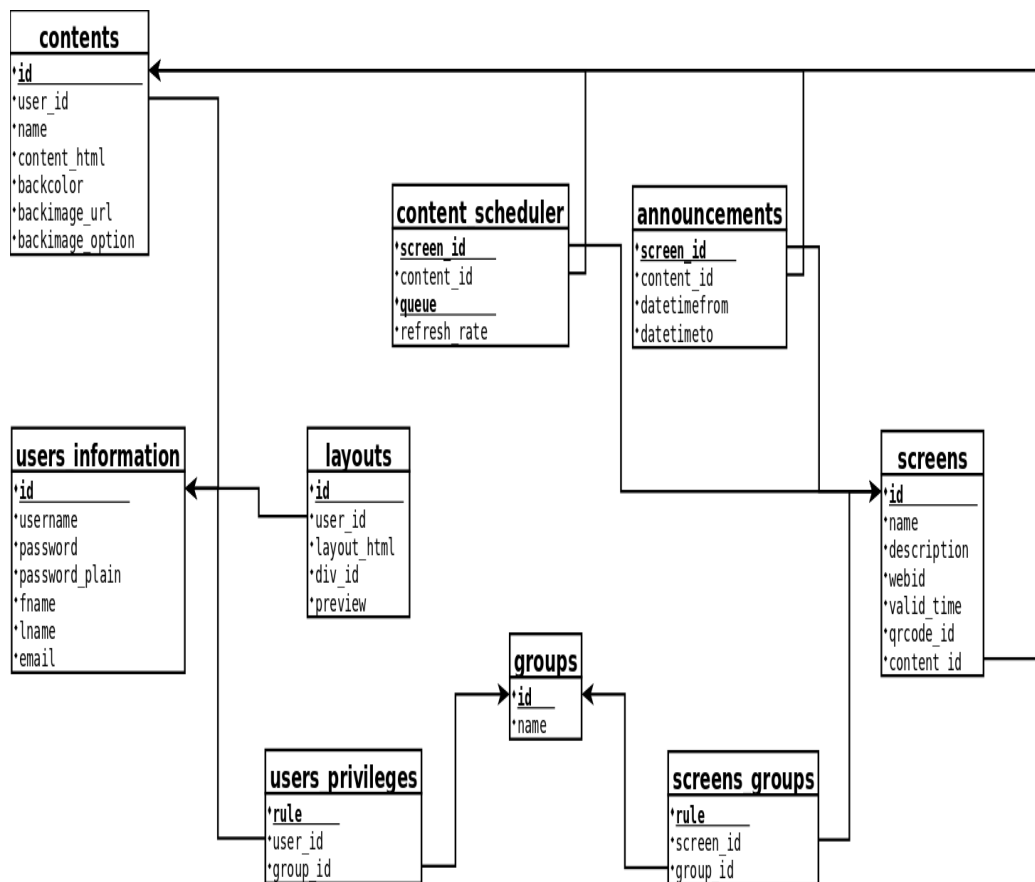
Πίνακας 4.9: Πίνακας *screens_groups*

- rule: Μοναδικός αύξων αριθμός για κάθε συνδυασμό οθόνης και ομάδα διαχείρισης. Είναι ένας ακέραιος αριθμός με μέγιστο όριο το 4294967295. Το rule αυξάνεται αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή στον πίνακα.
- screen_id: Ο αύξων αριθμός της οθόνης.
- group_id: Ο αύξων αριθμός της ομάδας διαχείρισης.

4.3.10 Διάγραμμα οντοτήτων-συσχετίσεων

Με τον όρο οντότητα εννοούμε ένα αντικείμενο ενός συστήματος το οποίο έχει διακριτή ύπαρξη σε σχέση με τα υπόλοιπα αντικείμενα. Στην παρούσα διπλωματική εργασία οντότητες είναι οι πίνακες της βάσης δεδομένων. Κάθε οντότητα περιγράφεται από ένα σύνολο γνωρισμάτων. Τα γνωρίσματα για τους πίνακες είναι τα πεδία τους. Με τον όρο συσχέτιση εννοούμε τη σχέση μεταξύ δύο ή περισσότερων οντοτήτων [31].

Ένα διάγραμμα οντοτήτων-συσχετίσεων [32] είναι η γραφική αναπαράσταση ενός συστήματος που δείχνει τις σχέσεις μεταξύ των αντικειμένων αυτού του συστήματος. Στη δική μας περίπτωση, δείχνει τη σχέση μεταξύ των πεδίων των πινάκων της βάσης δεδομένων.



Σχήμα 4.2: Το διάγραμμα οντοτήτων-συσχετίσεων της βάσης δεδομένων

Όπως έγινε αντιληπτό και από την ανάλυση των πινάκων, κάποια πεδία εξαρτώνται από άλλα. Αυτό εξασφαλίζει ότι, στα εξαρτώμενα πεδία μπορούν να εισαχθούν μόνο τιμές που έχουν εισαχθεί στα ανεξάρτητα πεδία και ότι αν διαγραφεί ένα ανεξάρτητο πεδίο θα διαγραφτεί και το εξαρτώμενο.

Οι εξαρτήσεις που υπάρχουν, είναι:

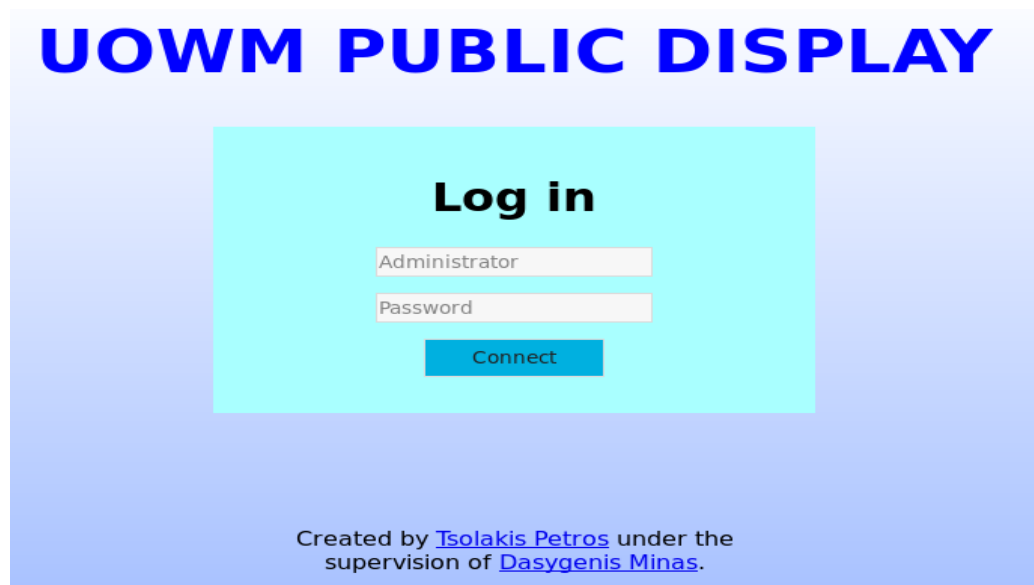
- Το πεδίο *id* του πίνακα *users_information*, συσχετίζεται με το πεδίο *user_id* των πινάκων *layouts*, *contents*, *users_privileges*.
- Το πεδίο *id* του πίνακα *screens*, συσχετίζεται με το πεδίο *screen_id* των πινάκων *content_scheduler*, *announcements*, *screens_groups*.
- Το πεδίο *id* του πίνακα *contents*, συσχετίζεται με το πεδίο *content_id* των πινάκων *screens*, *content_scheduler*, *announcements*.
- Το πεδίο *id* του πίνακα *groups*, συσχετίζεται με το πεδίο *group_id* των πινάκων *users_privileges*, *screens_groups*.

4.4 Λειτουργίες ιστοτόπου

Στην ενότητα αυτή θα αναλυθεί ποιες δυνατότητες δίνονται στο χρήστη για την ανάπτυξη του περιεχομένου και τη διαχείριση των οθονών. Για την οποιαδήποτε ενέργεια του χρήστη απαιτείται να έχει συνδεθεί στο σύστημα. Αλλιώς δεν μπορεί να επιδράσει με κανένα τρόπο.

4.4.1 Η αρχική σελίδα

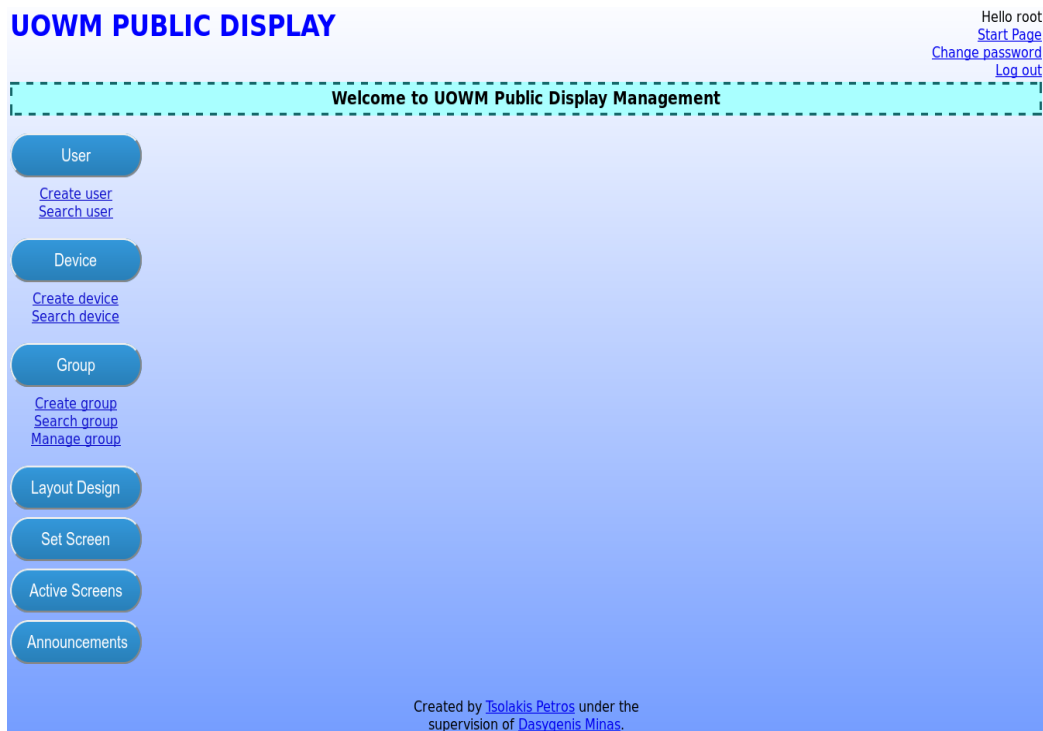
Το σημείο αυτό είναι η αρχή του ιστοτόπου. Ο χρήστης δεν μπορεί να προχωρήσει παρακάτω χωρίς να συνδεθεί. Απαιτείται να έχει ένα έγκυρο όνομα χρήστη και το σωστό συνθηματικό. Σε περίπτωση λάθους στοιχείων, του επιστρέφεται ένα μήνυμα για να του το επισημάνει. Στην παρακάτω εικόνα είναι η αρχική σελίδα.



Σχήμα 4.3: Η σελίδα σύνδεσης

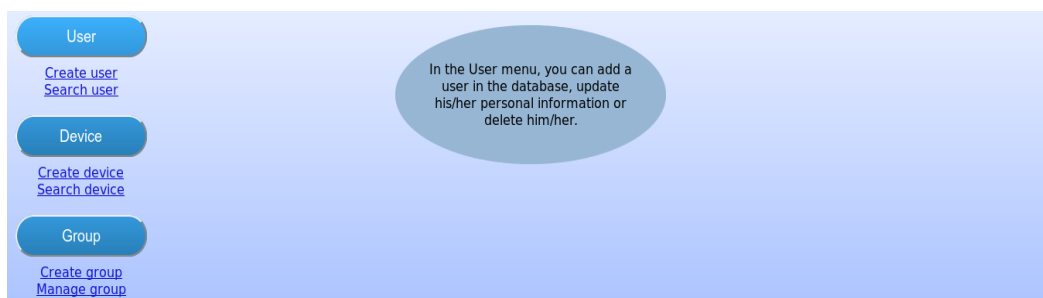
4.4.2 Η κεντρική σελίδα ελέγχου

Αφού συνδεθεί επιτυχώς, ο χρήστης, αναλόγως αν είναι ο υπερχρήστης ή κάποιος άλλος χρήστης, οδηγείται και στην αντίστοιχη κεντρική σελίδα ελέγχου. Όπως επισημάνθηκε σε προηγούμενη ενότητα, ο υπερχρήστης έχει παραπάνω δικαιώματα και λειτουργίες από τους υπόλοιπους χρήστες. Στην παρακάτω εικόνα φαίνεται η κεντρική σελίδα ελέγχου του υπερχρήστη.



Σχήμα 4.4: Η κεντρική σελίδα ελέγχου του υπερχρήστη

Στα αριστερά φαίνονται οι επιλογές του διαχειριστή ενώ, πάνω και δεξιά είναι οι επιλογές που δεν αφορούν άμεσα το σύστημα διαχείρισης οθονών. Κάθε φορά που ο κέρσορας του ποντικιού (mouse) περνά πάνω από κάποιο στοιχείο της αριστερής στήλης, εμφανίζονται επιπλέον πληροφορίες για αυτό. Στην παρακάτω εικόνα, φαίνονται οι πληροφορίες για το χρήστη.



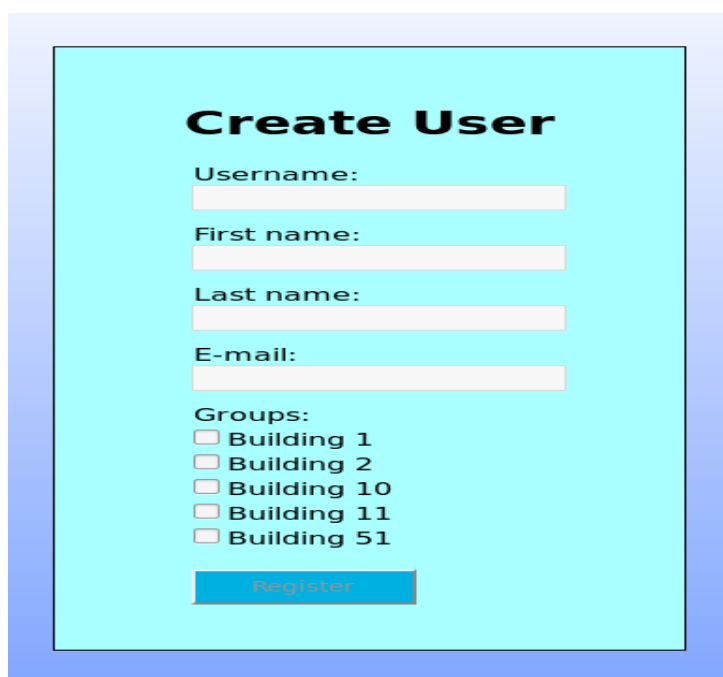
Σχήμα 4.5: Οι επιπλέον πληροφορίες του μενού User

Αρχικά θα αναφερθούν οι λειτουργίες του υπερχρήστη και μετά οι υπόλοιπες που είναι κοινές για όλους τους χρήστες του συστήματος.

4.4.3 Λειτουργίες υπερχρήστη

Οι λειτουργίες του υπερχρήστη αφορούν τη δημιουργία και διαχείριση των χρηστών, των οθονών και των ομάδων διαχείρισης.

Η δημιουργία νέων λογαριασμών χρηστών είναι ένα προνόμιο του υπερχρήστη. Γιατί πρόκειται για ένα σύστημα πραγματικού κόσμου, θεωρούμε ότι υπάρχει ήδη μια επαφή μεταξύ του υπερχρήστη και του υποψήφιου νέου χρήστη για την ανταλλαγή των βασικών στοιχείων. Τα στοιχεία που θα εισαχθούν, συμβαδίζουν με τον τύπο δεδομένων της βάσης δεδομένων όπως εξηγήθηκε σε προηγούμενη ενότητα. Για τη δημιουργία ενός χρήστη απαιτείται ένα έγκυρο και μοναδικό όνομα χρήστη, το μικρό του όνομα, το επώνυμο και η ηλεκτρονική διεύθυνση ταχυδρομείου. Προαιρετικά, ο υπερχρηστής μπορεί να προσθέσει τον χρήστη σε μια ή περισσότερες ομάδες διαχείρισης. Στην παρακάτω εικόνα βλέπουμε τη φόρμα εγγραφής χρήστη.



Create User

Username:

First name:

Last name:

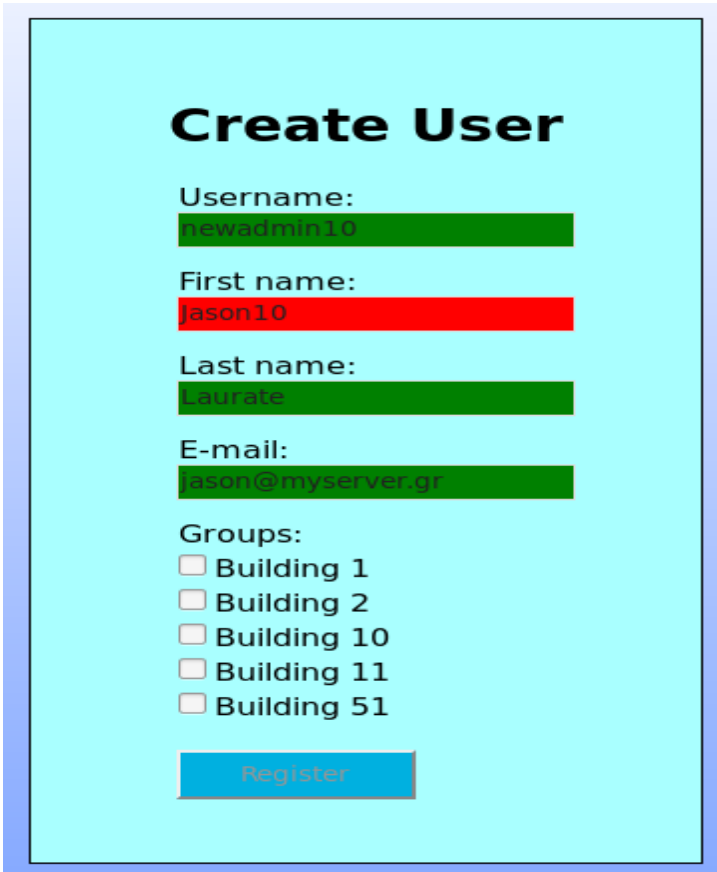
E-mail:

Groups:

- Building 1
- Building 2
- Building 10
- Building 11
- Building 51

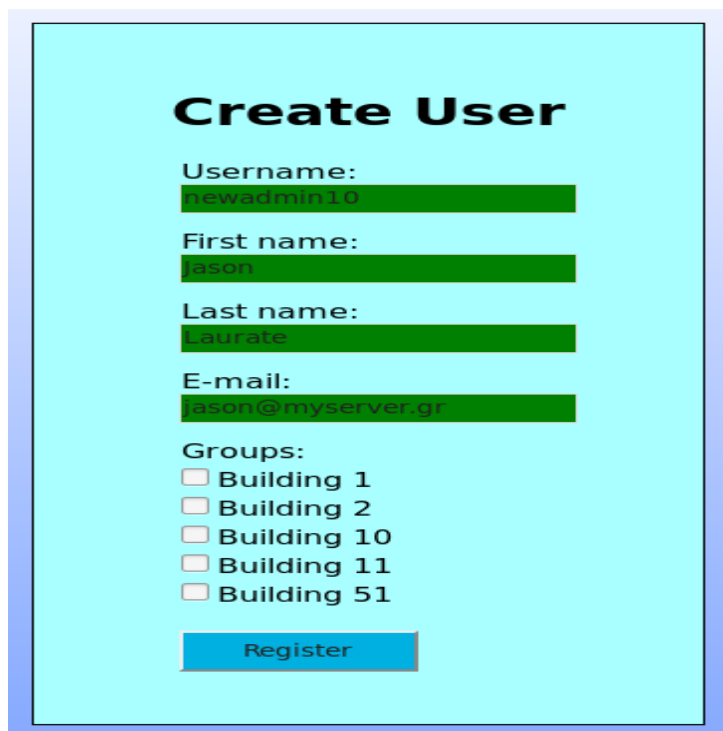
Σχήμα 4.6: Η φόρμα εγγραφής χρήστη

Όπως φαίνεται το κουμπί εγγραφής είναι απενεργοποιημένο. Για να ενεργοποιηθεί, πρέπει όλα τα παραπάνω πεδία να έχουν μια έγκυρη τιμή. Τα πεδία με έγκυρη τιμή έχουν πράσινο φόντο και αυτά με μη έγκυρη τιμή έχουν κόκκινο, όπως φαίνεται και στις παρακάτω εικόνες.



The image shows a web form titled "Create User" on a light blue background. The form contains several input fields and a list of checkboxes. The "Username:" field contains "newadmin10" and has a green background. The "First name:" field contains "jason10" and has a red background. The "Last name:" field contains "Laurate" and has a green background. The "E-mail:" field contains "jason@myserver.gr" and has a green background. Below these fields is a "Groups:" section with five checkboxes: "Building 1", "Building 2", "Building 10", "Building 11", and "Building 51", all of which are unchecked. At the bottom of the form is a blue "Register" button.

Σχήμα 4.7: Λανθασμένη εισαγωγή τιμής



Create User

Username:
newadmin10

First name:
jason

Last name:
Laurate

E-mail:
jason@myserver.gr

Groups:

- Building 1
- Building 2
- Building 10
- Building 11
- Building 51

Register

Σχήμα 4.8: Έγκυρες τιμές εισαγωγής

Ο έλεγχος των δεδομένων γίνεται με τη χρήση της JavaScript. Αυτό δεν εξασφαλίζει ότι τα δεδομένα είναι απόλυτα συμβατά για την εισαγωγή στο σύστημα. Το πιο συνηθισμένο σφάλμα είναι η ύπαρξη του επιλεγμένου ονόματος χρήστη ή ηλεκτρονικής διεύθυνσης ταχυδρομείου. Έτσι, η τελική απόφαση γίνεται πάντα από το διακομιστή κάνοντας χρήση της γλώσσας PHP. Αφού ολοκληρωθεί η εγγραφή και ο χρήστης καταχωρηθεί στη βάση δεδομένων, ο υπερχρήστης ειδοποιείται κατάλληλα και ένα ηλεκτρονικό μήνυμα στέλνεται στον καινούριο χρήστη. Σε περίπτωση σφάλματος, η εγγραφή χρήστη αποτυγχάνει και ο υπερχρήστης ενημερώνεται με το κατάλληλο μήνυμα.



Create User

Username:
[text input]

First name:
[text input]

Last name:
[text input]

E-mail:
[text input]

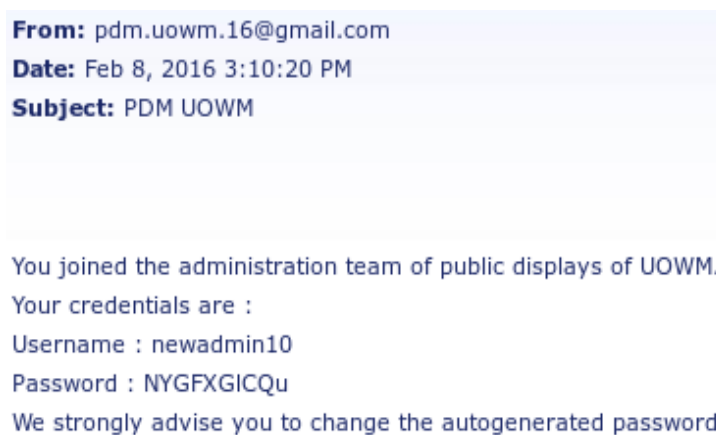
Groups:

- Building 1
- Building 2
- Building 10
- Building 11
- Building 51

[Register](#)

New user registered. An e-mail have sent to user.

Σχήμα 4.9: Επιτυχής εγγραφή χρήστη



From: pdm.uowm.16@gmail.com
Date: Feb 8, 2016 3:10:20 PM
Subject: PDM UOWM

You joined the administration team of public displays of UOWM.
Your credentials are :
Username : newadmin10
Password : NYGFXGICQu
We strongly advise you to change the autogenerated password

Σχήμα 4.10: Μήνυμα στο νέο χρήστη

Ο κωδικός που στέλνεται στο χρήστη παράγεται αυτόματα και αποτελείται από μια τυχαία συμβολοσειρά αριθμών, πεζών και κεφαλαίων λατινικών γραμμάτων, μεγέθους 10 χαρακτήρων.

Επιπλέον, ο υπερχρήστης μπορεί να αναζητήσει τους χρήστες στη βάση δεδομένων, να δει τα στοιχεία τους και να τα επεξεργαστεί ή να τους διαγράψει. Η

αναζήτηση των χρηστών γίνεται με βάση το όνομα χρήστη, το μικρό όνομα ή το επώνυμο. Στην εικόνα φαίνεται η αναζήτηση με το όρο "ste".

Username	First Name	Last Name	E-mail	Groups	Modify
admin2	Stefanos	Tatsis	stef.tatsis@lalala.com	Building 10 Building 1	<input type="button" value="Press"/>
admin22	Stephen	King	fantasy@book1234.com	Building 10	<input type="button" value="Press"/>

Σχήμα 4.11: Αναζήτηση χρήστη

Σε αυτό το σημείο έχει δύο επιλογές: είτε να πατήσει σε μια από τις ομάδες που ανήκει ο χρήστης για να δει και τα υπόλοιπα μέλη της, είτε να πατήσει το κουμπί "Press" και να επεξεργαστεί τα στοιχεία του χρήστη. Στη δεύτερη περίπτωση θα εμφανιστεί μια φόρμα παρόμοια με την φόρμα δημιουργίας χρήστη, όπου θα του επιτρέψει ή να ενημερώσει τα στοιχεία του ή να τον διαγράψει. Στην περίπτωση της ενημέρωσης των στοιχείων, ισχύουν οι κανόνες και οι περιορισμοί που τέθηκαν στη δημιουργία χρήστη.

Ένα ακόμη προνόμιο του υπερχρήστη, είναι η προσθήκη οθονών στο σύστημα. Η διαδικασία είναι παρόμοια με αυτή της δημιουργίας χρήστη. Ο υπερχρήστης εισάγει ένα όνομα για την οθόνη και μια σύντομη περιγραφή και προαιρετικά επιλέγει σε ποιες ομάδες διαχείρισης ανήκει. Ο έλεγχος των δεδομένων γίνεται με την JavaScript και αφού ελεγχθούν στέλνονται για επιβεβαίωση στο διακομιστή, όπως και στη δημιουργία χρήστη. Βασική προϋπόθεση το όνομα της οθόνης να είναι μοναδικό και η περιγραφή της να μην ξεπερνά τους 300 χαρακτήρες. Σε διαφορετική περίπτωση επιστρέφεται μήνυμα λάθους και δεν εισάγεται η οθόνη στη βάση δεδομένων. Παρακάτω φαίνεται μια επιτυχή δημιουργία νέας οθόνης.

Σχήμα 4.12: Δημιουργία οθόνης

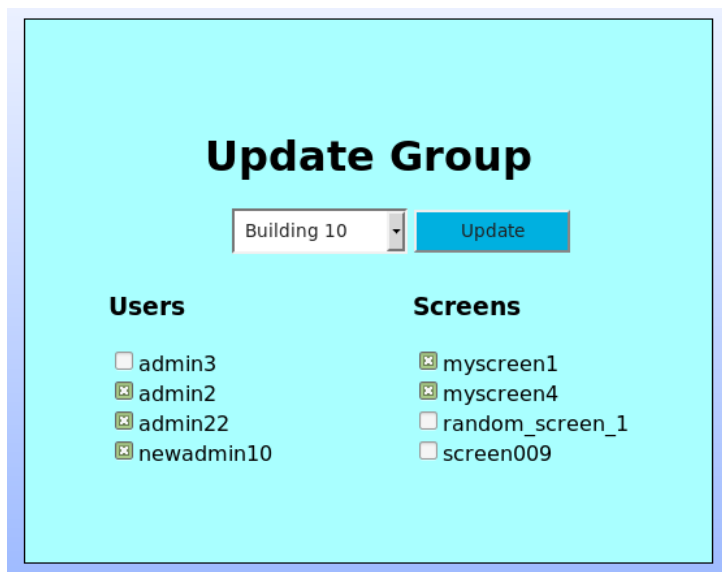
Όπως και για τους χρήστες, έτσι και για τις οθόνες ο υπερχρήστης μπορεί να αναζητήσει για αυτές ψάχνοντας στο όνομα τους ή στην περιγραφή τους. Στα αποτελέσματα που του επιστρέφονται, μπορεί να δει το όνομα, την περιγραφή και σε ποιες ομάδες ανήκει μια οθόνη. Πατώντας το κουμπί "Press" μπορεί να ενημερώσει αυτά τα στοιχεία ή να διαγράψει μια οθόνη. Πατώντας σε ένα από τα ονόματα των ομάδων διαχείρισης μπορεί να επεξεργαστεί τα άτομα και τις οθόνες αυτής της ομάδας. Στην παρακάτω εικόνα φαίνεται το αποτέλεσμα της αναζήτησης με τον όρο "test".

Screen Name	Description	Groups	Modify
myscreen1	a test xcxcxc	Building 10	Press
myscreen2	another test screen	Building 51	Press
testscreen_10	A random description....		Press

Σχήμα 4.13: Αναζήτηση οθόνης

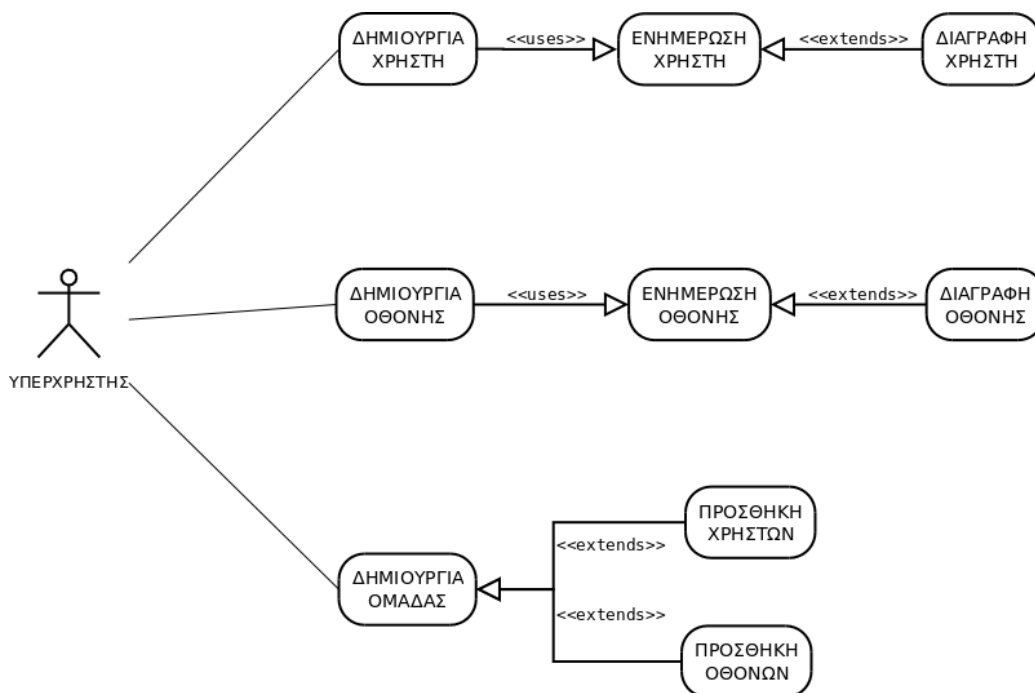
Τέλος, το τελευταίο προνόμιο του υπερχρήστη είναι η δημιουργία και ο έλεγχος των ομάδων διαχείρισης. Για τη δημιουργία μιας νέας ομάδας, απαιτείται ένα

όνομα το οποίο πρέπει να είναι μοναδικό και μια περιγραφή της ομάδας. Μετά τη δημιουργία της ομάδας ο υπερχρήστης μπορεί να δει τα μέλη της και να προσθέσει ή να αφαιρέσει χρήστες και οθόνες. Σε αυτή τη σελίδα οδηγείται ο υπερχρήστης, όταν έρχεται από την αναζήτηση ενός χρήστη ή οθόνης και επιλέξει το όνομα της ομάδας διαχείρισης. Στην παρακάτω εικόνα βλέπουμε την ομάδα με το όνομα "Building 2" με τα μέλη της.



Σχήμα 4.14: Διαχείριση ομάδας

Συνοπτικά, οι λειτουργίες του υπερχρήστη φαίνονται στο παρακάτω διάγραμμα.



Σχήμα 4.15: Διάγραμμα use cases του υπερχρήστη

4.4.4 Λειτουργίες χρηστών

Όλες οι υπόλοιπες λειτουργίες, είναι διαθέσιμες για όλους τους χρήστες.

Αρχικά, κάθε χρήστης μπορεί να δει μέχρι πότε είναι ενεργές και ποιο περιεχόμενο προβάλλουν εκείνη τη στιγμή οι οθόνες, οι οποίες ανήκουν στις ίδιες ομάδες με αυτόν. Ο υπερχρήστης μπορεί να δει όλες τις ενεργές οθόνες. Στην παρακάτω εικόνα, φαίνονται οι ενεργές οθόνες που διαχειρίζεται ένας χρήστης.

Screen	Description	Content	Expire on
random_screen_1	lal ao lal al oo ao pap oap p	music	Tuesday 23 February 2016 17:47 EET
screen009	This is a screen for this and that purpose	text	Tuesday 23 February 2016 17:49 EET

Σχήμα 4.16: Ενεργές οθόνες

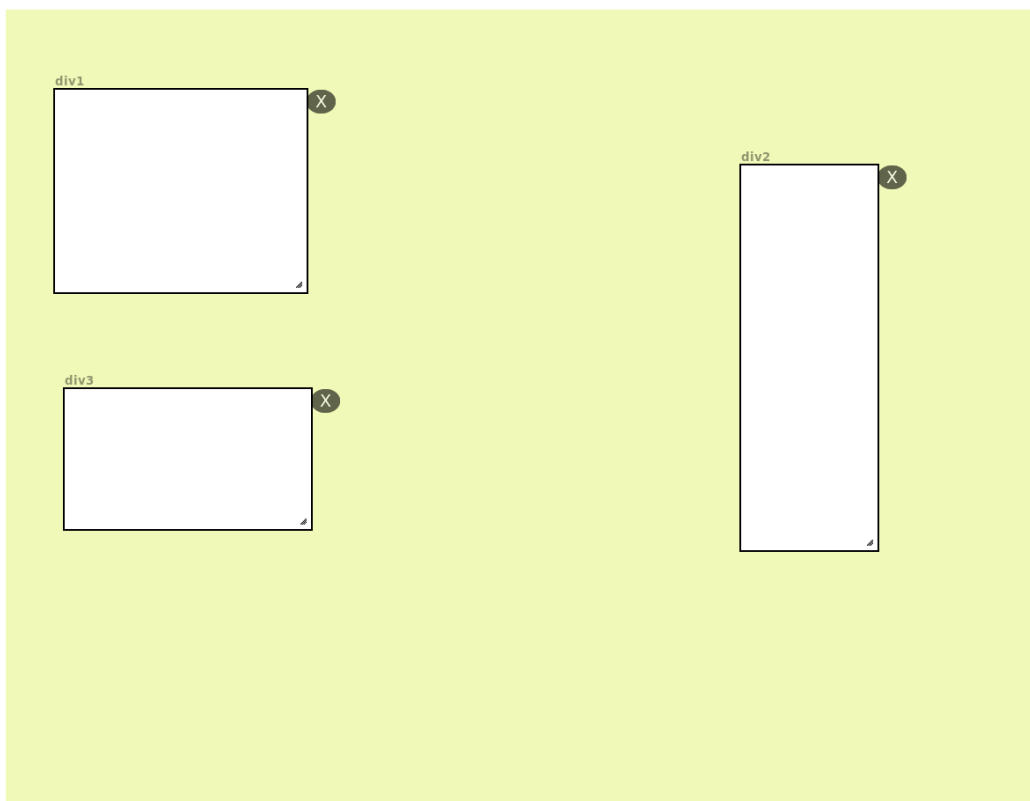
Έπειτα, ο χρήστης μπορεί να επιθεωρήσει και να αλλάξει τα έκτακτα περιεχόμενα. Όπως αναφέρθηκε προηγουμένως, έκτακτα είναι τα περιεχόμενα που προβάλλονται την κατάλληλη χρονική στιγμή παρακάμπτοντας την κανονική σειρά εμφάνισης. Αν επιλέξει μια ομάδα ή μια οθόνη, μπορεί να διαλέξει το περιεχόμενο και την περίοδο εμφάνισής του. Ο υπερχρήστης μπορεί να δει όλα τα έκτακτα περιεχόμενα όλων των οθονών. Στην παρακάτω εικόνα, φαίνονται τα έκτακτα περιεχόμενα που έχουν ορισθεί για τις οθόνες ενός χρήστη.

Group	Screen	Content	From	To
Building 1	myscreen3			
	random_screen_1	video	29/01/2016 08:22	29/01/2016 09:00
	screen009			
Building 10	myscreen1	music	29/01/2016 09:22	29/01/2016 09:36
	myscreen4	music	29/01/2016 09:22	29/01/2016 09:36

Σχήμα 4.17: Έκτακτα περιεχόμενα

Η σημαντικότερη λειτουργία είναι η δημιουργία των περιεχομένων. Κάθε χρήστης μπορεί να φτιάξει τις δικές του διατάξεις και μετά να προσθέσει σε αυτές τα δεδομένα που επιθυμεί και έτσι να δημιουργήσει ένα περιεχόμενο.

Για την δημιουργία μιας διάταξης, ο χρήστης σέρνει το ποντίκι του πάνω στην επιφάνεια σχεδίασης και φτιάχνεται ένα <div>, δηλαδή μια περιοχή στην οποία μπορεί να προσθέσει δεδομένα. Η περιοχή μετακινείται και αυξομειώνεται καθώς και διαγράφεται. Η τεχνολογία που χρησιμοποιήθηκε, βασίζεται στο πρόσθετο της jQuery, το jQuery UI που δίνει αυτές τις ιδιότητες στα δημιουργηθέντα στοιχεία. Στην παρακάτω εικόνα, φαίνεται η επιφάνεια σχεδίασης με κάποια δημιουργηθέντα στοιχεία.



Σχήμα 4.18: Η περιοχή σχεδίασης διατάξεων

Αφού το αποθηκεύσει, μπορεί να δει μια μικρογραφία του και να φορτώσει την αντίστοιχη διάταξη. Κάθε χρήστης έχει πρόσβαση μόνο στις διατάξεις που έχει φτιάξει ο ίδιος, εκτός από τον υπερχρήστη που μπορεί να τις χρησιμοποιήσει όλες. Τα αποθηκευμένα `<div>`, είναι ρυθμισμένα με τέτοιο τρόπο ώστε, σε οποιαδήποτε ανάλυση είναι η οθόνη του χρήστη, αυτά να τοποθετηθούν ανάλογα. Το ίδιο ισχύει και για το περιεχόμενο που θα προβάλλεται σε μια ενεργή οθόνη. Στην παρακάτω εικόνα, φαίνονται κάποιες αποθηκευμένες διατάξεις ενός χρήστη.



Σχήμα 4.19: Οι αποθηκευμένες διατάξεις

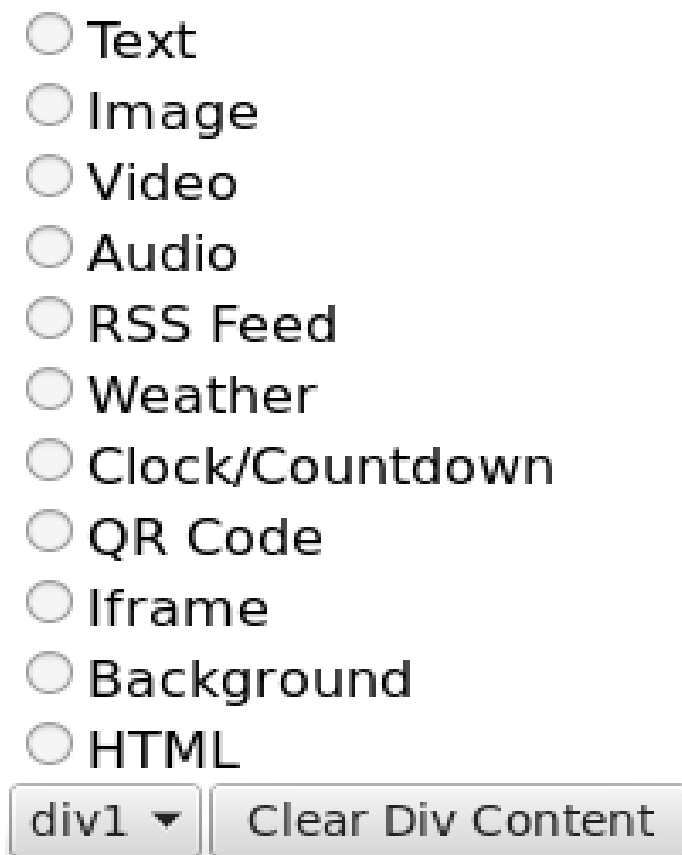
Αφού έχει τελειώσει με τη σχεδίαση και την αποθήκευση της διάταξης, ο χρήστης περνάει στο επόμενο στάδιο που είναι η τοποθέτηση του περιεχομένου. Οι επιλογές δεδομένων που του δίνονται για κάθε `<div>` είναι:

- Κείμενο. Μπορεί να εισάγει κείμενο και να το μορφοποιήσει κατάλληλα χρησιμοποιώντας τον CKEditor [33]. Επιπλέον μπορεί να κάνει το κείμενο του να μετακινείται από τη δεξιά πλευρά του `<div>` στην αριστερή.
- Εικόνα. Μπορεί να εισάγει μια ή περισσότερες εικόνες οι οποίες εναλλάσσονται σε χρόνο ορισμένο από τον ίδιο.
- Βίντεο. Μπορεί να εισάγει έναν ή περισσότερους συνδέσμους από τον ιστότοπο <https://www.youtube.com/> ή ένα αρχείο κειμένου που περιλαμβάνει συνδέσμους σε βίντεο αρχεία. Οι υποστηριζόμενοι τύποι αρχείων βίντεο είναι `.mp4`, `.webm` και `.ogg`. Πριν την έναρξη της αναπαραγωγής, μπορεί να υπάρχει μια χρονική καθυστέρηση. Η προβολή του βίντεο γίνεται μέσω της HTML5 χωρίς τη χρήση κάποιου flash player. Το βίντεο θα αναπαραχθεί

μόνο κατά την προβολή στην οθόνη και όχι κατά την σχεδίαση του περιεχόμενου.

- **Ήχος.** Μπορεί να εισάγει ένα αρχείο κειμένου με συνδέσμους σε αρχεία ήχου. Τα υποστηριζόμενα αρχεία είναι .mp3, .wav και .ogg. Η αναπαραγωγή τους γίνεται, όπως και στο βίντεο, μέσω της HTML5. Επίσης, μπορεί να υπάρχει μια αρχική καθυστέρηση πριν την έναρξη της αναπαραγωγής. Ο ήχος θα αναπαραχθεί μόνο κατά την προβολή στην οθόνη και όχι κατά την σχεδίαση του περιεχομένου.
- **Τροφοδοσία RSS (RSS feed).** Μπορεί να εισάγει ένα σύνδεσμο RSS και αυτός να ανανεώνεται κάθε φορά που υπάρχει ένα νέο αντικείμενο. Η διαδικασία αυτή γίνεται χρησιμοποιώντας τη βιβλιοθήκη SimplePie [34] που είναι γραμμένη σε PHP. Ο χρήστης μπορεί να επιλέξει πόσα από τα πιο πρόσφατα αντικείμενα, θέλει να εμφανίζονται. Η τροφοδοσία RSS φαίνεται μόνο κατά την προβολή στην οθόνη όπου το κείμενο μετακινείται αυτόματα για να μπορούν να αναγνωστούν όλα τα αντικείμενα.
- **Καιρός.** Εισάγει ένα πλαίσιο από τον ιστότοπο <http://www.deltiokairou.gr/>, που δείχνει τον καιρό σε διάφορες πόλεις της Ελλάδος.
- **Ρολόι και αντίστροφη χρονομέτρηση.** Μπορεί να εμφανίσει ένα εικοσιτετράωρο ρολόι ή να ορίσει μια αντίστροφη χρονομέτρηση για μια χρονική στιγμή στο μέλλον. Για την εμφάνιση των ρολογιών χρησιμοποιήθηκε η βιβλιοθήκη FlipClock.js [35] της jQuery.
- **Κώδικας QR.** Εισάγει ένα κώδικα QR που μπορεί να ρυθμιστεί ώστε όταν σαρωθεί να προβάλλει ένα κείμενο, συνήθως ένα σύνδεσμο, είτε να οδηγήσει σε μια σελίδα ώστε να επιλεγεί ένα άλλο περιεχόμενο για την οθόνη. Για τη δημιουργία των κωδίκων QR, χρησιμοποιήθηκε το πρόσθετο της jQuery `jquery.qrcode.js` [36].
- **Iframe.** Εισάγει ένα παράθυρο το οποίο εμφανίζει, έναν εξωτερικό σύνδεσμο μιας άλλης σελίδας.
- **Παρασκήνιο.** Εδώ μπορεί να ρυθμιστεί το χρώμα του παρασκήνιου ή να τοποθετηθεί μια εικόνα που είτε θα καλύπτει όλο το παρασκήνιο είτε θα επαναλαμβάνεται.
- **Κώδικας HTML.** Αν ο χρήστης έχει γνώσεις HTML, μπορεί να γράψει κώδικα σε αυτήν και να τον προσθέσει σε ένα `<div>`.

Συνοπτικά, οι επιλογές δεδομένων φαίνονται στην παρακάτω εικόνα:



Text

Image

Video

Audio

RSS Feed

Weather

Clock/Countdown

QR Code

Iframe

Background

HTML

div1 ▼ Clear Div Content

Σχήμα 4.20: Επιλογές δεδομένων περιεχομένου

Ακόμη, υπάρχουν κάποιες ρυθμίσεις σχετικά με την εμφάνιση των στοιχείων `<div>`, οι οποίες ρυθμίζονται μέσω της CSS και της JavaScript:

- Ρύθμιση ύψους (z-index). Όλα τα στοιχεία έχουν z-index 1 που σημαίνει, ότι βρίσκονται στο ίδιο επίπεδο στο δισδιάστατο χώρο. Αν ο δείκτης αυτός αυξηθεί ή μειωθεί τότε το εκάστοτε στοιχείο θα ανέβει ή θα κατέβει ένα επίπεδο, αντίστοιχα.
- Αδιαφάνεια. Όλα τα στοιχεία έχουν αδιαφάνεια ίση με 1 που είναι η μέγιστη τιμή. Όσο η τιμή αυτή πλησιάζει στο 0, τόσο το αντικείμενο αυτό γίνεται λιγότερο ορατό μέχρι να γίνει τελείως διάφανο.

- Χρώμα παρασκηνίου. Ο χρήστης μπορεί να επιλέξει το χρώμα παρασκηνίου του στοιχείου.
- Πλαίσιο. Εξ ορισμού, κάθε στοιχείο έχει ένα πλαίσιο το οποίο μπορεί να αφαιρεθεί.
- Χρονοπρογραμματισμός στοιχείων. Εξ ορισμού, όλα τα στοιχεία του περιεχομένου εμφανίζονται καθ' όλη τη διάρκεια που η οθόνη είναι ενεργή και προβάλλεται αυτό το περιεχόμενο. Αυτή η συμπεριφορά μπορεί να αλλάξει και ο χρήστης να ορίσει, πότε θέλει ένα στοιχείο να είναι ορατό ή όχι, επιλέγοντας τις μέρες σε μια χρονική περίοδο που το συγκεκριμένο στοιχείο εμφανίζεται ή εξαφανίζεται. Για την καλύτερη διαχείριση αυτής της δυνατότητας, προστέθηκε μια επιλογή που εξαφανίζει προσωρινά το εκάστοτε <div> και αποδεσμεύει το χώρο, ώστε να τοποθετηθεί εκεί ένα άλλο <div>.

Οι ρυθμίσεις για το κάθε <div> φαίνονται στην παρακάτω εικόνα:

div2 ▾ Fade Div Fade Border Rename Div

z-index (default 1) 1

opacity (default 1) 1



background color

DIV SCHEDULER



DATE	HOUR	DAYS
from	from	Monday
<input type="text" value="01/03/2016"/>	<input type="text" value="11:00"/>	Tuesday
to	to	Wednesday
<input type="text" value="07/03/2016"/>	<input type="text" value="11:00"/>	Thursday
		Friday
		Saturday
		Sunday

Show Hide

div1 show 27/02/2016 29/02/2016 08:30 14:00

mon,tue,wed,thu,fri  

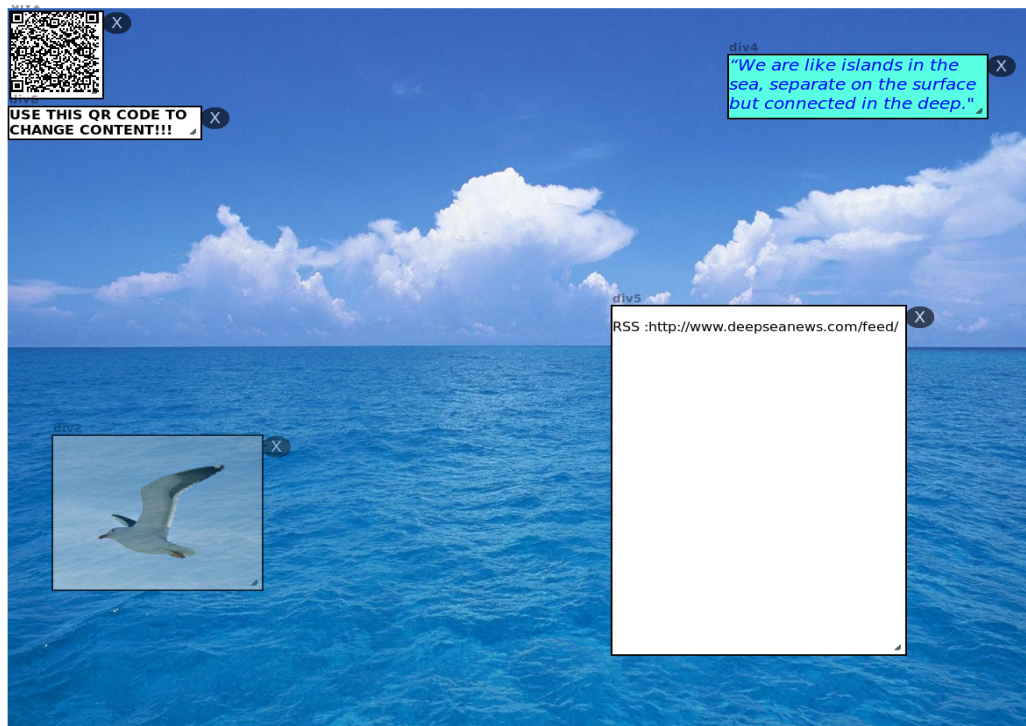
div2 hide 01/03/2016 07/03/2016 11:00 11:00

mon,tue,wed,thu,fri  

Σχήμα 4.21: Ρυθμίσεις στοιχείων <div>

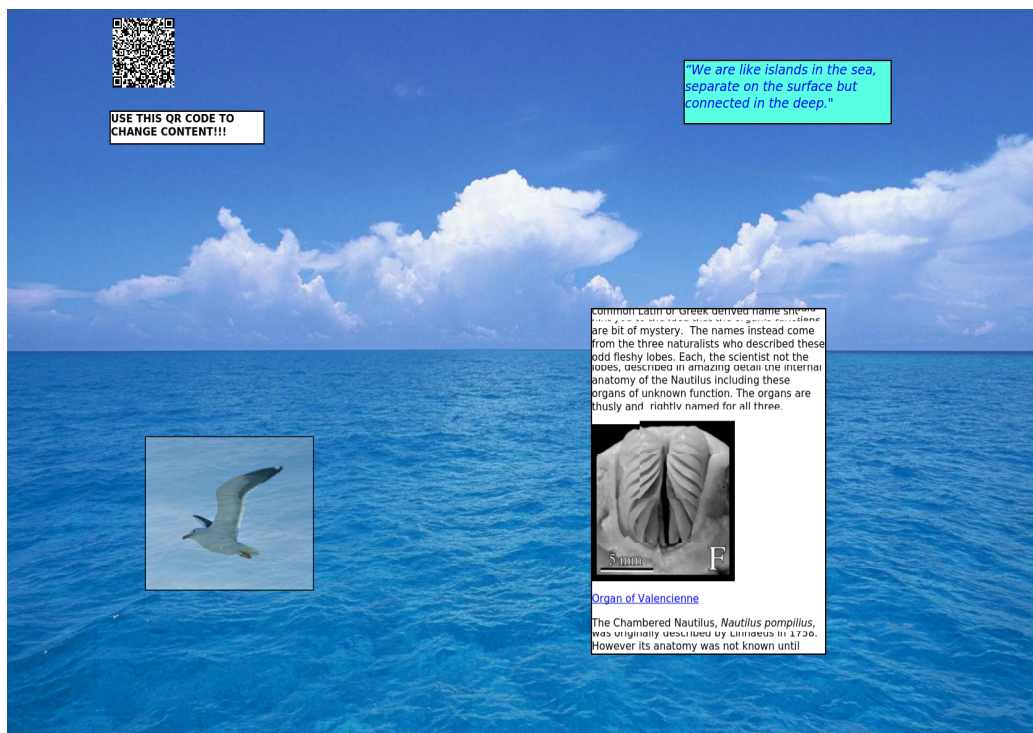
Αφού, τελειώσει με την εισαγωγή του περιεχομένου και τις ρυθμίσεις των <div>, ο χρήστης μπορεί να αποθηκεύσει το περιεχόμενο ή να ενημερώσει ένα

παλιό. Κάθε χρήστης έχει πρόσβαση μόνο στα δικά του περιεχόμενα, εκτός από τον υπερχρήστη που έχει πρόσβαση σε όλα. Στην παρακάτω εικόνα, φαίνεται ένα περιεχόμενο όπως εμφανίζεται στην επιφάνεια σχεδίασης.



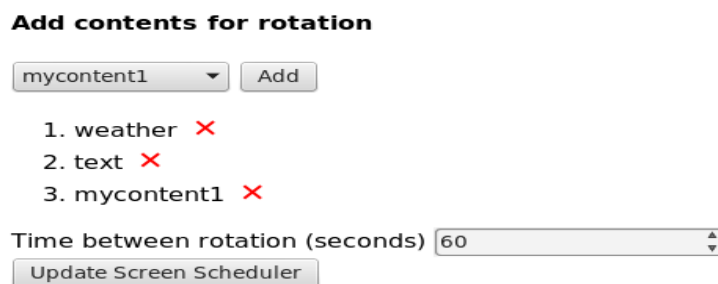
Σχήμα 4.22: Σχεδίαση περιεχομένου

Εφόσον, έχει ολοκληρωθεί η αποθήκευση του περιεχομένου, ο χρήστης μπορεί να το προωθήσει σε μια οθόνη ή σε μια ομάδα οθονών. Το περιεχόμενο θα προωθηθεί αμέσως και η οθόνη θα ανανεωθεί. Στην παρακάτω εικόνα, βλέπουμε το περιεχόμενο που σχεδιάστηκε στην εικόνα 4.22 πως προβάλλεται σε μια οθόνη.



Σχήμα 4.23: Προβαλλόμενο περιεχόμενο σε οθόνη

Η τελευταία δυνατότητα στο περιβάλλον σχεδίασης, είναι η αυτόματη εναλλαγή των περιεχομένων. Ο χρήστης μπορεί να επιλέξει τα περιεχόμενα που θέλει, να τα ορίσει σε μια σειρά εμφάνισης και να θέσει ένα χρονικό όριο που το καθένα θα εμφανίζεται. Έτσι η οθόνη αλλάζει περιεχόμενα αυτόματα και υπάρχει ένα δυναμικό περιβάλλον. Για αυτήν την επιλογή, προτείνεται η χρησιμοποίηση του κώδικα QR για αλλαγή περιεχομένου από το χρήστη. Στην παρακάτω εικόνα φαίνεται, το μενού διαχείρισης εναλλαγής περιεχομένων.



Σχήμα 4.24: Μενού εναλλαγής περιεχομένων

Να σημειωθεί πως, όπου χρειάστηκε η εισαγωγή μια ώρας από το χρήστη χρησιμοποιήθηκε το πρόσθετο Timericker [37] και όπου χρειάστηκε η επιλογή χρώματος, χρησιμοποιήθηκε το πρόσθετο colorpick [38], και τα δύο γραμμένα στην jQuery.

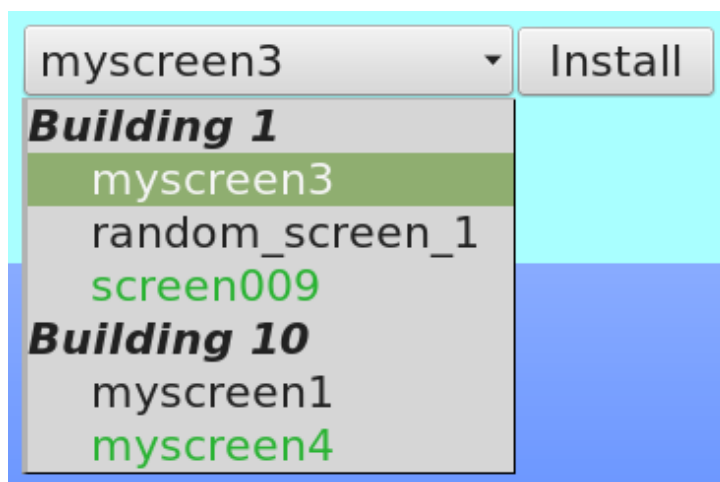
Η τελευταία λειτουργία του ιστότοπου είναι η ενεργοποίηση της οθόνης. Ο χρήστης μπορεί να επιλέξει μια από τις οθόνες που είναι ομαδοποιημένες, ως προς την ομάδα διαχείρισης. Μια οθόνη μπορεί να ενεργοποιηθεί παραπάνω από μια φορά παράλληλα. Αυτό σημαίνει ότι, πολλές οθόνες μπορούν να δείχνουν το ίδιο περιεχόμενο ταυτόχρονα. Οι ήδη ενεργοποιημένες οθόνες συμβολίζονται με πράσινη γραμματοσειρά. Ο κάθε χρήστης μπορεί να επιλέξει μόνο από τις οθόνες που είναι σε κάποια ομάδα διαχείρισης που ανήκει και ο ίδιος. Ο υπερχρήστης μπορεί να επιλέξει οποιαδήποτε οθόνη.

Όταν η οθόνη ενεργοποιηθεί, εγκαθίσταται ένα cookie σε αυτόν τον υπολογιστή με ένα μοναδικό αναγνωριστικό για κάθε οθόνη, που ξεκινά την επικοινωνία με τον διακομιστή, για την λήψη δεδομένων. Το cookie αυτό έχει διάρκεια τριών χρόνων και όταν πάψει να υφίσταται, η οθόνη δεν μπορεί να λάβει νέα δεδομένα αλλά θα συνεχίσει να προβάλλει το τελευταίο περιεχόμενο. Η τεχνολογία που χρησιμοποιείται για την επικοινωνία με τον διακομιστή ονομάζεται Server-Sent Events (SSE) [39]. Συνήθως, για την επικοινωνία μεταξύ του φυλλομετρητή και του διακομιστή, γίνεται ένα αίτημα από τον πρώτο στο δεύτερο και κάθε φορά που θέλει να ζητήσει νέα δεδομένα επαναλαμβάνεται η διαδικασία. Με την τεχνολογία SSE όμως, γίνεται ένα αίτημα από τον φυλλομετρητή στο διακομιστή αλλά, η σύνδεση δεν κλείνει και παραμένει ενεργή. Έτσι, κάθε φορά που υπάρχουν νέα δεδομένα αυτά μπορούν να προωθηθούν στον φυλλομετρητή. Ο έλεγχος για νέα δεδομένα γίνεται στο διακομιστή κατά το σταθερό χρονικό διάστημα των 8 δευτερολέπτων. Η τεχνολογία SSE λειτουργεί σε όλους τους γνωστούς φυλλομετρητές εκτός, του Internet Explorer. Έτσι, για την λειτουργία της οθόνης αποκλείεται η

χρήση του. Η διαδικασία παύει να εκτελείται, όταν το cookie που έχει οριστεί, καταργηθεί δηλαδή κατά την ημερομηνία λήξης του.

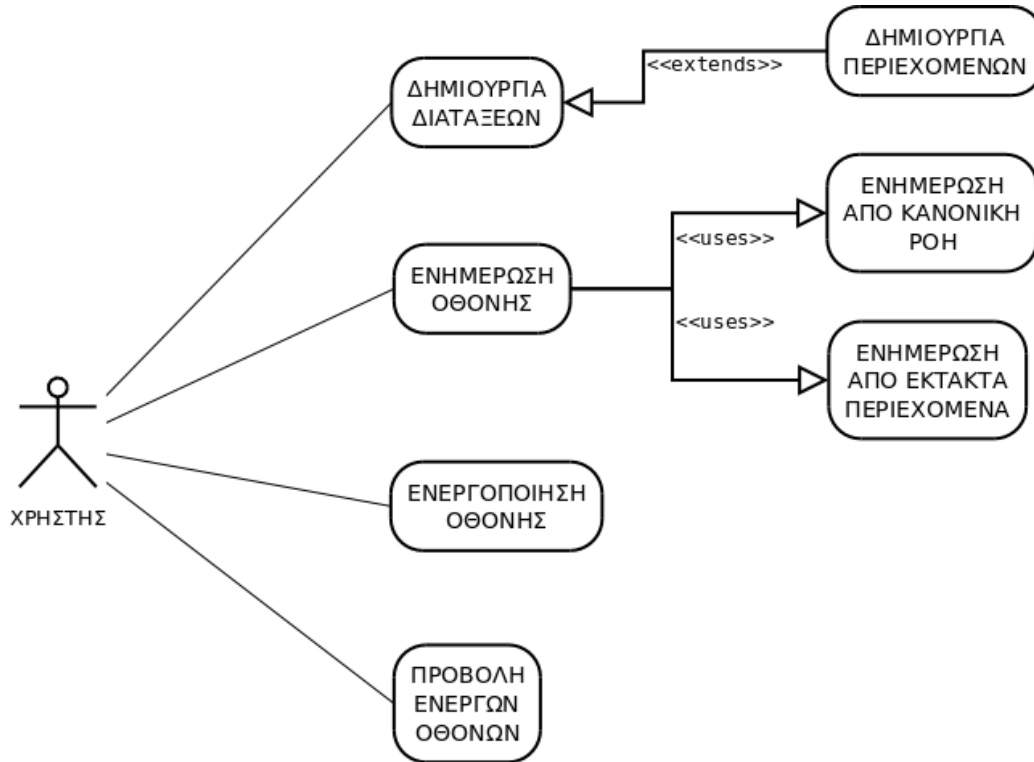
Για τη σωστή ενημέρωση του κάθε χρήστη, το σύστημα αναλαμβάνει κάθε μέρα να του στέλνει ένα μήνυμα ηλεκτρονικού ταχυδρομείου όπου αναγράφονται οι οθόνες οι οποίες είναι απενεργοποιημένες. Αυτό γίνεται μέσω του cron, ενός δαίμονα (daemon) ο οποίος αναλαμβάνει να εκτελέσει σε σταθερά χρονικά διαστήματα μια εργασία. Οι εργασίες αυτές γράφονται στο αρχείο crontab (cron table) [40].

Τέλος, μια οθόνη μπορεί να ενεργοποιηθεί χωρίς την επιλογή από την λίστα, αν συμπληρωθεί στο τέλος της διεύθυνσης της ιστοσελίδας display.php η συμβολοσειρά "?name=unique_id", αντικαθιστώντας το "unique_id" με το μοναδικό αναγνωριστικό κάθε οθόνης. Η επιλογή της οθόνης φαίνεται στην παρακάτω εικόνα.



Σχήμα 4.25: Επιλογή οθόνης

Συνοπτικά, οι λειτουργίες των χρηστών φαίνονται στο παρακάτω διάγραμμα.



Σχήμα 4.26: Διάγραμμα use cases των χρηστών

4.5 Ασφάλεια συστήματος

Η ασφάλεια ενός συστήματος είναι ένας σημαντικός τομέας και πρέπει να δοθεί η ανάλογη προσοχή. Τα μέτρα που πάρθηκαν στη συγκεκριμένη εργασία είναι:

- Η σύνδεση του χρήστη γίνεται με κωδικό. Ο κωδικός αυτός δεν μπορεί να είναι κενός και αποθηκεύεται σε κατακερματισμένη μορφή. Στην εγγραφή χρήστη, το σύστημα παράγει έναν κωδικό 10 χαρακτήρων και είναι στην ευχέρεια του χρήστη αν θέλει να τον αλλάξει.
- Καμιά ενέργεια δεν μπορεί να γίνει χωρίς σύνδεση του χρήστη. Κάθε φορά που υπάρχει επικοινωνία με το διακομιστή ελέγχεται αν ο χρήστης είναι συνδεδεμένος, ώστε να μπορέσει να διεκπεραιώσει την εκάστοτε ενέργεια.

- Ο χρήστης αποσυνδέεται αυτόματα μετά από 24 λεπτά χωρίς να πραγματοποιήσει κάποια ενέργεια.
- Οι λειτουργίες του υπερχρήστη είναι αποκλειστικές σε αυτόν και οποιοσδήποτε άλλος επιχειρήσει να πάρει πρόσβαση σε αυτές, ανακατευθύνεται.
- Τα δεδομένα που προέρχονται από εισαγωγή του χρήστη, φιλτράρονται πριν εισαχθούν στη βάση δεδομένων. Για αυτή την διαδικασία χρησιμοποιήθηκε η συνάρτηση `filter_var()` της PHP, η οποία "καθαρίζει" τα δεδομένα από ανεπιθύμητους χαρακτήρες. Επίσης, χρησιμοποιήθηκαν κανονικές εκφράσεις (regular expressions), ώστε τα δεδομένα να εισάγονται στην κατάλληλη μορφή για την αναγνώριση τους από το σύστημα.
- Για την σύνδεση και εκτέλεση ερωτημάτων στη βάση δεδομένων χρησιμοποιήθηκαν τα Αντικείμενα Δεδομένων PHP (PHP Data Objects, PDO). Τα αντικείμενα PDO υποστηρίζουν prepared statements [41] τα οποία είναι χρήσιμα για την αποφυγή επιθέσεων sql injection.

4.6 Ανάλυση κώδικα

Σε αυτήν την ενότητα, θα αναφερθούμε σε κάποια σημεία του κώδικα που χρήζουν ιδιαίτερη προσοχή και αναφέρθηκαν προηγουμένως.

4.6.1 Σύνδεση χρήστη

Αφού ο χρήστης πληκτρολογήσει τα στοιχεία του, ελέγχουμε στη βάση δεδομένων αν υπάρχει το όνομα του και αν ταιριάζει με το συνθηματικό του. Όπως είχαμε αναφέρει στην ενότητα της ασφάλειας, τα SQL ερωτήματα γίνονται με τη χρήση των PDO. Η επιβεβαίωση του συνθηματικού γίνεται με τη συνάρτηση `password_verify()` της PHP, που ελέγχει αν η κατακερματισμένη μορφή του συνθηματικού ταιριάζει με αυτή στη βάση δεδομένων. Αν ναι, ξεκινά ένα session και ανακατευθύνει το χρήστη, στην αρχική σελίδα του ιστοτόπου. Σε διαφορετική περίπτωση επιστρέφεται ένα μήνυμα λάθους στην σελίδα σύνδεσης.

```

4  $sql_query=$conn->prepare("SELECT password FROM users_information WHERE username=?");
5  $sql_query->bindParam(1,$user);
6  $sql_query->execute();
7  $result=$sql_query->fetch();
8
9  if (!empty($result['password']))
10 {
11     if (password_verify($pass,$result['password'])) {
12         session_start();
13         $_SESSION["admin"] = $user;
14
15         header('Location: control_panel.php');
16     } else {
17         $error = "Username or password is not correct.";
18     }
19 } else {
20     $error = "Username or password is not correct.";
21 }

```

Σχήμα 4.27: Σύνδεση του χρήστη στο σύστημα - Κώδικας

4.6.2 Φιλτράρισμα δεδομένων

Κάθε είσοδος που γίνεται από το χρήστη, πρέπει να φιλτράρετε ώστε να διαγραφτούν διάφοροι ανεπιθύμητοι χαρακτήρες. Αυτοί οι χαρακτήρες ονομάζονται χαρακτήρες ελέγχου (control characters) [42], δεν είναι εκτυπώσιμοι και χρησιμοποιούνται για τον έλεγχο διάφορων συσκευών. Ο καθαρισμός των εισόδων γίνεται με τη συνάρτηση `filter_var` της PHP, όπως φαίνεται στην παρακάτω εικόνα.

```

$screen=filter_var($_POST['scrname'],FILTER_SANITIZE_STRING,FILTER_FLAG_STRIP_LOW);
$descr=filter_var($_POST['description'],FILTER_SANITIZE_STRING,FILTER_FLAG_STRIP_LOW);

```

Σχήμα 4.28: Φιλτράρισμα δεδομένων - Κώδικας

4.6.3 Περιβάλλον σχεδίασης

Στο περιβάλλον σχεδίασης, ο χρήστης μπορεί να σχεδιάσει στοιχεία `<div>` κρατώντας κρατημένο το αριστερό κουμπί του ποντικιού και σέρνοντας το. Για τη δημιουργία του κάθε `<div>`, χρησιμοποιείται η βιβλιοθήκη jQuery UI, με την οποία μπορούμε να χειριστούμε τα συμβάντα αλλαγής κατάστασης (events), δηλαδή όταν ο χρήστης πατήσει το αριστερό κλικ και όταν το αφήσει, και να δημιουργήσουμε το ανάλογο στοιχείο. Σε αυτό το διάστημα καταγράφουμε την αρχική και τελική θέση του κέρσορα ώστε, να υπολογίσουμε το ύψος και το μήκος του. Τέλος, πρέπει να υπολογίσουμε την αρχική θέση, το μήκος και το πλάτος του στοιχείου σε ποσοστό επί της επιφάνειας για να μπορεί να προβληθεί σε κάθε

δυνατή ανάλυση. Στην παρακάτω εικόνα φαίνεται η τελική δημιουργία του στοιχείου `<div>`.

```
width = Math.abs(end_x - start_x);
height = Math.abs(end_y - start_y);

$(this).append('<div class="draw_div" id ="div' + i +' "></div>');

$(newdiv).css({
  "width" : (100* width / $("#draw_area").width())+'%',
  "height" : (100*height / ($("#draw_area").height())+'%',
  "left" : (100*start_x / ($("#draw_area").width())+'%',
  "top" : (100*start_y / ($("#draw_area").height())+'%',
  "background-color" : "#FFFFFF",
  "position" : "absolute",
  "border" : "1px solid",
  "z-index" : 1
});
```

Σχήμα 4.29: Σχεδίαση ενός στοιχείου `<div>` - Κώδικας

4.6.4 Λειτουργία οθόνης

Η λήψη των δεδομένων στην οθόνη γίνεται με τη χρήση των Server-Sent Events. Η οθόνη συνδέεται με το διακομιστή και λαμβάνει δεδομένα, κρατώντας ανοιχτή τη σύνδεση. Αφού συνδεθεί, η απάντηση από το διακομιστή έρχεται σε μορφή JSON. Οι πιθανές απαντήσεις που μπορεί να πάρει είναι:

1. Αν τα δεδομένα είναι σε μορφή HTML και CSS, να τα επικολλήσει στην ιστοσελίδα και να τα διαμορφώσει ανάλογα με τον τύπο τους.
2. Αν τα δεδομένα είναι κενά, αυτό δηλώνει πως δεν έχει οριστεί κάποιο περιεχόμενο για αυτήν την οθόνη ακόμη.
3. Αν τα δεδομένα είναι της μορφής "NOMOREDATA", αυτό σημαίνει πως έφτασε η ώρα της λήξης του cookie ενεργοποίησης της οθόνης και άρα κλείνει τη σύνδεση.
4. Αν τα δεδομένα είναι το "0", μην κάνει τίποτα γιατί είναι ένα αναγνωριστικό byte, για το αν παραμένει η σύνδεση ενεργή.

Στην εικόνα που ακολουθεί βλέπουμε την παραπάνω λειτουργία σε κώδικα jQuery. Πρέπει να σημειωθεί, ότι παραλείφθηκαν οι συναρτήσεις διαμόρφωσης του περιεχόμενου.

```
if(typeof(EventSource) !== "undefined") {  
    var source = new EventSource("update_screen.php");  
    source.onmessage = function(event) {  
        var data = JSON.parse(event.data);  
        console.log(data);  
        if (data == null) {  
            $("body").empty();  
        } else if (data.html == "NOMOREDATA") {  
            source.close();  
        } else if (data != "0") {  
            functions();  
        }  
    }  
}
```

Σχήμα 4.30: Η λειτουργία της οθόνης - Κώδικας

4.6.5 Ενημέρωση οθόνης

Αρχικά, ο διακομιστής ελέγχει αν υπάρχει έκτακτο περιεχόμενο για αυτήν την οθόνη ή όχι. Αν υπάρχει το προωθεί αλλιώς, συνεχίζει με την κανονική ροή του πίνακα *contents_scheduler* και προωθεί το ανάλογο περιεχόμενο. Η επιλογή του περιεχομένου γίνεται μέσω του μοναδικού *webid* κάθε οθόνης. Από το περιεχόμενο, επιλέγονται ο κώδικας HTML και οι πληροφορίες για το παρασκήνιο που είναι στη CSS. Κάθε φορά που επιλέγεται ένα περιεχόμενο, πριν σταλθεί ελέγχεται ότι είναι διαφορετικό από το τελευταίο που στάλθηκε. Σε διαφορετική περίπτωση δεν στέλνεται. Όλη αυτή η διαδικασία που περιγράφηκε, επαναλαμβάνεται κάθε 8 δευτερόλεπτα, μέσα σε ένα βρόγχο επανάληψης *while* που εκτελείται όσο η οθόνη παραμένει ενεργή.

Ο κώδικας PHP που περιγράφει το παραπάνω φαίνεται στην επόμενη εικόνα.

```

while (time() <= $valid_time) {
    choose_content();

    $sql_query=$conn->prepare("SELECT content_id, qrcode_id FROM screens WHERE webid = ?");
    $sql_query->bindParam(1,$id);
    $sql_query->execute();
    $screen=$sql_query->fetch();

    $sql_query=$conn->prepare("SELECT content_html, bgcolor, backimage_url, backimage_option FROM contents
    WHERE id=?");
    $sql_query->bindParam(1,$screen['content_id']);
    $sql_query->execute();
    $result=$sql_query->fetch();

    $content["html"] = $row['content_html'];
    $content["bg_color"] = $row['bgcolor'];
    $content["bg_img"] = $row['backimage_url'];
    $content["bg_opt"] = $row['backimage_option'];
    $content["qr"] = $screen['qrcode_id'];

    if ($content != $old_content || !(isset($old_content))) {
        echo 'data: '.json_encode($content)."\n\n";
        ob_end_flush();
        flush();

        $old_content = $content;
    }

    sleep(8);
}

```

Σχήμα 4.31: Ενημέρωση οθόνης - Κώδικας

4.7 Μετρικά στοιχεία κώδικα

Ο κώδικας αποτελείται από αρχεία της HTML, της CSS, της JavaScript και της PHP. Στον πίνακα που ακολουθεί, αναγράφονται μόνο τα στοιχεία που γράφτηκαν από εμάς και όχι από τα διάφορα πρόσθετα.

	Αριθμός αρχείων	Γραμμές	Μέγεθος (KB)
HTML	1	18	0.3
CSS	13	840	15
JavaScript	9	2592	97
PHP	40	3389	107

Πίνακας 4.10: Πίνακας μετρικών στοιχείων κώδικα

4.8 Δοκιμές

Αφού ολοκληρώθηκε η ανάπτυξη του κώδικα, έπρεπε να δοκιμάσουμε το σύστημα και σε πραγματικές συνθήκες. Καθ' όλη τη διάρκεια της ανάπτυξης, γινόταν συνεχείς έλεγχοι σε έναν επιτραπέζιο υπολογιστή. Ο υπολογιστής αυτός λειτουργούσε παράλληλα και ως διακομιστής και ως σταθμός βάσης για την οθόνη. Οτιδήποτε πρόβλημα προέκυπτε έπρεπε να αντιμετωπιστεί. Έτσι, όταν το σύνολο των προβλημάτων λύθηκε, έπρεπε να μεταβούμε και στο πραγματικό περιβάλλον. Σε αυτό μας εξυπηρέτησε το BeagleBoard-xM που λειτούργησε ως σταθμός βάσης και συνδέθηκε η οθόνη. Αρχικά εγκαταστήσαμε το λειτουργικό σύστημα Debian και το φυλλομετρητή Chromium [43] στο BeagleBoard-xM, ρυθμίσαμε τη σύνδεση στο λειτουργικό σύστημα χωρίς προτροπή και την αυτόματη έναρξη του φυλλομετρητή, με αρχική σελίδα τη σελίδα προβολής περιεχομένου και το συνδέσαμε στο διαδίκτυο. Έπειτα, μεταφέρθηκαν τα αρχεία του διακομιστή και η βάση δεδομένων, στην υπηρεσία okeanos του GRNET [44] και αφού εγκαταστήσαμε το πακέτο LAMP, ελέγξαμε τα παρακάτω:

- Η συσκευή μπορεί, όντως, να συνδεθεί αυτόματα χωρίς την παρεμβολή του χρήστη και να πλοηγηθεί στον ιστότοπο.
- Η οθόνη μπορεί να ενεργοποιηθεί και να προβάλλει το εκάστοτε περιεχόμενο.
- Όλα τα δυνατά δεδομένα που μπορούν να εισαχθούν από το χρήστη και όλες οι παραμετροποιήσεις αυτών, λειτουργούν σωστά.
- Ενεργοποιήθηκαν άλλες δυο οθόνες, οι οποίες ήταν συνδεδεμένες σε επιτραπέζιους υπολογιστές και ελέγξαμε τη λειτουργία τους.
- Ο διακομιστής μπορεί να στείλει σωστά, το ανάλογο περιεχόμενο στην κάθε οθόνη.
- Για λόγους ευκολίας, η διάρκεια ζωής της κάθε οθόνης ορίστηκε από μερικά λεπτά ως μερικές ώρες για να μπορέσουμε να ελέγξουμε ότι τερματίζει σωστά την επικοινωνία με το διακομιστή. Όντως, στο χρόνο λήξης, η οθόνη δεν μπορούσε να αλλάξει περιεχόμενο.

Στην παρακάτω εικόνα, φαίνεται μια ενεργή οθόνη, συνδεδεμένη στο BeagleBoard-xM.



Σχήμα 4.32: BeagleBoard-xM και οθόνη

Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο, περιγράψαμε πως από τις απαιτήσεις του συστήματος εξάχθηκαν τα ανάλογα συμπεράσματα ώστε να γίνει η υλοποίηση σε ένα προγραμματιστικό περιβάλλον. Αναφερθήκαμε στους πίνακες που αποτελούν τη βάση δεδομένων, στις δυνατότητες των χρηστών και πως μπορούν να επιδράσουν με το σύστημα και στα μέτρα ασφαλείας που πάρθηκαν για την ομαλή λειτουργία του διακομιστή και την εμπειρία του χρήστη. Τέλος, δόθηκαν μερικά κομμάτια κώδικα που αναφέρθηκαν στο κεφάλαιο ώστε, να γίνει καλύτερα κατανοητή η λειτουργία του ιστοτόπου και παρουσιάστηκαν, το είδος των αρχείων και το μέγεθός τους, που αποτελούν το σύστημα.

Κεφάλαιο 5

Συμπεράσματα και μελλοντικές επεκτάσεις

Στην παρούσα διπλωματική εργασία, αναπτύχθηκε και υλοποιήθηκε ένα σύστημα διαχείρισης δημόσιων οθονών. Το σύστημα αποτελείται από έναν ιστότοπο όπου ο χρήστης συνδέεται και μπορεί να δημιουργήσει το δικό του περιβάλλον το οποίο αργότερα θα προωθήσει σε μια οθόνη. Η οθόνη είναι συνδεδεμένη σε ένα υπολογιστή με διαδικτυακή σύνδεση για να μπορεί να λάβει τα δεδομένα. Το σύστημα μπορεί να υποστηρίζει πολλαπλές οθόνες.

5.1 Σύνοψη και συμπεράσματα

Το σύστημα που υλοποιήθηκε αποτελεί μια γενική επιλογή για την εγκατάσταση ενός συστήματος δημόσιων οθονών. Τα κριτήρια, με βάση τα οποία αναπτύχθηκε το σύστημα, καλύπτουν την πλειοψηφία των ανθρώπων που θα είχαν την ανάγκη ενός τέτοιου συστήματος. Πιθανώς, κάποιοι πιο έμπειροι χρήστες, να νιώσουν την έλλειψη κάποιας εξειδικευμένης δυνατότητας. Θεωρούμε, όμως, πως και σε αυτήν την περίπτωση οι δυνατότητες και ο τρόπος που αυτές προσφέρονται, μετριάζουν αυτήν την πιθανή ανεπάρκεια σε ένα χαρακτηριστικό.

Το σύστημα διαθέτει ένα φιλικό περιβάλλον, που βοηθά στην αλληλεπίδραση με το χρήστη. Πάρθηκαν τα κατάλληλα μέτρα προστασίας ώστε τα διαχειριζόμενα δεδομένα να αποθηκεύονται στο σύστημα σε μια ασφαλή μορφή. Όλες οι τεχνολογίες που χρησιμοποιήθηκαν είναι ευρέως διαδεδομένες και υπάρχει άμεση υποστήριξη.

Σε σύγκριση με διάφορες εμπορικές εφαρμογές, θεωρούμε ότι η παρούσα υλο-

ποίηση έχει αρκετά πλεονεκτήματα. Αρχικά, οι εφαρμογές απαιτούν εγκατάσταση του δικού τους προγράμματος. Χρειάζονται δυο προγράμματα, ένα για τη σχεδίαση και ένα για την προβολή του περιεχομένου που είναι κλειστού κώδικα και συνήθως απαιτούν λειτουργικό σύστημα Microsoft Windows. Σε σχέση με αυτές τις εφαρμογές, το παρών σύστημα άρει αυτούς τους περιορισμούς. Το σύστημα είναι σχεδιασμένο με βάση τις τεχνολογίες διαδικτύου όπου το μόνο πρόγραμμα που απαιτείται είναι ένας φυλλομετρητής και δεν υπάρχουν απαιτήσεις ως προς το λειτουργικό σύστημα.

Ένα άλλο πλεονέκτημα είναι ως προς το κόστος. Οι εμπορικές εφαρμογές κοστίζουν αρκετά χρήματα μόνο για την απόκτηση της άδειας. Σε περίπτωση που ο χρήστης χρειάζεται και άλλες οθόνες πέρα από μία, τότε θα αναγκαστεί να αγοράσει και άλλη άδεια για κάθε οθόνη. Στο υπάρχον σύστημα, δεν υπάρχει κάποια οικονομική επιβάρυνση. Τα έξοδα του χρήστη, είναι η αγορά των οθονών και των υπολογιστών βάσεων που είναι μια καθαρά προσωπική επιλογή με βάση τις δυνατότητές του.

Τέλος, για τη δημιουργία του περιεχομένου ο χρήστης περιορίζεται στις επιλογές που του δίνει η εφαρμογή. Σε αυτήν την υλοποίηση, αν και υπάρχουν οι βασικές επιλογές που μπορεί ο καθένας να χρησιμοποιήσει, ο χρήστης με λίγες γνώσεις HTML μπορεί να γράψει το δικό του κώδικα και να αναπτύξει το περιεχόμενο με βάση τα δικά του κριτήρια.

5.2 Προβλήματα κατά την ανάπτυξη

Τα περισσότερα προβλήματα που προέκυψαν ήταν προγραμματιστικής φύσεως και έπρεπε να επιλυθούν. Χρειάστηκαν αρκετές επαναλήψεις και δοκιμές σε συγκεκριμένα κομμάτια κώδικα για την επιβεβαίωση της ορθής τους λειτουργίας. Έπρεπε να καλυφθεί κάθε δυνατό σενάριο που μπορεί να προέκυπτε. Κάποιες φορές γινόταν ασυνείδητα κάποιο λογικό ή συντακτικό λάθος και ενώ φαινόταν πως ο αλγόριθμος είναι σωστά γραμμένος, δεν απόφερε τα ανάλογα αποτελέσματα.

Ως προς το υλικό, ένας περιορισμός του BeagleBoard-xM είναι η σύνδεση του με μια οθόνη. Απαιτεί, αποκλειστικά, σύνδεση HDMI με DVI-D. Έτσι θα έπρεπε να βρεθεί και η κατάλληλη οθόνη. Όπου και πάλι, όμως, δεν είναι όλες συμβατές. Αυτοί οι περιορισμοί δεν τέθηκαν από εμάς, αλλά υπάρχουν στη συγκεκριμένη συσκευή. Τέλος, η εγκατάσταση του λειτουργικού συστήματος στο BeagleBoard-xM, αν και εύκολη γιατί δεν απαιτούνταν εξειδικευμένες ρυθμίσεις και υπήρχε και γραφικό περιβάλλον, ήταν αρκετά χρονοβόρα φτάνοντας στις 11 ώρες.

5.3 Μελλοντικές επεκτάσεις

Σίγουρα υπάρχουν κάποιες δυνατότητες δεν μπόρεσαν να προστεθούν, σε αυτήν την πρώτη φάση ανάπτυξης. Το σύστημα είναι, όμως, έτσι δομημένο ώστε να μπορεί να δεχθεί πολλές τροποποιήσεις στο μέλλον. Κάποιες από αυτές παρουσιάζονται παρακάτω:

- Επιπλέον επιλογές περιεχομένου. Αν και οι υπάρχοντες επιλογές καλύπτουν ένα μεγάλο σύνολο των χρηστών, εν τούτοις στις παρούσες επιλογές θα μπορούσαν να προστεθούν και άλλες δυνατότητες παραμετροποίησης ή νέες επιλογές δεδομένων.
- Διαδραστικότητα οθόνης. Αυτήν την στιγμή, η μόνη επαφή μεταξύ ενός θεατή και της οθόνης είναι η επιλογή ενός άλλου περιεχόμενου. Στο μέλλον θα μπορούσε να προστεθεί ένα δίκτυο αισθητήρων, οι οποίοι συλλέγουν κάποια δεδομένα και έπειτα από επεξεργασία, τα παρουσιάζουν εκείνη τη στιγμή.
- Δημιουργία εφαρμογής για smartphones. Η εφαρμογή αυτή θα μπορούσε να βοηθήσει στην επιλογή των περιεχομένων για μια οθόνη απομακρυσμένα. Αν και τώρα μπορεί να διαλέξει ο χρήστης ένα περιεχόμενο μέσα από το κινητό του τηλέφωνο, καλό θα ήταν η ύπαρξη μιας εφαρμογής με το κατάλληλο περιβάλλον.
- Ανάπτυξη του ιστοτόπου σε εφαρμογή. Αν και όπως τονίστηκε αυτό είναι αρνητικό γιατί έχει περιορισμούς, θα μπορούσε να δημιουργηθεί μια εφαρμογή για τη σχεδίαση των περιεχομένων και μια για την προβολή τους. Προφανώς, θα ήταν ανοιχτού κώδικα και εκτελέσιμες σε ένα εύρος λειτουργικών συστημάτων.
- Δημιουργία και άλλων βαθμίδων δικαιωμάτων χρηστών. Αυτή τη στιγμή, υπάρχει ο υπερχρήστης και όλοι οι υπόλοιποι χρήστες. Στο μέλλον θα μπορούσε οι ομάδες χρηστών να διασπαστούν σε μικρότερες υπο-ομάδες, η κάθε μια με συγκεκριμένα δικαιώματα.
- Άρση των περιορισμών ρεύματος. Το σύστημα δεν μπορεί να λειτουργήσει χωρίς ρεύμα. Ούτε η οθόνη θα λειτουργεί ούτε ο δρομολογητής που θα είναι συνδεδεμένος ο υπολογιστής θα μπορεί να επικοινωνήσει με το διακομιστή. Έτσι, θα μπορούσε να χρησιμοποιηθεί ένα UPS ή κάποια άλλη εναλλακτική τροφοδοσία, ώστε να είναι συνεχώς διαθέσιμο. Αυτό θα έλυσε και το πρόβλημα υπέρτασης του ρεύματος που μπορεί να βλάψει τις συσκευές.

Σύνοψη κεφαλαίου

Στο αυτό το τελευταίο κεφάλαιο, αναφερθήκαμε στα γενικά συμπεράσματα που προέκυψαν μετά την ολοκλήρωση της εργασίας. Αναφερθήκαμε στην κάλυψη των στόχων και των προδιαγραφών που τέθηκαν, στα προβλήματα που αντιμετωπίσαμε και επιλύσαμε και τέλος, στο εξέλιξη που θα μπορούσε να έχει το σύστημα στο μέλλον.

Παράρτημα

Στο παρών τμήμα της εργασίας, θα παρουσιάσουμε πως μπορεί κάποιος, να εγκαταστήσει τον ιστότοπο στο δικό του σύστημα ώστε να μπορέσει να τον λειτουργήσει και να κάνει τις δικές του δοκιμές. Στο πρώτη ενότητα θα αναφερθούμε στο κομμάτι του διακομιστή και στη δεύτερη στο κομμάτι του σταθμού βάσης.

Εγκατάσταση αρχείων στο διακομιστή

Θεωρούμε ότι ο διακομιστής έχει ήδη εγκατεστημένο έναν εξυπηρετή ιστού, τη γλώσσα PHP και το σύστημα βάσης δεδομένων MySQL. Όπως αναφέρθηκε στο Κεφάλαιο 2, ο εξυπηρετής ιστού που εγκαταστήσαμε ήταν ο apache. Οι εκδόσεις των προγραμμάτων που χρησιμοποιήθηκαν είναι αυτές που αναφέρονται στο Κεφάλαιο 2. Πιστεύουμε πως και σε παλιότερες ή πιο σύγχρονες εκδόσεις τα αποτελέσματα θα είναι ίδια. Ειδικά για την PHP, όμως, η έκδοση θα πρέπει να είναι τουλάχιστον η 5^η για την υποστήριξη των PDO.

Τα βήματα που ακολουθούμε είναι τα παρακάτω.

1. Αντιγράφουμε όλους τους υποκαταλόγους, του καταλόγου "ergasia", στον κατάλογο που είναι ο εξυπηρετής μας. Για τον apache και σε λειτουργικό σύστημα Ubuntu 14.04 είναι ο /var/www/html.
2. Τα δικαιώματα όλων των αρχείων θα πρέπει να είναι 755. Άρα, σε ένα σύστημα Unix πληκτρολογούμε "chmod -R 755 *".
3. Μέσω της κονσόλας, συνδεόμαστε στη βάση δεδομένων MySQL και εκτελούμε, με έναν διαχειριστή με δικαιώματα να δημιουργεί βάσεις δεδομένων, την εντολή `source /location/to/file/ergasia/PDM.sql`. Το αρχείο αυτό θα δημιουργήσει όλους τους πίνακες που χρειάζονται μαζί με τις συσχετίσεις τους. Η μονή εγγραφή που υπάρχει, είναι του χρήστη root στον πίνακα *users_information*.

4. Αφού γίνει η εισαγωγή της βάσης δεδομένων, ο χρήστης μπορεί να συνδεθεί στο σύστημα. Προτείνεται η αλλαγή του κωδικού και της ηλεκτρονικής διεύθυνσης ταχυδρομείου, μέσω του ιστοτόπου.
5. Σε αυτό το σημείο μπορεί να δημιουργήσει νέους χρήστες και οθόνες ή να σχεδιάσει κάποια περιεχόμενα.

Εγκατάσταση λειτουργικού στο σταθμό βάσης

Θα περιγράψουμε, πως εγκαταστάθηκε το λειτουργικό σύστημα Debian στο BeagleBoard-xM και πως ρυθμίστηκε κατάλληλα ώστε να ξεκινάει αυτόματα και να ανοίγει τον φυλλομετρητή απευθείας στη σελίδα εμφάνισης περιεχομένου. Αυτό το σημείο, μπορεί να παραληφθεί από τους χρήστες που σκοπεύουν να χρησιμοποιήσουν έναν άλλο σταθμό βάσης.

Για την εγκατάσταση του λειτουργικού συστήματος Debian, χρειαζόμαστε μια κάρτα micro sd και ένα αναγνώστη καρτών μνήμης. Η κάρτα micro sd προτείνεται να είναι τουλάχιστον 4GB. Θα ακολουθήσουμε τη μέθοδο NetInstall, η οποία περιλαμβάνει τα βασικά πακέτα του λειτουργικού και τα υπόλοιπα πακέτα, τα κατεβάζει από το διαδίκτυο. Η εγγραφή του Debian NetInstall γίνεται στην κάρτα μνήμης μέσω ενός συστήματος Unix. Τα βήματα για την εγκατάσταση του λειτουργικού στη κάρτα μνήμης, βρίσκονται στον ακόλουθο σύνδεσμο <http://elinux.org/BeagleBoardDebian>

1. Εκτελούμε την εντολή
`git clone "https://github.com/RobertCNelson/netinstall.git"`.
2. Εκτελούμε την εντολή `"cd netinstall"`.
3. Τοποθετούμε την κάρτα micro sd στον αναγνώστη καρτών μνήμης.
4. Εκτελούμε την εντολή `"sudo ./mk_mmc.sh -probe-mmc"` και βλέπουμε σε ποιο device έχει οριστεί η κάρτα μνήμης μας. Μεγάλη προσοχή σε αυτό το βήμα γιατί ενδέχεται να διαγράψουμε ένα άλλο αποθηκευτικό μέσο, αν διαλέξουμε λάθος συσκευή.
5. Εκτελούμε την εντολή `"sudo ./mk_mmc.sh -mmc /dev/sdX -dtb omap3-beagle-xm -distro jessie"` και αντικαθιστούμε το X με το γράμμα της συσκευής της κάρτας μνήμης. Στην παράμετρο dtb ορίσαμε τη συσκευή του

BeagleBoard-xM και στην παράμετρο `distro` την διανομή Debian που θέλουμε να εγκαταστήσουμε και συγκεκριμένα την τελευταία σταθερή έκδοση, την Jessie.

6. Αφού ολοκληρωθεί η διαδικασία (περίπου 10 λεπτά) αφαιρούμε την κάρτα μνήμης από τον αναγνώστη και την τοποθετούμε στο BeagleBoard-xM.
7. Πάνω στο BeagleBoard-xM συνδέουμε ακόμη μια οθόνη αυστηρά με είσοδο DVI-D, μέσω της θύρας HDMI, ένα καλώδιο ethernet που καταλήγει σε ένα δρομολογητή με πρόσβαση στο διαδίκτυο, και ένα πληκτρολόγιο. Αφού συνδέσουμε αυτά τοποθετούμε το φορτιστή στο ρεύμα και στην υποδοχή της συσκευής.
8. Εγκαθιστούμε το λειτουργικό Debian. Η διαδικασία είναι σχετικά απλή, ακόμη και για αρχάριους χρήστες. Παράλληλα, όμως, είναι και χρονοβόρα γιατί απαιτούνται κάπου στις 11 ώρες. Όταν μας ζητήσει να επιλέξουμε ποιο γραφικό περιβάλλον να εγκατασταθεί, προτείνετε η επιλογή του Xfce4.
9. Μετά την εγκατάσταση, το πιθανότερο είναι η συσκευή να μην ξεκινάει. Αν συμβαίνει αυτό, αφαιρούμε τη κάρτα μνήμης από το BeagleBoard-xM, την τοποθετούμε στον αναγνώστη καρτών μνήμης, και με έναν επεξεργαστή κειμένου ανοίγουμε το αρχείο `/boot/uEnv.txt` που βρίσκεται στο `partition rootfs` της κάρτας μνήμης.
10. Στο αρχείο αυτό υπάρχει η γραμμή, `root=/dev/mmcblk0pX ro`, όπου X ένα ακέραιος αριθμός. Αφού την εντοπίσουμε, σβήνουμε το `ro`.
11. Τοποθετούμε πάλι την κάρτα μνήμης στο BeagleBoard-xM και τώρα θα μπορούμε να πάρουμε πρόσβαση στο σύστημα.
12. Σε αυτό το σημείο μπορούμε να συνδέσουμε και ένα ποντίκι, για την καλύτερη εξυπηρέτησή μας.

Ρυθμίσεις λειτουργικού στο σταθμό βάσης

Οι οδηγίες αυτές, αν και αναφέρονται στο λειτουργικό σύστημα Debian και στο γραφικό περιβάλλον Xfce4, προτείνεται να ακολουθηθούν, από όλους με τα ανάλογα βήματα για κάθε λειτουργικό σύστημα.

1. Εγκαθιστούμε το φυλλομετρητή Chromium. Η εγκατάσταση γίνεται με την εντολή "sudo apt-get install chromium-browser".
2. Απενεργοποιούμε την ερώτηση για σύνδεση με όνομα χρήστη και συνθηματικό. Για να το κάνουμε αυτό, ανοίγουμε με έναν επεξεργαστή κειμένου το αρχείο /etc/lightdm/lightdm.conf, βρίσκουμε τη γραμμή που γράφει #autologin-user=, αφαιρούμε το σύμβολο της δίεσης που δηλώνει τη γραμμή ως σχόλιο και μετά το σύμβολο της ισότητας, προσθέτουμε το όνομα χρήστη που θέλουμε να κάνουμε αυτόματα σύνδεση.
3. Απενεργοποιούμε το σβήσιμο της οθόνης. Στο γραφικό περιβάλλον Xfce4, αυτό γίνεται ως εξής: επιλέγουμε Applications menu → Settings → Power Managements και στην καρτέλα System ρυθμίζουμε τις επιλογές "Blank after", "Put to sleep after", "Switch off after" στο never.



Απενεργοποίηση σβησίματος οθόνης

4. Ορίζουμε να ξεκινάει ο Chromium, μαζί με το λειτουργικό σύστημα σε λειτουργία πλήρης οθόνης και με αρχική σελίδα το

`http://path/to/server/display.php?name=unique_id`. Ως `unique_id` ορίζουμε το μοναδικό αλφαριθμητικό κάθε οθόνης, ώστε να μην χρειάζεται να την επιλέξουμε μόνοι μας. Σε ένα τερματικό, γράφουμε την εντολή `crontab -e` και στο τέλος του αρχείου που άνοιξε, προσθέτουμε τη γραμμή `@reboot /usr/sbin/chromium-browser -kiosk http://path/to/server/display.php?name=unique_id`.

5. Προαιρετικά, μπορούμε να κρύψουμε και τον κέρσορα του ποντικιού. Εγκαθιστούμε το πακέτο `unclutter` με την εντολή `sudo apt-get install unclutter`. Ανοίγουμε πάλι το `crontab` με την εντολή `crontab -e` και στο τέλος του αρχείου, προσθέτουμε τη γραμμή `@reboot /usr/bin/unclutter -idle 0`.
6. Επανεκκινούμε το σύστημα και θα δούμε, πως συνδεθήκαμε χωρίς να δώσουμε τα στοιχεία μας, ο φυλλομετρητής άνοιξε στη σωστή σελίδα και σε πλήρης οθόνη, και αν για αυτήν την οθόνη έχει οριστεί ένα περιεχόμενο, θα το δούμε να προβάλλεται.

Βιβλιογραφία

- [1] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5716771>.
- [2] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5633972>.
- [3] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6815260>.
- [4] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6373665.
- [5] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5741322.
- [6] *IEEE Xplore Abstract*. Φεβρουάριος 2016. URL: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1364701>.
- [7] *Xibo Open Source Digital Signage*. Φεβρουάριος 2016. URL: <http://xibo.org.uk/>.
- [8] *Scala Digital Signage Software*. Φεβρουάριος 2016. URL: <http://scala.com/>.
- [9] *Rise Vision*. Φεβρουάριος 2016. URL: <https://www.risevision.com/>.
- [10] *Techopedia*. Φεβρουάριος 2016. URL: <https://www.techopedia.com/definition/23889/web-development>.
- [11] *Webopedia*. Φεβρουάριος 2016. URL: http://www.webopedia.com/TERM/C/client_side.html.
- [12] *Upwork*. Φεβρουάριος 2016. URL: <https://www.upwork.com/hiring/development/server-side-scripting-back-end-web-development-technology/>.
- [13] *W3Schools*. Φεβρουάριος 2016. URL: http://www.w3schools.com/html/html_intro.asp.

- [14] *W3Schools*. Φεβρουάριος 2016. URL: http://www.w3schools.com/css/css_intro.asp.
- [15] *W3Schools*. Φεβρουάριος 2016. URL: http://www.w3schools.com/js/js_intro.asp.
- [16] *jQuery*. Φεβρουάριος 2016. URL: <https://jquery.com/>.
- [17] *jQuery UI*. Φεβρουάριος 2016. URL: <http://jqueryui.com/>.
- [18] *PHP: Hypertext Preprocessor*. Φεβρουάριος 2016. URL: <http://php.net/manual/en/intro-what-is.php>.
- [19] *LinkedIn*. Φεβρουάριος 2016. URL: <https://www.linkedin.com/pulse/20140726092427-231927153-what-is-ajax-technology>.
- [20] *MySQL*. Φεβρουάριος 2016. URL: <https://www.mysql.com/>.
- [21] *Webopedia*. Φεβρουάριος 2016. URL: <http://www.webopedia.com/TERM/L/LAMP.html>.
- [22] *Geany*. Φεβρουάριος 2016. URL: <http://www.geany.org/Main/About>.
- [23] *phpMyAdmin*. Φεβρουάριος 2016. URL: <https://www.phpmyadmin.net/>.
- [24] *GIMP*. Φεβρουάριος 2016. URL: <https://www.gimp.org/>.
- [25] *BeagleBoard.org*. Φεβρουάριος 2016. URL: <http://beagleboard.org/beagleboard-xm>.
- [26] *GitHub*. Φεβρουάριος 2016. URL: https://github.com/CircuitCo/BeagleBoard-xM-RevC/blob/master/BeagleBoard-xM_revC_SRM.pdf?raw=true.
- [27] *Debian*. Φεβρουάριος 2016. URL: <https://www.debian.org/intro/about>.
- [28] *Samsung*. Φεβρουάριος 2016. URL: http://downloadcenter.samsung.com/content/UM/200807/20080729150425140_BN59-00720B-Eng.pdf.
- [29] *PHP: Hypertext Preprocessor*. Φεβρουάριος 2016. URL: <http://php.net/manual/en/function.password-hash.php>.
- [30] *Wikipedia*. Φεβρουάριος 2016. URL: https://en.wikipedia.org/wiki/Unix_time.
- [31] Johannes Gehrke Raghu Ramakrishnan. *ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ*. ed. by Εκδόσεις Τζιόλα. 2012, p. 29.
- [32] *TechTarget*. Φεβρουάριος 2016. URL: <http://searchcrm.techtarget.com/definition/entity-relationship-diagram>.

- [33] *CKEditor.com*. Φεβρουάριος 2016. URL: <http://ckeditor.com/>.
- [34] *SimplePie*. Φεβρουάριος 2016. URL: <http://simplepie.org/>.
- [35] *FlipClock.js*. Φεβρουάριος 2016. URL: <http://flipclockjs.com/>.
- [36] *jquery.qrcode.js*. Φεβρουάριος 2016. URL: <http://jeromeetienne.github.io/jquery-qrcode/>.
- [37] *Trent Richardson - practical web design & development*. Φεβρουάριος 2016. URL: <http://trentrichardson.com/examples/timepicker/>.
- [38] *GitHub*. Φεβρουάριος 2016. URL: <https://github.com/josedvq/colpick-jQuery-Color-Picker>.
- [39] *World Wide Web Consortium (W3C)*. Φεβρουάριος 2016. URL: <https://www.w3.org/TR/eventsourcing/>.
- [40] *Admin's Choice - Choice of Unix and Linux administrators*. Φεβρουάριος 2016. URL: <http://www.adminschoice.com/crontab-quick-reference>.
- [41] *PHP: Hypertext Preprocessor*. Φεβρουάριος 2016. URL: <http://php.net/manual/en/pdo.prepared-statements.php>.
- [42] *Webopedia*. Φεβρουάριος 2016. URL: http://www.webopedia.com/TERM/C/control_character.html.
- [43] *The Chromium Projects*. Φεβρουάριος 2016. URL: <https://www.chromium.org/Home>.
- [44] *Home | okeanos IAAS*. Φεβρουάριος 2016. URL: <https://okeanos.grnet.gr/home/>.