

Τ.Ε.Ι ΗΠΕΙΡΟΥ
Τ.Ε.Ι OF EPIRUS



ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ & ΟΙΚΟΝΟΜΙΑΣ(Σ.Δ.Ο)
ΤΜΗΜΑ ΤΗΛΕΠΛΗΡΟΦΟΡΙΚΗΣ & ΔΙΟΙΚΗΣΗΣ

SCHOOL OF MANAGEMENT AND ECONOMICS
DEPARTMENT OF COMMUNICATIONS,
INFORMATICS AND MANAGMENT

ΤΕΙ ΗΠΕΙΡΟΥ

Τμήμα : Τηλεπληροφορικής και Διοίκησης

ΠΟΡΙΚΗΣ ΑΓΓΕΛΟΣ
ΧΗΡΑΣ ΑΝΑΡΓΥΡΟΣ

“ Ανάπτυξη Ηλεκτρονικού Δήμου “

Οκτώβριος 2004

“ Ανάπτυξη Ηλεκτρονικού Δήμου “

Πορίκης Άγγελος
Χήρας Ανάργυρος

11/10/2004

Πτυχιακή Εργασία μέρος των απαιτήσεων
Του τμήματος Τηλεπληροφορικής και Διοίκησης

Λίγα λόγια...

Η εφαρμογή που παρουσιάζεται παρακάτω διαπραγματεύεται και υλοποιεί σκέψεις και ιδέες για το πώς θα μπορούσαν οι υπηρεσίες ενός Δήμου να δουλεύουν μέσω Internet ώστε να γίνουν πιο ανταγωνιστικές και να εξυπηρετούν καλύτερα το κοινό. Επιλέχθηκε λοιπόν η υπηρεσία του Δημοτολογίου και η διαχείριση των υπαλλήλων που εκτελούν βασικές λειτουργίες του Δήμου. Έτσι δίνεται δυνατότητα στους διαχειριστές να οργανώνουν το σύστημα και τους υπαλλήλους, στους υπαλλήλους να διαχειρίζονται το δημοτολόγιο, στους δημότες να παίρνουν οποιοδήποτε πιστοποιητικό αφορά το δημοτολόγιο και όλα αυτά μέσω Internet.

Επιπλέον διαπραγματεύονται και αντιμετωπίζονται τα διάφορα θέματα ασφαλείας που προκύπτουν καθώς δεν πρέπει να έχουν πρόσβαση όλοι στις ηλεκτρονικές υπηρεσίες για ευνόητους λόγους. Εξηγείται αναλυτικά η λειτουργία της εφαρμογής και πως αυτή μπορεί να προσαρμοστεί στην ιστοσελίδα οποιουδήποτε Ελληνικού Δήμου.

Υπάρχει επίσης παρουσίαση των εργαλείων που χρησιμοποιήθηκαν για το στήσιμο της εφαρμογής.

Περιεχόμενα

	Γενικά.....	1
1	Πρόλογος	
1.1	Οργανωτική παρουσίαση ενός τυπικού Δήμου.....	4
1.2	Προβλήματα από την οργάνωση ενός τυπικού Δήμου.....	6
1.3	Το πρόβλημα που αντιμετωπίζει η παρούσα εργασία.....	6
2	Απαιτούμενο λογισμικό	
2.1	Ο διακομιστής ιστοσελίδων Apache.....	10
2.1.1	Ιστορικά στοιχεία.....	11
2.1.2	Εγκατάσταση.....	13
2.1.3	Παραμετροποίηση.....	14
2.2	Η γλώσσα προγραμματισμού php	
2.2.1	Ιστορικά στοιχεία.....	15
2.2.2	Εγκατάσταση.....	17
2.2.3	Συνεργασία με τον Apache.....	17
2.3	Το σύστημα διοικήσεως βάσεων δεδομένων MySQL.....	18
2.3.1	Ιστορικά στοιχεία.....	18
2.3.2	Εγκατάσταση.....	19
2.3.3	Συνεργασία mysql και php.....	20
3	Οργάνωση εφαρμογής	
3.1	Κεντρική σελίδα.....	21
3.2	Ενημερωτικό μέρος-Είσοδος σαν επισκέπτης.....	22
3.3	Είσοδος σαν δημότης.....	22
3.4	Είσοδος σαν υπάλληλος.....	25
3.5	Είσοδος σαν προϊστάμενος.....	25
3.6	Διαχείριση συστήματος.....	30
3.7	Είσοδος σαν υπάλληλος άλλων υπηρεσιών.....	32
3.8	Εγκατάσταση.....	34
4	Συμπέρασμα	
4.1	Κρίση αποδοτικότητας της εφαρμογής – Μελλοντικές Κατευθύνσεις.....	35
5	Παράρτημα (Κώδικας Εφαρμογής).....	

ΔΗΛΩΣΗ ΠΕΡΙ ΛΟΓΟΚΛΟΠΗΣ

Όλες οι προτάσεις οι οποίες παρουσιάζονται σ' αυτό το κείμενο και οι οποίες ανήκουν σε άλλους αναγνωρίζονται από τα εισαγωγικά και υπάρχει η σαφής δήλωση του συγγραφέα. Τα υπόλοιπα γραφόμενα είναι επινόηση των γραφόντων οι οποίοι φέρουν και την καθολική ευθύνη γι αυτό το κείμενο και δηλώνουμε υπεύθυνα ότι δεν υπάρχει λογοκλοπή γι αυτό το κείμενο.

Όνοματεπώνυμο

Υπογραφή..... Ημερομηνία.....

1. Πρόλογος

1.1 Οργανωτική παρουσίαση ενός τυπικού δήμου

Πριν ξεκινήσουμε να αναφέρουμε την τυπική οργάνωση ενός Δήμου ας αναφέρουμε ποιος είναι ο σκοπός ενός δήμου .

Ο σκοπός ενός δήμου λοιπόν είναι :

A. Η δημιουργία κοινοφελών έργων

B. Η εξυπηρέτηση των κατοίκων (Δημοτών) σε θέματα που τους ενδιαφέρουν όπως την έκδοση βεβαιώσεων και πιστοποιητικών .

Μετά την εκλογή του ο/η Δήμαρχος σκοπό έχει, να υλοποιήσει το πρόγραμμα που είχε ανακοινώσει προεκλογικά στους δημότες. Βοήθεια στο πρόγραμμα αυτό έχει από τους Δημοτικούς συμβούλους. Για να υλοποιηθούν τα κοινοφελή έργα αυτά (π.χ. κατασκευή δρόμων, λιμανιών κ.α) χρειάζονται μελέτες οι οποίες συντάσσονται από μηχανικούς της περιφέρειας του δήμου, του τμήματος Τ.Υ.Δ.Κ. της Νομαρχιακής αυτοδιοίκησης και στην συνέχεια στέλνονται στην Νομαρχία ώστε να δοθεί το κονδύλι για την υλοποίησή τους .

Ο κάθε Δήμος προσπαθεί μέσα από το υπαλληλικό προσωπικό που διαθέτει να εξυπηρετήσει στο έπακρο τους δημότες του (π.χ. Έκδοση κάθε είδους πιστοποιητικού, γεννήσεως, οικογενειακής καταστάσεως κ.α.). Επίσης στην έκδοση Τ.Α.Π. (Τέλος Ακίνητης Περιουσίας) και κάθε άλλου είδους βεβαίωσης που αφορούν τα περιουσιακά και άλλα στοιχεία του.

Για να λειτουργήσει ο δήμος απαραίτητη είναι η σύσταση κάποιων Υπηρεσιών (χωρίς αυτές τις υπηρεσίες δεν νοείται δήμος) . Αυτές είναι :

A. Ταμειακές Υπηρεσίες

Η Ταμειακή Υπηρεσία σκοπό έχει την διεκπαιρέωση των πληρωμών που πρέπει να κάνει ο δήμος προς τρίτα φυσικά ή νομικά πρόσωπα

B. Λογιστήριο

Σκοπό έχει την συγκέντρωση όλων των απαραίτητων δικαιολογητικών (τιμολόγια, αποφάσεις, συμφωνητικά κ.α.) για την προώθηση τους προς το Δημοτικό Συμβούλιο (Δ.Σ.) για να παρθεί η απόφαση της πληρωμής των οφειλών .

Γ. Δημοτολόγια

Σκοπό έχει την έκδοση κάθε είδους πιστοποιητικού και την διεκπαιρέωση των εκλογών (εκλογικοί κατάλογοι).

Δ. Τεχνική Υπηρεσία

Σκοπό έχει την υλοποίηση των έργων .

Ε. Διοικητική Υπηρεσία

Είναι υπεύθυνη για το προσωπικό του Δήμου (Διοικητικό – Εργατικό) και ασχολείται με θέματα προσλήψεων με σκοπό την σωστή λειτουργία του Δήμου.

Ζ. Υπηρεσία Προσόδων

Ασχολείται με την διαδικασία είσπραξης των τελών του Δήμου.

Ο κάθε δήμος για να μπορεί να εξυπηρετεί το έργο του δημιουργεί επιχειρήσεις νομικού προσώπου μη κερδοσκοπικού χαρακτήρα (τα χρήματα που παίρνουν οι υπηρεσίες αυτές πηγαίνουν για την πληρωμή του υπαλληλικού προσωπικού καθώς και για την υλοποίηση έργων) οι οποίες ως στόχο έχουν την καλύτερη οργάνωση του δήμου . Ενδεικτικά αναφέρουμε τις επιχειρήσεις : Ύδρευσης και Αποχέτευσης, Αναπτυξιακές Επιχειρήσεις κ.α. ανάλογα με το μέγεθος και τις ανάγκες του δήμου.

1.2 Ιεραρχική Δομή ενός Δήμου

Ιεραρχικά η δομή του Δήμου έχει ως εξής :

A. Δήμαρχος

Υπεύθυνος για την ομαλή λειτουργία του δήμου

B. Αντιδήμαρχος

Εκτός από τις δικές του αρμοδιότητες ο αντιδήμαρχος αντικαταστέι τον δήμαρχο στην λήψη αποφάσεων, όταν αυτός είναι απών.

Γ. Πρόεδρος Δημοτικού Συμβουλίου (Δ.Σ.)

Είναι δημοτικός σύμβουλος και εκλέγεται τις περισσότερες φορές ο πρώτος (1^{ος}) σε ψηφοδέλτια. Αυτός προεδρεύει σε κάθε συνεδρίαση του Δ.Σ. αφού πρώτα με πρόσκληση του έχει καλέσει σε συμβούλιο τους δημοτικούς συμβούλους

Δ. Δημοτικοί Σύμβουλοι

Παίρνουν μέρος στις συνεδριάσεις του Δ.Σ. για να αποφασίσουν ομόφωνα ή κατά πλειοψηφία για διάφορες υποθέσεις του δήμου. Ο κάθε δημοτικός σύμβουλος μπορεί να είναι και πρόεδρος και σε κάποια από τις επιχειρήσεις του δήμου.

Ε. Πρόεδρος Τοπικού Συμβουλίου (Δημοτικού Διαμερίσματος –Τ.Σ.)

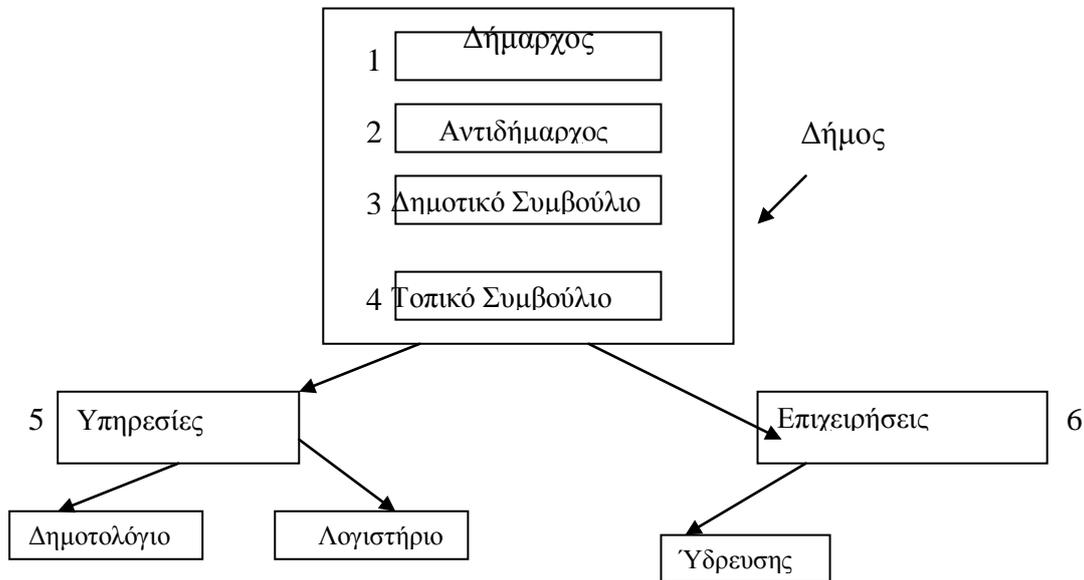
Είναι ο πρώτος (1^{ος}) σε ψήφους από το τοπικό συμβούλιο σε κάθε δημοτικό διαμέρισμα.

Ο πρόεδρος είναι και δημοτικός σύμβουλος

Στ. Τοπικό Συμβούλιο.

Εκλέγεται από τους δημότες του συγκεκριμένου δημοτικού διαμερίσματος. Το συμβούλιο σκοπό έχει μέσα από τις συνεδριάσεις του να εισηγήτε διάφορα θέματα που υπάρχουν στο διαμέρισμα δια του προέδρου του προς το δημοτικό συμβούλιο για να αποφασίσει σχετικά με την υλοποίηση ή όχι των αιτημάτων αυτών .

Βάση των όσων προαναφέραμε, διαγραμματικά η οργάνωση ενός δήμου έχει ως εξής :



(Σχήμα 1: Οργανωτικής Παρουσίασης Δήμου)

1.3 Προβλήματα από την οργάνωση ενός τυπικού δήμου

Όπως είναι γνωστό οι δήμοι σήμερα είναι σχηματισμένοι σύμφωνα με το πρόγραμμα «Ι. ΚΑΠΟΔΙΣΤΡΙΑΣ» του Υπουργείου Εσωτερικών Δημόσιας Διοίκησης και Αποκέντρωσης . Σύμφωνα με το πρόγραμμα αυτό οι Δήμοι έχουν αναδιαμορφωθεί και οι κοινότητες έχουν ενωθεί με τους Δήμους σχηματίζοντας έναν ενιαίο Δήμο με τις πρώην κοινότητες των χωριών να αποτελούν Δημοτικά διαμερίσματα του νέου Δήμου. Εδώ πρέπει να σημειώσουμε ότι μπορεί και μεγάλες (πρώην) κοινότητες να έχουν μετατραπεί σε δήμους ενώ κάποιες περιοχές (δήμοι/κοινότητες) εξαιρούνται από το πρόγραμμα αυτό (συνήθως κοινότητες με ιστορική σημασία).

Περισσότερες πληροφορίες παρέχει η ιστοσελίδα του Υπουργείου Εσωτερικών Δημόσιας Διοίκησης και Αποκέντρωσης (www.ypes.gr)

Σκοπός του προγράμματος αυτού είναι :

A. Η ενδυνάμωση των Δήμων

B. Η ενίσχυση της τοπικής αυτοδιοίκησης

Γ. Να παρέχουν καλύτερες υπηρεσίες προς τον δημότη

Δ. Η αύξηση των κρατικών επιχορηγήσεων του Δήμου (αφού οι επιχορηγήσεις είναι τόσο μεγαλύτερες όσο μεγαλύτερος είναι ο Δήμος)

E. Να δίνουν λύσεις στα προβλήματα όλου –πια- του ενιαίου Δήμου

(Στοιχεία από την ιστοσελίδα του Υπουργείου Εσωτερικών . Πρόγραμμα «Ι. ΚΑΠΟΔΙΣΤΡΙΑΣ»)

Παρότι αυτή η οργάνωση είχε σκοπό την καλύτερη λειτουργία του δήμου , έχουν παρουσιαστεί αρκετά προβλήματα στην διοικητική οργάνωση του. Η ενσωμάτωση πολλών κοινοτήτων σε έναν ενιαίο δήμο έχει ως αποτέλεσμα την αύξηση του αριθμού των υπαλλήλων που διαθέτει τώρα ο δήμος και μειώνει την δυνατότητα της σωστής παρακολούθησης και έλεγχου. Για παράδειγμα σε ζωτικής σημασίας υπηρεσίες όπως το Δημοτολόγιο δεν υπάρχει δυνατότητα ελέγχου των δεδομένων που εισάγονται στην βάση του Δημοτολογίου, έτσι σε περίπτωση κακόβουλων πράξεων δεν γίνεται αντιληπτό παρά μόνο όταν ζητηθούν τα δεδομένα αυτά .

Πρέπει να αναφέρουμε εδώ πως το πρόβλημα αυτό είναι διαχρονικό. Η διαφορά του τότε με το σήμερα είναι ότι ο έλεγχος ήταν τότε πιο εύκολος (μικρότερος αριθμός δημοτών) και έτσι τα λάθη ήταν σχετικά εύκολα να εντοπιστούν .

Επίσης υπάρχει πρόβλημα στην λήψη των αποφάσεων, αφού κάθε δημοτικό διαμέρισμα ζητάει να ικανοποιηθούν τα δικά του αιτήματα κάτι που είναι αδύνατον αφού τα έσοδα και οι χρηματοδοτήσεις είναι περιορισμένες .

Άλλο ένα μεγάλο πρόβλημα που προέκυψε είναι ότι σε πολλά μικρά χωριά (κοινότητες), ενώ πριν την ένωση διέθεταν δημοτικό κατάστημα ώστε να έχουν την δυνατότητα να εξυπηρετηθούν (έκδοση πιστοποιητικών και λοιπών βεβαιώσεων), με την συνένωση έπαψαν να διαθέτουν κατάστημα ώστε να μπορούν να πάρουν τα έγγραφα που θέλουν και να είναι αναγκασμένοι να μεταβαίνουν στην έδρα ή σε άλλο δημοτικό διαμέρισμα του δήμου για να εξυπηρετηθούν. Επίσης πολλές υπηρεσίες των πρώην κοινοτήτων/δήμων έπαψαν να λειτουργούν –αφού ενσωματώθηκαν σε μία κεντρική- και έτσι το πρόβλημα των δημοτών έχει διευρυνθεί ακόμη περισσότερο.

1.4 Το πρόβλημα που αντιμετωπίζει η παρούσα εργασία

Η παρούσα πτυχιακή εργασία ετοιμάστηκε στα πλαίσια του εκσυγχρονισμού της δημόσιας διοίκησης προς την κατεύθυνση της ηλεκτρονικής διακυβέρνησης (e-Government) που ενώ αυτός ο εκσυγχρονισμός έχει επιτευχθεί σε πολλά κράτη της Ευρώπης και της Αμερικής , στην χώρα μας βρίσκεται στα πρώτα βήματα της, κυρίως

λόγω της νομοθεσίας που δεν επιτρέπει κάτι τέτοιο και στην νοοτροπία του Έλληνα να μην αποδέχεται εύκολα καινοτομικές αλλαγές .

Σκοπό της εργασίας είναι η επίλυση διάφορων προβλημάτων που υπάρχουν κατά την εξυπηρέτηση των δημοτών από τις δημοτικές υπηρεσίες (μη δυνατότητα άμεσης λήψης πιστοποιητικών, βεβαιώσεων κ.α. σε μικρά δημοτικά διαμερίσματα από έλλειψη υπαλλήλων του δήμου, μεγάλη ταλαιπωρία λόγω αναμονής σε μεγάλους δήμους για την λήψη διάφορων εγγράφων κ.α.), καθώς αντιμετώπιση των προβλημάτων που αντιμετωπίζουν οι υπάλληλοι των υπηρεσιών του δήμου (π.χ. αργό το σύστημα σε μεγάλους αλλά και σε μικρούς δήμους, μη δυνατότητα έλεγχου εσφαλμένων λειτουργιών υπαλλήλων κ.α.)

Η παρούσα εφαρμογή παρέχει τις παρακάτω δυνατότητες ανάλογα το είδος του χρήστη :

1. Δημότης

Û Άμεση λήψη των πιστοποιητικών που χρειάζεται ο δημότης .

Με τον τρόπο αυτό αντιμετωπίζονται τα προβλήματα που έχουν παρουσιαστεί με την τρέχουσα οργάνωση του δήμου, αφού έχοντας πρόσβαση στο διαδίκτυο μπορεί να συνδεθεί στο ηλεκτρονικό σύστημα εξυπηρέτησης δημοτών και να πάρει το έγγραφο που τον ενδιαφέρει χωρίς να είναι αναγκασμένος να περιμένει σε ουρές ή να μεταβαίνει σε άλλο δημοτικό διαμέρισμα για την έκδοση του εγγράφου που θέλει . Έτσι γλυτώνει πολύτιμο χρόνο και χρήμα, που αυτό συνεπάγεται σε κέρδος (χρόνο, χρήμα) του δημότη

Û Δυνατότητα αποστολής παρατηρήσεων ή παραπόνων προς τις υπηρεσίες του δήμου.

2. Υπάλληλος Υπηρεσιών Δήμου

Û Μεγαλύτερη ασφάλεια πρόσβασης στο σύστημα μέσω ελέγχου IP Address και κωδικού πρόσβασης .

Û Πιο γρήγορη εκτέλεσης εργασιών, αφού το σύστημα βασίζεται στην Web Based αρχιτεκτονική που συνεπάγεται μη απαίτηση ειδικού ιδιωτικού δικτύου (οπότε μπορεί να γίνουν σφάλματα στο δίκτυο όπως να «πέσει»). Όπως γνωρίζουμε ένα σύστημα που βασίζεται στην αρχιτεκτονική αυτή, είναι ένα σύστημα client-Server που κάνει χρήση του internet ως μέσω επικοινωνίας. Από αυτό καταλαβαίνουμε πως αν έχουμε μια γρήγορη γραμμή επικοινωνίας (π.χ. 128 ISDN) ο client θα επικοινωνεί πιο γρήγορα με το server, χωρίς το κόστος να είναι τόσο μεγάλο όσο μιας μισθωμένης γραμμής που απαιτούν τα συστήματα των Δήμων σήμερα . Επιπλέον ο φόρτος εργασίας πέφτει αποκλειστικά στο όλο σύστημα (Web Server-Database server), οπου εκεί κάνοντας χρήση του Apache (Web Server) και της MySQL (Database Server) πετυχαίνουμε το καλύτερο δυνατό αποτέλεσμα.

Û Άμεση καταγραφή μη επιτρεπτών λειτουργιών των υπαλλήλων

3. Προϊστάμενος Υπηρεσιών Δήμου .

Û Ανάλογα με την υπηρεσία που ανήκει παίρνει και διαφορά στοιχεία (π.χ. για το Δημοτολόγιο έχουμε : Πλήρη στατιστικά δημογραφικά στοιχεία συγκεντρωτικά για όλο το δήμο, αλλά και ξεχωριστά για κάθε δημοτικό διαμέρισμα)

Û Διαχείριση Υπαλλήλων του τμήματος .

Δηλαδή ο προϊστάμενος του τμήματος γίνεται υπεύθυνος για την πρόσληψη ή απόλυση υπαλλήλων .

4. Διαχειριστής Συστήματος

- Û Δυνατότητα Διαχείρισης υπηρεσιών δήμου. Αυτό σημαίνει πως ο διαχειριστής έχει την δυνατότητα να προαφαιρέσει έναν τμήμα καθώς να αλλάξει τον κωδικό και τον προϊστάμενο του.
- Û Διαχείριση Δημοτών. Για να έχουν πρόσβαση στο σύστημα οι δημότες πρέπει να «ζητήσουν» τον κωδικό πρόσβασης τους από τον διαχειριστή του συστήματος . Αυτό θα γίνεται από ένα τμήμα που θα προμηθεύει τους κωδικούς στους δημότες . Επίσης σε περίπτωση διαγραφής ή θανάτου κάποιου δημότη που είχε πρόσβαση στο σύστημα ο διαχειριστής είναι υπεύθυνος για την διαγραφή του/της από το σύστημα.
- Û Διαχείριση Υπαλλήλων . Προσθαφαίρεση κωδικών πρόσβασης υπαλλήλων τμημάτων
- Û Έλεγχο Δημοτικών διαμερισμάτων .
- Û Εμφάνιση λάθος ενεργειών υπαλλήλων

Σήμερα στην Ελλάδα δεν υπάρχουν τέτοια πακέτα λογισμικού στην αγορά , αλλά ούτε σε ακαδημαϊκό επίπεδο έχουν γίνει τέτοιες εργασίες, παρότι συνέδρια για τρόπους ηλεκτρονικής διακυβέρνησης έχουν γίνει πολλά τα τελευταία χρόνια και τα αποτελέσματα αυτών οδήγησαν στην δημιουργία των Κέντρων Εξυπηρέτησης Πολιτών (Κ.Ε.Π.) όσο αφορά την τοπική αυτοδιοίκηση.

Η κατάσταση που επικρατεί σήμερα στις ιστοσελίδες των Ελληνικών δήμων κινείται–στους περισσότερους- προς την τουριστική προβολή του δήμου, για την προσέλκυση τουριστών. Έτσι, σε γενικές γραμμές, προσφέρουν πληροφορίες για τα προϊόντα του τόπου τους, για τα τουριστικά αξιοθέατα, για τα καταλύματα, εστιατόρια και χώρους διασκέδασης, χάρτες της περιοχής και χρήσιμα τηλέφωνα. Υπάρχουν βέβαια και ιστοσελίδες δήμων –αν και ελάχιστες- που εκτός της τουριστικής προβολής διαθέτουν και υπηρεσίες προς τους δημότες όπως είναι χρήσιμες πληροφορίες για τις ώρες λειτουργίας των δημοτικών επιχειρήσεων και υπηρεσιών του δήμου και την τοποθεσία τους , τι δικαιολογητικά πρέπει να προσκομίσουν οι δημότες στην περίπτωση που θέλουν την έκδοση κάποιου εγγράφου, ώστε να αποφευχθεί η ταλαιπωρία τους. Τέλος, τους δίνεται η δυνατότητα να αποστείλουν ηλεκτρονική αλληλογραφία (e-mail) προς τις υπηρεσίες του δήμου για παράπονα ή παρατηρήσεις (ενδεικτικά αναφέρουμε την ιστοσελίδα του δήμου Πρεβέζης που διαθέτει τέτοιες υπηρεσίες www.dimosprevezas.gr).

2. Απαιτούμενο Λογισμικό

2.1 Αρχιτεκτονική των εφαρμογών Διαδικτύου

1. Φυλλομετρητής(Web Browser)

Οι περισσότερες διαδικτυακές εφαρμογές έχουν έναν ενιαίο πελάτη : τον Web Browser που τρέχει στον υπολογιστή κάθε χρήστη που συνδέεται στο διαδίκτυο και επισκέπτεται διάφορες ιστοσελίδες. Η βασική γλώσσα των Browsers είναι η γλώσσα σήμανσης υπερκειμένων(HTML) αλλά όσο πιο καινούργιοι είναι διαθέτουν υποστήριξη για Javascripts και άλλα.

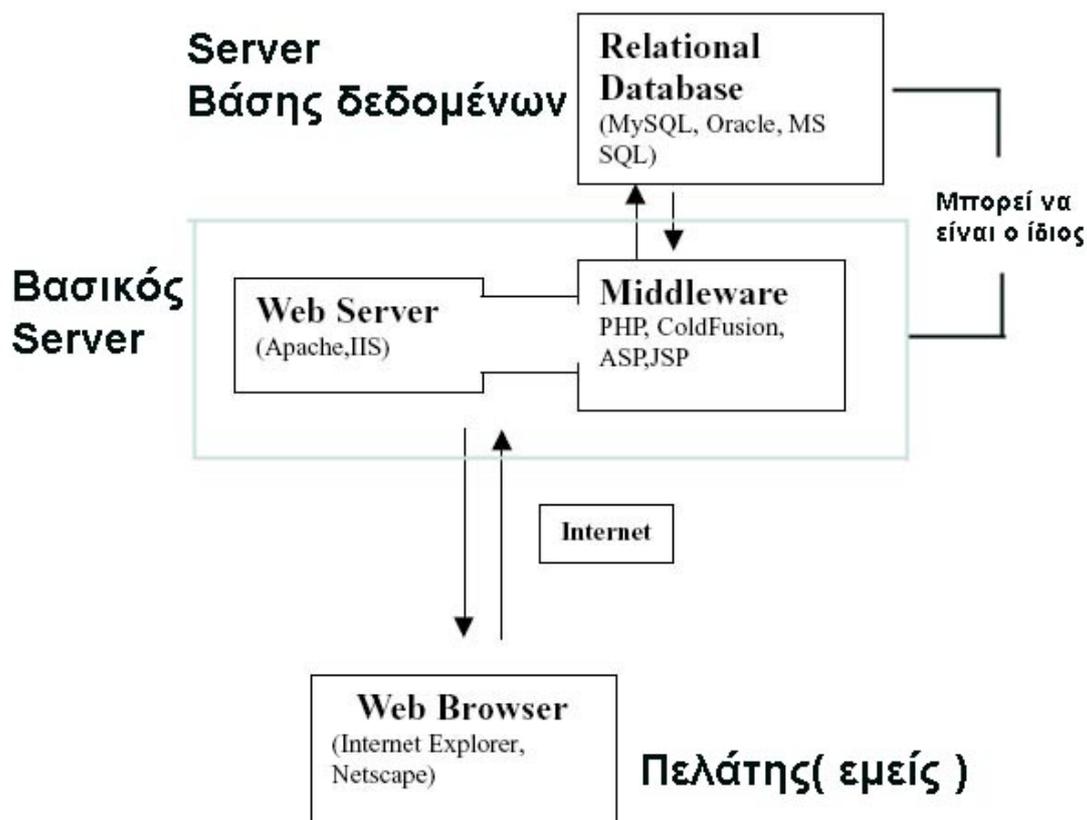
Γνωστοί browsers είναι ο Internet Explorer, ο Netscape Navigator, ο Opera και ο Konqueror(Linux).

2. Ο Server

Σχεδόν όλη η εργασία των διαδικτυακών εφαρμογών πραγματοποιείται στον Server. Μια συγκεκριμένη εφαρμογή, ο γνωστός **Web Server**, είναι αρμόδιος για την επικοινωνία με τον Browser μας. Ένας συγγενικός Server βάσεων δεδομένων αποθηκεύει οτιδήποτε πληροφορίες χρειάζεται η εφαρμογή.

Τέλος χρειάζεστε μια γλώσσα προγραμματισμού που να παρέχει επικοινωνία μεταξύ του Server και της Βάσης δεδομένων όπως επίσης να εκτελεί τις προγραμματιστικές εργασίες που έρχονται από τον Web Server αλληλεπιδρώντας με τον Browser σας.

Στο παρακάτω σχήμα φαίνεται ένα γενικό πλάνο της αλληλεπίδρασης των εφαρμογών (Σχήμα 2):



Όπως βλέπουμε και στο σχήμα μπορούμε να διαλέξουμε ανάμεσα σε διάφορους τύπους Βάσεων , Web Servers και γλώσσες Προγραμματισμού. Στην εφαρμογή μας θα επιλέξουμε τον συνδυασμό PHP-MySQL-Apache Server που είναι από τους πιο διαδεδομένους στο διαδίκτυο και αποδεικνύεται πολύ αξιόπιστος. Ακολουθεί περαιτέρω ανάλυση για την κάθε εφαρμογή.

2.2 Ο διακομιστής ιστοσελίδων Apache

2.2.1 Ιστορικά στοιχεία

2.2.1.1 Ο Web Server Apache

Όπως είπαμε ο Web Server είναι μια εφαρμογή που εκτελείται στον Server και επικοινωνεί με τον browser μας. Ο Apache είναι ένας από τους πιο γνωστούς και δοκιμασμένους Web Servers. Η δημιουργία του προήλθε από την προσπάθεια ανάπτυξης λογισμικού μιας ομάδας ανθρώπων που μέσα από συνεργασία προσπάθησαν να δημιουργήσουν μια εφαρμογή υλοποίησης κώδικα πηγής ενός HTTP Server.

Η εφαρμογή ρυθμίζεται από κοινού από μια ομάδα εθελοντών που βρίσκονται σε όλο τον κόσμο, χρησιμοποιώντας το Διαδίκτυο για να επικοινωνούν, να προγραμματίζουν, και να αναπτύσσουν τον server και τη σχετική βιβλιογραφία. Αυτοί οι εθελοντές είναι γνωστοί ως ομάδα Apache. Επιπλέον, εκατοντάδες χρήστες έχουν συμβάλει στις ιδέες, τον κώδικα, και την βιβλιογραφία για το πρόγραμμα.

2.2.1.2 Η δημιουργία του Apache

Το Φεβρουάριο του 1995, το δημοφιλέστερο λογισμικό Server στο Διαδίκτυο ήταν ο public domain HTTP daemon που αναπτύχθηκε από τον Rob McCool στο εθνικό κέντρο για Supercomputing εφαρμογές, του πανεπιστήμιο του Ιλλινόις. Ο httpd υποστηρίζει το γνωστό πρωτόκολλο http που είναι υπεύθυνο για την εμφάνιση των hypertext κειμένων στον browser με την μορφή ιστοσελίδων. Εντούτοις, η ανάπτυξη του httpd είχε χρονοτριβήσει αφότου άφησε ο Rob την NCSA στα μέσα του 1994 και πολλοί webmasters είχαν αναπτύξει τις δικές τους επεκτάσεις και διορθωτικά πακέτα που χρειαζόταν. Μια μικρή ομάδα αυτών των webmasters, που ήρθε σε επαφή μέσω ηλεκτρονικού ταχυδρομείου, συγκεντρωθήκαν μαζί με σκοπό να συντονίσουν και να ενώσουν τις αλλαγές τους υπό μορφή "patches". Ο Brian Behlendorf και Cliff Skolnick έφτιαξαν μαζί έναν κατάλογο διευθύνσεων, χώρο κοινών πληροφοριών και logins για τους υπεύθυνους για την ανάπτυξη του λογισμικού. Έτσι οκτώ προγραμματιστές που ήταν υπεύθυνοι για το λογισμικό ίδρυσαν την ομάδα Apache. Αυτοί ήταν οι:

Brian Behlendorf, Roy T. Fielding, Rob Hartill, David Robinson, Cliff Skolnick, Randy Terbush, Robert S. Thau, Andrew Wilson

με πρόσθετη συνεισφορά από τους:

Eric Hagberg, Frank Peters, Nicolas Pioch

Με τη χρησιμοποίηση httpd1.3 της NCSA ως βάση, προσθέσαν όλες τις δημοσιευμένες διορθώσεις και ότι άλλο αξιόλογο μπόρεσαν να βρουν, δοκίμασαν

το αποτέλεσμα στους κεντρικούς υπολογιστές τους και έκαναν την πρώτη επίσημη δημόσια έκδοση (0.6.2) του Apache Server τον Απρίλιο του 1995.

2.2.1.3 Εξέλιξη του Apache

Ο Apache 1.3 είναι ένας από τους πιο σταθερούς και γρήγορους Web Servers στο διαδίκτυο προσφέροντας παράλληλα σημαντικά εργαλεία που βελτιώνουν την λειτουργικότητά του σε θέματα συμβατότητας, διαχείρισης και ασφάλειας. Παρόλα αυτά στην πορεία έγιναν και άλλες βελτιώσεις με αποτέλεσμα να φτάσουμε στον Apache 2.0 που είναι η πιο πρόσφατη έκδοση του Apache Server. Η αρχιτεκτονική του περιέχει σημαντικές αναβαθμίσεις σχετικά με τις σειρές 1.3. Οι ακόλουθες είναι μερικές από αυτές.

Ενότητες πολυ-επεξεργασίας : Ο Apache 2.0 αφαιρεί την αρχιτεκτονική επεξεργασίας αιτήματος στα ειδικά modules των Server, αποκαλούμενα modules πολυ-επεξεργασίας (MPMs). Αυτό σημαίνει ότι ο Apache μπορεί να διαμορφωθεί για να είναι καθαρά process-based Server, καθαρά threaded Server ή ένα μίγμα των δύο προτύπων.

Ενότητες πρωτοκόλλου : Ο χειρισμός πρωτοκόλλου έχει τοποθετηθεί σε δικό του στρώμα στον Apache 2.0. Αυτό σημαίνει ότι είναι δυνατό να γραφτούν ενότητες για να εξυπηρετήσει πρωτόκολλα εκτός από το HTTP, όπως το POP3 για το ηλεκτρονικό ταχυδρομείο ή το FTP για τη μεταφορά αρχείων.

Modules και αρχιτεκτονική φίλτρων: Ο Apache 2.0 διατηρεί τη modular αρχιτεκτονική του 1.3 και προσθέτει έναν πρόσθετο μηχανισμό: φίλτρα επέκτασης. Τα φίλτρα επιτρέπουν στα modules να τροποποιήσουν το περιεχόμενο που παράγεται από άλλα modules. Μπορούν να κρυπτογραφήσουν, να πραγματοποιήσουν ανιχνεύση για ιούς ή να συμπιέσουν όχι μόνο τα στατικά αρχεία αλλά και το δυναμικά δημιουργημένο περιεχόμενο.

Ζητήματα συμβατότητας : Αν και το module API είναι παρόμοιο μεταξύ των εκδόσεων, δεν είναι ακριβώς ίδια στις δύο εκδόσεις και έτσι τα modules του Apache 1.3 πρέπει να είναι στη νέα αρχιτεκτονική. Τα περισσότερα επικρατούντα modules όπως η PHP και mod_perl έχουν ήδη εκδόσεις Apache 2.0 και είναι τώρα μέρος του πακέτου του Server. Η εκτέλεση των modules σε μια threaded αρχιτεκτονική απαιτεί τις συγκεκριμένες αλλαγές σε αυτά. Τα modules που διανέμονται με τον Apache έχουν υποβληθεί στις απαραίτητες αλλαγές και θεωρούνται ασφαλή για την threaded αρχιτεκτονική, αλλά τα modules ή βιβλιοθήκες τρίτων δεν μπορούν. Εάν χρειάζεται αυτή η αρχιτεκτονική, η εκτέλεση του Apache γίνεται ως καθαρά process-based Server.

Εφαρμογή σε Windows και Unix συστήματα : Ο Apache εκτελείται εξίσου καλά στα Windows όσο στις πλατφόρμες Unix χάρη στη φορητή βιβλιοθήκη χρόνου εκτέλεσης (APR) που διαθέτει. Αφαιρεί τις διαφορές μεταξύ των λειτουργικών συστημάτων, όπως η πρόσβαση στα APIs αρχείων ή δικτύων. Η μεταφορά του Apache σε μια νέα πλατφόρμα αποτελεί πλέον πολύ απλή διαδικασία.

2.2.2 Εγκατάσταση

2.2.2.1. Εγκατάσταση apache σε συστήματα Unix

Απαιτήσεις : Οι ακόλουθες απαιτήσεις υπάρχουν για την οικοδόμηση του Apache:

Χρειάζονται περίπου 12 MB του προσωρινού ελεύθερου χώρου των δίσκων διαθέσιμα. Μετά από την εγκατάσταση ο Apache καταλαμβάνει περίπου 3 MB του δίσκου (ο πραγματικός απαραίτητος χώρος εξαρτάται από τα εγκατεστημένα modules κλπ...).

Είναι απαραίτητο να υπάρχει εγκατεστημένος ένας μεταγλωττιστής ANSI-C. Συστήνεται ο μεταγλωττιστής GNU C (GCC) από το ελεύθερο ίδρυμα λογισμικού (FSF). Η αρχική σελίδα του GNU είναι η < <http://www.gnu.org/> > όπου υπάρχουν περισσότερες πληροφορίες.

Εγκατάσταση : Οι διάφορες εκδόσεις του Apache υπάρχουν διαθέσιμες για εγκατάσταση στην σελίδα < <http://httpd.apache.org/download.cgi> > .

Αναλυτικές πληροφορίες σχετικά με την εγκατάσταση υπάρχουν στη σελίδα : <<http://httpd.apache.org/docs/install.html#apaci>>.

Εκτός από τον κύριο httpd Server ο Apache περιλαμβάνει διάφορα προγράμματα υποστήριξης. Αυτά δεν μεταγλωτίζονται εξ ορισμού. Τα προγράμματα υποστήριξης είναι στον κατάλογο υποστήριξης της διανομής. Για να μεταγλωτιστούν τα προγράμματα υποστήριξης, μπαίνουμε στον κατάλογο και πληκτρολογούμε “ make “.

Δοκιμή του πακέτου : Για να εκκινήσει ο Apache Server γράφουμε

```
$PREFIX/bin/apachectl start
```

και έπειτα εμφανίζεται η κεντρική του σελίδα μέσω του URL <http://localhost/> (όταν έχει εγκατασταθεί ο Apache ως root) ή <http://localhost:8080/> (όταν έχει εγκατασταθεί ο Apache ως κανονικός χρήστης). Ο Server τερματίζεται με την εντολή:

```
$ PREFIX/bin/apachectl stop
```

Όπου \$ PREFIX το path που έχει εγκατασταθεί ο Apache.

* Περισσότερες πληροφορίες για την πιο πρόσφατη έκδοση Apache μπορούν να βρεθούν στην ιστοσελίδα του Apache Web Server < <http://httpd.apache.org/>>.

2.2.2.2 Εγκατάσταση Apache σε συστήματα Windows

Απαιτήσεις : Ο Apache 1.3 είναι σχεδιασμένος να εκτελείται στα WINDOWS NT 4.0 και στα Windows το 2000. Ο Apache μπορεί επίσης να εκτελεστεί στα 95 και 98, αλλά αυτό δεν έχει δοκιμαστεί. Εάν εκτελείται σε NT 4.0, συστήνεται η εγκατάσταση του Service Pack 3 ή 6 .

Εγκατάσταση : Η τελευταία έκδοση του Apache υπάρχει διαθέσιμη για εγκατάσταση στη σελίδα < <http://httpd.apache.org/download.cgi> >. Μετα την εκτέλεση του αρχείου που κατεβάσατε αυτό εμφανίζει διάφορες επιλογές:

- Εάν θέλετε ή όχι να εκτελείται ο Apache για όλους τους χρήστες (εγκατάσταση Apache ως υπηρεσία), ή να εγκατασταθεί ώστε να εκτελείται σε ένα παράθυρο κονσόλας όταν επιλέγετε το εικονίδιο έναρξής του.
- Το όνομα του Server, το Domain name και διαχειριστικό λογαριασμό ηλεκτρονικού ταχυδρομείου.
- Κατάλογο για να εγκατασταθεί ο Apache (η προεπιλογή είναι C:\Program Files\Apache Group\Apache και μπορεί να αλλάξει σε οποιοδήποτε κατάλογο).
- Τύπο εγκατάστασης. Η "πλήρης" επιλογή τα εγκαθιστά όλα, συμπεριλαμβανομένου και του κώδικα πηγής εάν έγινε εγκατάσταση του πακέτου "src.msi".

Αναλυτικές πληροφορίες σχετικά με την εγκατάσταση υπάρχουν στη σελίδα :

<<http://httpd.apache.org/docs/windows.html>>.

Κατά τη διάρκεια της εγκατάστασης, ο Apache θα διαμορφώσει τα αρχεία στον "conf" κατάλογο, για τον επιλεγμένο κατάλογο εγκαταστάσεων. Εντούτοις εάν οποιαδήποτε από τα αρχεία σε αυτόν τον κατάλογο υπάρχουν ήδη δεν θα επικαλυφθούν.

Δοκιμή του πακέτου : Για να γίνει σύνδεση με τον Server και να δοκιμαστεί , ανοίγεται κάποιος browser και εισάγεται το URL:

```
http://localhost/
```

Αυτό πρέπει να αποκριθεί με μια ευπρόσδεκτη σελίδα, και μια σύνδεση με το εγχειρίδιο Apache. Εάν τίποτα δεν συμβαίνει ή εμφανίζει ένα λάθος τότε εάν ο host δεν συνδέεται σε δίκτυο, μπορεί να χρειάζεται να γίνει σύνδεση με αυτό το URL:

```
http://127.0.0.1/
```

Περισσότερες πληροφορίες σχετικά με την εγκατάσταση υπάρχουν στη σελίδα <<http://httpd.apache.org/docs/windows.html>>

2.2.2.3 Ασφάλεια

Ο Apache παρέχει διάφορες ενότητες σχετικές με την ασφάλεια για την εξασφάλιση και τον περιορισμό της πρόσβασης στον Server.

Πιστοποίηση : Τα Authentication modules επιτρέπουν τον καθορισμό της ταυτότητας ενός χρήστη, συνήθως με την επαλήθευση ενός ονόματος χρήστη και ενός κωδικού πρόσβασης . Ο Apache περιλαμβάνει modules για πιστοποίηση μέσα από αρχεία κειμένων και βάσεις δεδομένων.

Έλεγχος πρόσβασης : Ο Apache παρέχει το module mod_access που μπορεί να περιορίσει την πρόσβαση στους πόρους, βασιζόμενο στις παραμέτρους του αιτήματος των χρηστών, όπως η παρουσία μιας συγκεκριμένης επιγραφής ή της διεύθυνσης IP ή το hostname του πελάτη. Άλλα modules σας επιτρέπουν να περιορίσετε την πρόσβαση στους χρήστες που συμπεριφέρονται απρεπώς.

SSL/TLS : Τα ασφαλή πρωτόκολλα ασφάλειας "Sockets Layer/Transport Layer" επιτρέπουν στα στοιχεία μεταξύ του Web server και του πελάτη να κρυπτογραφηθούν. Στον Apache 1.3 τα πρωτόκολλα εφαρμόζονται από το mod_ssl, το οποίο διανέμεται χωριστά από το mod_ssl website <<http://www.modssl.org/>> και απαιτεί "patches" στον Server. Αυτό ήταν απαραίτητο λόγω των κανονισμών εξαγωγής στην κρυπτογράφηση. Οι περισσότεροι από εκείνους τους περιορισμούς έχουν ξεπεραστεί και αρχίζοντας από τον Apache 2.0 το mod_ssl συμπεριλαμβάνεται τώρα ως βασικό module του Apache.

2.2.3 Παραμετροποίηση

2.2.3.1 Εικονική φιλοξενία

Ο Apache παρέχει εκτενή υποστήριξη virtual hosting που σημαίνει ότι μπορεί να εξυπηρετήσει πολλαπλά websites από έναν Server. Στον Apache 2.0, με το ανά υποδιεργασία MPM μπορούν να υπάρχουν πολλαπλές υποδιεργασίες, με τη καθεμία να εξυπηρετεί ένα διαφορετικό domain κάτω από διαφορετικούς χρήστες Unix . Αυτό

είναι πολύ σημαντικό για την ασφάλεια στα κοινά host, δεδομένου ότι επιτρέπει να απομονώνονται οι χρήστες μεταξύ τους.

2.2.3.2 Πιστοποίηση

Basic authentication : Όταν μια συγκεκριμένη πληροφορία του συστήματος προστατευθεί χρησιμοποιώντας τη βασική επικύρωση και κάποιος προσπαθήσει να αποκτήσει πρόσβαση σε αυτή, ο Apache στέλνει μια πιστοποιητική επιγραφή "401" με την απάντηση στο αίτημα, ειδοποιώντας τον χρήστη ότι πρέπει να στείλει τις απαραίτητες πιστοποιήσεις για να επιστραφεί η πληροφορία όπως ζητείται.

Επάνω στη λήψη μιας απαντητικής επιγραφής "401", ο φυλομετρητής του πελάτη, εάν υποστηρίζει τη βασική πιστοποίηση, θα ζητήσει από το χρήστη να συμπληρώσει όνομα χρήστη και έναν κωδικό πρόσβασης για να σταλούν στον Server. Εάν χρησιμοποιείται ένας γραφικός φυλομετρητής, όπως Netscape ή Internet Explorer, θα εμφανιστεί ένα παράθυρο που ζητάει το όνομα χρήστη και τον κωδικό πρόσβασης, που στέλνονται πίσω στον Server. Εάν το όνομα χρήστη είναι στον εγκεκριμένο κατάλογο και εάν ο παρεχόμενος κωδικός πρόσβασης είναι σωστός, η πληροφορία θα επιστραφεί στον χρήστη.

Επειδή το πρωτόκολλο HTTP δεν έχει ταυτότητα, κάθε αίτημα θα αντιμετωπιστεί με τον ίδιο τρόπο, ακόμα κι αν είναι από τον ίδιο πελάτη. Δηλαδή για κάθε πληροφορία που ζητείται από τον Server θα πρέπει να παρέχονται τα πιστοποιητικά επικύρωσης ξανά προκειμένου να παραληφθεί η πληροφορία.

Ο φυλομετρητής φροντίζει τις λεπτομέρειες εδώ, έτσι ώστε να πρέπει να πληκτρολογηθεί το όνομα χρήστη και ο κωδικός πρόσβασης μία φορά ανα επίσκεψη - δηλαδή πρέπει να πληκτρολογηθεί πάλι την επόμενη φορά που ανοίγει ο φυλομετρητής και μπαίνει στον ίδιο ιστοχώρο.

Μαζί με την απάντηση "401", ορισμένες άλλες πληροφορίες θα περάσουν πίσω στον χρήστη. Ειδικότερα, στέλνεται ένα όνομα που συνδέεται με την προστατευόμενη ζώνη του ιστοχώρου. Αυτό καλείται η *σφαίρα*, ή απλά όνομα πιστοποίησης. Ο φυλομετρητής των χρηστών αποθηκεύει προσωρινά όνομα χρήστη και κωδικό πρόσβασης που συμπληρώθηκαν, και τα αποθηκεύει μαζί με το όνομα πιστοποίησης που είχε σταλεί από τον Server έτσι ώστε εάν άλλοι πόροι ζητούνται από το ίδιο όνομα, το ίδιο όνομα χρήστη και κωδικός πρόσβασης μπορούν να επιστραφούν για να επικυρώσουν το αίτημα, χωρίς απαίτηση να δακτυλογραφηθούν πάλι από τον χρήστη. Αυτή η προσωρινή αποθήκευση των κωδικών είναι συνήθως για την τρέχουσα σύνοδο φυλομετρητή, αλλά μερικοί φυλομετρητές επιτρέπουν να αποθηκευτούν μόνιμα, έτσι ώστε να μην είναι απαραίτητο να πληκτρολογηθεί ο κωδικός πρόσβασης πάλι.

Digest authentication : Η digest authentication εφαρμόζεται από το module `mod_auth_digest`. Εφαρμόζεται παρόμοια διαδικασία με την basic authentication. Χρησιμοποιώντας όμως την digest authentication, ο κωδικός πρόσβασης δεν στέλνεται ποτέ στο δίκτυο όπως είναι, αλλά διαβιβάζεται πάντα ως MD5 κωδικοποίηση του κωδικού πρόσβασης του χρήστη. Κατ' αυτό τον τρόπο, ο κωδικός πρόσβασης δεν μπορεί να διαβάσει απλά με την απορρόφηση και επεξεργασία της κυκλοφορίας δικτύων.

2.2.3.3 Εγκατάσταση των modules

Επιλέγουμε τα modules που θέλουμε να εκτελεστούν με τον Apache στο αρχείο διαμόρφωσης `src/Configuration.tmpl`. Οι γραμμές χωρίς σχόλια αντιστοιχούν στα προαιρετικά modules που μπορούν να συμπεριληφθούν (μεταξύ των γραμμών

AddModule στο τέλος του αρχείου), ή εισάγονται νέες γραμμές που αντιστοιχούν στα πρόσθετα modules. Καλό είναι να μην πειράζουμε τα modules προεπιλογής.

Στη συνέχεια κανουμε compile τον Apache με την εντολή make. Επιπλέον modules υπάρχουν στην σελίδα : <<http://www.apache.org/dist/httpd/contrib/modules/>>.

2.2.3.4 CGI-scripts

Η CGI(Common Gateway Interface) είναι το μέρος του Web Server που επικοινωνεί με άλλα προγράμματα που τρέχουν στον Server . Με την CGI, ο Web Server μπορεί να καλέσει ένα πρόγραμμα, περνώντας τα συγκεκριμένα στοιχεία κάθε χρήστη στο πρόγραμμα (αυτά που παρέχει ο χρήστης έχει χρησιμοποιώντας φόρμες HTML). Το πρόγραμμα έπειτα επεξεργάζεται τα στοιχεία και ο Server στέλνει την απάντηση του προγράμματος πίσω στον browser.

Προκειμένου να δουλέψουν σωστά τα προγράμματα CGI, πρέπει να διαμορφωθεί ο Apache να επιτρέπει την εκτέλεση CGI scripts. Υπάρχουν διάφοροι τρόποι να γίνει αυτό.

ScriptAlias : Η οδηγία ScriptAlias ενημερώνει τον Apache ότι ένας ειδικός κατάλογος υπάρχει κατά μέρος για τα προγράμματα CGI .Ο Apache θα υποθέσει ότι κάθε αρχείο σε αυτόν τον κατάλογο είναι ένα πρόγραμμα CGI, και θα προσπαθήσει να το εκτελέσει, όταν ζητείται εκείνο το συγκεκριμένο στοιχείο από έναν χρήστη.

Η εντολή ScriptAlias:

```
" ScriptAlias /cgi-bin/ /usr/local/apache/cgi-bin/ "
```

Παραδείγματος χάριν εάν τώρα ζητείται το URL <http://www.example.com/cgi-bin/test.pl> , ο Apache θα προσπαθήσει να εκτελέσει το αρχείο `usr/local/apache/cgi-bin/test.pl` και να επιστρέψει την έξοδο. Φυσικά το αρχείο θα πρέπει να υπάρχει να είναι εκτελέσιμο και επιστρέφει κάτι συγκεκριμένο αλλιώς ο Apache θα επιστρέψει λάθος.

Χρησιμοποίηση επιλογών : Μπορεί να γίνει διαμόρφωση των επιλογών, μέσα στο κύριο αρχείο διαμόρφωσης του Server, για να διευκρινιστεί ότι η εκτέλεση των CGI scripts επιτράπηκε σε έναν ιδιαίτερο κατάλογο:

```
<Directory /usr/local/apache/htdocs/somedir>  
    Options +ExecCGI  
</Directory>
```

Η ανωτέρω οδηγία λέει στον Apache να επιτρέψει την εκτέλεση των αρχείων CGI στον κατάλογο " somedir " .

2.3 Η γλώσσα προγραμματισμού PHP

2.3.1 Ιστορικά Στοιχεία

Η γλώσσα προγραμματισμού PHP (Hypertext Preprocessor) δημιουργήθηκε για να καλύψει τις ανάγκες των προγραμματιστών (Web Developers) να αναπτύξουν εφαρμογές για το Web και την δημιουργία δυναμικών σελίδων οι οποίες μπορούν εύκολα να ενσωματωθούν σε HTML σελίδες , μιας και η γλώσσα HTML που χρησιμοποιούταν για την κατασκευή Ιστοσελίδων (Web Sites) δεν είχε –και δεν έχει - αυτή την δυνατότητα (Static).

Η σύνταξη της γλώσσας αυτής παίρνει στοιχεία των C, Java και Perl και είναι σχετικά εύκολη στην μάθηση. Η PHP είναι διάδοχος μιας άλλης γλώσσας της

PHP/FI. Η PHP/FI η οποία δημιουργήθηκε από τον Ramsus Lerdorf το 1995, που αρχικά την χρησιμοποιούσε ως ένα απλό Perl script για τον έλεγχο πρόσβασης των χρηστών του διαδικτύου (Internet) στην σελίδα του βιογραφικού του. Ονόμασε αυτό το σύνολο από τα scripts 'Personal Home Page Tools'. Επειδή οι απαιτήσεις του σε λειτουργικότητα αυξάνονταν ο Ramsus έγραψε ένα μεγαλύτερο σετ κώδικα σε C, το οποίο είχε την δυνατότητα να επικοινωνεί με βάσεις και έδινε την δυνατότητα στους χρήστες να αναπτύσσουν δυναμικές εφαρμογές Web (Dynamic Web Applications).

Στην συνέχεια ο Ramsus έδωσε το πηγαίο κώδικα της PHP/FI, ώστε με την βοήθεια των χρηστών (προγραμματιστών) του διαδικτύου να μπορέσουν να την βελτιώσουν και να την αναβαθμίσουν. Η PHP/FI είχε κάποιες από τις βασικές λειτουργίες της σημερινής PHP. Είχε μεταβλητές που έμοιαζαν με μεταβλητές της Perl και την δυνατότητα της ενσωμάτωσης σε κώδικα HTML. Η σύνταξη της αν και πολύ πιο περιορισμένη και απλή έμοιαζε με αυτήν της Perl. Μέχρι το 1997, αναπτύχθηκε η PHP/FI v.2.0 η οποία είχε εγκατασταθεί σε περίπου 50,000 domains παγκοσμίως.

Η πρώτη έκδοση της PHP στην μορφή που την γνωρίζουμε σήμερα ήταν η PHP 3.0. Δημιουργήθηκε από τους Andi Gutmans και Zeev Suraski το 1997 όταν διαπιστώθηκε σε μια εργασία τους στο Πανεπιστήμιο τους, ότι με την PHP/FI 2.0 ήταν αδύνατο να δημιουργήσουν εφαρμογή ηλεκτρονικού εμπορίου (eCommerce Application).

Ένα από τα δυνατότερα σημεία της PHP 3.0 ήταν η επεκτασιμότητα. Με την βοήθεια εκατοντάδων προγραμματιστών (developers) μπόρεσαν και δημιούργησαν modules για την λειτουργία της PHP σε διαφορετικά υπολογιστικά συστήματα και βάσεις δεδομένων. Επίσης προστέθηκε και ακόμη μια ιδιότητα στην γλώσσα αυτή, της αντικειμενοστραφούς σύνταξης (object oriented syntax) που την έκαναν ακόμη πιο δυνατή και σταθερή. Η νέα αυτή γλώσσα ονομάστηκε PHP και δόθηκε για χρήση σε κάθε ενδιαφερόμενο.

Το γεγονός ότι η νέα αυτή γλώσσα μπορούσε να χρησιμοποιηθεί από κάθε χρήστη –διότι είναι ανοιχτού κώδικα λογισμικό με μεγάλες δυνατότητες προγραμματισμού, υπολογίζεται ότι περίπου 10,000 χρήστες και εκατοντάδες χιλιάδες ιστοσελίδες χρησιμοποιούσαν την γλώσσα αυτή μέχρι τα τέλη του 1998. Αυτό αντιστοιχεί στο σημαντικό ποσοστό του 10 % των Web Servers παγκοσμίως να χρησιμοποιούν την γλώσσα αυτή. Πρέπει να επισημάνουμε εδώ ότι επισήμως η PHP 3.0 δόθηκε για χρήση τον Ιούνιο του 1998, αφού πρώτα είχε δοκιμαστεί από τους χρήστες του διαδικτύου και του προγραμματιστές για 9 μήνες.

Μέχρι το Χειμώνα του ίδιου έτους, οι Andi Gutmans και Zeev Suraski άρχισαν να ξαναγράφουν τον πυρήνα (core) της PHP με σκοπό να βελτιώσουν την απόδοση περίπλοκων εφαρμογών που PHP 3.0 δεν μπορούσε να χειριστεί αποτελεσματικά. Έτσι στα μέσα του 1999 παρουσιάστηκε η PHP 4.0, η οποία χρησιμοποιούσε έναν νέο πυρήνα (core/engine) με όνομα "Zend Engine" (Zend- Zeev και Andi) και κάποια νέα χαρακτηριστικά. Τον Μάιο του 2000 δόθηκε επισήμως για χρήση, σχεδόν 2 χρόνια μετά την πρώτη έκδοση της PHP 3.0. Μέχρι και σήμερα αυτή είναι η έκδοση που χρησιμοποιούμε αν και έχει κάνει την εμφανισή της η νέα έκδοση PHP 5.0.

Τέλος αξίζει να αναφέρουμε ότι σήμερα είναι μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού για εφαρμογές Web. Το γεγονός ότι προσφέρεται χωρίς κόστος χρήσης την καθιστά υποψήφια να γίνει η πιο δημοφιλή γλώσσα για Web Developing στο άμεσο μέλλον.

2.3.2 Εγκατάσταση .

Εγκατάσταση σε Win32 Συστήματα

Ο Windows PHP installer, είναι διαθέσιμος στην σελίδα www.php.net . Κάνοντας χρήση αυτού του εργαλείου πετυχαίνουμε εύκολη εγκατάσταση της php στο σύστημα , όμως περιορίζεται σε πολλούς τομείς, όπως για παράδειγμα δεν υποστηρίζεται η αυτόματη ρύθμιση των extensions . Έτσι στην περίπτωση που χρησιμοποιείται για Web Server ο Apache πρέπει να γίνουν τροποποιήσεις στο αρχείο php.ini για την σωστή λειτουργία του Web Server καθώς και ρυθμίσεις στο apache.conf .

Εγκατάσταση σε Linux/Unix Συστήματα

Πολλές διανομές των Linux διαθέτουν κάποιο είδος εγκατάστασης μέσω πακέτων, όπως είναι το RPM . Αυτά μπορούν να βοηθήσουν να ρυθμιστεί μια κανονική εγκατάσταση, αλλά αν χρειαστεί ένα διαφορετικό σύνολο από χαρακτηριστικά (όπως για παράδειγμα έναν secure server), πιθανόν να χρειαστεί να γίνει build η PHP και πολλές φορές ο Server.

Ο πιο συνηθισμένος τρόπος εγκατάστασης της PHP σε Linux είναι να γίνει download ενός πακέτου της php από την τοποθεσία www.php.net και στην συνέχεια με τα ενσωματωμένα εργαλεία εγκατάστασης του Linux να εγκατασταθεί η php στο σύστημα σας.

2.3.3 Συνεργασία με τον Apache

Υπάρχουν δύο (2) τρόποι συνεργασίας της php με τον apache web server σε συστήματα Win32 και Linux/Unix . Αυτοί οι τρόποι είναι :

CGI binary
Apache module DLL

Εγκατάσταση ως Apache Module DLL

Για να εγκαταστήσουμε την php ως module πρέπει να εισάγουμε τις παρακάτω γραμμές ρυθμίσεων στο httpd.conf του Apache :

LoadModule php4_module "c:\php\sapi\php4apache2.dll
AddType application/x-httpd-php .php

Στην συνέχεια μετακινήστε το αρχείο php4ts.dll (που βρίσκεται στο φάκελο της php) στο winnt/system32 (για τα Win2000) ή στο windows/system32 (για τα WinXP), κάνοντας overwriting τυχόν παλαιότερο αρχείο.

Εγκατάσταση ως CGI binary

Ομοίως για να χρησιμοποιήσουμε την php να λειτουργεί ως CGI binary πρέπει να προσθέσουμε κάποιες γραμμές με ρυθμίσεις στο httpd.conf του Apache . Αυτές οι γραμμές φαίνονται παρακάτω

ScriptAlias /php/ "c:/php/"
AddType application/x-httpd-php .php
Action application/x-httpd-php "/php/php.exe"

Μετά την τοποθέτηση αυτών των γραμμών στο αρχείο httpd.conf, ξεκινήστε πάλι τον apache Web Server.

2.4 Το σύστημα διοικήσεως βάσεων δεδομένων MySQL

2.4.1 Ιστορικά στοιχεία

2.4.1.1 Δημιουργία της MySQL

Η MySQL ήταν αποτέλεσμα της προσπάθειας μιας ομάδας προγραμματιστών που χρησιμοποίησαν την mSQL για να συνδέσουν τους πίνακες χρησιμοποιώντας γρήγορες χαμηλού επιπέδου ρουτίνες (ISAM). Εντούτοις, μετά από δοκιμές, καταλήξαν στο συμπέρασμα ότι η mSQL δεν ήταν αρκετά γρήγορη ούτε αρκετά εύκαμπτη για τις απαιτούμενες ανάγκες. Αυτό οδήγησε σε μια νέα διεπαφή SQL με τη βάση δεδομένων αλλά σχεδόν την ίδια διεπαφή API με την mSQL. Αυτή η διεπαφή επιλέχτηκε για να επιτρέψει τον κώδικα τρίτων που γράφτηκε για τη χρήση σε mSQL να είναι συμβατός με MySQL.

Η παραγωγή του ονόματος MySQL δεν είναι σαφής. Ο κατάλογος της βάσης και ένας μεγάλος αριθμός βιβλιοθηκών και εργαλείων είχαν το πρόθεμα "my" για πάνω από 10 έτη. Εντούτοις, η κόρη ενός εκ των ιδρυτών Monty Widenius ονομάζεται επίσης "My".

Σήμερα η MySQL θεωρείται ένα από τα δημοφιλέστερα Ανοιχτού Κώδικα Λογισμικό (Open Source Software), Σύστημα Διαχείρισεως Βάσεων Δεδομένων (DBMS- Data Base Management System) στον κόσμο. Παρέχεται από την MySQL AB (www.mysql.org), η οποία έχει τα δικαιώματα του συστήματος αυτού και παρέχει υποστήριξη και άλλες υπηρεσίες (με αμοιβή) στους χρήστες της MySQL (συνήθως σε αυτούς που έχουν αγοράσει το προϊόν για εμπορικούς λόγους).

* Η mSQL(μίνι SQL) είναι ένα σύστημα βάσεων δεδομένων που σχεδιάστηκε για να παρέχει υψηλής ταχύτητας πρόσβαση σε μικρά σύνολα στοιχείων της βάσης . Ο αρχικός στόχος ήταν να εκτελούνται 100 βασικές διαδικασίες ανά δευτερόλεπτο σε έναν μέσο τερματικό σταθμό Unix σε μικρά σύνολα στοιχείων που χρησιμοποιούν πολύ λίγους πόρους συστήματος (δηλ. μνήμη και CPU).

2.4.1.2 Η MySQL και η Oracle

Η MySQL γενικά είναι γρήγορη, αξιόπιστη και εύκολη στην εγκατάσταση και στην διαχείριση . Η MySQL σχεδιάστηκε να διαχειρίζεται μεγάλες σε όγκο βάσεις δεδομένων και να παρέχει γρήγορα αποτελέσματα. Επίσης έχει ένα παρά πολύ καλό σύστημα πρόσβασης και ασφαλείας καθώς προσφέρει και συνδεσιμότητα (connectivity) που την κάνουν ιδανική για χρήση σε εφαρμογές Web. Επίσης δεν

απαιτεί μεγάλο αποθηκευτικό χώρο για την εγκατάσταση της, όπως η ORACLE που απαιτεί περίπου 1 Gb για μια τυπική εγκατάσταση. Σε σχέση με άλλες σχεσιακές βάσεις δεδομένων και ειδικότερα με την ORACLE έχει αρκετές διαφορές .

- Η MySQL επιτρέπει τις αλλαγές στον τύπο των πεδίων και θα προσπαθήσει να μετατρέψει τα στοιχεία. Στην Oracle μια παρόμοια αλλαγή απαιτεί κατάργηση και επαναδημιουργία πεδίων και στοιχείων.
- Διαφορές τύπων στοιχείων:
 - § Τύποι κειμένων: στην Oracle ένα VARCHAR μπορεί να είναι 4000 bytes, στην MySQL το όριο είναι 255. Οι Long text τύποι κειμένων λειτουργούν διαφορετικά στις δύο βάσεις.
 - § Τύποι ημερομηνίας: Η Oracle περιλαμβάνει την ώρα , η MySQL έχει χωριστά ημερομηνία, ώρα και τους αντίστοιχους τύπους. Οι ρουτίνες χειρισμού ημερομηνίας είναι διαφορετικές.
 - § Αριθμοί: Η Oracle έχει έναν αριθμητικό τύπο, "number" που κρατά έως 38 ψηφία. Η MySQL έχει πολλούς διαφορετικούς αριθμητικούς τύπους συμπεριλαμβανομένου του "int", του "decimal" και του "float" αλλά όχι "number".
 - § Αυτόματη αρίθμηση : Οι ακολουθίες της Oracle είναι χωριστά στοιχεία σε μια βάση δεδομένων και χρησιμοποιούνται για να ορίσουν τις τιμές στα πεδία. Στην MySQL το auto_increment είναι μια ιδιότητα ενός πεδίου και μπορεί να υπάρξει μόνο μια στήλη auto_increment ανά πίνακα.
- Στην αναζήτηση κειμένων στην Oracle διακρίνονται κεφαλαία και μικρά. Στην MySQL η προεπιλογή στην αναζήτηση κειμένου δεν διακρίνει κεφαλαία και μικρά.
- Η Oracle παρέχει πολύ καλή υποστήριξη στην αποκατάσταση της βάσης σε περίπτωση σφάλματος, δυνατότητα για transactions (Commit / Rollback) σε περίπτωση αδυναμίας του συστήματος μας , κλείδωμα στα ανώτατα επίπεδα, όρια συναλλαγών, καταγραφή συναλλαγών και προστασία ακεραιότητας συναλλαγών. Η MySQL παρέχει περιορισμένες δυνατότητες σε αυτούς τους τομείς πράγμα που την κάνει αρκετά επικίνδυνη όταν έχουμε να χειριστούμε ευαίσθητα δεδομένα όπως είναι τραπεζικές συναλλαγές .
- Επίσης η Oracle υποστηρίζει τα views, τα queries με sub-selects, τις αποθηκευμένες διαδικασίες, τα triggers, κ.λπ. ενώ η MySQL όχι.
- Η MySQL δεν παρέχει ενσωματωμένη γλώσσα προγραμματισμού , όπως PS/SQL ή Java που διαθέτει η ORACLE.

Γενικά έχουν παρατηρηθεί προβλήματα(σπάνια βέβαια) στη λειτουργία της MySQL σε περιπτώσεις που η βάση δεδομένων είναι εξαιρετικά πολύ μεγάλη .Θα λέγαμε ότι ORACLE συνιστάται κυρίως για πάρα πολύ μεγάλες εταιρίες που μετακινούν και χειρίζονται μεγάλο όγκο πληροφοριών και αξίζει να δαπανηθεί τόσο μεγάλο χρηματικό ποσό για την απόκτηση της άδειας χρήσης.

2.4.2 Εγκατάσταση

2.4.2.1 Εγκατάσταση σε Linux

Ο συνιστώμενος τρόπος να εγκατασταθεί η MySQL σε Linux είναι με τη χρησιμοποίηση RPM. Τα MySQL RPMs μπορούν να εγκατασταθούν σε όσες εκδόσεις Linux υποστηρίζουν τα rpm και τη χρήση glibc.

Τα πιο δημοφιλή RPM είναι:

- **MySQL-VERSION.i386.rpm** ο MySQL Server. Θα χρειαστείτε αυτό εκτός αν θέλετε μόνο να συνδεθείτε με έναν Server MySQL που τρέχει σε έναν άλλο υπολογιστή .
- **MySQL-client- VERSION.i386.rpm** τυποποιημένα προγράμματα πελατών MySQL.(για σύνδεση με Server MYSQL).

Οι παραπάνω εκδόσεις υπάρχουν στη σελίδα < www.mysql.com/downloads/ >.

2.4.2.2 Εγκατάσταση σε Windows

Η τελευταία έκδοση υπάρχει στη σελίδα

<www.mysql.com/downloads/mysql-3.2.23.html>

Τρέξτε το " setup.exe " για να αρχίσετε τη διαδικασία εγκαταστάσεων. Εάν θέλετε να εγκαταστήσετε σε άλλο κατάλογο από την προεπιλογή " c:", χρησιμοποιήστε το Browse κουμπί για να διευκρινίσετε τον προτιμημένο κατάλόγο σας.

Εάν θέλετε να χρησιμοποιήσετε πίνακες μεγαλύτερους από 4G, πρέπει να εγκαταστήσετε την MySQL σε ένα NTFS ή ένα νεότερο filesystem.

2.4.3 Συνεργασία MySQL και PHP

Η συνεργασία της PHP με την Βάση δεδομένων MySQL αποτελεί το πιο δημοφιλές εργαλείο ανάπτυξης δυναμικών ιστοσελίδων στο Internet και από ότι φαίνεται λειτουργεί άψογα . Μέσα από την PHP γίνεται η σύνδεση με τη βάση και στη συνέχεια παρέχονται πολλές δυνατότητες για επεξεργασία των πινάκων.

Έτσι όταν ένας χρήστης θελήσει κάποια στοιχεία από την βάση, ο browser του επικοινωνεί με τον Web Server, ο οποίος περνάει τα στοιχεία στη PHP που αναλαμβάνει να εκτελέσει το κατάλληλο Query στην MySQL. Μπορεί να ενσωματωθεί στην PHP οποιοδήποτε Query της MySQL μέσω σχετικών συναρτήσεων ,κάνοντας έτσι εύκολη την επεξεργασία των πινάκων της βάσης και την διαχείριση των στοιχείων της. Το μόνο που χρειάζεται να κάνει κάποιος είναι να χρησιμοποιήσει τις συναρτήσεις της PHP που επικοινωνούν με την MySQL.

mysql connect('localhost', 'username', 'password') : Συνδέει την php με την mysql. Η πρώτη παράμετρος είναι το hostname του MySQL Server . Η δεύτερη είναι το όνομα χρήστη και η τρίτη ο κωδικός πρόσβασης.

\$db = mysql select db('database name') : Επιλέγεται ποια βάση δεδομένων θα χρησιμοποιηθεί .

mysql query(\$query) : Εκτελείται το query. πχ \$query ="SELECT * FROM ΠΙΝΑΚΑ";

Παρέχονται πολλές συναρτήσεις για επεξεργασία των αποτελεσμάτων των Queries όπως π.χ :

mysql num rows(\$query) : Επιστρέφει τον αριθμό των αποτελεσμάτων του Query.

mysql fetch row(\$query) : Επιστρέφει έναν πίνακα με στοιχεία ,τα στοιχεία της γραμμής του πίνακα της βάσης που αντιστοιχούν στο συγκεκριμένο Query.

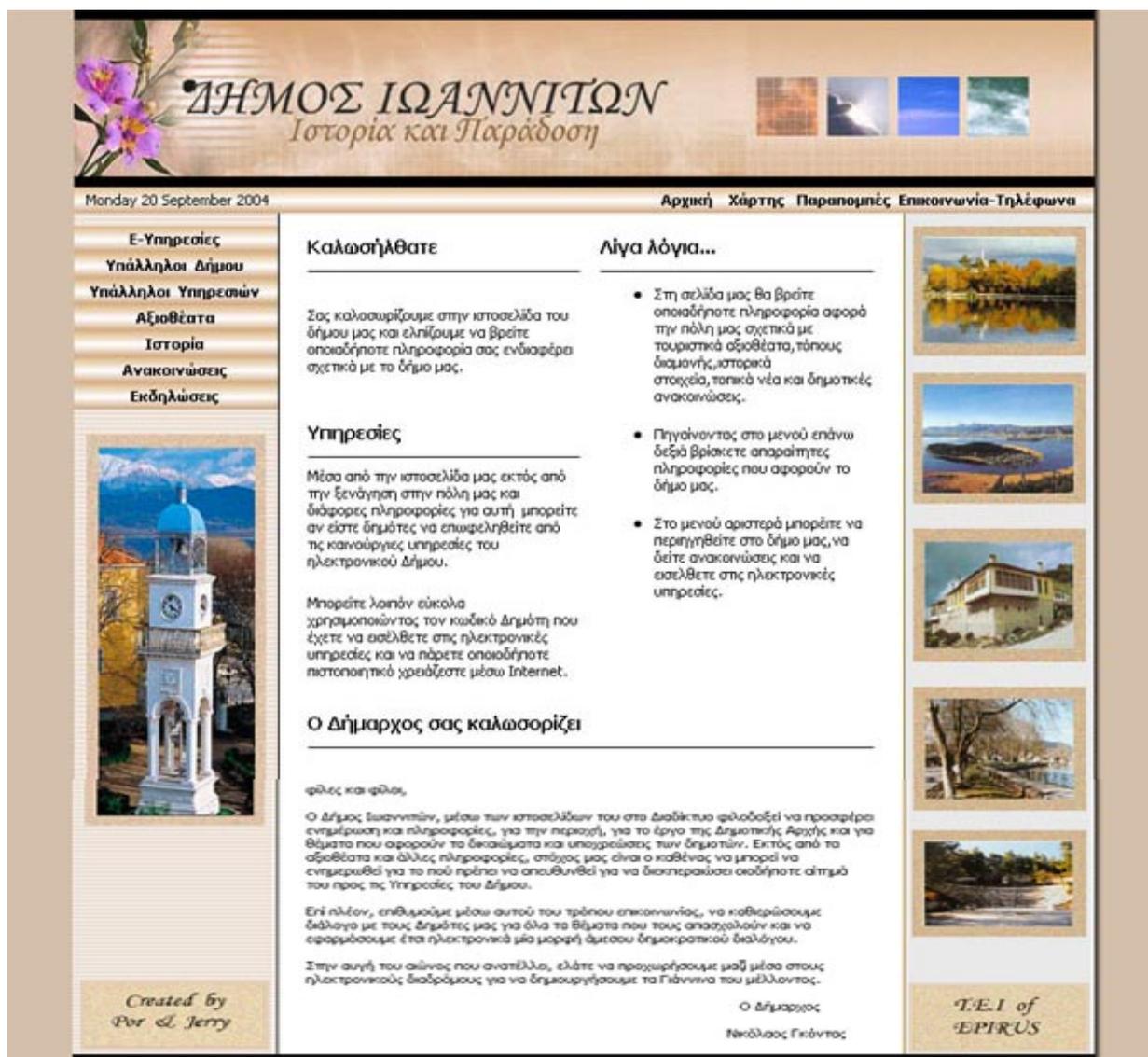
3. Οργάνωση εφαρμογής

3.1 Κεντρική σελίδα

Αρχικά να επισημάνουμε ότι αυτή η εφαρμογή δημιουργήθηκε για να προσαρμοστεί στο site οποιουδήποτε Δήμου το επιθυμεί αλλάζοντας ελάχιστα έως καθόλου την υπάρχουσα ιστοσελίδα του .

Κατασκευάστηκε λοιπόν μια ιστοσελίδα του Δήμου Ιωαννιτών όπου προσαρμόστηκε η εφαρμογή όπως θα προσαρμοζόταν σε οποιαδήποτε ιστοσελίδα άλλου Δήμου. Η ιστοσελίδα αυτή περιέχει τα απαραίτητα στοιχεία και εμφάνιση που πιστεύουμε ότι χρειάζεται να έχει μια σύγχρονη αλλά ταυτόχρονα απλή και εύκολη στη πλοήγηση σελίδα που ταιριάζει σε έναν Δήμο. Η σελίδα έχει τη δομή και τη λειτουργικότητα που θα απαιτούνταν αν ήταν πράγματι κεντρική σελίδα του Δήμου απλά δεν έχει προστεθεί ο απαραίτητος όγκος πληροφοριών που θα ήταν αναγκαίος για να καλύψει όλο το φάσμα του Δήμου. Και αυτό γιατί σκοπός μας δεν ήταν να κατασκευάσουμε την σελίδα του Δήμου αλλά να δείξουμε πως η εφαρμογή “e-Dimos” θα προσαρμοζόταν και θα δούλευε στην ιστοσελίδα.

Περνάμε στη κεντρική σελίδα η οποία φαίνεται στην εικόνα(Σχήμα 3) :



Σχήμα 3(Κεντρική σελίδα)

Έτσι όποιος επισκευθεί υποτιθέμενα το URL < www.ioannina.gr > θα βρεθεί σε αυτή τη σελίδα από όπου μπορεί εύκολα να αναζητήσει την πληροφορία που θέλει.

Η κεντρική σελίδα όπως βλέπετε παρέχει απευθείας πληροφορίες όπως η ημερομηνία και ευανάγνωστες διασυνδέσεις(links) που οδηγούν σε περισσότερο υλικό.Υπάρχει ένα ενημερωτικό κείμενο για το τι μπορεί να κάνει κάποιος μέσα απ'τη σελίδα. Επίσης υπάρχει το καλοσώρισμα του Δημάρχου προς τους επισκέπτες. Έτσι μπορεί οποιοσδήποτε να περιηγηθεί αναζητώντας τουριστικές πληροφορίες όπως αξιοθέατα, χάρτες κλπ.Υπάρχουν όμως και κάποιοι τομείς που είναι για τους δήμοτες και υπαλλήλους του Δήμου. Ακολουθεί επιμέρους ανάλυση στη συνέχεια.

3.2 Ενημερωτικό μέρος - Είσοδος σαν επισκέπτης

Υπάρχουν οι ακόλουθες διασυνδέσεις για όποιον θέλει να περιηγηθεί στη σελίδα είτε είναι δημότης ή απλώς επισκέπτης.

Αξιοθέατα : Εδώ υπάρχουν πληροφορίες και φωτογραφίες για τα κυριότερα αξιοθέατα της πόλης .

Ιστορία : Τα κυριότερα ιστορικά στοιχεία και γεγονότα για την πόλη από τα αρχαία χρόνια εως σήμερα.

Ανακοινώσεις : Εδώ αναρτούνται τα κυριότερα θέματα επικαιρότητας σχετικά με το Δήμο καθώς και χρήσιμες ανακοινώσεις που αφορούν τους πολίτες όπως προγραμματισμένες διακοπές ηλεκτρικού ρεύματος ,νερού κλπ.

Εκδηλώσεις : Οι προγραμματισμένες πολιτιστικές εκδηλώσεις (συναυλίες, φεστιβάλ, εκθέσεις) που θα γίνουν στη πόλη.

Χάρτης : Ο πανοραμικός χάρτης της πόλης από όπου μπορεί κάποιος να δει αναλυτικά τους δρόμους και να κατευθυνθεί στον προορισμό του.

Παραπομπές : Άλλες ιστοσελίδες που περιέχουν πληροφορίες σχετικά με το Δήμο και την γύρω περιοχή.

Επικοινωνία : Εδώ θα υπάρχουν τηλέφωνα για τις κυριότερες υπηρεσίες του Δήμου καθώς και όλα τα χρήσιμα τηλέφωνα διαφόρων φορέων και υπηρεσιών της πόλης.

Επίσης πατώντας στα κουτάκια “ Created by.. “ και “T.E.I of EPIRUS “ στο τέλος της σελίδας στο μεν πρώτο εμφανίζει μια φόρμα για αποστολή email στους κατασκευαστές της σελίδας και το δεύτερο πηγαίνει στη κεντρική σελίδα του T.E.I Ηπείρου. Έτσι αν θέλει κάποιος μπορεί να επικοινωνήσει μαζί μας για οποιαδήποτε παρατηρήση έχει σχετικά με τη σελίδα ή οτιδήποτε άλλο μας ενδιαφέρει .

3.3 Είσοδος σαν Δημότης

Με τις e-Υπηρεσίες ο Δήμος αποκτά ένα νέο σύγχρονο πρόσωπο κάνοντας την αρχή στην παροχή διαδικτυακών υπηρεσιών . Καθώς οι χρήστες του διαδικτύου αυξάνονται κατακόρυφα στη χώρα μας και η χρήση των ηλεκτρονικών υπολογιστών αποτελεί πλέον κομμάτι της καθημερινότητας οι δημότες γρήγορα θα αποδεχτούν και θα χρησιμοποιήσουν την νέα υπηρεσία.Το μόνο που έχει να κάνει ο Δήμος είναι να γνωστοποιήσει την νέα υπηρεσία στους δημότες μέσα από τα τοπικά μέσα ενημέρωσης προβάλλοντας την απλότητα της διαδικασίας και πως οποιοσδήποτε εφόσον είναι δημότης μπορεί να αποκτήσει τη δυνατότητα να χρησιμοποιήσει τις ηλεκτρονικές υπηρεσίες.

e-Υπηρεσίες

Η πρόσβαση θα είναι εφικτή για κάποιον εφόσον αποκτήσει τον κωδικό δημότη που καταχωρείται στο ονομά του τη στιγμή που θα τον ζητήσει. Τον κωδικό μπορεί να τον καταχωρήσει ή να τον αλλάξει μόνο ο διαχειριστής του συστήματος όπως

παρουσιάζεται αναλυτικά παρακάτω. Τον κωδικό μπορεί να τον παραλάβει κάποιος ταχυδρομικώς ή στα γραφεία του Δήμου, επιδεικνύοντας την αστυνομική του ταυτότητα. Θα του δοθεί ένα έντυπο με τα ακριβή στοιχεία που πρέπει να συμπληρώσει για να εισέλθει στις ηλεκτρονικές υπηρεσίες. Σε περίπτωση που κάποιος τα ξεχάσει ή καταλάβει ότι τα έχουν μάθει και άλλοι μπορεί να απευθυνθεί πάλι στο Δήμο για να του δώσει καινούργιο κωδικό.

Αυτό το κομμάτι αποτελεί το σημαντικότερο της εφαρμογής καθώς ενσωματώνει την διεπαφή της εφαρμογής με τους δημότες. Προσφέρει μια εντελώς νέα υπηρεσία στα χρονικά των Ελληνικών Δήμων που σκοπό έχει την διευκόλυνση των πολιτών και την καλύτερη λειτουργία του Δήμου. Πλέον ο πολίτης θα μπορεί να κάνει κάποιες γραφειοκρατικές συναλλαγές με το Δήμο μέσω του Διαδικτύου παρακάπτοντας έτσι πολλές χρονοβόρες διαδικασίες που κοστίζουν σε αυτόν και τον Δήμο.

Το μόνο που έχει να κάνει κάποιος είναι να μπει στη σελίδα του Δήμου και να πατήσει το link “ Ε-Υπηρεσίες ” που θα τον οδηγήσει στη σελίδα εισόδου που φαίνεται παρακάτω(Σχήμα 4) :

Σχήμα 4 (Είσοδος στις ηλεκτρονικές υπηρεσίες)

Από εδώ μπορεί κάποιος, συμπληρώνοντας τα στοιχεία του και τον κωδικό που έχει παραλάβει από τον Δήμο, να εισέλθει στις ηλεκτρονικές υπηρεσίες εφόσον τα στοιχεία που θα συμπληρώσει είναι σωστά . Αν όχι το σύστημα τον προτρέπει να τα ξανα-συμπληρώσει. Για οποιαδήποτε προβλήματα στη συμπλήρωση των στοιχείων η σημείωση από κάτω προτρέπει για είσοδο στην σελίδα που περιέχει πληροφορίες για το τι πρέπει να κάνει κάποιος όταν έχει πρόβλημα ή πως θα αποκτήσει κωδικό πρόσβασης(δίνεται και τηλέφωνο για περισσότερες πληροφορίες). Οι οποιοσδήποτε αποτυχημένες ή επιτυχημένες προσπάθειες εισόδου καταγράφονται στο σύστημα.

Όταν κάποιος εισάγει στοιχεία στη φόρμα εισόδου και τα αποστέλλει εκτός του ότι καταγράφονται, οι εγγραφές περνούν πρώτα από έλεγχο για την αποφυγή εισαγωγής κακόβουλων στοιχείων(html tags, java tags) που μπορεί να έχουν ως αποτέλεσμα κατάρρευση του συστήματος και της βάσης δεδομένων της εφαρμογής.

Τα στοιχεία που πρέπει να συμπληρώσει κάποιος για να εισέλθει όπως φαίνεται και στο σχήμα είναι Όνομα, Επίθετο, Όνομα Πατρός και κωδικός. Εφόσον εισαχθούν σωστά στοιχεία τότε το σύστημα συνδέεται με την κεντρική σελίδα των υπηρεσιών που φαίνεται παρακάτω(Σχήμα 5) :

ΔΗΜΟΣ ΙΩΑΝΝΙΝΩΝ
Ιστορία και Παράδοση

Thursday 30 September 2004 Εξοδος

Καλωσήθατε στις ηλεκτρονικές υπηρεσίες

Αγαπητοί Δημότες από εδώ μπορείτε να εκτυπώσετε το πιστοποιητικό που σας ενδιαφέρει ή απλά να δείτε τα στοιχεία του. Πατώντας σε ένα από τα παρακάτω πιστοποιητικά θα μπορέσετε να δείτε τα στοιχεία σας όπως είναι καταχωρημένα στον Δήμο μας. Στη συνέχεια πατώντας στο πλήκτρο " Εκτύπωση " μπορείτε να εκτυπώσετε αν διαθέτετε εκτυπωτή.

Προσοχή! Όταν πατήσετε το πλήκτρο " Εκτύπωση " το έγγραφο γίνεται επίσημο έγγραφο του Δήμου και παίρνει αριθμό πρωτοκόλλου. Μπορείτε να πάρετε **εώς 3** επίσημα πιστοποιητικά από κάθε είδος το μήνα.

- [Πιστοποιητικό Γέννησης](#)
- [Πιστοποιητικό Οικογενειακής Κατάστασης](#)
- [Π.Γέννησης Δελτίου Αστυνομικής Ταυτότητας](#)
- [Πιστοποιητικό Κληρονόμων](#)
- [Παρατηρήσεις προς το Δήμο](#)

* Το πιστοποιητικό που θα πάρετε μπορείτε να το προσκομίσετε σε οποιαδήποτε υπηρεσία το χρειάζεται. Ο υπάλληλος που θα το παραλάβει θα ελέγξει την εγκυρότητά του απλά συμπληρώνοντας στον αντίστοιχο τομέα της σελίδας μας τον αριθμό πρωτοκόλλου και αριθμό εγγράφου του πιστοποιητικού.

Σχήμα 5 (Κεντρική σελίδα ηλεκτρονικών υπηρεσιών).

Τα κείμενα που υπάρχουν στη σελίδα ενημερώνουν τους Δημότες για το τι μπορούν να κάνουν μέσα από αυτή τη σελίδα. Συγκεκριμένα μπορούν απλά να δούν τα στοιχεία τους ή να εκτυπώσουν πιστοποιητικά Γέννησης, δελτίου Αστυνομικής ταυτότητας, Οικογενειακής κατάστασης, Κληρονόμων καθώς και να διατυπώσουν επώνυμα τα παραπονά τους προς το Δήμο.

Έτσι πατώντας πάνω σε κάποιο πιστοποιητικό εμφανίζεται το πιστοποιητικό με όλα τα στοιχεία επάνω εκτός από αριθμό πρωτοκόλλου και αριθμό εγγράφου. Αν θέλει κάποιος λοιπόν βλέπει απλώς τα στοιχεία του. Αν όμως θελήσει να εκτυπώσει το πιστοποιητικό και αυτό να αποτελεί νόμιμο έγγραφο του Δήμου τότε όπως επισημαίνεται και στη σελίδα χρειάζεται να πατήσει το πλήκτρο εκτύπωση στο τέλος τη σελίδας οπότε αυτόματα το πιστοποιητικό παίρνει αριθμό πρωτοκόλλου και αριθμό εγγράφου που το καθιστούν νόμιμο και έτοιμο για χρήση. Επίσης ανοίγει αυτόματα το πλαίσιο διαλόγου για επιλογή εκτυπωτή.

Τα στοιχεία αριθμός Πρωτοκόλλου και Εγγράφου καταχωρούνται στη βάση μαζί με άλλα στοιχεία ώστε να δείχνουν το συγκεκριμένο πιστοποιητικό του Δημότη που το παρέλαβε. Με άλλα λόγια το πιστοποιητικό καταχωρείται κατά κάποιον τρόπο στη βάση. Τα στοιχεία αυτά θα χρησιμοποιηθούν αργότερα όπως θα δούμε παρακάτω από υπαλλήλους άλλων υπηρεσιών που θα καταλήξει το πιστοποιητικό. Κάθε δήμοτης έχει δικαίωμα να πάρει μέχρι τρία νόμιμα πιστοποιητικά το μήνα που σημαίνει ότι μπορεί να πατήσει εκτύπωση μόνο όταν χρειάζεται ένα έγκυρο πιστοποιητικό και όχι απλά αν θέλει να εκτυπώσει τα στοιχεία του. Αν έχει συμπληρωθεί αυτό το όριο τότε το σύστημα του εμφανίζει σχετικό μήνυμα.

Πατώντας στον σύνδεσμο *Παράπονα προς το Δήμο* εμφανίζεται μια φόρμα όπου ο Δημότης μπορεί να γράψει οποιαδήποτε παρατήρηση ή παράπονα προς το Δήμο και να

τα στείλει. Τα κείμενα καταχωρούνται σε πίνακα μαζί με τα στοιχεία του Δημότη και τα βλέπει ο αρμόδιος υπάλληλος. Επίσης δίνεται και τηλέφωνο σε περίπτωση που κάποιος θέλει να κάνει τηλεφωνική καταγγελία-παρατήρηση.

Εφόσον ο δημότης έχει τελειώσει τις εργασίες του στις e-Υπηρεσίες πατώντας τον σύνδεσμο *Έξοδος* το σύστημα τον βγάζει από τις υπηρεσίες και τον πηγαίνει στο login(σελίδα εισόδου στο σύστημα). Ακόμη για μεγαλύτερη ασφάλεια, όταν ο χρήστης δεν χρησιμοποιεί το σύστημα ενώ έχει εισέλθει στις υπηρεσίες για χρονικό διάστημα μεγαλύτερο των δέκα(10) λεπτών, αν πατήσει κάτι το σύστημα τον βγάζει αυτόματα από τις υπηρεσίες και τον πηγαίνει στο login.

3.4 Είσοδος σαν Υπάλληλος

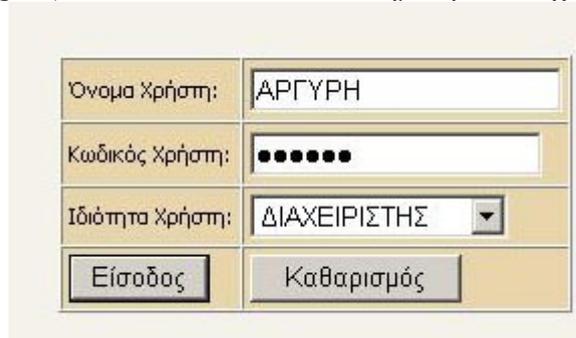
Οι υπάλληλοι των Υπηρεσιών του Δήμου χωρίζονται σε δύο (2) κατηγορίες δικαιωμάτων :

- Ø Οι υπάλληλοι που έχουν δικαίωμα προϊστάμενου
- Ø Απλοί υπάλληλοι

(Στην συγκεκριμένη εφαρμογή έχει αναπτυχθεί η υπηρεσία του Δημοτολογίου .)

Για την ασφάλεια του συστήματος και την πρόληψη τυχόν παραβιάσεων , η υπηρεσία δεν είναι προσβάσιμη από χρήστες που δεν έχουν δηλώσει την IP Address τους στο σύστημα. Έτσι οι υπάλληλοι πρέπει πρώτα να δηλώσουν την IP Address – αυτή την λειτουργία όπως θα δούμε αργότερα την αναλαμβάνει ο διαχειριστής του συστήματος- ώστε να έχουν πρόσβαση στο σύστημα . Επίσης πρέπει να πληκτρολογήσουν το όνομα χρήστη και τον κωδικό πρόσβασης που τους παρέχονται από τον προϊστάμενο του τμήματος.

Επίσης για την ασφαλέστερη λειτουργία, όταν ο χρήστης/υπάλληλος του συστήματος δεν χρησιμοποιεί το σύστημα ενώ έχει εισέλθει στις υπηρεσίες για χρονικό διάστημα μεγαλύτερο των δεκαπέντε (15) λεπτών , τότε τον βγάζει αυτόματα από το σύστημα και πηγαίνει στο login (σελίδα εισόδου στο σύστημα- βλέπε Σχ.6) .



(Σχήμα 6)

Όπως προαναφέραμε οι υπάλληλοι χωρίζονται σε απλούς υπαλλήλους και σε προϊστάμενους, με ξεχωριστές λειτουργίες ο καθένας .

Προϊστάμενος

Οι υπάλληλοι που έχουν δικαίωμα προϊστάμενου για να κάνουν είσοδο στο σύστημα πρέπει να δώσουν το κατάλληλο όνομα χρήστη (username) και συνθηματικό (password) καθώς και να επιλέξουν από την λίστα την ιδιότητα «ΠΡΟΪΣΤΑΜΕΝΟΣ» (βλέπε Σχ.7)

Χρήστης : ΙΩΑΝΝΗΣ ΧΗΡΑΣ			
Υπηρεσία Δημοτολογίου			
<u>Παρουσίαση Δημογραφικών Στοιχείων</u>		<u>Πληθυσμιακή Κατανομή</u>	
<u>Γεννήσεων</u>		<u>Ανά Φύλο</u>	
<u>Θανάτων</u>		<u>Δημοτικό Διαμέρισμα</u>	
<u>Γάμων</u>		<u>Ηλικία</u>	
<u>Διαζυγίων</u>			
Διαχείριση Τμήματος			
<u>Πρόσληψη Υπαλλήλου</u>			
<u>Μεταβολή Κατάστασης Υπαλλήλου</u>			
Σημ : Τα στοιχεία πρέπει να συμπληρώνονται με ΚΕΦΑΛΑΙΑ ΓΡΑΜΜΑΤΑ			

(Σχήμα. 7)

Το σύστημα αφού ελέγξει τα στοιχεία που έχουν πληκτρολογήσει με τα στοιχεία που είναι αποθηκευμένα στην βάση δεδομένων του Δήμου επιτρέπει ή απορρίπτει την είσοδο του/της.

Αφού επιτραπεί η πρόσβαση του προϊστάμενου του τμήματος δημοτολογίου μεταφέρεται στην αρχική σελίδα του τμήματος . Είναι λογικό ο προϊστάμενος του τμήματος να ενδιαφέρεται για διαφορετικές πληροφορίες από αυτές ενός υπαλλήλου, έτσι και το κεντρικό μενού είναι διαφορετικό από αυτό του υπαλλήλου του δημοτολογίου (βλέπε σχ.2)

Οι λειτουργίες και οι πληροφορίες που μπορεί να πάρει ο προϊστάμενος είναι :

- I. Παρουσίαση Δημογραφικών Στοιχείων
 - a. Γεννήσεων
 - b. Θανάτων
 - c. Γάμων
 - d. Διαζυγίων
- II. Παρουσίαση Κατανομής Πληθυσμού
 - a. Φύλο
 - b. Δημοτικό Διαμέρισμα
 - c. Ηλικία
- III. Διαχείριση Υπαλλήλων Τμήματος
 - a. Πρόσληψη
 - b. Μεταβολή Κατάστασης Υπαλλήλου

Ανάλυση Λειτουργιών

I. Παρουσίαση Δημογραφικών Στοιχείων

Σημείωση:

1. Για την παρουσίαση των στοιχείων αυτών γίνεται χρήση μιας φόρμας εισαγωγής ημερομηνιών αρχικής και τελικής χρονολογίας .

2. Σε περίπτωση που ο χρήστης δώσει λάθος ημερομηνίες δηλαδή η αρχική ημερομηνία είναι μεγαλύτερη από την τελική τότε η ενέργεια αυτή καταγράφεται στον πίνακα συμβάντων της βάσης δεδομένων.

a. Γεννήσεων

Ο προϊστάμενος αφού δώσει το χρονολογικό διάστημα για το οποίο ζητά τις γεννήσεις που θέλει, γίνεται αναζήτηση στην βάση και συγκεκριμένα στο πίνακα residents (δημότες) τους δημότες οι οποίοι έχουν γεννηθεί στο διάστημα αυτό .

Στην συνέχεια παρουσιάζονται τα στοιχεία του ονόματος, επωνύμου, όνομα πατρός, όνομα μητρός , γένος και ημερομηνία γεννήσεως , με αύξουσα σειρά γεννήσεως.

b. Θανάτων

Ομοίως έχουμε εδώ την παρουσίαση των θανάτων κατά την χρονολογική περίοδο που δώσαμε, με τα στοιχεία του ονοματεπωνύμου του θανούντα καθώς την ημερομηνία θανάτου και τον αριθμό ληξιαρχικής πράξης.

c. Γάμων

Εδώ γίνεται παρουσίαση των γάμων που έχουν τελεστή κατά την περίοδο που θέσαμε . Η αναζήτηση εδώ γίνεται σε έναν άλλο πίνακα της βάσεως μας marriage οπου εκεί βρίσκονται οι καταχωρήσεις των γάμων .Τα στοιχεία που εμφανίζονται εδώ είναι τα μοναδικά κλειδιά των δημοτών (*resid male* , *resid female*) και την ημερομηνία τελέσεως του γάμου .

d. Διαζυγίων

Εδώ η αναζήτηση των διαζυγίων γίνεται και αυτή στον πίνακα marriage στο πεδίο *date_broken* . Αν το πεδίο αυτό έχει τιμή τότε σημαίνει ότι έχει προκύψει διαζύγιο στον συγκεκριμένο γάμο. Τα στοιχεία που μας παρουσιάζονται εδώ είναι τα μοναδικά κλειδιά των δημοτών (*resid male* , *reside female*) καθώς και η ημερομηνία που εκδόθηκε το διαζύγιο.

II. Παρουσίαση Κατανομής Πληθυσμού

a. Φύλο

Εδώ έχουμε συγκεντρωτική παρουσίαση σε όλο τον δήμο, το σύνολο του πληθυσμού συγκεντρωτικά αλλά και ανά φύλο (άνδρες/γυναίκες) . Επίσης παρουσιάζεται και το ποσοστό που καταλαμβάνουν τα δύο (2) φύλα στο δήμο.

b. Δημοτικό Διαμέρισμα

Εδώ έχουμε παρουσίαση καταμερισμού του πληθυσμού ανά δημοτικό διαμέρισμα . Ο χρήστης πρέπει να επιλέξει για πιο δημοτικό διαμέρισμα θέλει να πάρει πληροφορίες και αφού κάνει την επιλογή του , εμφανίζεται η κατανομή του πληθυσμού στο συγκεκριμένο διαμέρισμα.

c. Ηλικία

Στην περίπτωση αυτή έχουμε έξι (6) κατηγορίες ηλικιών οι οποίες είναι

Κατηγορία	Από	Εώς
Παιδιά	0	13
Έφηβοι	14	18
Νέοι/Νέες	19	25
Ενήλικες	26	37
Μεσήλικες	38	49
Ηλικιωμένοι	50	Και άνω (π.χ, 130)

Αφού γίνει η επιλογή της κατηγορίας που θέλει ο χρήστης γίνεται παρουσίαση των ονομάτων, επωνύμων και ημερομηνιών γεννήσεων των δημοτών που ανήκουν στην κατηγορία αυτή .

Παρατήρηση :

Όπως φαίνεται από τα στοιχεία που μπορεί να συλλέξει ο προϊστάμενος του τμήματος δημοτολογίου, οι στατιστικές αυτές είναι ιδιαίτερα σημαντικές για την σωστή εξυπηρέτηση των δημοτών και την προσφορά υπηρεσιών . Για παράδειγμα αν τα στατιστικά στοιχεία των ηλικιών δείξουν ότι έχουμε αύξηση της κατηγορίας των

παιδιών σε σχέση με τις άλλες κατηγορίες, μπορεί αυτό να αξιοποιηθεί από τον δήμο με την δημιουργία περισσότερων παιδικών σταθμών ή αντιθέτως αν τα άτομα που ανήκουν στην κατηγορία των ηλικιωμένων είναι περισσότερα ίσως να χρειάζεται να δημιουργηθούν περισσότερα Κ.Α.Π.Η. (Κέντρα Ανοιχτής Προστασίας Ηλικιωμένων).

III. Διαχείριση Υπαλλήλων Τμήματος

Ο προϊστάμενος κάθε τμήματος είναι υπεύθυνος για την διαχείριση του ανθρώπινου δυναμικού του. Έτσι μπορεί :

A. Να κάνει πρόσληψη ενός υπαλλήλου

Η πρώτη ενέργεια που πρέπει να κάνει είναι, να επιλέξει αν ο υπάλληλος που προτίθεται να προσλάβει είναι μόνιμος ή συμβασιούχος .

Στην περίπτωση τώρα που είναι μόνιμος ο χρήστης πρέπει να συμπληρώσει τα ακόλουθα στοιχεία : *Όνομα, Επώνυμο, Αριθμό Φορολογικού Μητρώου (Α.Φ.Μ.). Επίσης στοιχεία που αφορούν τον τόπο κατοικίας του και τέλος τα στοιχεία που αφορούν την εργασία του καθώς και το δικαίωμα που έχει στο σύστημα δηλαδή προϊστάμενος ή υπάλληλος.* Αφού συμπληρωθούν τα στοιχεία αυτά –τα οποία είναι όλα υποχρεωτικά- το σύστημα ελέγχει για πιθανή ίδια εγγραφή και ενημερώνει τον χρήστη για την επιτυχία ή μη της ενέργειας του.

Στην δεύτερη (2^η) περίπτωση που ο υπάλληλος είναι συμβασιούχος εκτός από τα παραπάνω στοιχεία ο χρήστης πρέπει να δηλώσει και *την ημερομηνία πρόσληψης καθώς και την ημερομηνία λήξης της σύμβασης.* Ομοίως γίνεται έλεγχος των στοιχείων αυτών και καταχώρηση στην βάση δεδομένων του δήμου.

B. Μεταβολή Κατάστασης Υπαλλήλου

Με τον όρο αυτό εννοούμε την δυνατότητα επεξεργασίας της σύμβασης ενός υπαλλήλου με το να μπορεί να κάνει *Ανανέωση Σύμβασης, Αλλαγή Μισθού ή Απόλυση ενός συμβασιούχου υπαλλήλου.* Έτσι με τον τρόπο αυτό ο προϊστάμενος του τμήματος καθίσταται αποκλειστικός υπεύθυνος για την σωστή λειτουργία και οργάνωση της υπηρεσίας που ανήκει .

Υπάλληλος

Σημείωση : Πριν ξεκινήσουμε να αναφέρουμε τις δυνατότητες ενός υπαλλήλου πρέπει να αναφέρουμε ότι στην συγκεκριμένη εφαρμογή έχει αναπτυχθεί το τμήμα Δημοτολογίου. Οπότε όταν αναφερόμαστε σε υπαλλήλους τμημάτων θεωρούμε ότι αναφερόμαστε στους υπαλλήλους της υπηρεσίας Δημοτολογίου.

Οι υπάλληλοι αφού πληκτρολογήσουν και αυτοί το κατάλληλο όνομα χρήστη και το συνθηματικό (που είναι διαφορετικό για τον καθένα), μεταφέρονται στο κεντρικό μενού του τμήματος, με την διαφορά ότι το μενού είναι διαφορετικό από αυτό των προϊσταμένων του τμήματος.

Οι λειτουργίες που μπορούν να κάνουν οι υπάλληλοι του δημοτολογίου είναι (βλ. Σχ.3):

- A. Εισαγωγή Νέου Μέλους Οικογένειας
- B. Άνοιγμα Νέας Οικογενειακής Μερίδας
- Γ. Διαγραφή Δημότη λόγω Μεταδημότευσης
- Δ. Διαγραφή Δημότη λόγω Θανάτου
- E. Αλλαγή Στοιχείων Δημότη



(Σχήμα 8)

Ανάλυση λειτουργιών

A. Εισαγωγή Νέου Μέλους Οικογένειας

Για να γίνει η εισαγωγή ενός νέου μέλους, ο υπάλληλος πρέπει να κάνει πρώτα αναζήτηση της οικογενειακής μερίδας στην οποία ανήκει το νέο μέλος. Η αναζήτηση αυτή γίνεται βάση των εξής στοιχείων : *Επώνυμο και όνομα πατρός, όνομα και επώνυμο μητρός* .

Αφού βρεθεί ο αριθμός οικογενειακής μερίδας ο υπάλληλος οδηγείται στην φόρμα εισαγωγής του νέου μέλους. Στην συνέχεια γίνεται έλεγχος των στοιχείων αυτών και εισαγωγή της νέας εγγραφής στην βάση δεδομένων.

B. Άνοιγμα Νέας οικογενειακής Μεριδας .

Εδώ γίνεται αυτόματη ανάθεση οικογενειακής μερίδας από το σύστημα με βάση τη χρονολογία εισαγωγής της νέας εγγραφής και στην συνέχεια έλεγχος και αποθήκευση της νέας εγγραφής στο σύστημα .

Γ. Διαγραφή Δημότη λόγω Μεταδημότευσης .

Για να γίνει διαγραφή λόγω μεταδημότευσης ο υπάλληλος πρέπει να κάνει αναζήτηση στο σύστημα της εγγραφής δημοτολογίου που θέλει να κάνει διαγραφή. Τα στοιχεία με τα οποία κάνει αναζήτηση είναι τα εξής : Όνομα, Επώνυμο, Όνομα πατρός, όνομα μητρός, γένος , ημερομηνία γεννήσεως και ημερομηνία διαγραφής.

Αφού δώσει τα στοιχεία αυτά το σύστημα αναζητεί για την ύπαρξη της εγγραφής αυτής και στο τέλος γίνεται «διαγραφή» της εγγραφής αυτής από το σύστημα.

Δ. Διαγραφή Δημότη λόγω Θανάτου .

Και στην περίπτωση αυτή ο υπάλληλος πρέπει να κάνει αναζήτηση της εγγραφής για την οποία ενδιαφέρεται . Τα στοιχεία με τα οποία κάνει αναζήτηση είναι τα εξής για αυτή την περίπτωση : όνομα, επώνυμο ,όνομα πατρός, όνομα μητρός, γένος, ημερομηνία γέννησης, ημερομηνία θανάτου και αριθμός ληξιαρχικής πράξης.

Στην συνέχεια γίνεται αναζήτηση της εγγραφής αυτής και εν συνεχεία «διαγραφή λόγω θανάτου» .

E. Αλλαγή Στοιχείων Δημότη

Ο Υπάλληλος της υπηρεσίας είναι υπεύθυνος για την αλλαγή –όποτε απαιτείται– των στοιχείων ενός Δημότη (π.χ. στην περίπτωση που το παιδί μιας οικογένειας, αποκτά όνομα- βαπτίζεται). Τα στοιχεία που μπορούν να αλλάξουν είναι συγκεκριμένα και καμία αλλαγή δεν γίνεται σε αλλά στοιχεία του δημοτολογίου. Τα στοιχεία που μπορούν να αλλάξουν είναι : *Όνομα, Επίθετο και Θρήσκευμα* .

ΣΤ. Προβολή Παρατηρήσεων Δημοτών

Οι υπάλληλοι του δήμου (τμήματος Δημοτολογίου) μπορούν, να ενημερώνονται για τα διάφορα προβλήματα και παρατηρήσεις των δημοτών, στην προσπάθεια για καλύτερη εξυπηρέτησή τους. Έτσι αφού διαβάσουν τα παράπονα/παρατηρήσεις τους μπορούν να αντιμετωπίσουν τα τυχόν προβλήματα που παρουσιάζονται .

Z. Καταχώρηση στοιχείων Γάμου

Από εδώ γίνεται καταχώρηση ενός γάμου στο δημοτολόγιο. Ο υπάλληλος καταχωρεί σε μια φόρμα τα αναλυτικά στοιχεία του άνδρα και της γυναίκας καθώς και τα στοιχεία του γάμου. Μετά την αποστολή των στοιχείων το σύστημα ψάχνει να βρει αν είναι δημότες. Αν είναι τότε καταχωρείται ο γάμος και καινούργια οικογενειακή μερίδα αν χρειάζεται ενώ αν δεν είναι καταχωρούνται σαν καινούργιοι δημότες και ανοίγει νέα οικογενειακή μερίδα. Το σύστημα εκτελεί αναλόγως αν μόνο ένας από τους δύο είναι δημότης.

Παρατηρήσεις :

1. Κατά την διαγραφή ενός δημότη λόγω Μεταδημότευσης ή Θανάτου η εγγραφή του δημοτολογίου δεν διαγράφεται από την βάση . Για να γίνει μη ενεργή και να ξεχωρίζει από τις υπόλοιπες εγγραφές, στην περίπτωση της Μεταδημότευσης προστίθεται ένα “D” μπροστά από το πεδίο που θεωρείτε ως πρωτεύον κλειδί της εγγραφής (resid) και ένα “P” στην περίπτωση που έχει γίνει διαγραφή λόγω θανάτου .
2. Σε όλες τις λειτουργίες των υπαλλήλων οι εγγραφές περνούν πρώτα από έλεγχο για την αποφυγή εισαγωγής κακόβουλων στοιχείων (html tags, java tags) που μπορεί να έχουν ως αποτέλεσμα κατάρρευση του συστήματος και της βάσης δεδομένων της εφαρμογής.

3.5 Διαχείριση Συστήματος

Είσοδος του διαχειριστή του συστήματος γίνεται από την κοινή σελίδα (login.php) των υπαλλήλων, επιλέγοντας ως ιδιότητα χρήστη το «ΔΙΑΧΕΙΡΙΣΤΗ». Ο διαχειριστής του συστήματος είναι ο υπεύθυνος για την σωστή λειτουργία της εφαρμογής (ηλεκτρονικός δήμος) και για το λόγω αυτό πρέπει απαραίτητων να γνωρίζει το Σύστημα Διαχείρισης Βάσεων Δεδομένων MySQL DBMS (Data Base Management System), ώστε να είναι σε θέση να αντιμετωπίσει πιθανά μικροπροβλήματα που ίσως παρουσιαστούν.

Ο διαχειριστής του συστήματος έχει τις παρακάτω δυνατότητες (βλέπε Σχημ.9) :

- A. Διαχείριση Δημοτών/ Υπαλλήλων
 - a. Έλεγχος Δημοτών
 - b. Έλεγχος Υπαλλήλων
 - c. Έλεγχος προϊσταμένων τμημάτων
- B. Διαχείριση Υπηρεσιών και Δημοτικών Διαμερισμάτων
 - a. Διαχείριση Τμημάτων/Υπηρεσιών Δήμου
 - b. Διαχείριση Δημοτικών Διαμερισμάτων
 - c. Αλλαγή Προέδρων Δημοτικών Διαμερισμάτων
- Γ. Προβολή Σφαλμάτων
 - a. Προβολή λάθος ενεργειών Υπαλλήλων
 - b. Έλεγχος Πρόσβασης άλλων Υπηρεσιών

Χρήστης : ΑΡΓΥΡΗΣ ΧΗΡΑΣ

Διαχείριση Ηλεκτρονικού Δήμου

<p>Υπάλληλοι / Δημότες</p> <p><u>Έλεγχος Δημοτών</u></p> <p><u>Έλεγχος Υπαλλήλων</u></p> <p><u>Έλεγχος Προϊσταμένων Τμημάτων</u></p>	<p>Υπηρεσίες και Δημοτικά Διαμερίσματα</p> <p><u>Διαχείριση Τμημάτων/Υπηρεσιών</u></p> <p><u>Διαχείριση Δημοτικών Διαμερισμάτων</u></p> <p><u>Αλλαγή Προέδρου Δ/Δ</u></p>
---	--

Προβολή Σφαλμάτων

Προβολή Λάθος Ενεργειών Υπαλλήλων

Έλεγχος Πρόσβασης άλλων Υπηρεσιών

Σημ : Τα στοιχεία πρέπει να συμπληρώνονται με ΚΕΦΑΛΑΙΑ ΓΡΑΜΜΑΤΑ

(Σχήμα 9)

Ανάλυση λειτουργιών

Παρακάτω αναλύονται –σε γενικές γραμμές- οι λειτουργίες του διαχειριστή που προαναφέραμε .

A. Διαχείριση Δημοτών/ Υπαλλήλων

a. Έλεγχος Δημοτών

Ο διαχειριστής του συστήματος είναι υπεύθυνος για την εισαγωγή των κωδικών πρόσβασης των δημοτών στο σύστημα, μετά από αίτηση των δευτέρων να έχουν πρόσβαση στις ηλεκτρονικές υπηρεσίες .

Επίσης μπορεί στην περίπτωση που θέλει ο/η δημότης να γίνει η αλλαγή του κωδικού πρόσβασης με ένα καινούριο .

b. Έλεγχος Υπαλλήλων

Ομοίως ο διαχειριστής είναι αυτός που περνάει στο σύστημα τους κωδικούς πρόσβασης, το όνομα χρήστη των υπαλλήλων καθώς και την IP Address των ηλεκτρονικών υπολογιστών που θα έχουν αποκλειστική πρόσβαση στο σύστημα .

c. Έλεγχος Προϊστάμενων Υπηρεσιών

Εδώ ο διαχειριστής έχει την δυνατότητα να αλλάξει τον προϊστάμενο ενός τμήματος και να του δώσει πρόσβαση στο σύστημα .

B. Διαχείριση Υπηρεσιών και Δημοτικών Διαμερισμάτων

a. Διαχείριση Τμημάτων/Υπηρεσιών

Αυτή η ενέργεια παίρνει μέρος κατά την εγκατάσταση του συστήματος σε κάποιον Δήμο. Ο διαχειριστής είναι υπεύθυνος για την προσθήκη των υπηρεσιών του δήμου στο νέο σύστημα .

b. Διαχείριση Δημοτικών Διαμερισμάτων

Ομοίως προσθέτει τα δημοτικά διαμερίσματα από τα οποία αποτελείται ο δήμος καθώς και τους προέδρους αυτών των διαμερισμάτων

c. Αλλαγή Προέδρων Δημοτικών Διαμερισμάτων

Αυτή η λειτουργία επιτρέπει την αλλαγή των προέδρων των δημοτικών διαμερισμάτων. Όπως είναι γνωστό αυτή η λειτουργία λαμβάνει μέρος συνήθως κάθε τέσσερα (4) χρόνια μετά τις εκλογές .

Γ. Προβολή Λάθος Ενεργειών Υπαλλήλων

Η λειτουργία αυτή επιτρέπει στο διαχειριστή του συστήματος να εντοπίζει λάθος ενέργειες των υπαλλήλων που καταγράφει το σύστημα, έτσι ώστε σε περίπτωση που προκύψει κάποιο σφάλμα στη καλή λειτουργία του συστήματος από λάθος ενέργεια των υπαλλήλων να είναι σε θέση να γνωρίζει το λόγο που έγινε και να το διορθώνει .

Δ. Έλεγχος πρόσβασης άλλων Υπηρεσιών .

Για μεγαλύτερη ασφάλεια, στην περίπτωση που οι υπάλληλοι άλλων υπηρεσιών (π.χ. Αστυνομία), πληκτρολογήσουν λάθος πάνω από πέντε (5) φορές τον ειδικό αριθμό του πιστοποιητικού/βεβαίωσης που θέλουν να ελέγξουν για εγκυρότητα, το σύστημα δεν τους επιτρέπει να συνεχίσουν τον έλεγχο κλειδώνοντας την IP Address τους . Στην περίπτωση αυτή πρέπει να επικοινωνήσουν στα τηλέφωνα που τους δίνονται ώστε ο διαχειριστής του συστήματος χρησιμοποιώντας αυτή την λειτουργία να τους επιτρέψει την πρόσβαση και να τους κάνει χρήστες του συστήματος και να μην έχουν πάλι το ίδιο πρόβλημα.

Γενικές Παρατηρήσεις :

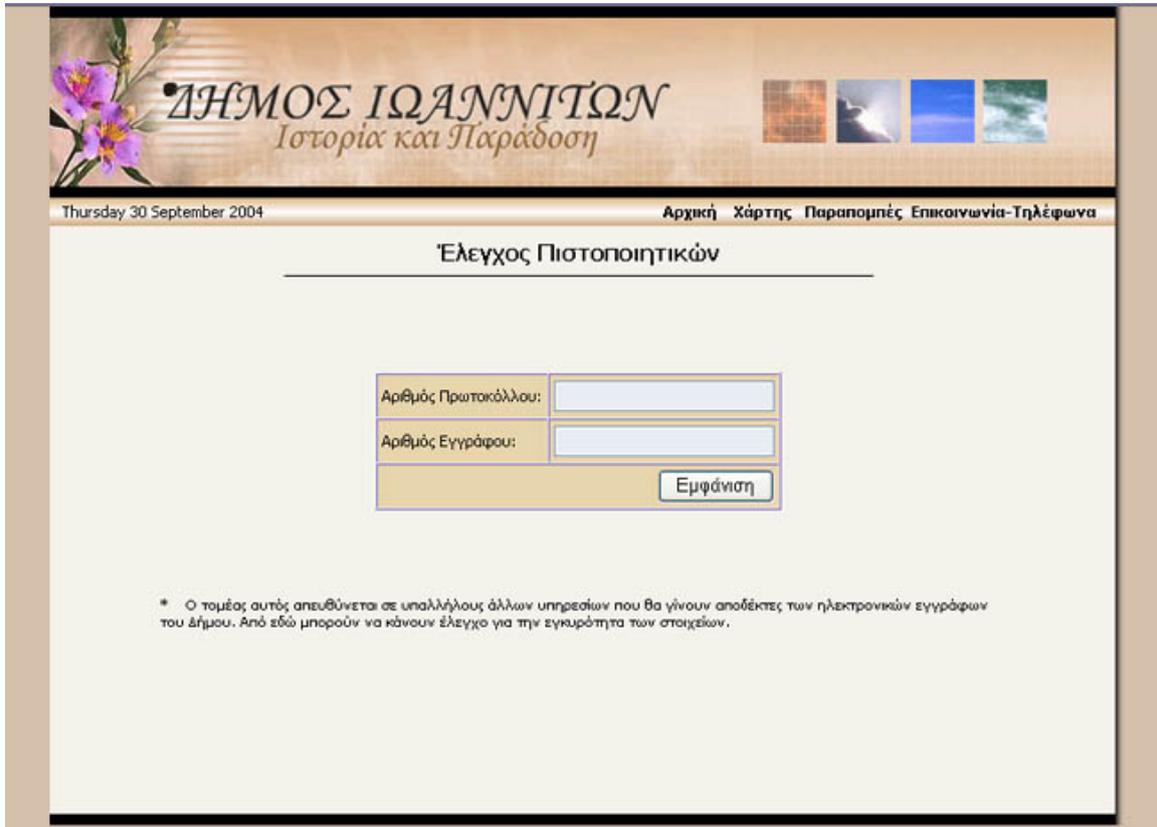
Παρότι το σύστημα είναι θωρακισμένο να αποτρέπει την πληκτρολόγηση και εισαγωγή στο σύστημα δεδομένων που μπορεί να προκαλέσουν σφάλματα στο σύστημα, δεν εγγυάται την ορθότητα αυτών των στοιχείων. Αυτό παρότι μπορεί να φαίνεται ως αρνητικό στοιχείο της όλης εφαρμογής είναι σίγουρο ότι **KAMIA** εφαρμογή δεν μπορεί να αποτρέψει κάτι τέτοιο . Έτσι η βαρύτητα των δεδομένων που εισάγονται στην βάση δεδομένων του συστήματος βαραίνουν αποκλειστικά και μόνο τους υπαλλήλους του κάθε τμήματος και στην περίπτωση τις εφαρμογής μας, η εισαγωγή τέτοιων δεδομένων από τους υπαλλήλους αποτελούν ποινικά αδικήματα .

3.6 Είσοδος σαν υπάλληλος άλλων υπηρεσιών

Τα πιστοποιητικά που παίρνουν οι δημότες φυσιολογικά θα καταλήξουν σε κάποια υπηρεσία ή σε κάποιο οργανισμό που τα χρειάζεται. Θα αναρωτηθεί κάποιος πως θα καταλάβει ο κάθε υπάλληλος αυτών των υπηρεσιών αν το πιστοποιητικό που θα του

προσκομίσει κάποιος δημότης είναι αυθεντικό ή όχι, αφού δεν φέρει καμία σφραγίδα ή υπογραφή.

Το μόνο που έχει να κάνει ο υπάλληλος είναι να μπει στη σελίδα του Δήμου και να πάει στον σύνδεσμο *Υπάλληλοι Υπηρεσιών* που θα τον οδηγήσει στην παρακάτω σελίδα (Σχίμα 10) :



ΔΗΜΟΣ ΙΩΑΝΝΙΝΩΝ
Ιστορία και Παράδοση

Thursday 30 September 2004 Αρχική Χάρτης Παραπομπές Επικοινωνία-Τηλέφωνα

Έλεγχος Πιστοποιητικών

Αριθμός Πρωτοκόλλου:

Αριθμός Εγγράφου:

Εμφάνιση

* Ο τομέας αυτός απευθύνεται σε υπαλλήλους άλλων υπηρεσιών που θα γίνουν αποδέκτες των ηλεκτρονικών εγγράφων του Δήμου. Από εδώ μπορούν να κάνουν έλεγχο για την εγκυρότητα των στοιχείων.

Σχίμα 10 (Έλεγχος Πιστοποιητικών)

Εδώ μόλις πάρει στα χέρια του ο υπάλληλος το πιστοποιητικό απλά θα συμπληρώσει τον αριθμό πρωτοκόλλου και τον αριθμό εγγράφου του πιστοποιητικού στην φόρμα και το σύστημα θα του επιστρέψει το πιστοποιητικό του δημότη με τα στοιχεία του όπως είναι καταχωρημένα εκείνη τη στιγμή. Οπότε εύκολα μπορεί να συγκρίνει τα στοιχεία και να διαπιστώσει την εγκυρότητα του εγγράφου.

Στη φόρμα συμπλήρωσης των στοιχείων έχουν πρόσβαση όλοι καθώς δεν μπορεί να προβλεφθεί σε ποια υπηρεσία και σε ποιο μέρος της χώρας μπορεί να καταλήξει το πιστοποιητικό. Έτσι όποιος κατέχει λοιπόν τον αριθμό πρωτοκόλλου και αριθμό εγγράφου οπότε και το πιστοποιητικό μπορεί να ελέγξει για την εγκυρότητά του.

Για μεγαλύτερη ασφάλεια αν κάποιος δώσει 5 φορές λανθασμένα στοιχεία μέσα σε 15 λεπτά τότε το σύστημα διακόπτει την είσοδο του στον έλεγχο πιστοποιητικών καθώς μπορεί να είναι αποτέλεσμα κακόβουλων ενεργειών. Στην συνέχεια τον προτρέπει να επικοινωνήσει με τον διαχειριστή του συστήματος σε σχετικό τηλέφωνο. Έτσι αν κάποιος είναι υπάλληλος κάποιας υπηρεσίας προτρέπεται να επικοινωνήσει από το σταθερό τηλέφωνο της υπηρεσίας του ώστε ο διαχειριστής να τον προσθέσει στους χρήστες του συστήματος και να αποκτήσει πάλι πρόσβαση και επιπλέον να μην διακόπτεται η εισόδός του με την εισαγωγή λάθος στοιχείων. Αν δεν το κάνει αυτό δεν θα μπορέσει να μπει στον έλεγχο πιστοποιητικών με την ίδια *ip διεύθυνση* αφού έχει καταχωρηθεί στο σύστημα. Για την καλύτερη συνεργασία με το σύστημα οι υπηρεσίες χρειάζεται να έχουν στατικές ip διευθύνσεις.

3.7 Εγκατάσταση σε άλλο Δήμο

Για να εγκαταστήσουμε την εφαρμογή μας σε κάποιο άλλο δήμο, απαιτείται πρώτα η εγκατάσταση κάποιων προγραμμάτων (software) στα οποία θα βασίζεται η εφαρμογή μας . Αυτά είναι :

1. MySQL DBMS

Σ' αυτό το σύστημα Διαχείρισης Βάσεων Δεδομένων έχει στηθεί η εφαρμογή . Εδώ θα αποθηκεύονται οι πληροφορίες που θα περνούν από τους υπαλλήλους του δήμου. Για να δηλώσουμε κάποιον άλλο δήμο πρέπει να γίνουν μερικές μεταβολές σε κάποια αρχεία. Συγκεκριμένα στα αρχεία *db_connection.php* και *l_config.php* πρέπει να δηλωθεί το όνομα του καινούργιου Δήμου στις σχετικές μεταβλητές και να εισαχθούν τα καινούργια στοιχεία σύνδεσης με τη βάση δεδομένων (*IP,Username,Password*).

2. Web Server / PHP

Δεν νοείται η εφαρμογή μας , αν δεν υπάρχουν αυτά τα δύο (2) στοιχεία . Παρότι η εφαρμογή μας έχει δοκιμαστεί να δουλεύει με τον Apache Web Server , κάνοντας τις απαραίτητες αλλαγές στο αρχείο *php.ini* είναι δυνατόν να δουλέψει με οποιοδήποτε Web διακομιστή (server) .

3. Εμφάνιση

Χρειάζεται να γίνουν μικρές αλλαγές στα χρώματα των σελίδων των υπηρεσιών ώστε να ταιριάζουν στο στυλ της σελίδας του Δήμου που θα ενσωματωθεί η εφαρμογή.

Μετά την ολοκλήρωση της εγκατάστασης , ο διαχειριστής του συστήματος πρέπει να περάσει χειροκίνητα στην βάση το όνομα χρήστη και τον κωδικό πρόσβασης του. Στην συνέχεια μέσω της εφαρμογής γίνεται η προσθήκη των δημοτικών διαμερισμάτων, υπαλλήλων, προϊσταμένων και νέων δημοτών του δήμου. Παρόλα αυτά σήμερα κάνοντας χρήση του ODBC (Open Data Base Connectivity) είναι δυνατόν να κάνουμε εξαγωγή από οποιαδήποτε βάση τα στοιχεία που περιέχει και να τα περάσουμε στο σύστημα (MySQL) δαπανώντας ελάχιστο χρόνο, κάνοντας όπου χρειάζεται κάποιες αλλαγές .

4. Συμπεράσματα

Η εφαρμογή αυτή αποτελεί σίγουρα μια καλή προσέγγιση στη μελλοντική πραγματικότητα όπου θα έχουν αξιοποιηθεί οι τεχνολογικές δυνατότητες που εξελίσσονται ραγδαία σήμερα. Ειδικά για την Ελλάδα που τώρα αρχίζουν να γίνονται τα πρώτα βήματα προς αυτή την κατεύθυνση αποτελεί μια πολύ καλή πρόταση για την ανάπτυξη των Δήμων και άλλων δημόσιων υπηρεσιών που λειτουργούν με παρόμοιο τρόπο. Τα οφέλη από έναν τέτοιο εκσυγχρονισμό είναι μεγάλα τόσο για τους υπαλλήλους όσο και για τους Δημότες-Πολίτες .

Θα υπάρξει ουσιαστική αλλαγή στη διαχείριση των δεδομένων και τον τρόπο που αυτά διανέμονται στους Δημότες με αποτέλεσμα να ελαττωθεί το φαινόμενο της γραφειοκρατίας στο ελάχιστο. Αυτό αποτελεί μεγάλη επιτυχία αφού η γραφειοκρατία αποτελεί εδώ και χρόνια το αγκάθι των ελληνικών δημόσιων υπηρεσιών. Επομένως ο χρόνος που χάνεται σε τέτοιες διαδικασίες από τους υπαλλήλους μπορεί να αξιοποιηθεί αποδοτικότερα. Επιπλέον οι Δημότες δεν θα χρειάζεται να περιμένουν σε ουρές και να πηγαίνουν από γραφείο σε γραφείο όπως γίνεται σήμερα. Αντιθέτως με συνοπτικές διαδικασίες θα εξυπηρετούνται χάνοντας λιγότερο χρόνο και μένοντας περισσότερο ευχαριστημένοι.

Με την πάροδο του χρόνου και αφού οι Δημότες εξοικειωθούν με το σύστημα , μπορεί να επεκταθεί και σε άλλες υπηρεσίες του Δήμου εκτός από το δημοτολόγιο και την έκδοση σχετικών πιστοποιητικών. Έτσι με σχετική μελέτη μπορεί να επεκταθεί στην πολεοδομία για την διαχείριση και έκδοση αδειών, τις οφειλές προς το Δήμο και άλλα θέματα.

Η εφαρμογή όπως είναι στημένη μπορεί να εξυπηρετήσει Δήμους που έχουν πληθυσμό μέχρι εκατό χιλιάδες εγγεγραμμένους δημότες. Στην περίπτωση πολύ μεγαλύτερων Δήμων πρέπει να γίνουν μικρές αλλαγές στον τομέα των κωδικών και κάποιων άλλων στοιχείων του συστήματος. Μπορεί να χρειαστούν περισσότερες αλλαγές στην περίπτωση που το σύστημα διαχείρισης της βάσης(MySQL) δεν αποδώσει καλά όταν χρειαστεί να διαχειριστεί μια εξαιρετικά μεγάλη βάση δεδομένων όπως είναι αυτή των μεγαλύτερων Ελληνικών Δήμων. Τότε μπορεί να χρειαστεί να αντικατασταθεί με ένα που να έχει καλύτερη απόδοση στην περίπτωση πολύ μεγάλων βάσεων όπως είναι η ORACLE (αναμφίβολα και πολύ μεγαλύτερο κόστος).

Όταν κάποιος ακούσει για τις ηλεκτρονικές υπηρεσίες αμέσως θα αναρωτηθεί για την ασφάλεια στη διακίνηση πιστοποιητικών και εγγράφων αφού δεν φέρουν καμία σφραγίδα. Μην ξεχνάμε όμως ότι, όπως είπαμε παραπάνω, φέρουν κάποιους μοναδικούς αριθμούς που δίνουν τη δυνατότητα στον καθένα να ελέγξει για την εγκυρότητά τους. Επιπλέον θα λέγαμε ότι είναι πιο ασφαλής τρόπος από ότι μια σφραγίδα, καθώς έχουμε δει επανειλημμένως να πλαστοποιούνται σφραγίδες και υπογραφές ενώ το ηλεκτρονικό πιστοποιητικό είναι μέσα στη βάση οπότε κάποιος για να το αλλάξει πρέπει να παραβιάσει συστήματα με τεράστια επίπεδα ασφαλείας, που είναι σχεδόν αδύνατο.

Επιμένουμε λοιπόν ότι η εφαρμογή αποτελεί μια ενδιαφέρουσα πρόταση για τη μελλοντική αναβάθμιση των υπηρεσιών που παρέχει ένας Δήμος .

ΑΝΑΦΟΡΕΣ - ΒΙΒΛΙΟΓΡΑΦΙΑ

Βιβλία

1. “*MySQL/PHP Database Applications*” by Jay Greenspan and Brad Bulger.
2. “*PHP MANUAL*” by Stig Sather Bakken and Egon Schmid.

Ιστοσελίδες

1. <http://www.php.net>
2. <http://www.apache.org>
3. <http://dev.mysql.com>
4. <http://www.ypes.gr>

Ευχαριστούμε θερμά τις υπηρεσίες του “ Δήμου Σκοπέλου ” , του “ Δήμου Άρτας ” και του “ Δήμου Ιωαννιτών ” για τα στοιχεία που μας δόθηκαν .

ΠΑΡΑΡΤΗΜΑ

(Κώδικας Εφαρμογής)

1. Είσοδος(Login)

Login.php

```
<?php
session_start();
include("l_config.php");
include("l_functions.php");

if (@$_POST['name'] && @$_POST['lname'] && @$_POST['fname']) // βαζουμε @
για να μην μας βγαζει μηνυμα λαθους αν δεν υπαρχουν οι μεταβλητές
    {
// οταν μπαινουμε στο login.php πριν στείλουμε στοιχεία απο τη φορμα

$stoxeia=secure_globals($_POST['name'],$_POST['lname'],$_POST['fname'],$_POS
T['password']);

        //$onoma=strtoupper ($stoxeia[0]);
        $onoma=kefalaia($stoxeia[0]);
        $epitheto=kefalaia ($stoxeia[1]);
        $onoma_pat=kefalaia($stoxeia[2]);

$pswd = get_password ($onoma, $epitheto,$onoma_pat);

$loginsuccess = 0;
if ($pswd) // prevent empty password
    {
        foreach($pswd as $v)
            {
                if ($v==$stoxeia[3])
                    {
                        $loginsuccess = 1;
                        break;
                    }
            }
    }

if ($loginsuccess)
    {
        set_values("access ok",$onoma,
$epitheto,$_POST['ip'],$_POST['date_time']);
        $_SESSION["verified_user"] =get_resid ($onoma,$stoxeia[3]) ;
        $_SESSION["lasttime"] = time();
        Header("Location: $starturl?".SID);
        setcookie ($l_cookie , $onoma, time()+3600*24*3); // 3
days valid
    }
    else
    {
        set_values("access Denied",$onoma,
$epitheto,$_POST['ip'],$_POST['date_time']);
```

```

        $_l_error = "wrong_pwd";
    }
}

$message="Είσοδος στις ηλεκτρονικές υπηρεσίες";

include("emfanisi_login2.php");

$REMOTE_ADDR = $_SERVER['REMOTE_ADDR'];

if(isset($_GET['l_error']))
    $emf="1";
else
    $emf="0";
?>
<form name="login" action="<?php echo $_SERVER['PHP_SELF']?>"
method="post">

    <table border=0 align="center" cellspacing=0 cellpadding=4>
<?php
if (@$_l_error)
{
    echo"<tr >";
    echo"<td colspan='2'>";
    if ($_l_error == "wrong_pwd")
        echo"Τα στοιχεία που δώσατε δεν είναι σωστά! Παρακαλούμε
συμπληρώστε ξανά.";
    }
elseif ($emf=="1")
{
    echo"<tr >";
    echo"<td colspan='2'>";
        echo"Παρακαλούμε συμπληρώστε ξανά τα στοιχεία! ";
    }
?>
    </td>
</tr>
</table>

    <table bgcolor="#E8D5AC" align="center" border="1" cellspacing="0"
cellpadding="3">
<tr><td>Όνομα:</td><td>
<input type="text" name="name" maxlength="15" value="">
</td></tr>
<tr><td>Επίθετο:</td><td>
<input type="text" name="lname" maxlength="15" value="">
</td></tr>
<tr><td>Όνομα Πατρός:</td><td>
<input type="text" name="fname" maxlength="15" value="">
</td></tr>
<tr><td>Κωδικός:</td><td>
<input type="password" name="password" maxlength="15" value="">
</td></tr>
<tr><td colspan="2" align="right">

```


I_config.php

```
<?php
$dbhost = "localhost";      // you MySQL host
$dbname = "edimos";        // MySQL database name
$dbuser = "";               // MySQL login
$dbpwd = "";                // MySQL password

$dbtable = "residents";
$userfield = "ResID";
$userfield1 = "Name";
$userfield2 = "Lname";
$userfield3 = "Fname";
$pwdfield = "Code";

$loginurl = "login.php";
$logouturl = "login.php";
$starturl = "ypiresies.php";

$_I_cookie = "psluser";
$_I_timeout = 20*60;        // in seconds
?>
```

L_functions.php

```
<?php
include("I_config.php");

function secure_globals($name1,$lname1,$fname1,$password1)
{
    $name=strip_tags($name1);
    $lname=strip_tags($lname1);
    $fname=strip_tags($fname1);
    $password=strip_tags($password1);
    $secure=array($name,$lname,$fname,$password);
    return $secure;
}

function get_password ($name,$lname,$fname)
{
    include("connect_db.php");
    $result = mysql_query ("SELECT $GLOBALS[pwdfield] FROM
$GLOBALS[dbtable]
                                WHERE $GLOBALS[userfield1]='$name'
and
$GLOBALS[userfield2]='$lname' and
$GLOBALS[userfield3]='$fname'", $db);
    if (!$result) return "";
    $passwords=array();
```

```

        while($row=mysql_fetch_row($result))//gia na pairnei o pinakas to
password kathe grammis
        {
            foreach($row as $v)
            {
                array_push($passwords,$v);
            }
        }
        mysql_close ($db);
        return $passwords;
    }
function get_resid ($name,$code)
{
    include("connect_db.php");
    $result = mysql_query ("SELECT ResID FROM $GLOBALS[dbtable]
                                WHERE Name='$name' and
                                Code='$code'", $db);

    if (!$result) return "";

    $myrow = mysql_fetch_row($result);
    mysql_close ($db);
    return $myrow[0];
}
function set_values($log,$username,$userIname,$ip,$date_time)
{
    include("connect_db.php");
    $query="INSERT INTO login VALUES (NULL,'$date_time',
'$username','$userIname',
                                '$ip','$log)";

    $result = mysql_query ($query,$db);
    mysql_close ($db);
}
function str_split1($str,$num = '1')
{
    if($num < 1) return FALSE;

    $arr = array();

    for ($j = 0; $j < strlen($str); $j= $j+$num)
    {
        $arr[] = substr($str,$j,$num);
    }

    return $arr;
}
function kefalaia($str)
{
    $arr=str_split1($str);

    $arr1=Array();
    $output_str="";
}

```

```

$counter=count($arr);

for($i=0;$i<$counter;$i++)
{
    switch($arr[$i])
    {
        case 'a':
        case 'á':
        case 'A':
        case 'Á':
            $arr1[$i]='A';
            break;
        case 'b':
        case 'B':
            $arr1[$i]='B';
            break;
        case 'g':
        case 'Γ':
            $arr1[$i]='Γ';
            break;
        case 'd':
        case 'Δ':
            $arr1[$i]='Δ';
            break;
        case 'e':
        case 'E':
        case 'é':
        case "E'":
            $arr1[$i]='E';
            break;
        case 'z':
        case 'Z':
            $arr1[$i]='Z';
            break;
        case 'h':
        case 'h́':
        case "H'":
        case 'H':
            $arr1[$i]='H';
            break;
        case 'o':
        case 'O':
            $arr1[$i]='O';
            break;
        case 'i':
        case 'í':
        case "I'":
        case 'I':
            $arr1[$i]='I';
            break;
        case 'k':
        case 'K':
            $arr1[$i]='K';
            break;
        case 'λ':
    }
}

```

```
case '^':
    $arr1[$i]='^';
    break;
case 'μ':
case 'M':
    $arr1[$i]='M';
    break;
case 'v':
case 'N':
    $arr1[$i]='N';
    break;
case 'ξ':
case 'Ξ':
    $arr1[$i]='Ξ';
    break;
case 'o':
case 'ò':
case 'O':
case 'O':
    $arr1[$i]='O';
    break;
case 'n':
case 'Π':
    $arr1[$i]='Π';
    break;
case 'p':
case 'P':
    $arr1[$i]='P';
    break;
case 'σ':
case 'ç':
case 'Σ':
    $arr1[$i]='Σ';
    break;
case 'T':
case 'T':
    $arr1[$i]='T';
    break;
case 'u':
case 'ù':
case "Y":
case 'Y':
    $arr1[$i]='Y';
    break;
case 'φ':
case 'Φ':
    $arr1[$i]='Φ';
    break;
case 'χ':
case 'X':
    $arr1[$i]='X';
    break;
case 'ψ':
case 'Ψ':
    $arr1[$i]='Ψ';
    break;
```

```

        case 'ω':
        case 'ώ':
        case "Ω":
        case 'Ω':
            $arr1[$i]='Ω';
            break;
    }
}

foreach($arr1 as $v)
    $output_str=$output_str.$v;

return $output_str;
}
function getpost($num)
{
    $ar=$num;

    return $ar;
}
?>

```

Checklogin.php

```

<?php
session_start();
require("l_config.php");

$timeout = "";

if (time()- @$_SESSION["lasttime"] >= $!_timeout and @$_SESSION["lasttime"]) //
την πρώτη φορά που μπαίνουμε δεν υπάρχει το @$_SESSION["lasttime"]) για αυτό
βαζουμε @
    $timeout = "?l_error='timeout'";

if (@!$_SESSION["verified_user"] or $timeout)
    {
        // echo " not regisrtrekjdfsag";

        Header("Location:
http://localhost/php/edimos/secure_login/login.php$timeout");
        exit;
    }
$_SESSION["lasttime"] = time(); //οπότε μπαίνεις στη σελίδα κρατάει τη χρονική
στιγμή
?>

```

Db_connection.php

```
<?php
include("l_config.php");

$db = mysql_connect ($GLOBALS['dbhost'], $GLOBALS['dbuser'],
    $GLOBALS['dbpwd']);
    mysql_select_db ($GLOBALS['dbname'],$db);

?>
```

Logout.php

```
<?php
require("l_config.php");
session_start();
session_destroy();
Header("Location: $logouturl");

?>
```



```

        case '1':
            include("message.php");
            break;
        case '2':
            include("form_paratiriseis.php");
            break;
        case '3':
            include("message1.php");
            break;
        case '4':
            include("message4.php");
            break;
        case '5':
            include("message_error.php");
            break;
    }
}
else
    include("ypir_pistop.php");

?>

```

```

</td>
<td width="15%">&nbsp;  </td>
</tr>
<tr>
<td width="14%" height="72">&nbsp;  </td>
<td width="37%" height="72">&nbsp;  </td>
<td width="31%" height="72">&nbsp;  </td>
<td width="15%" height="72">&nbsp;  </td>
</tr>
<tr>
<td colspan="4" height="38"><span lang="el">
<font size="1" face="Tahoma"> <b> * </b> &nbsp;  &nbsp;  > Το
πιστοποιητικό που θα πάρετε μπορείτε να το
προσκομίσετε σε
οποιαδήποτε υπηρεσία το χρειάζεται . Ο υπάλληλος που θα το
παραλάβει θα ελέγξει την
εγκυρότητά του απλά συμπληρώνοντας στον αντίστοιχο
τομέα της σελίδας μας τον
αριθμό πρωτοκόλλου και αριθμό εγγράφου του
πιστοποιητικού.</font></span></td>
</tr>
</table>
</div>
</td>
<td background=" ../dimos/right_shadow.gif" width="10"></td>
</tr>
</table>
<table width="760" border="0" cellspacing="0" cellpadding="0">
<tr>

```

```

        <td bgcolor="#000000"></td>
        <td background="../../dimos/right_shadow.gif"></td>
    </tr>
</table>
</center>
</div>
</body>
</html>

```

Functions_pistop.php

```
<?php
```

```

function get_stoixeia_gamou($resid_male,$resid_female,$table)
{
    include("../localhost/php/edimos/secure_login/connect_db.php");
    $result = mysql_query ("SELECT * FROM $table WHERE
ResID_male='$resid_male' and ResID_female='$resid_female' ",$db);
    //if (mysql_num_rows($result)==0)
        //return "";
        if (!$result) return "";
    $myrow = mysql_fetch_row($result); //epistrefei pinaka me ola ta stoixeia
    tou trexontos xristi
    mysql_close ($db);
    return $myrow;
}
function resid_info1($resid) //epistrefei to proto grama toy resid π.χ αν είναι
"Π" είναι πεθαμένος
{
    $temp= substr($resid, 0, 1);

    return $temp;
}
function resid_info2($familynumber,$ar_melous) //epistrefei to resid kai ta
stoixeia tou.(an zei ,an pethane ,
{
//an diagrafike)
    anazitontas tous pinakes me basi to familynumber

    include("../localhost/php/edimos/secure_login/connect_db.php");

    $a_resid = mysql_query ("SELECT ResID FROM residents where
Familynumber_now='$familynumber' and

Ar_melous_now='$ar_melous' ",$db);
    $resid = mysql_fetch_row($a_resid); //aytoi poy tairiazei to
family number

    $d_resid = mysql_query ("SELECT ResID FROM residents where
Familynumber_first='$familynumber' and

Ar_melous_first='$ar_melous' ",$db);

```

```

$resid3 = mysql_fetch_row($d_resid);

//$info=array();

if($resid[0]!=NULL)
{
    $fsit=resid_info1($resid[0]);           //Elegxei to arxiko
    gamma tou resid

    if($fsit=='P')
        $info=Array($resid[0],'resdead');   //analogws
    kataxwrei

    if($fsit=='D')
        $info=Array($resid[0],'resdeleted');

    if($fsit!='D' && $fsit!='P')
        $info=Array($resid[0],'residents1');

}
if($resid3[0]!=NULL)
{
    $fsit=resid_info1($resid3[0]);           //Elegxei to arxiko
    gamma tou resid

    if($fsit=='P')
        $info=Array($resid3[0],'resdead');   //analogws kataxwrei

    if($fsit=='D')
        $info=Array($resid3[0],'resdeleted');

    if($fsit!='D' && $fsit!='P')
        $info=Array($resid3[0],'residents2');

}

return $info;
}

function resid_ginaikas($familynumber,$ar_melous)
{
    include("//localhost/php/edimos/secure_login/connect_db.php");

    $res1 = mysql_query ("select ResID_female from marriage where
    Familynumber='$familynumber' and Ar_melousfemale='$ar_melous'", $db);

    $resid=mysql_fetch_row($res1);

    return $resid[0];
}

function get_stoixeia($resid,$table)
{
    include("//localhost/php/edimos/secure_login/connect_db.php");

```

```

        $result = mysql_query ("SELECT * FROM $table WHERE
ResID='$resid'", $db);

        if (!$result) return "";

        // if(mysql_num_rows($result)==0)
        // return "";
        $myrow = mysql_fetch_row($result); //epistrefei pinaka me ola ta stoixeia
tou trexontos xristi
        /*$count=1;
        while(mysql_fetch_row($result)) $count++;
        echo " count :$count <br>";*/
        mysql_close ($db);
        return $myrow;
    }

function protocol_number($Resid,$perigrafi)
{
    include("//localhost/php/edimos/secure_login/connect_db.php");

    $a_id= mysql_query ("SELECT MAX(id) FROM certificationinfo", $db);
    $b_id =mysql_fetch_row($a_id); // epilegoume to
megisto id

    $a_protocol = mysql_query ("SELECT Protocol FROM certificationinfo
where id='$b_id[0]' ", $db);
    $protocol = mysql_fetch_row($a_protocol); // epileg. ton
ar.protocolou gia max id

    $a_date= mysql_query ("SELECT Datecert FROM certificationinfo
where id='$b_id[0]'", $db);
    $b_date =mysql_fetch_row($a_date); // tin imerominia gia
max id

    $a_number=mysql_query ("SELECT Certificationnum FROM
certificationinfo where id='$b_id[0]'", $db);
    $b_number =mysql_fetch_row($a_number); //epilegetai o amesws
proigoymenos certification arithmos

    $str_date1=date("Y-m-d");
    $str_date=date("dmy");

    //sinthiki gia to protocol number

    if ($str_date1==$b_date[0]) //efoson einai i idia
mera
    {
        $temp = substr($protocol[0],7); //apo ton proigoymeno
arithmo protocolou epilegoume
        $temp1=$temp+1; //ta 3 teleytaia
psifia kai ta ayksanoyme kata 1
        $new_numb=str_pad($temp1,4,"000", STR_PAD_LEFT); //gia
epistrofi toy arithmoy me 4 psifia
    }
}

```

```

        $new_protocol=$str_date.$new_num; //kai ta proskoloume sti
trexousa imerominia
    }
    else
    {
        $new_num="0000"; //an allaksei i mera
ksekinaei apo to 0000
        $new_protocol=$str_date.$new_num;
    }

    //edw dimiourgeitai o certification number

    if($b_number[0]==999999)
        $new_number="000001";
    else
    {
        $temp2=$b_number[0]+1;
        $new_number=str_pad($temp2,6,"00000", STR_PAD_LEFT);
//gia epistrofi toy arithmoy me 6 psifia
    }

    $times=Array($new_protocol,$new_number); //pinakas pou
dioxeteyoume tis times pou brikame

    //pairname tis times sti basi kai ginontai oses allages prepei logo
//pareleusis kapoion xronikon orion

    $q= mysql_query ("insert into certificationinfo
values(null,'$Resid','$new_number','$new_protocol','$str_date1','$perigrifi')",$db)
;

    //Stis egrafes pou exoun diarkei zois megalyteri apo ena xrono
midenizetai
    //to certificationinfo oste na min einai diathesimes kai na paramenoun
ta stoixeia
    //protocolou gia apothikeysi.

    $str_date2=date("md");
    if($str_date2=="0229") //gia tin periptosi pou exei 29 o
flebouarios
    {
        $lastyear = mktime(0, 0, 0, 2, 28,date("Y")-1); //pigainoume
stis 28 tou proigoumenou etous
        $delete_date = date("Y-m-d", $lastyear);
    }
    else
    {
        $lastyear = mktime(0, 0, 0, date("m"), date("d"), date("Y")-1
);
        $delete_date = date("Y-m-d", $lastyear);
//katebainoume akribws 1 xrono
    }

    //bazoume "000000" ston arithmo eggrafou ota exei perasei 1 xronos
oste na min isxyei

```

```

//gia to koino alla na paramenei os back up sti basi

$z= mysql_query ("update certificationinfo set
Certificationnum='000000' where Datecert<'$delete_date'", $db);

mysql_close ($db);
return $times;
}
function elegxos_p($Resid,$perigrafi)
{

include("//localhost/php/edimos/secure_login/connect_db.php");

$temp = mktime(0, 0, 0, date("m"), 1,date("Y"));
$arxi_mina = date("Y-m-d", $temp); //pame stin arxi tou mina

$z= mysql_query ("select count(ResID) from certificationinfo where
Datecert>='$arxi_mina' and ResID='$Resid' and Perigrafi='$perigrafi'", $db);
$apot=mysql_fetch_row($z);
mysql_close ($db); //epistrefetai o arithmos ton pistopoiitikon apo
to sygekrimeno eidos //pou exei parei o sygekrimenos xristis
apo tin arxi tou mina

if($apot[0]<50) //an einai ligotera apo ton prokathorismeno
arithmo //epistrefetai 1 alliws 0
return 1;
else
return 0;
}
function emfanisi_date($imerominia) //epistrefei tin imerominia me opia morfi
theloume gia na //tin emfanisoume
{

$temp1 = substr($imerominia, 0, 4);
$temp2 = substr($imerominia, 5, 2);
$temp3 = substr($imerominia, 8, 2);

$date1=$temp3."/".$temp2."/".$temp1;

return $date1;
}
function emf_date_sigrisi($imerominia) //epistrefei tin imerominia me opia morfi
theloume gia na //tin emfanisoume
{

$temp1 = substr($imerominia, 0, 4);
$temp2 = substr($imerominia, 5, 2);
$temp3 = substr($imerominia, 8, 2);

$date1=$temp1.$temp2.$temp3;

return $date1;
}

```

```

}

function etos_mitrwo($imerominia)
{
    $etos = substr($imerominia, 0, 4);

    return $etos;
}

function geniki_adr($name)
{
    $kataliksi= substr($name, -2);    //κοβουμε τα 2 τελευταια της καταληξης
    $name1=$name;
    $kataliksi1="";

    if($kataliksi=='ης')                //αλλάζουμε την κατάληξη
    {
        $name1= substr($name, 0, -2);    //παιρνομε ως τα 2 τελευταία
        $kataliksi1='η';
    }

    if($kataliksi=='ος')
    {
        $name1= substr($name, 0, -2);
        $kataliksi1='ου';
    }

    if($kataliksi=='ας')
    {
        $name1= substr($name, 0, -2);
        $kataliksi1='α';
    }

    $name_g=$name1.$kataliksi1;    //και τα ξαναπροσθέτουμε

    return $name_g;
}

function geniki_gyn($name)
{
    $kataliksi= substr($name, -1);
    $name1=$name;
    $kataliksi1="";

    if($kataliksi=='α')                //αλλάζουμε την κατάληξη
    {
        $name1= substr($name, 0, -1);    //παιρνομε ως τα 2 τελευταία
        $kataliksi1='ας';
    }

    if($kataliksi=='η')
    {
        $name1= substr($name, 0, -1);
        $kataliksi1='ης';
    }
}

```

```

        $name_g=$name1.$kataliksi1;

        return $name_g;
    }
function getpost($num)
{

    $ar=$num;

    return $ar;
}
function getsession($sess,$num)
{

    if($num=='2')
        $res= substr($sess, 0, 6); //epilegontai ta 5 prota psifia otan mpainoyme
apo ypalliloys // gia na min yparxei idio session_id
    else
        $res=$sess;

    return $res;

}

?>

```

Pistopoiitiko.php(Γέννηση)

```

<?php

include("../localhost/php/edimos/secure_login/checklogin.php");
require("../globals_pistop.php");
require("../functions_pistop.php");

        $protocol_numb="Πατήστε εκτύπωση"; // Η τιμη του πεδίου πριν πατηθει
η εκτυπωση
        $ar_eggrafou= "Πατήστε εκτύπωση";

    $value="0";

    if(isset($_POST['ar']))
        $value=getpost($_POST['ar']);

    $dimotis=getsession( $_SESSION['verified_user'],$value);

    $v=get_stoixeia($dimotis,"residents");
    $v1=get_stoixeia($dimotis,"malereg"); //xrisimopoiisi synartisewn gia ton
trexon user

```

```

if($value=="1") // H timi poy fernei otan patisoyme
ektiposi
    {
        $el=elegxos_p($_SESSION['verified_user'],'Π.ΓΕΝΝΗΣΗΣ');
        //echo"ar proig pistopoitika : $el";
        if($el==0)
        {
            header("Location:
            ../../secure_login/ypiresies.php?ar1=1"); //elegxos gia to posa P.Gennisis exei
            parei o sygekrimenos
            exit; //xristis
            ayto to mina.An perissotera apo 5 exoyme el==0
            } //kai
            pigainoume sto header me to analogo minima

```

```

$k=protocol_number($_SESSION['verified_user'],'Π.ΓΕΝΝΗΣΗΣ');
    $protocol_num=$k[0];
    $ar_eggrafou= $k[1];
}

```

```

$date_birth=emfanisi_date($v[8]);//emfanisi imerominias me tin apaitoumeni
morfi
$date_ktisi=emfanisi_date($v[13]);

```

```
?>
```

```

<html>
<head>
<meta http-equiv="Content-Language" content="el">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-7">
<title>ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ</title>

```

```

<SCRIPT LANGUAGE="JavaScript">
<!--

```

```

function printPage() {
    if (window.print)
        window.print()
    else
        alert("Το συστημά σας δεν υποστηρίζει την αυτόματη εκτύπωση! Θα πρέπει να
        εκτυπώσετε το παρόν έγγραφο από τον Browser σας.");
}
// -->
</SCRIPT>

```

```

Include"gennisis.php";

```

```
?>
```

Functions_oikkatast.php (Π.Οικογενειακής κατάστασης)

```
<?php
include("../functions_pistop.php");
include("../globals_pistop.php");

function find_family($familynumber_father,$value)
{
    include("//localhost/php/edimos/secure_login/connect_db.php");

    $info_father=resid_info2($familynumber_father,1);
    //echo" family number:  $familynumber_father";
    //echo" resid_father:  $info_father[0]";

    $a_ginaikes = mysql_query ("SELECT Ar_melousfemale FROM
marriage where

ResID_male='$info_father[0]'

order by Datedeglare",$db);

// epileg. tis ginaikes pou exei pantreutei o Resid

    $ar_members1=mysql_query ("SELECT ResID from residents
where familynumber_now='$familynumber_father' or
familynumber_first='$familynumber_father'", $db);

//epilegei ta ResID tw n melwn tis oikogeneias

    $ar_members2=mysql_query ("SELECT ResID_female from
marriage where familynumber='$familynumber_father'", $db);

    $ginaikes_ar=Array();
    while($row=mysql_fetch_row($a_ginaikes))//gia na pairnei o
pinakas to proto stoixeio kathe grammis
    {

        foreach($row as $v)
        {
            array_push($ginaikes_ar,$v);
        }
    }

    $count_ginaikes=count($ginaikes_ar);           //metrame
poses ginaikes exei

    $ar_telytaias=$ginaikes_ar[$count_ginaikes-1];
//epilegei ton arithmo tis pio profatis ginaikas giati ton xreiazomaste
```

```

$a_stoixeia_teleytaiou_gamou= mysql_query ("SELECT
ResID_female,Date_broken FROM marriage where

ResID_male='$info_father[0]' and Ar_melousfemale='$ar_teleytaias'", $db);

$stoixeia_teleytaiou_gamou=mysql_fetch_row($a_stoixeia_teleytaiou_gamou);

$resid_teleytaias=$stoixeia_teleytaiou_gamou[0]; //epileg. to
resid tis pio profatis

$imerominia_dialisis=$stoixeia_teleytaiou_gamou[1]; //an den
einai null o gamos exei dialythei

sizigos      if($info_father[1]== 'resdead')           //blepoume an ze i o
              $sizigos_male="ΤΕΩΣ ΣΥΖΥΓΟΥ";
              else
              $sizigos_male="ΤΟΥ ΣΥΖΥΓΟΥ";

gamos        if($imerominia_dialisis==NULL)           //den exei liksei o
              {
                $pliroforia_emfanisis=1;
                $sizigos_female="ΤΗΣ ΣΥΖΥΓΟΥ";
              }
              else
              {
                $sit=resid_info1($resid_teleytaias); //me tin sinartisi
                epistrefei to proto gramma tou resid
                if($sit=='P')                          //an exei
                pethanei
                {
                  $a_deathtime=mysql_query("select Date_death
                  from resdead where ResID='$resid_teleytaias'", $db);
                  $deathtime=mysql_fetch_row($a_deathtime);
                  //blepoume pote pethane
                  $d_time=emf_date_sigrisi($deathtime[0]);
                  // fernoyme imerom. dialisis gamou
                  $d_broken=emf_date_sigrisi($imerominia_dialisis);
                  // kai imerom. thanatou me katallili morfi
                  if($d_broken<$d_time)
                  // kai tis sigrinoyme
                  $aitia="ΔΙΑΖΥΓΙΟ";
                  //an o gamos eixe dialithe i prin to thanato
                  else
                  $aitia="ΘΑΝΑΤΟΣ ΕΝΟΣ ΜΕΛΟΥΣ";
                }
              }
              else
              $aitia="ΔΙΑΖΥΓΙΟ";
              //efoson einai se zoi exoyme diazigio

              $sizigos_female="ΤΗΣ ΤΕΩΣ ΣΥΖΥΓΟΥ";
              $pliroforia_emfanisis=0;
            }

```

```

        $male_stoixeia=get_stoixeia($info_father[0],'residents');
//stoixeia pou theloyme na emfanisoyme
        $female_stoixeia=get_stoixeia($resid_telytaias,'residents');

$gamos_stoixeia=get_stoixeia_gamou($info_father[0],$resid_telytaias,'Marriage
');
        $male_stoixeia1=get_stoixeia($info_father[0],'malereg');

        $protocol_numb="Πατήστε εκτύπωση"; // Η τιμη του πεδιου πριν
πατηθει η εκτυπωση
        $ar_eggrafou= "Πατήστε εκτύπωση";

        if($value=="1") // Η timi poy fernei
otan patisoyme ektiposi
        {

$el=elegxos_p($_SESSION['verified_user'],'ΟΙΚ.ΚΑΤΑΣΤΑΣΗΣ');
        //echo"ar proig pistopoiitika : $el";
        if($el==0)
        {
            header("Location:
../../secure_login/ypiresies.php?ar1=1"); //elegxos gia to posa P.Gennisis exei
parei o sygekrimenos
            exit; //xristis
ayto to mina.An perissotera apo 5 exoyme el==0
        } //kai
pigainoume sto header me to analogo minima

$k=protocol_number($_SESSION['verified_user'],'ΟΙΚ.ΚΑΤΑΣΤΑΣΗΣ');
        $protocol_numb=$k[0];
        $ar_eggrafou= $k[1];
        }

        include("oik1.php"); //emfanizontai ta prota stoixeia

        if($pliroforia_emfanisis==0)
            include("parat_gamoy.php"); //an exei dialythei o
gamos exoyme sxetiki paratirisi

        $resid_members1=Array(); // ta resid apo
residents tis oikog.meridas
        $resid_members2=Array(); // ta resid apo
marriage tis oikog.meridas
        while($row=mysql_fetch_row($ar_members1))//gia na pairnei o
pinakas to proto stoixeio kathe grammis
        {
            foreach($row as $v)
            {
                array_push($resid_members1,$v);
            }
        }
    }

```

```

        while($row=mysql_fetch_row($ar_members2))//για να παίρνει ο
pinakas το proto stoixeio kathe grammis
    {
        foreach($row as $v)
        {
            array_push($resid_members2,$v);
        }
    }
    $result = array_merge($resid_members1, $resid_members2);
//ta ennonoume gia na exoume to sinolo tw n resid
    $result1 = array_unique($result); //που υπηρχαν
στην οικ.μεριδα και σβηθουμε τις διπλές τιμές (* για την περιπτωση που η
χωρισμενη συζυγος
//
δεν εχει αλλαξει οικ.μεριδα οποτε τη βρικουμε και στον marriage και ston
residents)

    $ar_members=count($result1); //arithmos melon
oikogeneias(μετραμε τα ResID)

```

```

        /* echo"<br>"; //DOKIMES
        echo"$familynumber_father";
        echo"<br>";
        echo"gfgg :$resid_members[0],";echo"gfgg
:$resid_members[1],";echo"gfgg :$resid_members[2],";echo"gfgg
:$resid_members[3],";
        echo"<br>";
        echo"ar teleyt :$ar_teleytaias";
        echo"<br>";
        echo"meli :$ar_members"; */

        if($ar_members>$ar_teleytaias) //για thn teleytai sisigo kai
ta paidia tis(an exei)
        {
            include("apektisan.php");
            include("stoixeia.php");

            for($j=$ar_teleytaias+1;$j<=$ar_members;$j=$j+1)
//για tous ar_melous > tis teleytaias(ta paidia tis)
            {
                $ar=$j;

                $info_child=resid_info2($familynumber_father,$j);

                $stoixeia_paidia=get_stoixeia($info_child[0],'residents');

                $stoixeia_paidia1=get_stoixeia($info_child[0],'malereg');
                include("emfanisi_stoix.php");
                //echo" pin :$info_child[1]";
                if($info_child[1]='residents2')

```

```

        {
            $aitia_diagrafis="Μεταφορά σε νέα
μερίδα";
            include("metafora.php");
        }
        if($info_child[1]== 'resdead')
        {
            $aitia_diagrafis="Διαγραφή λόγω
θανάτου";
            include("metafora.php");
        }
        if($info_child[1]== 'resdeleted')
        {
            $aitia_diagrafis="Διαγραφή λόγω
μεταδημότευσης";
            include("metafora.php");
        }
    }
}

if($ginaikes_ar!=NULL)
{
    for($j=0;$j<$count_ginaikes-1;$j=$j+1)
    {
        $k=$ginaikes_ar[$j+1] ;

        for($i=2;$i<$k;$i=$i+1)
        {
            if($i==$ginaikes_ar[$j])
            {
                include("o_anwtero.php");
                $resid_ginaikes=
resid_ginaikas($familynumber_father,$i);

                $stoixeia_ginaikes=get_stoixeia($resid_ginaikes,'residents');

                $gamos_stoixeia1=get_stoixeia_gamou($info_father[0],$resid_ginaikes,'Marriage
');
                include("gamoi.php");
                include("stoixeia.php");
            }
            if($i>$ginaikes_ar[$j])
            {
                $ar=$i;
                $info_child=resid_info2($familynumber_father,$i);

                $stoixeia_paidia=get_stoixeia($info_child[0],'residents');

                $stoixeia_paidia1=get_stoixeia($info_child[0],'malereg');
                include("emfanisi_stoix.php");
                //echo " pin :$info_child[1]";
            }
        }
    }
}

```

```

        //echo" rid  :$info_child[0]";
        if($info_child[1]=="residents2")
        {
            $aitia_diagrafis="Μεταφορά σε νέα
μερίδα";

            include("metafora.php");
        }
        if($info_child[1]=="resdead")
        {
            $aitia_diagrafis="Διαγραφή λόγω
θανάτου";

            include("metafora.php");
        }
        if($info_child[1]=="resdeleted")
        {
            $aitia_diagrafis="Διαγραφή λόγω
μεταδημότευσης";

            include("metafora.php");
        }
    }
}
}

include("telos.php");

include("../ekt_links.php");
}

function familystate($Resid,$value) //analyei kai emfanizei tin oikogeneiaki
katastasi
{

    include("../localhost/php/edimos/secure_login/connect_db.php");

    $a_familynumber = mysql_query ("SELECT Familynumber_now FROM
residents where ResID='$Resid' ",$db);
    $familynumber_now = mysql_fetch_row($a_familynumber);          //
epileg. ton ar. oikogeneiaki meridas gia to Resid

    $b_familynumber = mysql_query ("SELECT Familynumber_first FROM
residents where ResID='$Resid' ",$db);
    $familynumber_first = mysql_fetch_row($b_familynumber);        //
epileg. ton ar. oikogeneiaki meridas gia to Resid

    $a_familystate = mysql_query ("SELECT Familystate FROM residents where
ResID='$Resid' ",$db);
    $familystate1 = mysql_fetch_row($a_familystate);                // epileg. ton
ar. oikogeneiaki meridas gia to Resid

    if(($familystate1[0]=='ΑΓΑΜΟΣ') or ($familystate1[0]=='ΑΓΑΜΗ'))
    //if(($familystate1[0]=='agamos') or ($familystate1[0]=='agami'))

```

```
{
  if( $familynumber_first[0]!=NULL)//Einai agamos kai exei allaksei
oikogeneiaki merida
  {
    find_family($familynumber_first[0],$value);
  }
  else
  {
    find_family($familynumber_now[0],$value);
  }
}
else
{
  find_family($familynumber_now[0],$value);
}

}

?>
```

Υπάλληλοι Δήμου

Διαχειριστής

Funcdiaxeir.php

```
<?php
//To arxeio auto periechi functions gia emfanisei dimotikwn diamerismatwn kai
twv tmimatwn/ypiresiwn tou dimou
function diamerismata()
{
include "db_connection.php";
//query anazitisis dim diamerismatwn
$query=mysql_query("SELECT * FROM areas");
$count=mysql_num_rows($query);
    if ($count==0)
    {
        echo "Δεν υπάρχουν Διαμερίσματα στο Δήμο<br>";
        echo "Επικοινωνήστε με τον Διαχειριστή για αρχικοποίηση του συστήματος
Ηλεκτρονικού Δήμου";
    }
    else
    {
        echo"<div align='center'>";
        echo"<table border='2' cellpadding='1' cellspacing='1'
bordercolor='#00CCFF' width=600 height=250>" ;
        echo "<tr>";
        echo "Δημοτικά Διαμερίσματα Δήμου"; //teliki fasi me $dimos
        echo "</tr>";
        echo"<tr>";
        while($row=mysql_fetch_row($query))
        {
            echo"<tr>";
            foreach($row as $v)
            {
                echo"<td> $v </td>";
            }
            echo"<tr>";
        }
        echo"</table>";
    }
}
function tmimata()
{
include "db_connection.php";
//euresi
$query=mysql_query("SELECT * FROM departments");
$count=mysql_num_rows($query);
    if ($count==0)
    {
        echo "Δεν υπάρχουν Δηλωμένα Τμήματα στον Ηλεκτρονικό Δήμο";
    }
    else
```

```

    {
    echo"<div align='center'>";
    echo"<table border='2' cellpadding='1' cellspacing='1'      width=600
height=250>" ;
    echo"<tr>";
    echo "Τμήματα/Υπηρεσίες του Ηλεκτρονικού Δήμου";
    echo"</tr>";
    echo"<tr>";
    while($row=mysql_fetch_row($query))
    {
    echo"<tr>";
    foreach($row as $v)
        {
    echo"<td> $v </td>";
        }
    echo"<tr>";
    }
    echo"</table>";
    }
}
tmimata();
?>

```

Changecoderes.php

```

<?php
include "db_connection.php";
$query=mysql_query("SELECT * FROM residents WHERE Name='$name' AND
Lname='$lname' AND Code='$pass'");
$count=mysql_num_rows($query);
if ($count==0)
{
echo "Ο τρέχων κωδικός που δώσατε είναι λανθασμένος!!";
}
else
{
$compare=strcmp($newpswd,$confirmpswd); //elegchos tou neou password kai
του confirm του
    if ($compare==0)
    {
        //anazitisi an yparxei kapou allou o kwdikos autos
        $search=mysql_query("SELECT * FROM residents WHERE
Code='$newpswd'");
        $tot=mysql_num_rows($search);
        if ($tot==0)
        {
            $updatecode=mysql_query("UPDATE residents SET Code='$neocode' WHERE
Name='$name' AND Lname='$lname' AND Fname='$fname' AND
Genosm='$genos'");
            $records=mysql_affected_rows();
            echo "Ενημερώθηκαν οι κωδικοί σε $records εγγραφές";
        }
        else
        {

```

```

        echo "Ο νέος κωδικός που δώσατε υπάρχει αλλού
καταχωρημένος!Δοκιμάστε έναν άλλον";
    }
}
else
{
    echo "Παρακαλούμε ελέγξτε τον νέο κωδικό που εισάγατε ωστέ να
συμφωνούν μεταξύ τους τα πεδία";
}
}
?>

```

Υπάλληλοι

Funcstat.php (Για στατιστικά)

```

<?php
//Το αρχείο αυτό περιέχει τις functions για την εμφάνιση της κατανομής του
πληθυσμού και των στατιστικών στοιχείων του δήμου
function sex_stat() //Η function αυτή αναφέρει την κατανομή του πληθυσμού ανά φύλο
{
include "db_connection.php";
echo "<table border='1' align='center'>";
echo "<tr><td colspan='2'>";
print "Σύμφωνα με τα στατιστικά στοιχεία του Δήμου $dimos η κατανομή του
πληθυσμού ανά φύλο έχει ως εξής:";
echo "</td>";
echo "<td>";
echo "<b>ΠΟΣΟΣΤΟ</b>";
echo "</td></tr><tr><td>";
echo "ΣΥΝΟΛΙΚΟ ΑΡΙΘΜΟΣ:";
echo "</td><td>";
$tot=mysql_query("SELECT * FROM residents WHERE ResID NOT LIKE 'P%' AND
ResID NOT LIKE 'D%'");
$tot_res=mysql_num_rows($tot);
print "<b>$tot_res</b>";
echo "</td><td>";
print "100%";
echo "</td></tr><tr><td>";
print "ΑΝΔΡΕΣ";
echo "</td><td>";
$tot_men=mysql_query("SELECT * FROM residents,malereg WHERE
residents.ResID=malereg.ResID AND residents.ResID NOT LIKE 'P%' AND
residents.ResID NOT LIKE 'D%'");
$tot_man=mysql_num_rows($tot_men);
print "<b>$tot_man</b>";
echo "</td><td>";
$pos_men=$tot_man*100/$tot_res;
print "<b>$pos_men</b>";
echo "</tr><tr><td>";
echo "ΓΥΝΑΙΚΕΣ:";
echo "</td><td>";
$tot_woman=$tot_res-$tot_man;
print "<b>$tot_woman</b>";

```

```

echo "</td><td>";
$pos_women=$tot_woman*100/$tot_res;
print "<b>$pos_women</b>";
echo "</tr><td></table>";
}
function stat_dim() //ypologismos tw n dimotikwn diamerismatwn
{
include "db_connection.php";
$d_table=mysql_query("SELECT * FROM areas");
$tot_ar=mysql_num_rows($d_table);
if ($tot_ar==0)
{
print "Δεν υπάρχουν Δημοτικά Διαμερίσματα στην Βαση μας !! Επικοινωνηστε
άμεσα με τον Διαχειριστή";
}
else
{
$d=mysql_query("SELECT DimDiamesisma FROM areas");
$a=0;
while (list($DimDiamesisma)=mysql_fetch_array($d))
{
$a++;
print "<a href=\"action.php?b=$a\">$DimDiamesisma</a> ";
echo "<br>";
}
}
}
function apot_dim($b) //emfanisi katanomis pithismou analoga to epilegen
dimotiko diamesisma
{
include "db_connection.php";
$find_dim=mysql_query("SELECT DimDiamesisma FROM areas WHERE a='$b'");
while (list($DimDiamesisma)=mysql_fetch_array($find_dim))
{
$dim=$DimDiamesisma;
$tot=mysql_query("SELECT * FROM areas,residence WHERE
areas.DimDiamesisma=residence.Placevote AND residence.Placevote='$dim'");
$synolo=mysql_num_rows($tot); //To synolo politwn ana dim diamesisma
$tot_men=mysql_query("SELECT * FROM areas,residence,malereg,residents
WHERE areas.DimDiamesisma=residence.placevote AND
residence.ResID=malereg.ResID AND residents.ResID=malereg.ResID AND
residence.placevote='$dim' AND residents.ResID NOT LIKE 'P%' AND
residents.ResID NOT LIKE 'D%'");
$synolo_men=mysql_num_rows($tot_men); //Synolo andrwn stin perioxi auti

$synolo_women=$synolo-$synolo_men; //Synolo Gynaikwn stin perioxi auti
echo "<table border=\"1\" align=\"center\">";
echo "<tr><td>";
echo "Το Σύνολο των πολιτών στο Δημοτικό Διαμέρισμα $dim είναι: $synolo";
echo "</td></tr><tr><td>";
echo "Το σύνολο των ανδρών είναι : $synolo_men";
echo "</td></tr><tr><td>";
print "Το σύνολο των γυναικών είναι : $synolo_women";
echo "</td></tr></table>";
}
}

```

```

}
function stat_dead($stryear,$sendyear)
{
include "db_connection.php";
if ($stryear<$sendyear)
{
    $stday="$stryear-01-01";
    $sedday="$sendyear-01-01";
    $query=mysql_query("SELECT Name,Lname,Lixiarxiki_praxi,Date_death
FROM residents,resdead WHERE residents.ResID LIKE 'P%' AND
residents.ResID=resdead.ResID AND resdead.Date_death BETWEEN '$stday'
AND '$sedday'");
    $tot_dead=mysql_num_rows($query);
    if ($tot_dead==0)
    {
        echo "Δεν υπάρχουν καταγεγραμμένοι Θάνατοι";
    }
    else
    {
        echo"<div align='center'>";
        echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
        echo"<tr>";
        while($row=mysql_fetch_row($query))
        {
            echo"<tr>";
            foreach($row as $v)
            {
                echo"<td> $v </td>";
            }
            echo"<tr>";
        }
        echo"</table>";
    }
    }
    else
    {
        echo "Μη έγκυρες τιμές";
        $record="ΛΑΘΟΣ_ΣΤΑΤΙΣΤΙΚΟ";
        include "wronglogin.php";
    }
}
}

```

```

function stat_birth($stryear,$sendyear) //Function gia ton
{
include "db_connection.php";
if ($stryear<$sendyear) //Elegxos egkyrotitas twv timwn pou
{
    $stday="$stryear-01-01"; //format tin arxiki imerominia
    $sedday="$sendyear-01-01" ; //Format tin teliki imerominia
    $query=mysql_query("SELECT Name,Lname,Fname,Mname,Genosm,Birth
FROM residents WHERE Birth BETWEEN '$stday' AND '$sedday'");
    $tot_birth=mysql_num_rows($query);
    if ($tot_birth==0)
    {
        echo "ΔΕΝ ΥΠΑΡΧΟΥΝ ΓΕΝΝΗΣΕΙΣ ";
    }
}
}

```

```

    }
    else
    {
        echo "ΤΟ ΣΥΝΟΛΟ ΤΩΝ ΓΕΝΝΗΣΕΩΝ ΓΙΑ ΤΗΝ ΠΕΡΙΟΔΟ $stday ΕΩΣ
$edday ΕΙΝΑΙ :<b>$tot_birth</b>";
        echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($query))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
}
}
else
{
echo "ΔΕΝ ΔΩΣΑΤΕ ΕΓΚΥΡΕΣ ΤΙΜΕΣ ";
//echo "Η ενέργεια αυτή καταγράφηκε";
$record="ΛΑΘΟΣ_ΣΤΑΤΙΣΤΙΚΟ";
include "wronglogin.php";
}
}

```

```

function stat_gamoi($stryear,$endyear) //Function
{
include "db_connection.php";
if (strlen($stryear)==4 && strlen($endyear==4) && ($stryear<$endyear))
{
$stday="$stryear-01-01";
$edday="$endyear-01-01";
$query=mysql_query("SELECT ResID_male,ResID_female FROM marriage
WHERE Date_broken IS NULL AND Datedeglare BETWEEN '$stday' and
'$edday'");
$tot_gamoi=mysql_num_rows($query);
if ($tot_gamoi==0)
{
echo "ΔΕΝ ΥΠΑΡΧΟΥΝ ΓΑΜΟΙ";
}
else
{
echo "ΤΟ ΣΥΝΟΛΟ ΤΩΝ ΓΑΜΩΝ ΓΙΑ ΤΗΝ ΠΕΡΙΟΔΟ
$stday ΕΩΣ $edday ΕΙΝΑΙ :<b> $tot_day</b>";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($query))
{

```

```

echo"<tr>";
foreach($row as $v)
    {
echo"<td> $v </td>";
    }
echo"<tr>";
}
echo"</table>";
    }
else
    {
    echo "Μη έγκυρες τιμές ημερομηνίων. Η λάθος αυτή ενέργεια καταγράφηκε
για λόγους ασφαλείας";
    $record="ΗΜΕΡΟΜΗΝΙΑ";
    include "wronglogin.php";
    }
}

function stat_divorce($stryear,$endyear)
{
include "db_connection.php";
if ($stryear<$endyear)
{
$stday="$stryear-01-01";
$edday="$endyear-01-01";
$query=mysql_query("SELECT ResID_male,ResID_female,Date_broken FROM
marriage WHERE Date_broken BETWEEN '$stday' and '$edday'");
$tot_divorces=mysql_num_rows($query);
    if($tot_divorces==0)
    {
    echo "Σύμφωνα με τα στοιχεία του Δήμου δεν υπάρχουν καταχωρήσεις
Διαζυγίων για την συγκεκριμένη περίοδο";
    }
    else
    {
    echo "Το Σύνολο των Διαζυγίων για την περίοδο $stday ΕΩΣ $edday ΕΙΝΑΙ
:<b> $tot_day</b>";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' width=600 height=250>"
;
echo"<tr>";
while($row=mysql_fetch_row($query))
{
echo"<tr>";
foreach($row as $v)
    {
echo"<td> $v </td>";
    }
echo"<tr>";
}
echo"</table>";
    }
}
else
{

```

```

echo "Μη έγκυρες τιμες ημερομηνιών<br>";
echo "Η λάθος αυτή ενέργεια καταγράφηκε για λόγους ασφαλείας";
    $record="ΛΑΘΟΣ_ΣΤΑΤΙΣΤΙΚΟ";
include "wronglogin.php";
}
}

function stat_age($choice) //Pairnei tin epilogi apo tin forma kai emfanizei auto
pou zitame
{
include ("db_connection.php");
    switch ($choice)
    {
case ΠΑΙΔΙΑ:
$up_limit=0;
$down_limit=13;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");
$count=mysql_num_rows($result);
//ΕΜΦΑΝΙΣΗ
echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250> ";
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
case ΕΦΗΒΟΙ:
$up_limit=14;
$down_limit=18;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");
$count=mysql_num_rows($result);
//ΕΜΦΑΝΙΣΗ
echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";

```

```

echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
    case NEOI:
$up_limit=19;
$down_limit=25;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");
$count=mysql_num_rows($result); //synolo newn residents
//ΕΜΦΑΝΙΣΗ
echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
    case ΕΝΗΛΙΚΕΣ:
$up_limit=26;
$down_limit=37;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");
$count=mysql_num_rows($result);
//ΕΜΦΑΝΙΣΗ

```

```

echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
case ΜΕΣΗΛΙΚΕΣ:
$up_limit=38;
$down_limit=49;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");
$count=mysql_num_rows($result);
//ΕΜΦΑΝΙΣΗ
echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>" ;
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
case ΗΛΙΚΙΩΜΕΝΟΙ:
$up_limit=50;
$down_limit=130;
$down_year= mktime(0, 0, 0, date("m"), date("d"), date("Y")-$down_limit );
$downyear = date("Y-m-d", $down_year );
$up_year = mktime(0, 0, 0, date("m"), date("d"), date("Y")-$up_limit );
$upyear = date("Y-m-d", $up_year );
$result= mysql_query ("SELECT Name,Lname,Birth FROM residents WHERE Birth
BETWEEN '$downyear' AND '$upyear' AND ResID NOT LIKE 'P%' AND ResID
NOT LIKE 'D%'");

```

```

$count=mysql_num_rows($result);
//ΕΜΦΑΝΙΣΗ
echo"<h3 align='center'>Σύμφωνα με τα στοιχεία του Δήμου το σύνολο των
ατόμων ηλικίας απο $up_limit εως $down_limit ετών είναι :$count</h3> ";
echo"<div align='center'>";
echo"<table border='2' cellpadding='1' cellspacing='1' bordercolor='#00CCFF'
width=600 height=250>";
echo"<tr>";
while($row=mysql_fetch_row($result))
{
echo"<tr>";
foreach($row as $v)
{
echo"<td> $v </td>";
}
echo"<tr>";
}
echo"</table>";
break;
}
}
?>

```

Valid_new_record.php(Εισαγωγή νέου μέλους)

```

<?php
//Sto arxeio auto ginetai elegchos gia tin egkyrotita tis neas eggrafis
if (empty($katoikia)) //elegchos an exoun symplirwthei ola ta pedia
{
echo "ΠΑΡΑΚΑΛΟΥΜΕ ΣΥΜΠΛΗΡΩΣΤΕ ΟΛΑ ΤΑ ΠΕΔΙΑ";
EXIT;
}
else
{
include "eggrafes.php";//elegxo swstwn timwn pou pairnoun stin basi
//stin synexeia ginetai elegchos an yparxei idi i eggrafi
include "db_connection.php";
$query=mysql_query("SELECT ResID FROM residents WHERE
Name='$newval[4]' AND Lname='$newval[3]' AND Fname='$newval[5]' AND
Mname='$newval[7]' AND Genosm='$newval[8]'");
$result=mysql_num_rows($query);
if ($result==0)
{
$birthres="$newval[12]-$newval[11]-$newval[10]"; //Imerominia
gennisis
$imerktisi="$newval[19]-$newval[18]-$newval[17]"; //Imerominia
ktisis ithageneias
//ginetai elegchos tou pediou me to filo an einai arren i gynaika
if ($newval[22]=="APPEN")
{
$query=mysql_query("SELECT MAX(Malenumb) FROM malereg WHERE
Maleyear='$newval[12]'");
$search=mysql_num_rows($query);
if ($search==0)
{

```

```

    $newmale=1;
  }
  else
  {
    while(list($Malenumb)=mysql_fetch_array($query))
    {
      $newmale=$Malenumb+1; //Au3isi kata 1 tou neou male number
    }
  }
//Kataxwrisi neas eggrafis mitrwou arrenwn to 1,1 tha ta pairnei apo to arxeio
db_connection
$record_male=mysql_query("INSERT INTO malereg VALUES
('$newval[0]','1','1','$newmale','$newval[12]')");
//Kataxwrisi neas eggrafis dimotologiou
$query=mysql_query("INSERT INTO residents
(Resid,Name,Lname,Fname,Flname,Mname,Genosm,Familystate,Birth,Birthplace,Birt
h_nomos,Birth_xwra,Citizenship,Datektisi,Wayktisi,Religion,Familynumber_now,Ar_m
elous_now) VALUES
('$newval[0]','$newval[4]','$newval[3]','$newval[5]','$newval[6]','$newval[7]','$newv
al[8]','$newval[9]','$birthres','$newval[13]','$newval[14]','$newval[15]','$newval[16]',
'$imerktisi','$newval[20]','$newval[21]','$newval[2]','$newval[1]')");
//Kataxwrisi stin synexeia sto pinaka areas mia akomi eggrafi kai telos sto residence
$areas=mysql_query("SELECT Totalresidents FROM areas WHERE
Dimdiameterisma='$newval[23]");
while (list($Totalresidents)=mysql_fetch_array($areas))
{
  $new_res=$Totalresidents+1; //Au3isi kata 1 tous totalresidents tis perioxis
}
$update_areas=mysql_query("UPDATE areas SET Totalresidents='$new_res' WHERE
Dimdiameterisma='$newval[23]");
$new_residence=mysql_query("INSERT INTO residence (ResID,Placevote) VALUES
('$newval[0]','$newval[23]");
//kai episis kataxwrisi sto residence
}
else
{
  //Stin periptwsi pou einai gynaika apla ginetai nea kataxwrisi sto dimotologio
  $query=mysql_query("INSERT INTO residents
(Resid,Name,Lname,Fname,Flname,Mname,Genosm,Familystate,Birth,Birthplace,Birt
h_nomos,Birth_xwra,Citizenship,Datektisi,Wayktisi,Religion,Familynumber_now,Ar_m
elous_now) VALUES
('$newval[0]','$newval[4]','$newval[3]','$newval[5]','$newval[6]','$newval[7]','$newv
al[8]','$newval[9]','$birthres','$newval[13]','$newval[14]','$newval[15]','$newval[16]',
'$imerktisi','$newval[20]','$newval[21]','$newval[2]','$newval[1]')");
  //Kataxwrisi stin synexeia sto pinaka areas mia akomi eggrafi kai telos sto residence
  $areas=mysql_query("SELECT Totalresidents FROM areas WHERE
  Dimdiameterisma='$newval[23]");
  while (list($Totalresidents)=mysql_fetch_array($areas))
  {
    $new_res=$Totalresidents+1; //Au3isi kata 1 tous totalresidents tis perioxis
  }
  $update_areas=mysql_query("UPDATE areas SET Totalresidents='$new_res' WHERE
  Dimdiameterisma='$newval[23]");
  $new_residence=mysql_query("INSERT INTO residence (ResID,Placevote) VALUES
  ('$newval[0]','$newval[23]");
  //Omoiws kataxwrisi tis neas eggrafis sto pinaka residence

```

```
}
    }
    else
    {
        echo "Η ΕΓΓΡΑΦΗ ΠΟΥ ΠΡΟΣΠΑΘΕΙΤΕ ΝΑ ΚΑΤΑΧΩΡΗΣΕΤΕ ΥΠΑΡΧΕΙ
        ΗΔΗ ΣΤΑ ΑΡΧΕΙΑ ΤΟΥ ΔΗΜΟΤΟΛΟΓΙΟΥ";
        $record="ΥΠΑΡΧΟΥΣΑ_ΕΓΓΡΑΦΗ";
        include "wronglogin.php";//Η ενεργεια auti katagrafike gia logous
        asfaleias edw ena pedio tha mpainei automata pou tha lee ti lathos einai
    }
    EXIT;
}
?>
```