

321-88103 - Προγραμματισμός στο Διαδίκτυο

ΑΠΑΛΛΑΚΤΙΚΗ ΕΡΓΑΣΙΑ ΜΑΘΗΜΑΤΟΣ

ΕΙΔΟΣ: ΟΜΑΔΙΚΗ (ΟΜΑΔΕΣ ΜΕΧΡΙ 3 ΑΤΟΜΑ)

ΠΡΟΘΕΣΜΙΑ: 02/02/2025

ΣΤΟΧΟΙ

Οι στόχοι της εργασίας είναι: (α) η εξοικείωση των φοιτητών με τις τεχνολογίες προγραμματισμού στο διαδίκτυο και (β) η συνεργασία μεταξύ τους στα πλαίσια ενός έργου λογισμικού. Με το πέρας της εργασίας, οι φοιτητές θα έχουν την εμπειρία και τις ικανότητες να σχεδιάζουν και να υλοποιούν εφαρμογές ιστού (web applications) που βασίζονται σε αρχιτεκτονική 3-tier με την χρήση μοντέρνων τεχνολογιών και κατάλληλων γλωσσών επισήμανσης και προγραμματισμού.

ΘΕΜΑ

Το θέμα της εργασίας αφορά την σχεδίαση και υλοποίηση μιας εφαρμογής ιστού για την διαχείριση ιατρείου. Ειδικότερα, η εφαρμογή θα πρέπει να επιτρέπει την διαχείριση της πληροφορίας ασθενών, του ιστορικού τους και των ραντεβού τους με τον/την ιατρό. Η εφαρμογή θα μπορεί να χρησιμοποιηθεί από ασθενείς, τις γραμματείες του ιατρείου και τον/την ιατρό.

ΑΠΑΙΤΗΣΕΙΣ

ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ ΕΦΑΡΜΟΓΗΣ

Η διαχείριση των ασθενών θα αφορά τις ακόλουθες λειτουργίες:

- **εισαγωγή νέου ασθενούς:** εδώ θα πρέπει να εισάγεται η απαιτούμενη πληροφορία με βάση συγκεκριμένα πεδία φόρμας. Τα πεδία αυτά περιγράφονται λεπτομερώς στην ενότητα Μοντέλο Δεδομένων & Απαιτήσεις Επεξεργασίας Δεδομένων. Ο νέος ασθενής θα αποθηκεύεται στην εφαρμογή μόνο εφόσον είναι μοναδικός (όσον αφορά το ΑΜΚΑ του) και υπάρχουν έγκυρες τιμές στα πεδία του.
- **εισαγωγή πολλαπλών ασθενών:** ικανότητα εισαγωγής της πληροφορίας για πολλούς νέους ασθενείς με βάση ένα αρχείο CSV ή Excel. Η πληροφορία για κάθε ασθενή θα περιλαμβάνεται σε ξεχωριστή, αντίστοιχη γραμμή του αρχείου. Αν ο ασθενής υπάρχει ήδη ή υπάρχουν λάθη στις τιμές των πεδίων του, τότε η αντίστοιχη γραμμή του αρχείου θα αγνοείται.
- **εμφάνιση ασθενούς:** θα πρέπει να εμφανίζονται κατάλληλα όλες οι πληροφορίες για έναν ασθενή, εφόσον αυτός επιλεγεί από τον χρήστη. Η επιλογή αυτή θα

μπορούσε να γίνει στα πλαίσια μιας αναζήτησης ή στα πλαίσια εμφάνισης της λίστας των ασθενών. Προφανώς, κατά την εμφάνιση ενός ασθενούς, ο χρήστης, ανάλογα με τον ρόλο που παίζει, θα μπορεί επίσης να τον διαγράψει ή να τον τροποποιεί. Η εμφάνιση των πληροφοριών του ασθενούς μπορεί να περιλαμβάνει και την εμφάνιση του ιστορικού του. Το τελευταίο θα μπορεί να ιδωθεί μόνο από τον ιατρό και τον ίδιο τον ασθενή αλλά όχι από την γραμματέα.

- *ενημέρωση ασθενούς*: η λειτουργία αυτή είναι παρόμοια με την προηγούμενη ως προς την εμφάνιση. Η διαφορά είναι πως τα πεδία που θα εμφανίζονται θα έχουν ήδη τιμές, αυτές που είχαν δοθεί κατά την εισαγωγή του ασθενούς. Η τροποποίηση των τιμών των πεδίων ενός ασθενούς θα πρέπει να ελέγχεται ώστε μόνο έγκυρες τιμές να αποθηκεύονται στην εφαρμογή.
- *αναζήτηση ασθενούς*: η λειτουργία αυτή αφορά την αναζήτηση ενός ασθενούς με βάση το επίθετό του ή/και το ΑΜΚΑ του. Το αποτέλεσμα της αναζήτησης θα είναι η εμφάνιση μιας λίστας από ασθενείς.
- *διαγραφή ασθενούς*: αν και όχι τόσο συχνή ως λειτουργία, θα επιτρέπεται να πραγματοποιείται από μια γραμματέα ή τον ιατρό.

Παρομοίως, η διαχείριση του ιστορικού των ασθενών θα περιλαμβάνει τις εξής λειτουργίες:

- *εισαγωγή νέας εγγραφής*: Κατά τη διάρκεια ή μόλις ολοκληρωθεί η επίσκεψη και η αντίστοιχη εξέταση ενός ασθενούς από τον/την ιατρό, τότε θα μπορεί ο/η ιατρός να καταχωρήσει μια νέα εγγραφή στο ιστορικό του ασθενούς που θα αφορά τα προβλήματα υγείας που ανίχνευσε και την θεραπεία που συστήνει. Τα πεδία μιας εγγραφής ιστορικού ασθενούς περιγράφονται λεπτομερώς στην ενότητα Μοντέλο Δεδομένων & Απαιτήσεις Επεξεργασίας Δεδομένων.
- *μαζική εισαγωγή ιστορικού ασθενούς*: θα είναι δυνατή η μαζική εισαγωγή εγγραφών ιστορικού για ένα ασθενή από ένα αρχείο τύπου CSV ή Excel. Σε κάθε γραμμή του αρχείου αυτού θα περιγράφεται και μια εγγραφή ιστορικού. Σε περίπτωση «προβληματικής» εγγραφής (πχ. με μη έγκυρα δεδομένα), αυτή θα αγνοείται. Η μαζική εισαγωγή έχει νόημα και θα πρέπει να επιτρέπεται μόνο εφόσον το ιστορικό του ασθενούς είναι κενό.
- *ανανέωση εγγραφής ιστορικού*: θα μπορεί να αλλάζουν τα πεδία μόνο για την πιο πρόσφατη εγγραφή ιστορικού του ασθενούς.
- *διαγραφή εγγραφής ιστορικού*: θα μπορεί να πραγματοποιείται η διαγραφή της πιο πρόσφατης εγγραφής στο ιστορικό του ασθενούς.
- *εμφάνιση εγγραφής ιστορικού*: εμφάνιση λεπτομερειών για μια συγκεκριμένη εγγραφή στο ιστορικό ασθενούς, εφόσον την επιλέξει ο χρήστης.
- *εμφάνιση ιστορικού*: θα είναι δυνατή η εμφάνιση όλης της λίστας των εγγραφών του ιστορικού ασθενούς σε φθίνουσα χρονική σειρά (δηλ. από την τελευταία εγγραφή μέχρι την πρώτη). Έπειτα, ο χρήστης μπορεί να επιλέγει μια εγγραφή προς εμφάνιση.
- *αναζήτηση ιστορικού*: θα μπορεί ο ιατρός ή ο ασθενής να αναζητήσει μία ή περισσότερες εγγραφές από το ιστορικό του (ασθενούς) με βάση ένα εύρος ημερομηνιών ή/και το πρόβλημα υγείας που ανιχνεύθηκε.

Τέλος, θα πρέπει να υποστηρίζεται η διαχείριση ραντεβού ασθενών με τις εξής λειτουργίες προς υλοποίηση:

- *εισαγωγή νέου ραντεβού*: εισαγωγή νέου ραντεβού ασθενούς σε ημερομηνία και ώρα όπου ο ιατρός είναι διαθέσιμος.
- *ανανέωση ραντεβού*: θα είναι δυνατή η τροποποίηση των εξής στοιχείων: ημερομηνίας, ώρας ραντεβού, και της κατάστασης του ραντεβού. Τα δύο πρώτα στοιχεία θα μπορούν να αλλάζουν από τον ασθενή ή μια γραμματέα ή τον ιατρό ενώ το τελευταίο μόνο από μια γραμματέα ή τον ιατρό.
- *ακύρωση ραντεβού*: θα είναι δυνατή η ακύρωση ενός καταχωρημένου ραντεβού εφόσον δεν έχει πραγματοποιηθεί η επίσκεψη από τον χρήστη.
- *εμφάνιση ραντεβού*: εμφάνιση λεπτομερειών για ένα συγκεκριμένο ραντεβού.
- *αναζήτηση ραντεβού*: εφόσον ο χρήστης δεν δώσει εύρος ημερομηνιών ή επίθετο ή ΑΜΚΑ ασθενούς ή κατάσταση ραντεβού, τότε προκαθορισμένα θα του εμφανίζονται τα εκκρεμή ραντεβού της τρέχουσας ημέρας. Διαφορετικά, θα εμφανίζονται τα ραντεβού που ικανοποιούν τα κριτήρια αναζήτησης του χρήστη.
- *προσδιορισμός slots διαθεσιμότητας ιατρού*: ο ιατρός θα μπορεί να δηλώνει επιλέγοντας μια συγκεκριμένη ημερομηνία, τα slots όπου θα είναι διαθέσιμος. Προφανώς, μπορεί και να τα αλλάζει (δηλ. να προσδιορίζει σε ένα νέο slot πως είναι διαθέσιμος ή σε υπάρχων slot πως δεν είναι πλέον διαθέσιμος), εφόσον το επιθυμεί. Εναλλακτικά, η ίδια λειτουργία μπορεί να υποστηριχθεί μαζικά μέσω της χρήσης ενός CSV ή Excel αρχείου. Σημειώνεται πως αν ο ιατρός δεν το κάνει αυτό για μια ημέρα, τότε η μέρα αυτή δεν θα είναι διαθέσιμη για κλείσιμο/δημιουργία ραντεβού. Εναλλακτικά, μπορείτε να θεωρήσετε πως ο ιατρός είναι προκαθορισμένα διαθέσιμος μόνο Δευτέρα έως Παρασκευή σε εργάσιμες ώρες.

Για όλες τις σχετικές, προαναφερόμενες λειτουργίες, σε περίπτωση αποστολής φόρμας, θα πρέπει να ελέγχονται τα πεδία ως προς: (α) ύπαρξη τιμών για τα υποχρεωτικά πεδία, (β) την εγκυρότητα τιμών που παρέχονται τόσο για υποχρεωτικά όσο και προαιρετικά πεδία ενώ (γ) θα πρέπει να ελέγχεται και το URL αποστολής για τον αποκλεισμό παράνομων μεμονωμένων καθώς και συνόλων χαρακτήρων (ώστε να αποφύγουμε επιθέσεις XSS).

ΥΠΟΣΤΗΡΙΞΗ ΠΟΛΛΑΠΛΩΝ ΡΟΛΩΝ

Θα υπάρχουν 4 ρόλοι στο σύστημα με διαφορετικές αρμοδιότητες και ικανότητες, οι οποίοι θα μπορούν να βλέπουν διαφορετικές πλευρές των δεδομένων/οντοτήτων που αποθηκεύονται. Οι ρόλοι αυτοί είναι οι ακόλουθοι:

- *επισκέπτης (μη ταυτοποιημένος χρήστης)*: θα μπορεί να βλέπει μόνο ορισμένες γενικές πληροφορίες για το ιατρείο καθώς και τις σελίδες εγγραφής και αυθεντικοποίησης.
- *ασθενής*: θα μπορεί να δημιουργεί, τροποποιεί, ακυρώνει και βλέπει τα δικά του ραντεβού μόνο. Επομένως, όταν αναζητά ραντεβού, θα του επιστρέφεται μόνο μια λίστα με τα δικά του ραντεβού. Θα μπορεί να βλέπει και να αλλάζει τις πληροφορίες που τον αφορούν (στοιχεία ασθενούς). Τέλος, θα μπορεί να βλέπει το ιστορικό του και να πλοηγείται στις εγγραφές του (μόνο προς ανάγνωση).

- **γραμματέας:** θα μπορεί να διαχειρίζεται ραντεβού και τις πληροφορίες των ασθενών. Δεν θα μπορεί να δει καθόλου (και άρα να διαχειριστεί) το ιστορικό των ασθενών.
- **ιατρός:** Ουσιαστικά παίζει το ρόλο του υπερ-χρήστη του συστήματος διότι μπορεί να διαχειριστεί όλες τις βασικές οντότητες (ασθενής, ιστορικό, ραντεβού).

Ένας χρήστης θα μπορεί να παίζει μόνο έναν ρόλο. Ακόμη, θα πρέπει να υποστηρίζεται τόσο η σύνδεση όσο και η αποσύνδεση των χρηστών. Όταν αποσυνδέεται ένας χρήστης, τότε θεωρείται ότι παίζει το ρόλο του επισκέπτη. Η αποσύνδεση μπορεί να πραγματοποιείται ρητά μέσω της διεπαφής της εφαρμογής καθώς και χρονικά με τη λήξη της αντίστοιχης συνόδου. Η χρονική διάρκεια μιας συνόδου μπορεί να αποτελεί μια παράμετρος διαμόρφωσης της εφαρμογής.

Ο παρακάτω πίνακας συνοψίζει τα δικαιώματα πρόσβασης του κάθε ρόλου ως προς τις λειτουργίες (διαχείρισης) που θα πρέπει να υλοποιηθούν.

Ρόλος	Λειτουργίες
Επισκέπτης	Εμφάνιση γενικών πληροφοριών ιατρείου + ιστοσελίδων εγγραφής και αυθεντικοποίησης
Ασθενής	Ότι και ο επισκέπτης + Διαχείριση των ραντεβού του + Διαχείριση της πληροφορίας του + Εμφάνιση/Αναζήτηση του ιστορικού του
Γραμματέας	Διαχείριση Ραντεβού + Διαχείριση Ασθενών
Ιατρός	Όλες

ΜΟΝΤΕΛΟ ΔΕΔΟΜΕΝΩΝ & ΑΠΑΙΤΗΣΕΙΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ

Οι βασικές οντότητες στο σύστημά μας είναι οι εξής:

- **Χρήστης:** για αυτόν θα πρέπει να κρατάμε κυρίως πληροφορία διαπίστευσης, δηλαδή το *email & password* του. Θα πρέπει να έχει προφανώς *ονοματεπώνυμο* και *αριθμό ταυτότητας* (ΑΤ). Τέλος, θα πρέπει να αντιστοιχίζεται σε ένα ρόλο από τους εγγεγραμμένους (δηλ. ασθενούς, γραμματέα, ιατρού). Όλα τα πεδία είναι υποχρεωτικά.
- **Ασθενής:** Για έναν ασθενή, θα πρέπει να διατηρήσουμε την εξής έξτρα αλλά υποχρεωτική πληροφορία (σε σχέση με έναν χρήστη): *ΑΜΚΑ & ημερομηνία εγγραφής*. Η *ημερομηνία εγγραφής* θα παράγεται από το σύστημα αυτόματα όταν ολοκληρωθεί επιτυχώς η εγγραφή του ασθενούς.
- **Ιατρός:** Για τον ιατρό θα πρέπει να διατηρούμε την εξής έξτρα αλλά υποχρεωτική πληροφορία εκτός από αυτήν του χρήστη: *ειδικότητα ιατρού* (πχ. *ορθοπεδικός*).
- **Ραντεβού:** αναπαριστά ένα ραντεβού ασθενούς στο ιατρείο. Για ένα ραντεβού θα πρέπει να καταχωρήσουμε τα εξής: *αναγνωριστικό, ημερομηνία, ώρα, λόγος ραντεβού & ημερομηνία δημιουργίας*. Όλα τα πεδία είναι υποχρεωτικά ενώ το τελευταίο θα πρέπει να παράγεται αυτόματα από το σύστημα.
Επίσης, θα πρέπει να διατηρείται η *κατάσταση* ενός ραντεβού. Μόλις το ραντεβού δημιουργηθεί, η κατάστασή του είναι *Δημιουργημένο*. Εφόσον ο ασθενής πάει στο ραντεβού, η κατάσταση θα γίνει *Τηρημένο*. Μόλις ολοκληρωθεί η επίσκεψη του ασθενούς, τότε η κατάσταση θα γίνει *Ολοκληρωμένο*. Αν το ραντεβού είναι

Δημιουργημένο και είτε ακυρωθεί από έναν χρήστη ή παρέλθει η ημερομηνία του χωρίς την τήρησή του (από τον ασθενή), τότε το ραντεβού θα γίνει *Ακυρωμένο*. Οι καταστάσεις *Ολοκληρωμένο & Ακυρωμένο* είναι τελικές.

- *Ιστορικό*: αναπαριστά το ιστορικό ενός ασθενούς. Το ιστορικό ουσιαστικά έχει ως γνώρισμα μόνο το *αναγνωριστικό* του. Αλλά περιλαμβάνει και μια λίστα από εγγραφές ιστορικού, οι οποίες αναλύονται στην επόμενη οντότητα.
- *Εγγραφή ιστορικού*: αναπαριστά μια εγγραφή στο ιστορικό ενός ασθενούς. Η εγγραφή προφανώς αφορά μια συγκεκριμένη επίσκεψη στο ιατρείο. Για μια εγγραφή, θα πρέπει να καταχωρήσουμε τα εξής: *αναγνωριστικό εγγραφής, ημερομηνία δημιουργίας, ανιχνευμένα προβλήματα υγείας και θεραπεία*. Όλα τα πεδία είναι υποχρεωτικά. Η *ημερομηνία δημιουργίας* θα πρέπει να κατασκευάζεται αυτόματα από το σύστημα.
- *Διαθεσιμότητα ιατρού*: περιλαμβάνει μια σειρά από *slots* όπου ο ιατρός θα είναι διαθέσιμος. Για απλότητα, μπορείτε να θεωρήσετε πως ένα slot είναι ένα *dateTime* (δηλ. αναπαριστά ώρα και ημερομηνία μαζί) και πως η διάρκειά του είναι σταθερή (πχ. μισή ώρα).

Θα πρέπει να επισημανθεί πως πρέπει να ελέγχεται η μοναδικότητα μιας εγγραφής/σειράς δεδομένων (όπως είναι ένας ασθενής) ώστε να αποφεύγεται η αποθήκευση διπλο-εγγραφών στη βάση δεδομένων.

Επιπλέον, θα πρέπει να αποτρέπετε την πραγματοποίηση επιθέσεων SQL injection μέσω της χρήσης της κατάλληλης τεχνολογίας διασύνδεσης βάσεων δεδομένων (δηλαδή την χρήση prepared statements).

ΟΜΟΙΟΜΟΡΦΗ, ΧΡΗΣΤΙΚΗ ΚΑΙ ΑΠΟΚΡΙΤΙΚΗ ΔΙΕΠΑΦΗ ΧΡΗΣΗΣ

Θα πρέπει να σχεδιαστεί κατάλληλα το (HTML) περιεχόμενο της κάθε ιστοσελίδας της εφαρμογής ιστού και να εμφανίζεται είτε στην Αγγλική ή στην Ελληνική γλώσσα. Θα πρέπει να υπάρχει ένα ενιαίο στυλ εμφάνισης μέσω της χρήσης CSS. Επίσης, θα πρέπει οι ιστοσελίδες να φορτώνονται γρήγορα. Αυτό σημαίνει ότι θα πρέπει να υπάρχει ένα είδος ισορροπίας όσον αφορά την επικοινωνία με τον εξυπηρετητή μέσω της χρήσης της τεχνολογίας AJAX (συνεπώς, θα πρέπει να χρησιμοποιείται και Javascript στην πλευρά του πελάτη). Οι ιστοσελίδες θα πρέπει να είναι ευανάγνωστες και να εμφανίζονται χωρίς προβλήματα για διάφορα είδη συσκευών (desktops, laptops, tablets, κ.α.). Επιπλέον, θα πρέπει ο χρήστης να επιτελεί μια λειτουργία που παρέχεται από την εφαρμογή με τις λιγότερες δυνατές ενέργειες.

Όλες οι ιστοσελίδες θα πρέπει ευδιάκριτα και καταλλήλως να εμφανίζουν πληροφορία για το ιατρείο (π.χ., όνομα ιατρείου συν logo), καθώς και για τον χρήστη, εφόσον αυτός έχει ταυτοποιηθεί (πχ. το ονοματεπώνυμό του).

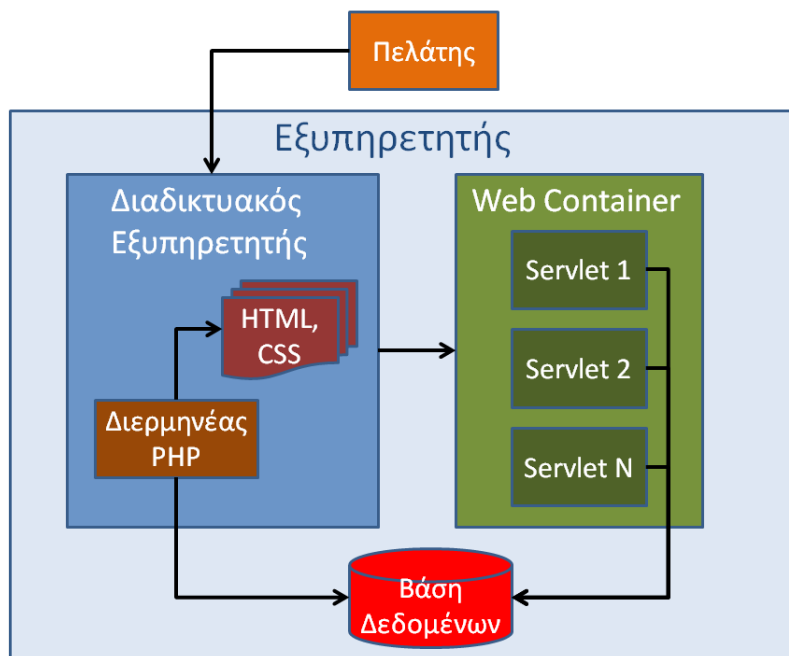
Σε κάθε ιστοσελίδα, οι πληροφορίες που παρέχονται θα πρέπει να δομούνται και να κατηγοριοποιούνται κατάλληλα. Καλό είναι όλες οι ιστοσελίδες να έχουν κάποια κοινά σταθερά μέρη και κάποια άλλα δυναμικά που θα αλλάζουν ανάλογα με την εκάστοτε ιστοσελίδα. Σε κάθε περίπτωση, επιβάλλεται ο χωρισμός της ιστοσελίδας σε διάφορα μέρη ανάλογα με το είδος της λειτουργικότητας που θα παρέχεται από το κάθε μέρος.

Σε περιπτώσεις λάθους (π.χ., στην αποστολή δεδομένων φορμών), ο χρήστης θα πρέπει να ενημερώνεται καταλλήλως με βάση το είδος του λάθους και να οδηγείται ως προς την σωστή συμπεριφορά (π.χ., αν γίνεται κάποιο λάθος σε πεδίο φόρμας, τότε θα μηδενίζεται το περιεχόμενο του πεδίου και θα εμφανίζεται το αντίστοιχο μήνυμα λάθους με κόκκινο χρώμα στη κατάλληλη θέση δίπλα στο / ή πάνω από το πεδίο). Καλό είναι στα πεδία φόρμας να παρέχεται και κάποιο είδος επεξήγησης (με οποιοδήποτε τρόπο πιστεύετε ότι είναι κατάλληλος) ώστε να οδηγείται ο χρήστης στην σωστή συμπλήρωση της τιμής τους.

Τέλος, θα πρέπει παρέχεται ένας κατάλληλος τρόπος περιήγησης τόσο στο εσωτερικό της ιστοσελίδας όσο και κατά μήκος των ιστοσελίδων με την χρήση κατάλληλων συνδέσμων, αγκύρων (anchors) & fragments. Θα πρέπει να εμφανίζονται με διαφορετικά χρώματα τους συνδέσμους που έχει ήδη επισκεφθεί ο χρήστης και αυτούς που δεν έχει επισκεφθεί.

ΚΑΤΑΛΛΗΛΗ ΑΡΧΙΤΕΚΤΟΝΙΚΗ & ΥΛΟΠΟΙΗΣΗ

Η εφαρμογή ιστού θα πρέπει να ακολουθήσει προφανώς μια αρχιτεκτονική 3-tier που θα χωρίζεται στα κομμάτια/επίπεδα του πελάτη, εξυπηρετητή και (βάσης) δεδομένων. Στην πλευρά του εξυπηρετητή θα πρέπει να υπάρχει σίγουρα ένας διαδικτυακός εξυπηρετητής (web server) και, ανάλογα με το είδος της υλοποίησης της λογικής της εφαρμογής, ένας web container (αν χρησιμοποιηθούν Java Servlets) ή ένας PHP Interpreter (ο οποίος μπορεί να παρέχεται με την μορφή module ή εκτελέσιμου CGI) ή και τα δύο. Συνεπώς, η αρχιτεκτονική της εφαρμογής θα έχει την μορφή που επιδεικνύεται στην παρακάτω εικόνα.



Εικόνα 1 - Πρότυπη αρχιτεκτονική διαδικτυακής εφαρμογής

Όσον αφορά την υλοποίηση, από την πλευρά του πελάτη, θα πρέπει να χρησιμοποιηθεί Javascript και η τεχνολογία AJAX. Από την πλευρά του εξυπηρετητή, μπορεί να χρησιμοποιηθεί η PHP, η Java (servlets ή ακόμη και REST υπηρεσίες), η Javascript (Node.js) ή οποιαδήποτε άλλη γλώσσα της προτίμησής σας, αρκεί να είναι συμβατή με τον

διαδικτυακό εξυπηρετητή. Ο τελευταίος μπορεί να είναι ο Apache ή οποιοσδήποτε άλλος όπου, όμως, θα πρέπει να εξυπηρετεί στατικό περιεχόμενο αλλά και να παράγει και δυναμικό σε συνεργασία με το κατάλληλο συστατικό μέρος (όπως είναι ο Διερμηνέας PHP ή ένας web container). Μπορεί όμως να είναι απλώς και μια διαδικασία Node.js, η οποία καλύπτει τόσο το στατικό όσο και το δυναμικό περιεχόμενο. Προτείνεται αλλά δεν επιβάλλεται ο web container να είναι ο Tomcat. Τέλος, προτείνεται αλλά δεν επιβάλλεται να χρησιμοποιήσετε σχεσιακή βάση δεδομένων (κατά προτίμηση την MySQL).

Εφόσον χρησιμοποιήσετε τον Apache, τότε προτείνεται η χρήση του XAMPP ώστε να επωφεληθείτε από την δυνατότητα ελέγχου και διάταξης επιπλέον συστατικών μερών όπως είναι ο tomcat ως web container ή η MySQL ως σχεσιακή βάση δεδομένων. Τέλος, είστε ελεύθεροι να χρησιμοποιήσετε οποιοδήποτε κειμενογράφο ή πλαίσιο ανάπτυξης εφαρμογών ιστού επιθυμείτε. Σε περίπτωση χρήσης Java, προτείνεται η χρήση του Netbeans ή του Eclipse ως IDE.

Να σημειωθεί ότι θα εκτιμηθούν δεόντως προσπάθειες αυτοματισμού της διαμόρφωσης και διάταξης του backend της εφαρμογής.

ΠΑΡΑΔΟΤΕΑ

Θα πρέπει να αποσταλεί ένα συμπιεσμένο αρχείο zip από το ένα από τα τρία ή δύο μέλη μιας ομάδας (εκτός και αν η ομάδα αποτελείται από ένα άτομο, οπότε μόνο αυτό είναι υπεύθυνο για την αποστολή των παραδοτέων) που θα έχει όνομα <AM1>_<AM2>_<AM3>.zip για μια ομάδα τριών ατόμων, <AM1>_<AM2>.zip για μια ομάδα δύο ατόμων και <AM>.zip για ομάδα ενός ατόμου. Το συμπιεσμένο αρχείο θα περιλαμβάνει:

- Μια αναφορά σε μορφοποίηση PDF που θα εξηγεί: την αρχιτεκτονική της διαδικτυακής εφαρμογής, τον τρόπο διαμόρφωσης, διάταξης και εκτέλεσής της (δηλ. αντίστοιχες οδηγίες), πως αντιμετωπίστηκαν όλες οι απαιτήσεις της εφαρμογής, σε ποια μέρη της εφαρμογής συνεισέφεραν τα μέλη της ομάδας (αυτό δεν απαιτείται για ομάδες ενός ατόμου) και σε ποιο βαθμό, και ποια είναι η εμπειρία που εισέπραξαν τα μέλη της ομάδας.
- Τον πηγαίο κώδικα της εφαρμογής δομημένο με κατάλληλο και ιεραρχικό τρόπο, ο οποίος θα πρέπει να περιλαμβάνει διάφορα είδη αρχείων που θα αφορούν το περιεχόμενο (π.χ. HTML), το στυλ (CSS) και την λειτουργικότητα/συμπεριφορά της εφαρμογής (π.χ. PHP, Java, Javascript).
- Το σχήμα της βάσης δεδομένων μαζί με τα δεδομένα που καταχωρήθηκαν σε αυτήν για την επίδειξη της εφαρμογής (λογικά στην μορφή ενός .sql αρχείου).

Επιπλέον, η κάθε ομάδα θα πρέπει να ετοιμάσει και να δημοσιεύσει σε κάποια δημόσια προσβάσιμη υπηρεσία (π.χ. Youtube) ένα βίντεο επίδειξης της υλοποιημένης εφαρμογής ιστού. Το βίντεο θα πρέπει να έχει διάρκεια 2 λεπτών και θα πρέπει να καλύψει όσο γίνεται καλύτερα ολόκληρη την λειτουργικότητα της εφαρμογής. Το URL για την θέασή του θα πρέπει να έχει επισημανθεί στην αναφορά του προαναφερόμενου παραδοτέου αρχείου.

Τέλος, οι ομάδες θα εξεταστούν πάνω στην εργασία τους και θα επιδείξουν τον κώδικά τους εξ αποστάσεως με βάση σενάρια χρήσης που θα πρέπει να σχεδιάσουν καταλλήλως και να καλύπτουν επί τω πλείστο όλη την λειτουργικότητα της εφαρμογής. Για τα σενάρια αυτά, θα πρέπει να χρειαστεί να μπει εκ των προτέρων κάποιο περιεχόμενο στη βάση δεδομένων της εφαρμογής. Η ημερομηνία της εξέτασης θα ανακοινωθεί εν ευθέτω χρόνο.

ΤΡΟΠΟΣ ΒΑΘΜΟΛΟΓΗΣΗΣ

Η εν λόγω εργασία είναι απαλλακτική και παίρνει 60% του τελικού βαθμού του μαθήματος. Η βαθμολόγηση της εργασίας θα πραγματοποιηθεί με βάση τα ακόλουθα κριτήρια, κάθε ένα από τα οποία αντιστοιχεί σε ένα ποσοστό του βαθμού της εργασίας:

ΚΡΙΤΗΡΙΟ	ΠΟΣΟΣΤΟ
Ομοιόμορφη, χρηστική και αποκριτική (responsive) διεπαφή χρήσης	30%
Κατάλληλη και πλήρης υλοποίηση λειτουργιών διαχείρισης ασθενών, ιστορικού και ραντεβού	35%
Αποθήκευση και χειρισμός δεδομένων	15%
Υποστήριξη πολλαπλών ρόλων (χρηστών)	10%
Αναφορά	05%
Βίντεο επίδειξης	05%

ΜΠΟΝΟΥΣ

Δίδεται δυνατότητα αύξησης της βαθμολογίας της εργασίας μέσω μπόνους που θα μετράνε μόνο εφόσον έχουν υλοποιηθεί σωστά. Προφανώς, η αύξηση της βαθμολογίας δεν μπορεί να ξεπεράσει το κατώφλι του μέγιστου βαθμού (100%). Τα μπόνους είναι τα ακόλουθα:

- *Παρουσίαση περιεχομένου σε μια επιπλέον γλώσσα (Αγγλικά) και δυνατότητα εναλλαγής μεταξύ γλωσσών: 10%.* Σημειώνεται πως εφόσον επιλεγεί μια γλώσσα, όλο το περιεχόμενο θα πρέπει να εμφανίζεται συνεχώς στην γλώσσα αυτή. Αν αυτό γίνει μόνο για την τρέχουσα ιστοσελίδα και όχι την επόμενη στην οποία θα πλοηγηθεί ο χρήστης, τότε δεν θα λάβετε όλο τον βαθμό του μπόνους.
- *Σελιδοποίηση αποτελεσμάτων αναζήτησης: 10% (3,33% ανά λειτουργία αναζήτησης).* Θα εκτιμηθεί δεόντως η ικανότητα σελιδοποίησης στην εμφάνιση αποτελεσμάτων αναζήτησης (για ασθενείς, εγγραφές ιστορικού και ραντεβού) ώστε να εμφανίζονται μόνο Χ αποτελέσματα (πχ. 10) και ο χρήστης να μπορεί να πλοηγείται σε μια επόμενη ή προηγούμενη σελίδα (αποτελεσμάτων) καθώς και στην αρχική και τελευταία σελίδα (αποτελεσμάτων).
- *Εκφόρτωση ιστορικού ασθενούς: 10%.* Θα πρέπει να μπορεί ο ιατρός να ζητήσει την εκφόρτωση σε πλήρη μορφή είτε μιας εγγραφής ή ολόκληρου του ιστορικού ενός ασθενούς σε μορφοποίηση Excel ή PDF.
- *Εμφάνιση & επιλογή ημερομηνίας & ώρας ραντεβού καθώς και slots διαθεσιμότητας ιατρού με βάση ημερολόγιο: 10%.* Εφόσον θα πρέπει να γίνει δημιουργία ή τροποποίηση ραντεβού, ο χρήστης μπορεί να βλέπει ένα ημερολόγιο με συγκεκριμένα slots ανά ημέρα και να επιλέγει ένα από τα slots που είναι διαθέσιμα. Επιπλέον, ο ιατρός μπορεί να δηλώνει τα slots όπου είναι διαθέσιμος.