



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Σύστημα τηλεγραμματοεπικοινωνίας για την διαδικτυακή
εξυπηρέτηση διάφορων επαγγελματικών ειδικοτήτων»



Φοιτητής

ΣΟΥΛΗΜΙΔΟΥ ΔΕΣΠΟΙΝΑ 516318

Επιβλέπων

Δρ. Κυριάκος Τσιακμάκης

Σεπτέμβριος 2023

Σύστημα τηλεγραμματοεπικοινωνίας για την διαδικτυακή εξυπηρέτηση διάφορων επαγγελματικών
ειδικοτήτων

Κωδικός: 23130

Φοιτήτρια: Σουλημίδου Δέσποινα

Εισηγητής: Δρ Κυριάκος Τσιακμάκης

Ημερομηνία ανάληψης Π.Ε. 08-03-2023

Ημερομηνία περάτωσης Π.Ε. 10-09-2023

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

*Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία της φοιτήτριας **ΣΟΥΛΗΜΙΔΟΥ ΔΕΣΠΟΙΝΑ** που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.*

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Περίληψη

Η εργασία αυτή αφορά τη μελέτη, η σχεδίαση και η υλοποίηση ενός συστήματος ηλεκτρονικής γραμματειακής υποστήριξης για ραντεβού για ειδικότητες όπως ο γιατρός και δικηγόρος. Το σύστημα διαθέτει διαχειριστικό για τον ιδιοκτήτη (γιατρός και δικηγόρος) και για τη γραμματεία του. Διαθέτει ιστοχώρο για να διευθετεί ο πελάτης τα ραντεβού του, να βλέπει τα διαθέσιμα και υπόλοιπα ραντεβού του. Ένα από τα σημαντικότερα χαρακτηριστικά είναι η δυνατότητα σύνδεσης του συστήματος με το website του ιδιοκτήτη-γιατρού. Το σύστημα έχει υλοποιηθεί με το Laravel, PHP και Mysql δίνοντας τη δυνατότητα μέσω αυτής της υλοποίησης την εκμάθηση ειδικών αλγοριθμικών τεχνικών, τεχνολογιών και frameworks και βελτίωση προγραμματιστικών δεξιοτήτων για web εφαρμογές.

« Appointment system for the online service of various professional specialties »

Abstract

This work concerns the study, design and implementation of a system of electronic support for appointments for specialties such as the doctor and the lawyer. The system has an administrative for the owner (doctor and lawyer) and for his secretary. It has a website for the client to arrange his appointments, to see the availability and remaining appointments. One of the most features is the ability to connect the system to the doctor's website. The system is designed with Laravel, PHP and Mysql give the opportunity to learn specific algorithmic techniques, technologies and frameworks and improve programming skills for develop web applications.

Ευχαριστίες

Θέλω να ευχαριστήσω τους γονείς μου για τη συμπαράσταση τους και τον επιβλέπων κ. Τσιακμάκη Κυριάκο για την καθοδήγηση και συμβολή του στον κώδικα της εφαρμογής.

Περιεχόμενα

Περίληψη	iv
Abstract.....	v
Ευχαριστίες.....	vi
Περιεχόμενα.....	vii
Κατάλογος Σχημάτων	viii
Κεφάλαιο 1ο: Εισαγωγή.....	10
1.1 Εισαγωγή.....	10
1.2 Δομή της εργασίας	11
Κεφάλαιο 2ο: Παρόμοια Συστήματα Διαδικτυακών Ραντεβού	12
2.1 Εισαγωγή.....	12
2.2 Συστήματα διαδικτυακών ραντεβού εκτός Ελληνικής αγοράς	13
2.3 Συστήματα διαδικτυακών ραντεβού εντός Ελληνικής αγοράς	17
Κεφάλαιο 3ο: Τεχνολογία και εργαλεία	22
3.1 PHP	22
3.2 Laravel	30
3.3 MySQL	35
3.4 Xampp.....	39
3.4.1 Apache Server	40
3.4.2 phpmyadmin.....	42
Κεφάλαιο 4ο: Το σύστημα Ηλεκτρονικής Γραμματειακής Υποστήριξης.....	43
4.1 Εισαγωγή στο σύστημα	43
4.1.1 Ιδιοκτήτης	45
4.1.2 Χρήστης	49
4.1.3 Γραμματεία.....	53
4.1.4 Διαγράμματα ροής.....	54
4.2 Η βάση δεδομένων	59
Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης	63
ΒΙΒΛΙΟΓΡΑΦΙΑ	64
ΠΑΡΑΡΤΗΜΑ Α.....	65

Κατάλογος Σχημάτων

Εικόνα 2.1: Acuity scheduling.....	13
Εικόνα 2.2: Calendly.....	14
Εικόνα 2.3: Setmore.....	15
Εικόνα 2.4: SimplyBook.....	16
Εικόνα 2.5: medi-booking.com.....	16
Εικόνα 2.5: book-online.gr.....	17
Εικόνα 2.7: easyappointments.org.....	18
Εικόνα 2.8: doctoranytime.gr.....	19
Εικόνα 2.9: https://doctornearyou.gr/	20
Εικόνα 2.10: anydoctor.gr.....	21
Εικόνα 3.1: PHP 8.....	23
Εικόνα 3.2: Σύγκριση χρήσης PHP με διάφορες γλώσσες.....	29
Εικόνα 3.3: Σύγκριση δημοτικότητας PHP με διάφορες γλώσσες.....	29
Εικόνα 3.4: Framework Laravel.....	30
Εικόνα 3.5: XAMPP control panel.....	39
Εικόνα 3.6: PHPMYAdmin web panel.....	42
Εικόνα 4.1: Το σύστημα Ηλεκτρονικής Γραμματειακής Υποστήριξης - Αρχική.....	43
Εικόνα 4.2: Εισαγωγή των στοιχείων πρόσβασης.....	44
Εικόνα 4.3: Εισαγωγή στοιχείων χρήστη για εγγραφή.....	44
Εικόνα 4.4: Κεντρική σελίδα με τα ραντεβού σε συγκεκριμένες ημερομηνίες.....	46
Εικόνα 4.5: Μπλε απεικόνιση – Ελεύθερο ραντεβού.....	46
Εικόνα 4.6: Ελεύθερο ραντεβού για δέσμευση ενός ραντεβού από τον ιδιοκτήτη.....	46
Εικόνα 4.7: Κόκκινη απεικόνιση – Δεσμευμένο ραντεβού.....	46
Εικόνα 4.8: Από Κόκκινη απεικόνιση – ο ιδιοκτήτης μπορεί να επιβεβαιώσει ή να ακυρώσει ένα δεσμευμένο ραντεβού από τον χρήστη.....	47
Εικόνα 4.9: Μαύρη απεικόνιση – Μη ενεργοποιημένο ραντεβού.....	47
Εικόνα 4.10: Πράσινη απεικόνιση – Επιβεβαιωμένο ραντεβού – Ο χρήστης μπορεί να προσέλθει στο ραντεβού του.....	47
Εικόνα 4.11: Από Πράσινο κουμπί – λεπτομέρειες του ραντεβού.....	48
Εικόνα 4.12: Πόσα ραντεβού έχει ένας χρήστης με ιδιοκτήτη.....	48
Εικόνα 4.13: Πόσα ραντεβού έχει ο ιδιοκτήτης.....	49
Εικόνα 4.14: Ιστοσελίδα με τα ραντεβού του 8A2B1F που βλέπει ο χρήστης.....	50
Εικόνα 4.15: Εισαγωγή.....	51
Εικόνα 4.16: Εισαγωγή.....	51
Εικόνα 4.17: Εισαγωγή.....	51
Εικόνα 4.18: Στη μπλε απεικόνιση – Ελεύθερο ραντεβού για τον χρήστη.....	52
Εικόνα 4.19: Ο χρήστης μπορεί να ζητήσει ραντεβού.....	52
Εικόνα 4.20: Μαύρη απεικόνιση – Μη επιβεβαιωμένο αλλά και μη ελεύθερο ραντεβού.....	52

Εικόνα 4.21: Αν ο χρήστης πατήσει στο μαύρο κουμπί θα προβληθεί μήνυμα μη επιβεβαίωσης του ραντεβού	52
Εικόνα 4.22: Κόκκινη απεικόνιση – δεσμευμένο ραντεβού	53
Εικόνα 4.23: Πίνακας αντιστοίχισης του id του ιδιοκτήτη με του γραμματέα	53
Εικόνα 4.24: Ποιες ιδιοκτήτες έχει αναλάβει για εξυπηρέτηση ο γραμματέας με id 4	53
Εικόνα 4.25: Διάγραμμα ροής χρήστη: για την είσοδο, εγγραφή, προβολή ραντεβού και ενέργειες. Χωρίς συγκεκριμένο αναγνωριστικό.....	54
Εικόνα 4.26: Διάγραμμα ροής χρήστη: για την είσοδο και προβολής κεντρικής σελίδας του ιδιοκτήτη με χρήση συγκεκριμένου αναγνωριστικού.....	55
Εικόνα 4.27: Διάγραμμα ροής χρήστη: για την είσοδο και προβολής κεντρικής σελίδας του ιδιοκτήτη με χρήση συγκεκριμένου αναγνωριστικού.....	56
Εικόνα 4.28: Διάγραμμα ροής για προβολή των ραντεβού του συγκεκριμένου ιδιοκτήτη. Ενέργειες για τα ραντεβού.....	57
Εικόνα 4.29: Διάγραμμα ροής για τον ιδιοκτήτη. Προβολή των ραντεβού του συγκεκριμένου ιδιοκτήτη. Ενέργειες για τα ραντεβού.	58
Εικόνα 4.30: Διάγραμμα ροής την/τον γραμματέα. Προβολή των ιδιοκτητών, επιλογή.....	59
Εικόνα 4.31: Η βάση δεδομένων με τους πίνακες που χρησιμοποιήθηκαν	59
Εικόνα 4.32: Η δομή του πίνακα user – οι χρήστες.....	60
Εικόνα 4.33: Το περιεχόμενο των χρηστών.....	60
Εικόνα 4.34: Η δομή για τον πίνακα appointment.....	60
Εικόνα 4.35: Το περιεχόμενο του πίνακα appointment – περιέχει τα ραντεβού.....	61
Εικόνα 4.36: Πίνακας secowner με αντιστοίχιση του ιδιοκτήτη με τον/την γραμματέα	62
Εικόνα 4.37: Περιεχόμενα-δείγμα του πίνακα secowner	62

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Τα συστήματα διαδικτυακών ραντεβού προσφέρουν πολυάριθμα οφέλη σε άτομα, επιχειρήσεις και οργανισμούς. Βελτιώνουν τη διαδικασία προγραμματισμού ραντεβού. Με την αυτοματοποίηση της διαχείρισης ραντεβού, αυτά τα συστήματα παρέχουν μια σειρά από πλεονεκτήματα. Για τις επιχειρήσεις, τα συστήματα διαδικτυακών ραντεβού ενισχύουν τη λειτουργική αποτελεσματικότητα. Εξαλείφουν την ανάγκη για μη αυτόματη κράτηση ραντεβού και μειώνουν την πιθανότητα σφαλμάτων προγραμματισμού ή διπλών κρατήσεων. Οι επιχειρήσεις μπορούν επίσης να αποκτήσουν πληροφορίες σχετικά με τις τάσεις των ραντεβού και τις προτιμήσεις των πελατών μέσω δεδομένων που συλλέγονται από το σύστημα.

Από την πλευρά του πελάτη, τα συστήματα διαδικτυακών ραντεβού προσφέρουν την ευκολία πρόσβασης στον προγραμματισμό 24/7. Οι πελάτες μπορούν να κλείσουν ραντεβού με την άνεσή τους, χωρίς τους περιορισμούς των παραδοσιακών ωρών γραφείου. Επιπλέον, αυτά τα συστήματα συχνά στέλνουν αυτοματοποιημένες υπενθυμίσεις μέσω email ή SMS, μειώνοντας τον κίνδυνο χαμένων ραντεβού. Αυτό βελτιώνει τη συνολική εμπειρία των πελατών.

Τα συστήματα διαδικτυακών ραντεβού μπορούν να οδηγήσουν σε αυξημένη παραγωγικότητα τόσο για τις επιχειρήσεις όσο και για τους πελάτες. Με τον χρόνο που εξοικονομείται από τον μη αυτόματο συντονισμό των ραντεβού, οι επιχειρήσεις μπορούν να κατανέμουν τους πόρους ή χρόνο τους πιο αποτελεσματικά. Οι πελάτες, από την άλλη πλευρά, ξοδεύουν λιγότερο χρόνο περιμένοντας σε αναμονή ή ανταλλάσσοντας email για να εξασφαλίσουν ραντεβού.

Επιπλέον, τα συστήματα διαδικτυακών ραντεβού συμβάλλουν σε μια προσέγγιση χωρίς χαρτί και φιλική προς το περιβάλλον. Οι παραδοσιακές μέθοδοι κράτησης ραντεβού συχνά περιλαμβάνουν γραφειοκρατία και φυσικά αρχεία. Με τη μετάβαση σε διαδικτυακά συστήματα, οι επιχειρήσεις μπορούν να μειώσουν το περιβαλλοντικό τους αποτύπωμα και να λειτουργήσουν με πιο βιώσιμο τρόπο.

Για πελάτες που αναζητούν ιατρικές ή νομικές υπηρεσίες, ένα σύστημα διαδικτυακών ραντεβού εξαλείφει την ανάγκη πραγματοποίησης τηλεφωνικών κλήσεων κατά τις ώρες αιχμής. Μπορούν να κλείσουν ραντεβού ανά πάσα στιγμή που τους βολεύει, μειώνοντας την αναμονή.

Οι γιατροί και οι δικηγόροι μπορούν να βελτιστοποιήσουν τα προγράμματά τους πιο αποτελεσματικά χρησιμοποιώντας συστήματα διαδικτυακών ραντεβού. Μπορούν να ορίσουν διαθέσιμα χρονικά διαστήματα, να καθορίσουν τύπους ραντεβού και να καταναείμουν τις κατάλληλες διάρκειες, διασφαλίζοντας ισορροπημένο φόρτο εργασίας.

Οι πελάτες μπορούν να επιλέξουν τον προτιμώμενο χρόνο με βάση τη διαθεσιμότητα, μειώνοντας τους χρόνους αναμονής και βελτιώνοντας τη συνολική τους εμπειρία. Αυτό συμβάλλει στην υψηλότερη ικανοποίηση και διατήρηση του ασθενούς/πελάτη.

Πολλά συστήματα διαδικτυακών ραντεβού μπορούν να ενσωματωθούν με ηλεκτρονικά αρχεία υγείας (EHR) για γιατρούς ή λογισμικό διαχείρισης υποθέσεων για δικηγόρους. Αυτή η απρόσκοπτη σύνδεση απλοποιεί τη συνολική ροή εργασίας και διασφαλίζει την ακριβή τήρηση αρχείων. [1]

Οι γιατροί και οι δικηγόροι μπορούν να έχουν καλύτερο έλεγχο των προγραμμάτων τους, καθορίζοντας πότε είναι διαθέσιμοι για ραντεβού. Αυτή η ευελιξία μπορεί να οδηγήσει σε βελτιωμένη ισορροπία μεταξύ εργασίας και προσωπικής ζωής και μειωμένο άγχος.

Ειδικά σε περιόδους ανησυχιών για την υγεία, όπως πανδημίες, ένα σύστημα διαδικτυακών ραντεβού επιτρέπει τον προγραμματισμό ανέπαφων, μειώνοντας τον κίνδυνο λοιμώξεων και αυξάνοντας την ασφάλεια τόσο για τους επαγγελματίες όσο και για τους πελάτες.

Ο κύριος στόχος αυτής της εργασίας είναι η μελέτη, η σχεδίαση και η υλοποίηση ενός συστήματος ηλεκτρονικής γραμματειακής υποστήριξης για ραντεβού για ειδικότητες όπως ο γιατρός και δικηγόρος.

Το σύστημα διαθέτει διαχειριστικό για τον ιδιοκτήτη (γιατρός και δικηγόρος) και για τη γραμματεία του. Διαθέτει ιστοχώρο για να διευθετεί ο πελάτης τα ραντεβού του, να βλέπει τα διαθέσιμα και υπόλοιπα ραντεβού του. Ένα από τα σημαντικότερα χαρακτηριστικά είναι η δυνατότητα σύνδεσης του συστήματος με το website του ιδιοκτήτη-γιατρού. Το σύστημα έχει υλοποιηθεί με το Laravel, PHP και Mysql δίνοντας τη δυνατότητα μέσω αυτής της υλοποίησης την εκμάθηση ειδικών αλγοριθμικών τεχνικών, τεχνολογιών και frameworks και βελτίωση προγραμματιστικών δεξιοτήτων για web εφαρμογές.

1.2 Δομή της εργασίας

Στο πρώτο κεφάλαιο παρουσιάζεται μια μικρή εισαγωγή της εργασίας και ο στόχος της.

Στο δεύτερο κεφάλαιο παρουσιάζονται παρόμοια συστήματα διαδικτυακών ραντεβού.

Στο τρίτο κεφάλαιο περιγράφεται η τεχνολογία και τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της ηλεκτρονικής γραμματείας..

Στο τέταρτο κεφάλαιο αναλύεται το σύστημα ηλεκτρονικής γραμματειακής υποστήριξης και η βάση που χρησιμοποιήθηκε.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα συμπεράσματα της εργασίας και θέματα για μελλοντική έρευνα ενώ στο τέλος της εργασίας παρατίθεται το παράρτημα με κάποιους από τους κώδικες που χρησιμοποιήθηκαν στην εργασία.

Κεφάλαιο 2ο: Παρόμοια Συστήματα Διαδικτυακών Ραντεβού

2.1 Εισαγωγή

Τα συστήματα διαδικτυακών ραντεβού προσφέρουν μια βολική και αποτελεσματική λύση για τη διαχείριση ραντεβού και κρατήσεων. Παρέχουν πολλά πλεονεκτήματα που τα καθιστούν πολύτιμο εργαλείο τόσο για επιχειρήσεις όσο και για ιδιώτες.

Για τις επιχειρήσεις, τα συστήματα διαδικτυακών ραντεβού απλοποιούν τη διαδικασία κράτησης, μειώνοντας την ανάγκη για μη αυτόματο προγραμματισμό και επικοινωνία. Αυτό εξοικονομεί χρόνο και ελαχιστοποιεί τις πιθανότητες σφαλμάτων που μπορεί να προκύψουν κατά την κράτηση ραντεβού μέσω τηλεφώνου ή μέσω email. Το σύστημα μπορεί να χειριστεί πολλές κρατήσεις ταυτόχρονα, διασφαλίζοντας τη βέλτιστη χρήση των πόρων και τη διαθεσιμότητα του προσωπικού.

Επιπλέον, τα συστήματα διαδικτυακών ραντεβού ενισχύουν την ικανοποίηση των πελατών προσφέροντας μια φιλική προς το χρήστη διεπαφή όπου οι πελάτες μπορούν να δουν τις διαθέσιμες χρονοθυρίδες και να επιλέξουν συναντήσεις που ταιριάζουν στο πρόγραμμά τους. Οι αυτοματοποιημένες υπενθυμίσεις μπορούν να σταλούν μέσω email ή SMS, μειώνοντας την πιθανότητα μη εμφάνισης και ακυρώσεων της τελευταίας στιγμής.

Από διοικητική άποψη, αυτά τα συστήματα παρέχουν μια κεντρική πλατφόρμα για τη διαχείριση των ραντεβού, την προβολή των επερχόμενων προγραμμάτων και την ανάλυση των τάσεων κρατήσεων. Αυτή η προσέγγιση βάσει δεδομένων δίνει τη δυνατότητα στις επιχειρήσεις να λαμβάνουν τεκμηριωμένες αποφάσεις σχετικά με την κατανομή πόρων και τη στελέχωση.

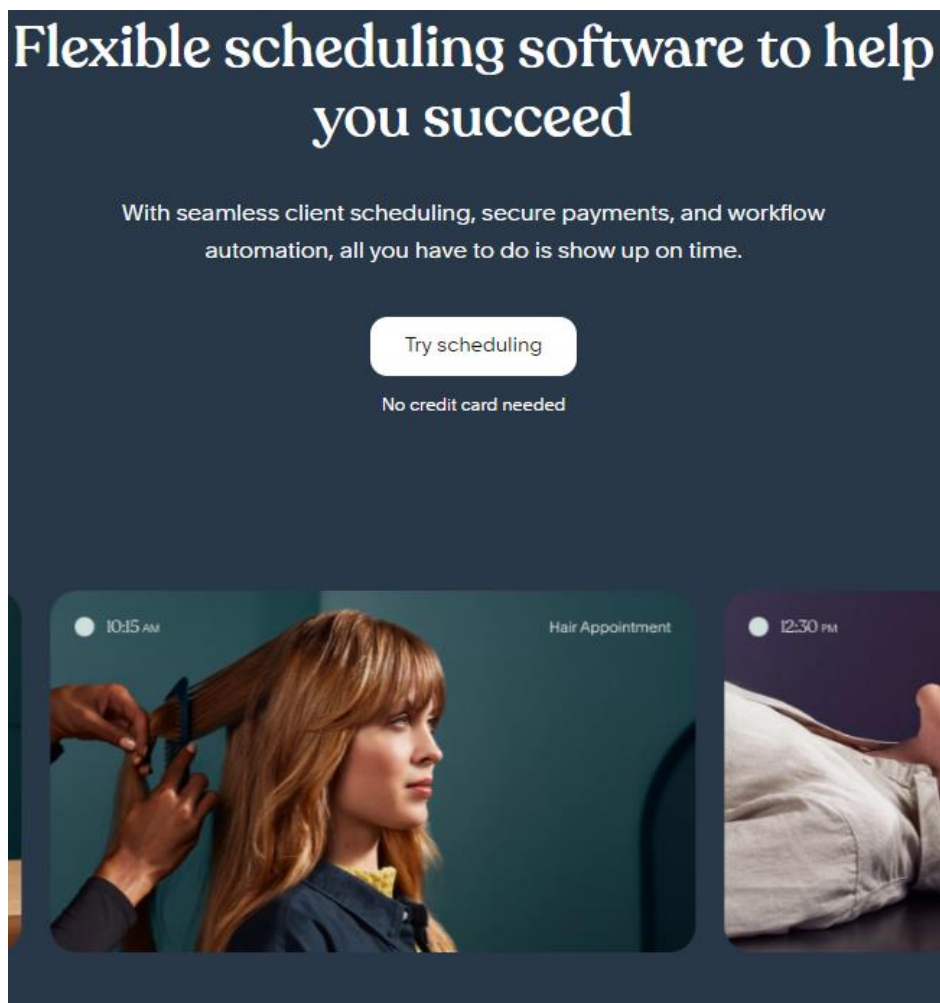
Τα άτομα επωφελούνται επίσης από τα συστήματα διαδικτυακών ραντεβού, καθώς παρέχουν την ευελιξία να κλείσουν ραντεβού ανά πάσα στιγμή, χωρίς να χρειάζεται να τηλεφωνούν κατά τις εργάσιμες ώρες. Αυτή η ευκολία είναι ιδιαίτερα πολύτιμη για όσους έχουν πολυάσχολα προγράμματα ή διαφορετικές ζώνες ώρας. Επιπλέον, η δυνατότητα λήψης αυτοματοποιημένων υπενθυμίσεων βοηθά τους πελάτες να παραμείνουν οργανωμένοι και διασφαλίζει ότι δεν χάνουν σημαντικά ραντεβού.

Τα συστήματα διαδικτυακών ραντεβού προσφέρουν αποτελεσματικότητα, ευκολία και βελτιωμένη ικανοποίηση πελατών. Απλοποιούν τη διαδικασία κράτησης, μειώνουν τον διοικητικό φόρτο και παρέχουν μια απρόσκοπτη εμπειρία τόσο για επιχειρήσεις όσο και για ιδιώτες.

2.2 Συστήματα διαδικτυακών ραντεβού εκτός Ελληνικής αγοράς

Acuity Scheduling:

Το Acuity Scheduling είναι ένα φιλικό προς τον χρήστη σύστημα διαδικτυακών ραντεβού που έχει σχεδιαστεί για να απλοποιεί τις διαδικασίες κράτησης για επιχειρήσεις όλων των μεγεθών. Προσφέρει προσαρμόσιμες επιλογές προγραμματισμού, επιτρέποντας στους πελάτες να βλέπουν τη διαθεσιμότητα σε πραγματικό χρόνο και να επιλέγουν θέσεις για ραντεβού που ταιριάζουν στις προτιμήσεις τους.



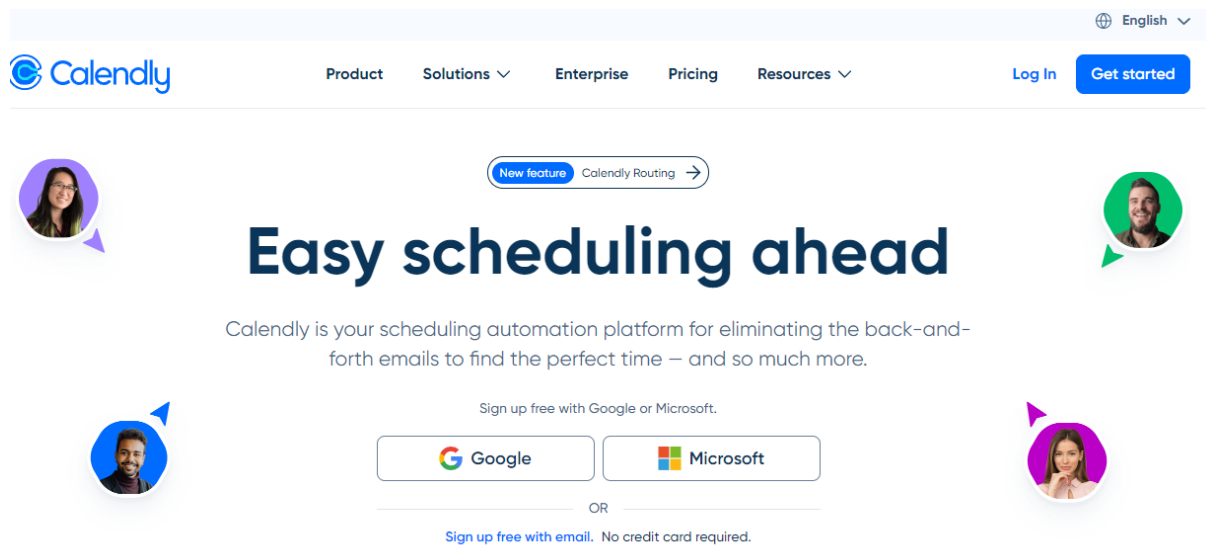
Εικόνα 2.1: Acuity scheduling

[<https://acuityscheduling.com/>]

Η πλατφόρμα προσφέρει αυτοματοποιημένες υπενθυμίσεις και ειδοποιήσεις, μειώνοντας τις μη εμφανίσεις. Επιπλέον, το Acuity Scheduling ενσωματώνεται με δημοφιλή ημερολόγια και πύλες πληρωμών, καθιστώντας το ένα ολοκληρωμένο εργαλείο για τη διαχείριση συναντήσεων, μαθημάτων και εργαστηρίων.

Calendly:

Το Calendly είναι ένα ευρέως χρησιμοποιούμενο σύστημα προγραμματισμού ραντεβού που βασίζεται στο διαδίκτυο, γνωστό για την απλότητα και την αποτελεσματικότητά του. Επιτρέπει στους χρήστες να ορίζουν τη διαθεσιμότητά τους και να δημιουργούν εξατομικευμένους συνδέσμους κράτησης που μπορούν να κοινοποιηθούν με πελάτες ή συναδέλφους.



Εικόνα 2.2: Calendly

[<https://calendly.com/>]

Οι παραλήπτες μπορούν στη συνέχεια να επιλέξουν μια βολική ώρα από τις διαθέσιμες επιλογές, συγχρονίζοντας αυτόματα το ραντεβού με τα ημερολόγια και των δύο μερών. Το Calendly υποστηρίζει την ενοποίηση με διάφορα ημερολόγια και προσφέρει λειτουργίες όπως αυτοματοποιημένες υπενθυμίσεις και ανίχνευση ζώνης ώρας.

Setmore:

Το Setmore είναι ένα διαισθητικό σύστημα διαδικτυακών ραντεβού σχεδιασμένο για επιχειρήσεις που αναζητούν μια ευέλικτη λύση προγραμματισμού. Προσφέρει δυνατότητες όπως προσαρμόσιμες σελίδες κράτησης, διαχείριση προσωπικού και συγχρονισμό σε πραγματικό χρόνο με δημοφιλή ημερολόγια.

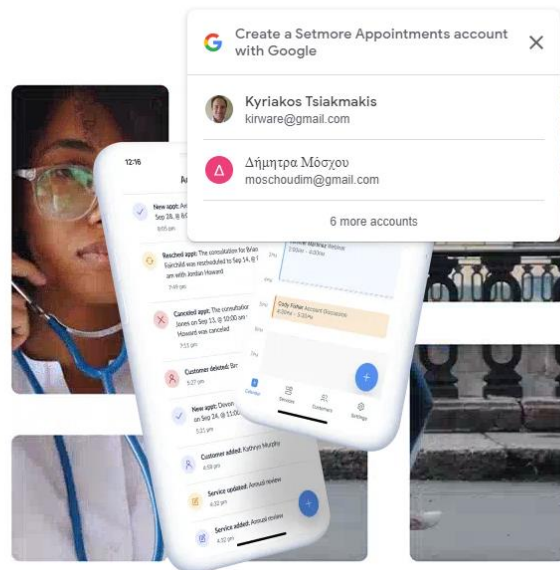
Free scheduling software.

Organize your business with 24/7 automated online booking, reminders, payments, and more.

Your email [Start FREE](#)

Or sign up with Google or Facebook

Excellent Trustpilot



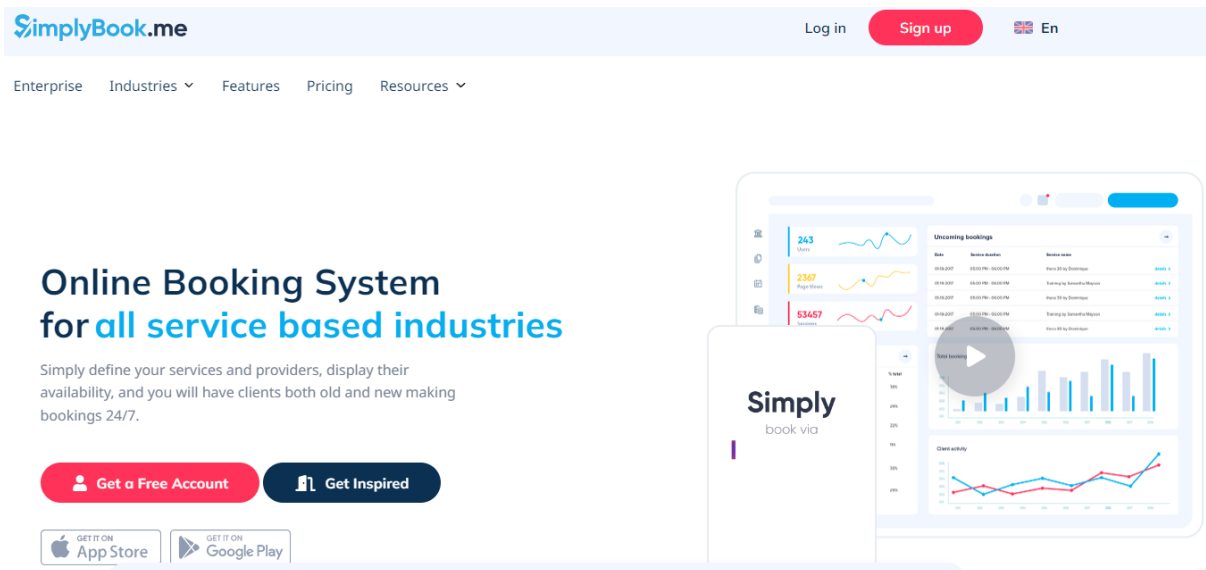
Εικόνα 2.3: Setmore

[<https://www.setmore.com/>]

Το Setmore υποστηρίζει επίσης την ηλεκτρονική επεξεργασία πληρωμών και παρέχει εργαλεία για τη διαχείριση των προφίλ πελατών και του ιστορικού συναντήσεων. Με τη φιλική προς τον χρήστη διεπαφή και την εφαρμογή για κινητά, οι επιχειρήσεις μπορούν εύκολα να διαχειρίζονται συναντήσεις εν κινήσει.

SimplyBook.me:

Το SimplyBook.me είναι ένα ευέλικτο σύστημα διαδικτυακών ραντεβού κατάλληλο για διάφορους κλάδους, όπως η υγειονομική περίθαλψη, η ευεξία, η εκπαίδευση και άλλα. Επιτρέπει στις επιχειρήσεις να δημιουργούν προσαρμοσμένους ιστότοπους κρατήσεων με πολλαπλές προσφορές υπηρεσιών και μέλη του προσωπικού.



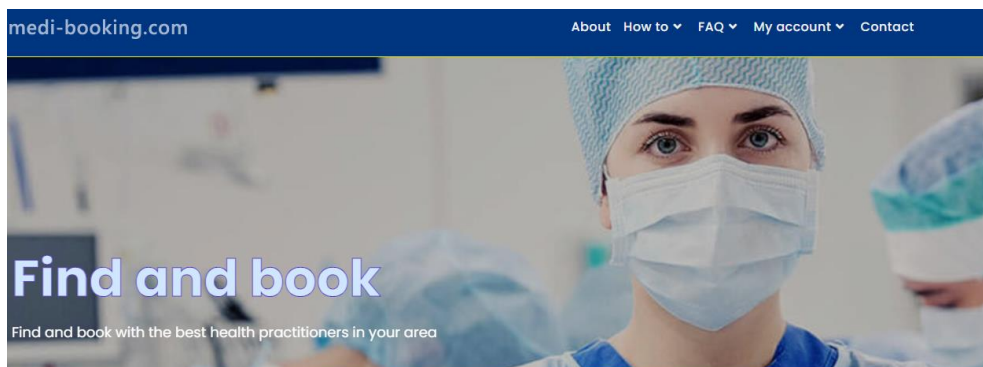
Εικόνα 2.4: SimplyBook

[<https://simplybook.me/en/>]

Οι πελάτες μπορούν να κλείσουν ραντεβού με βάση τη διαθεσιμότητα σε πραγματικό χρόνο και να λαμβάνουν αυτοματοποιημένες επιβεβαιώσεις και υπενθυμίσεις. Το σύστημα προσφέρει επίσης ενοποίηση με διαφορετικές πύλες πληρωμής, επιτρέποντας ασφαλείς διαδικτυακές συναλλαγές. Με τις ευέλικτες δυνατότητες του, το SimplyBook.me καλύπτει τις μοναδικές ανάγκες προγραμματισμού διαφορετικών επιχειρήσεων.

MediBooking:

Το MediBooking είναι μια διαδικτυακή πλατφόρμα που διευκολύνει τα ραντεβού γιατρού και τις ιατρικές υπηρεσίες. Οι χρήστες μπορούν να αναζητήσουν γιατρούς με βάση την ειδικότητα, την τοποθεσία και τη διαθεσιμότητα, γεγονός που διευκολύνει την εύρεση του κατάλληλου παρόχου υγειονομικής περίθαλψης.



Εικόνα 2.5: medi-booking.com

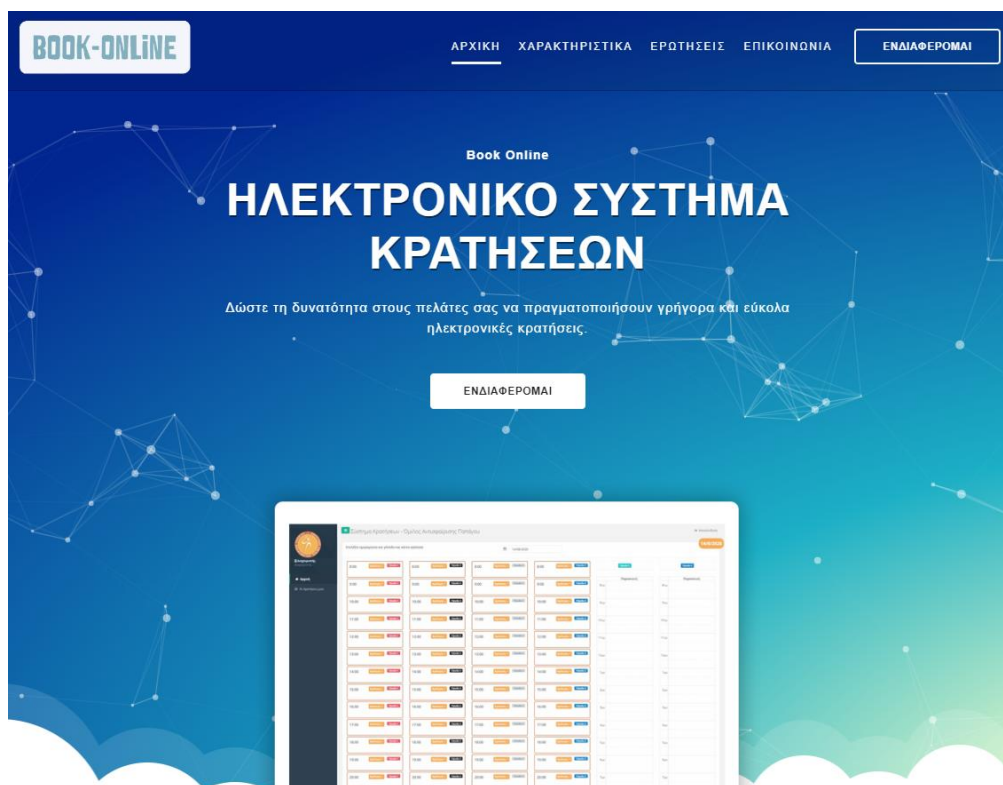
[<https://medi-booking.com/>]

Η πλατφόρμα προσφέρει προγραμματισμό σε πραγματικό χρόνο, δίνοντας τη δυνατότητα στους ασθενείς να επιλέξουν ένα ραντεβού που ταιριάζει στο πρόγραμμά τους. Επιπλέον, το MediBooking παρέχει λειτουργίες όπως υπενθυμίσεις ραντεβού και αποθήκευση ιατρικού ιστορικού, βελτιστοποιώντας τη συνολική εμπειρία υγειονομικής περίθαλψης.

2.3 Συστήματα διαδικτυακών ραντεβού εντός Ελληνικής αγοράς

book-online.gr:

Το BookOnline.gr είναι ένα δημοφιλές σύστημα διαδικτυακών ραντεβού στην Ελλάδα που προσφέρει μια φιλική προς τον χρήστη πλατφόρμα προγραμματισμού ραντεβού σε διάφορους κλάδους. Απευθύνεται σε επιχειρήσεις που κυμαίνονται από ιατρεία και ινστιτούτα αισθητικής έως γυμναστήρια. Το σύστημα επιτρέπει στους πελάτες να περιηγηθούν στις διαθέσιμες υπηρεσίες, να επιλέξουν τον πάροχο υπηρεσιών που προτιμούν και να επιλέξουν μια κατάλληλη χρονική περίοδο από το εμφανιζόμενο ημερολόγιο.



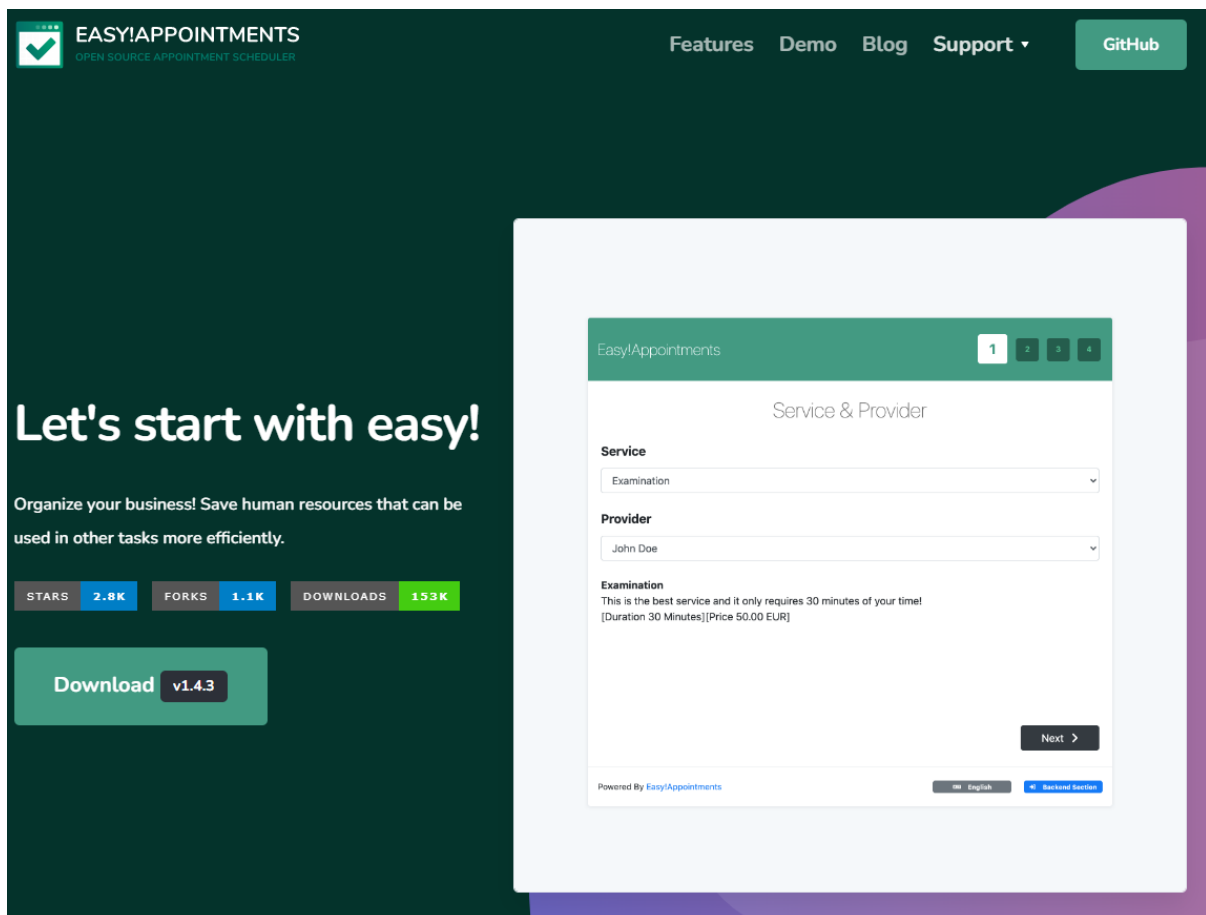
Εικόνα 2.6: book-online.gr

[<https://www.book-online.gr/>]

Επιπλέον, το BookOnline.gr παρέχει αυτοματοποιημένες υπενθυμίσεις και ειδοποιήσεις τόσο στους πελάτες όσο και στους παρόχους υπηρεσιών, μειώνοντας την πιθανότητα χαμένων ραντεβού. Η διαισθητική διεπαφή και η συμβατότητα με φορητές συσκευές της πλατφόρμας συμβάλλουν στη δημοτικότητά της μεταξύ των χρηστών που αναζητούν χωρίς προβλήματα προγραμματισμό ραντεβού.

EasyAppointments

Το EasyAppointments είναι ένα ευρέως διαδεδομένο διαδικτυακό σύστημα προγραμματισμού ραντεβού στην Ελλάδα, σχεδιασμένο να απλοποιεί τις διαδικασίες κράτησης για επιχειρήσεις όλων των μεγεθών. Αυτό το σύστημα ξεχωρίζει για την ευελιξία του, δίνοντας τη δυνατότητα σε παρόχους υπηρεσιών, όπως γιατρούς, συμβούλους και καθηγητές, να διαχειρίζονται αποτελεσματικά τα προγράμματά τους.



The image shows a screenshot of the Easy!Appointments website. The top navigation bar includes links for 'Features', 'Demo', 'Blog', 'Support', and a 'GitHub' button. The main content area features a large heading 'Let's start with easy!' and a sub-heading 'Organize your business! Save human resources that can be used in other tasks more efficiently.' Below this, there are statistics for 'STARS 2.8K', 'FORKS 1.1K', and 'DOWNLOADS 153K'. A prominent 'Download v1.4.3' button is visible. The central part of the screenshot displays a user interface for selecting a service and provider. The 'Service' dropdown is set to 'Examination', and the 'Provider' dropdown is set to 'John Doe'. Below these, there is a description for the 'Examination' service: 'This is the best service and it only requires 30 minutes of your time! [Duration 30 Minutes][Price 50.00 EUR]'. A 'Next >' button is located at the bottom right of the form. The footer of the interface includes 'Powered By Easy!Appointments', a language selector set to 'English', and a 'Backend Section' link.

Εικόνα 2.7: easyappointments.org

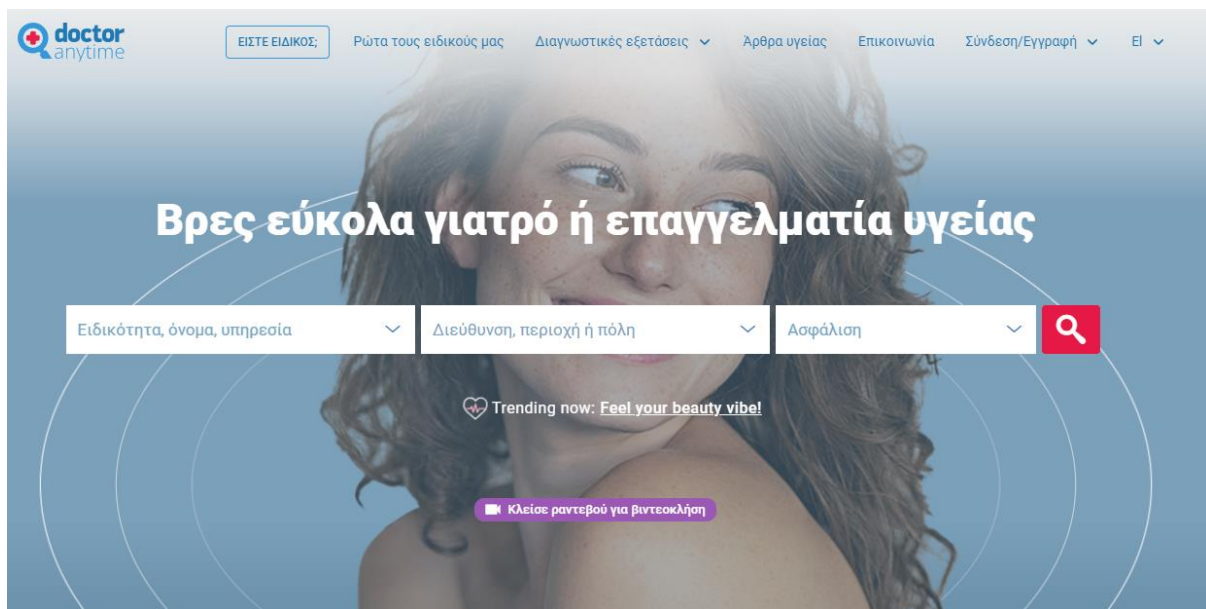
[<https://easyappointments.org/>]

Οι πελάτες μπορούν να έχουν πρόσβαση στο σύστημα για να δουν τη διαθεσιμότητα σε πραγματικό χρόνο, να επιλέξουν την υπηρεσία που επιθυμούν και να εξασφαλίσουν μια υποδοχή ραντεβού που

τους ταιριάζει. Το EasyAppointments ενσωματώνει μια προσαρμόσιμη φόρμα κράτησης που επιτρέπει στους παρόχους υπηρεσιών να συλλέγουν συγκεκριμένες πληροφορίες από τους πελάτες κατά τη διαδικασία κράτησης. Με έμφαση στην προσαρμογή και τη φιλική προς τον χρήστη πλοήγηση, η πλατφόρμα καλύπτει τις μοναδικές ανάγκες προγραμματισμού διαφορετικών επιχειρήσεων.

DoctorAnytime.gr:

Το DoctorAnytime.gr είναι μια ευρέως διαδεδομένη διαδικτυακή πλατφόρμα ραντεβού γιατρού στην Ελλάδα. Παρέχει στους χρήστες τη δυνατότητα αναζήτησης γιατρών, οδοντιάτρων και άλλων επαγγελματιών υγείας με βάση την ειδικότητά τους, την τοποθεσία και τη διαθεσιμότητά τους. Οι ασθενείς μπορούν να δουν τα προφίλ των παρόχων υγειονομικής περίθαλψης, να διαβάσουν κριτικές ασθενών και να κλείσουν ραντεβού απευθείας μέσω της πλατφόρμας.



Εικόνα 2.8: doctoranytime.gr

[<https://www.doctoranytime.gr/>]

Το σύστημα προσφέρει ευκολία και προσβασιμότητα, επιτρέποντας στους ασθενείς να διαχειρίζονται διαδικτυακά τα ραντεβού τους για την υγειονομική περίθαλψη.

doctornearyou.gr:

Το doctornearyou.gr είναι ένα διαδικτυακό σύστημα ραντεβού γιατρού που στοχεύει στη σύνδεση ασθενών με κατάλληλους επαγγελματίες υγείας στην Ελλάδα. Η πλατφόρμα προσφέρει μια φιλική προς

το χρήστη διεπαφή όπου οι ασθενείς μπορούν να αναζητήσουν γιατρούς, να διαβάσουν για την πείρα τους και να ελέγξουν τη διαθεσιμότητά τους.

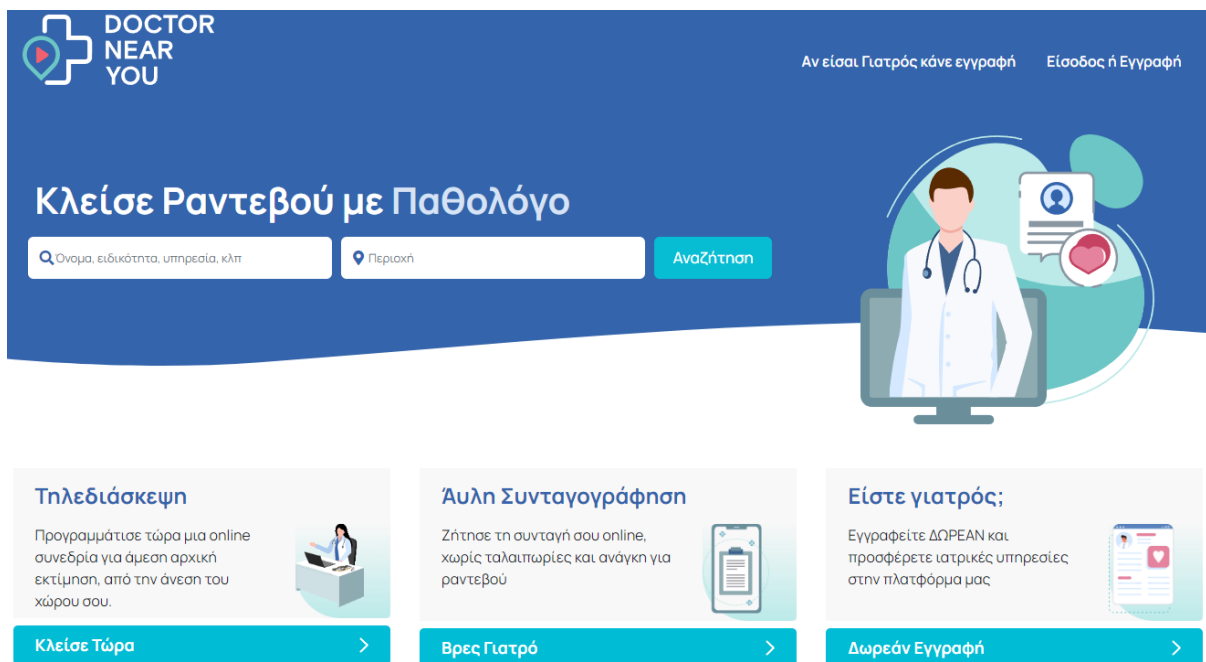
Παρέχει

εύρεση κατάλληλου επαγγελματία υγείας

εξερεύνηση Προφίλ

κλείσιμο Ραντεβού

επιβεβαίωση Ραντεβού



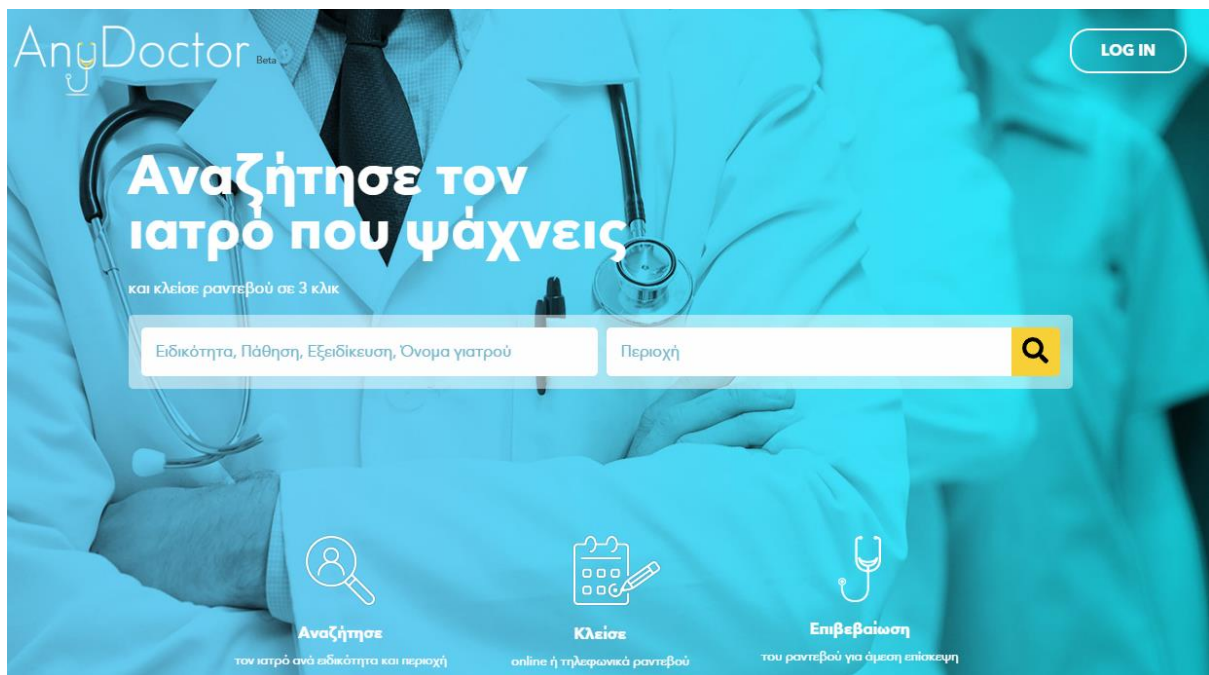
Εικόνα 2.9: <https://doctornearyou.gr/>

[<https://doctornearyou.gr/>]

Επιπλέον, οι ασθενείς μπορούν να δουν το κόστος των υπηρεσιών και να κλείσουν ραντεβού ηλεκτρονικά. Το doctornearyou.gr δίνει έμφαση στην ευκολία χρήσης και την αποτελεσματική επικοινωνία μεταξύ ασθενών και γιατρών.

anydoctor.gr:

Το anydoctor.gr παρέχει μια διαδικτυακή πλατφόρμα για τους ασθενείς να βρίσκουν και να προγραμματίζουν ραντεβού με γιατρούς στην Ελλάδα. Ο ιστότοπος διαθέτει μια εκτενή λίστα ιατρικών ειδικοτήτων και επιτρέπει στους χρήστες να φιλτράρουν τα αποτελέσματα με βάση την τοποθεσία, την ασφάλιση και τις γλωσσικές προτιμήσεις.



Εικόνα 2.10: anydoctor.gr

[<https://anydoctor.gr/>]

Η AnyDoctor είναι μια πλατφόρμα αναζήτησης ιατρών για όλη την Ελλάδα. Ο κάθε ενδιαφερόμενος μπορεί εύκολα και γρήγορα να αναζητήσει και να επιλέξει τον ιατρό που επιθυμεί, να διαβάσει πληροφορίες για το προφίλ του ιατρού και να δει σχετικές κριτικές καθώς και να κλείσει ραντεβού μόνο με 3 κλικ ή μέσω τηλεφώνου.

Η AnyDoctor είναι μια δυναμική πλατφόρμα που έρχεται να αλλάξει τα δεδομένα στον ιατρικό χώρο. Με στρατηγικό συνεργάτη την εταιρεία Forthright, που αποτελεί το μεγαλύτερο και πλέον εξειδικευμένο Digital Agency στο χώρο του ιατρικού marketing και έχει περισσότερες από 300 συνεργασίες με τα μεγαλύτερα ονόματα του ιατρικού χώρου, η AnyDoctor δημιουργεί ένα πλαίσιο που έχει σαν στόχο να καλύψει όλες τις ανάγκες τόσο του επισκέπτη-ασθενή όσο και του ιατρού!

Στην Anydoctor θα μπορεί ο κάθε ενδιαφερόμενος να βρει τους καλύτερους ιατρούς για την κάθε ειδικότητα, να ενημερωθεί για το προφίλ και τις πληροφορίες που σχετίζονται με την δραστηριότητά τους αλλά και να διαβάσει αξιολογήσεις ασθενών. Επιπλέον, σταδιακά θα δημοσιεύονται συνεντεύξεις και άρθρα των ιατρών μαζί με πλούσιο φωτογραφικό υλικό. [από το website]

Προσφέροντας μια σειρά από επιλογές και σχετικές πληροφορίες σχετικά με τους παρόχους υγειονομικής περίθαλψης, το anydoctor.gr απλοποιεί τη διαδικασία εύρεσης και κράτησης ιατρικών ραντεβού online.

Κεφάλαιο 3ο: Τεχνολογία και εργαλεία

Στο κεφάλαιο αυτό θα περιγράψει η τεχνολογία και τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της ηλεκτρονικής γραμματείας.

3.1 PHP

Η PHP (Hypertext Preprocessor) είναι μια ευέλικτη και ευρέως χρησιμοποιούμενη γλώσσα δέσμης ενεργειών από την πλευρά του διακομιστή που έχει σχεδιαστεί κυρίως για ανάπτυξη Ιστού. Δημιουργήθηκε από τον Rasmus Lerdorf στα μέσα της δεκαετίας του 1990, η PHP έχει εξελιχθεί σε ένα θεμελιώδες εργαλείο στη σφαίρα των δυναμικών διαδικτυακών εφαρμογών. Γνωστή για την απλότητα, την ευελιξία και την ευρεία υποστήριξη της κοινότητας, η PHP επιτρέπει στους προγραμματιστές να δημιουργούν διαδραστικές και πλούσιες σε δυνατότητες εμπειρίες ιστού.[2]

Στον πυρήνα της, η PHP λειτουργεί ως γλώσσα δέσμης ενεργειών ενσωματωμένη στην HTML, ενσωματώνοντας απρόσκοπτα τη δημιουργία δυναμικού περιεχομένου με στατικές ιστοσελίδες. Αυτή η δυνατότητα ήταν καθοριστική για την τροφοδοσία πολλών ιστότοπων, που κυμαίνονται από απλά προσωπικά ιστολόγια έως πολύπλοκες πλατφόρμες ηλεκτρονικού εμπορίου και δίκτυα μέσω κοινωνικής δικτύωσης.

Η δύναμη της PHP έγκειται στην ικανότητά της να αλληλεπιδρά με βάσεις δεδομένων, να χειρίζεται τις εισροές των χρηστών, να δημιουργεί δυναμικό περιεχόμενο, να διαχειρίζεται συνεδρίες και πολλά άλλα. Η φύση του ανοιχτού κώδικα έχει οδηγήσει σε πληθώρα βιβλιοθηκών, πλαισίων και εργαλείων που ενισχύουν την αποδοτικότητα και τη συντηρησιμότητα της ανάπτυξης. Η εισαγωγή πλαισίων PHP όπως τα Laravel, Symfony και Yii έχει εξορθολογίσει τη διαδικασία δημιουργίας ισχυρών, επεκτάσιμων και διατηρήσιμων εφαρμογών ιστού.

Με τα χρόνια, η PHP έχει υποστεί σημαντικές βελτιώσεις, τόσο ως προς την απόδοση όσο και ως προς τις δυνατότητες. Η εισαγωγή της PHP 7 έφερε σημαντικές βελτιώσεις στην απόδοση, κάνοντας τις εφαρμογές πιο γρήγορες και πιο αποδοτικές από πλευράς πόρων. Η γλώσσα συνεχίζει να εξελίσσεται, με περιοδικές ενημερώσεις που εισάγουν νέες λειτουργίες και βελτιώσεις ασφάλειας.

Σε αυτήν την εποχή διασύνδεσης, όπου η διαδικτυακή παρουσία είναι ζωτικής σημασίας τόσο για τις επιχειρήσεις όσο και για τα άτομα, η PHP παραμένει θεμελιώδης παράγοντας στο τοπίο ανάπτυξης ιστού. Η ευρεία χρήση, η εκτεταμένη τεκμηρίωση και η ζωντανή κοινότητα διασφαλίζουν ότι η PHP θα συνεχίσει να διαμορφώνει τον ψηφιακό κόσμο για τα επόμενα χρόνια. Είτε είστε έμπειρος προγραμματιστής είτε μόλις ξεκινάτε το ταξίδι κωδικοποίησης, η PHP προσφέρει μια ευέλικτη και προσβάσιμη πλατφόρμα για τη δημιουργία δυναμικών και διαδραστικών εφαρμογών Ιστού.



Εικόνα 3.1: PHP 8

[<https://www.capconnect.com/blog/en/php-8/>]

Μερικές από τις εκδόσεις της PHP είναι:

PHP 5.6 (Αύγουστος 2014):

Βελτιωμένη απόδοση και μειωμένη χρήση μνήμης.

Εισήγαγε σταθερές βαθμωτές εκφράσεις και τελεστή εκθέσεως.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 5.6. [2]

PHP 7.0 (Δεκέμβριος 2015):

Σημαντικές βελτιώσεις απόδοσης μέσω του Zend Engine 3.0.

Εισαγωγή βαθμωτών δηλώσεων τύπου και δηλώσεων τύπου επιστροφής.

Κατάργηση καταργημένων λειτουργιών και παλαιών και μη υποστηριζόμενων SAPI.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 7.0. [2]

PHP 7.1 (Δεκέμβριος 2016):

Μηδενικοί τύποι, επαναλαμβανόμενος ψευδοτύπος και τύπος άκυρης επιστροφής.

Λήψη πολλαπλών εξαιρέσεων σε ένα μόνο μπλοκ αλιευμάτων.

Βελτιώσεις στην κατηγορία σταθερής ορατότητας και δέσμευσης κλεισίματος.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 7.1. [2]

PHP 7.2 (Νοέμβριος 2017):

Εισαγωγή της υπόδειξης τύπου αντικειμένου.

Τύπος υπαινιγμού για παραμέτρους διεπαφών.

Νέες δυνατότητες όπως `random_bytes()` και `random_int()` για ασφαλή δημιουργία τυχαίων αριθμών.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 7.2. [2]

PHP 7.3 (Δεκέμβριος 2018):

Υποστήριξη για ακραία κόμματα στις κλήσεις συναρτήσεων και μεθόδων.

Αρκετές βελτιώσεις για τη διαχείριση σφαλμάτων, όπως η ευέλικτη σύνταξη `heredoc` και `nowdoc`.

Εισαγωγή της συνάρτησης `is_countable()`.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 7.3. [2]

PHP 7.4 (Νοέμβριος 2019):

Εισαγωγή συναρτήσεων βέλους και τελεστής εκχώρησης συγχώνευσης.

Προφόρτωση για καλύτερη απόδοση.

Υποστήριξη για δακτυλογραφημένες ιδιότητες σε κλάσεις.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 7.4. [2]

PHP 8.0 (Νοέμβριος 2020):

Σημαντικές βελτιώσεις συμπεριλαμβανομένου του μεταγλωττιστή JIT για βελτιστοποίηση απόδοσης.

Ονομασμένα ορίσματα, χαρακτηριστικά για μεταδεδομένα, τύπους ένωσης και έκφραση αντιστοίχισης.

Κατάργηση πολλών παλαιών και καταργημένων λειτουργιών.

Πηγή: Επίσημος ιστότοπος PHP, σημειώσεις έκδοσης PHP 8.0. [2]

Μέχρι σήμερα έχουν ανακοινωθεί οι εκδόσεις 8.2.9 και η 8.3.0 είναι σε beta έκδοση.

Η PHP, που σημαίνει "Προεπεξεργαστής Υπερκειμένου", είναι μια ευρέως χρησιμοποιούμενη γλώσσα δέσμης ενεργειών ανοιχτού κώδικα σχεδιασμένη για ανάπτυξη ιστού. Είναι ενσωματωμένο σε κώδικα HTML και εκτελείται από την πλευρά του διακομιστή. Ένα από τα κύρια πλεονεκτήματα της PHP είναι η ικανότητά της να δημιουργεί δυναμικό περιεχόμενο Ιστού επεξεργάζοντας αιτήματα χρηστών και αλληλεπιδρώντας με βάσεις δεδομένων.

Η κύρια περίπτωση χρήσης της PHP είναι η δέσμη ενεργειών από την πλευρά του διακομιστή, όπου επεξεργάζεται δεδομένα στον διακομιστή πριν στείλει το HTML που προκύπτει στο πρόγραμμα περιήγησης του πελάτη. Αυτή η προσέγγιση επιτρέπει στους προγραμματιστές να δημιουργούν διαδραστικούς και δυναμικούς ιστότοπους, όπως πλατφόρμες μέσω κοινωνικής δικτύωσης, ιστότοπους ηλεκτρονικού εμπορίου, συστήματα διαχείρισης περιεχομένου και άλλα. Ο κώδικας PHP ενσωματώνεται απευθείας σε HTML, χρησιμοποιώντας ετικέτες όπως `<?php ... ?>` για να περικλείουν τον κώδικα PHP. Αυτή η απρόσκοπτη ενοποίηση καθιστά εύκολη τη μίξη στατικού και δυναμικού περιεχομένου σε ένα μόνο έγγραφο. Η PHP έχει απλή σύνταξη που είναι παρόμοια με τη C και άλλες γλώσσες προγραμματισμού. Υποστηρίζει ένα ευρύ φάσμα λειτουργιών, συμπεριλαμβανομένου του χειρισμού αρχείων, της συνδεσιμότητας της βάσης δεδομένων, του χειρισμού φόρμας, της διαχείρισης περιόδων σύνδεσης και πολλά άλλα. Οι μεταβλητές στην PHP ξεκινούν με το σύμβολο \$ και η εκτεταμένη βιβλιοθήκη ενσωματωμένων συναρτήσεων απλοποιεί εργασίες όπως ο χειρισμός συμβολοσειρών, η μορφοποίηση ημερομηνίας και η επικύρωση δεδομένων. Επιπλέον, η PHP υποστηρίζει αντικειμενοστραφή προγραμματισμό (OOP), επιτρέποντας στους προγραμματιστές να δημιουργήσουν επαναχρησιμοποιήσιμες και οργανωμένες δομές κώδικα.

Η PHP παίζει καθοριστικό ρόλο στη σύνδεση εφαρμογών Ιστού με βάσεις δεδομένων, χρησιμοποιώντας συχνά επεκτάσεις όπως MySQL, PostgreSQL ή MongoDB. Αυτό επιτρέπει την αποθήκευση και την ανάκτηση δεδομένων απαραίτητων για τον έλεγχο ταυτότητας χρήστη, τη διαχείριση περιεχομένου και άλλες δυναμικές λειτουργίες. Ωστόσο, οι προγραμματιστές πρέπει να δώσουν προτεραιότητα στην ασφάλεια, καθώς οι ανεπαρκώς απολυμανμένες εισόδους ή ο ευάλωτος κώδικας μπορεί να οδηγήσουν σε παραβιάσεις της ασφάλειας. Οι προετοιμασμένες δηλώσεις, η επικύρωση εισόδου και ο σωστός έλεγχος ταυτότητας χρήστη είναι μεταξύ των πρακτικών που χρησιμοποιούνται για την ενίσχυση της ασφάλειας της εφαρμογής PHP.

Η PHP διαθέτει μια τεράστια και ενεργή κοινότητα προγραμματιστών που συμβάλλει στη συνεχή βελτίωσή της. Η κοινότητα της PHP έχει αναπτύξει πλαίσια όπως τα Laravel, Symfony και Yii, τα οποία προσφέρουν δομημένες μεθοδολογίες και εργαλεία για τον εξορθολογισμό της ανάπτυξης. Τα συστήματα διαχείρισης περιεχομένου όπως το WordPress και οι πλατφόρμες ηλεκτρονικού εμπορίου όπως το Magento είναι επίσης χτισμένα σε PHP. Επιπλέον, η συμβατότητα της PHP με διάφορα λειτουργικά συστήματα και διακομιστές ιστού, όπως ο Apache και ο Nginx, την καθιστά μια ευέλικτη επιλογή για φιλοξενία ιστοσελίδων. Με το πλούσιο οικοσύστημά της, η PHP συνεχίζει να είναι μια θεμελιώδης τεχνολογία στο τοπίο ανάπτυξης Ιστού.

Η PHP είναι μια ευέλικτη γλώσσα δέσμης ενεργειών από την πλευρά του διακομιστή που αποτελεί ακρογωνιαίο λίθο της ανάπτυξης ιστού για δεκαετίες. Η ικανότητά του να δημιουργεί δυναμικό περιεχόμενο, να αλληλεπιδρά με βάσεις δεδομένων και να ενσωματώνεται απρόσκοπτα με την HTML το έχει καταστήσει ιδανική επιλογή για τη δημιουργία διαδραστικών και πλούσιων σε δυνατότητες ιστοσελίδων και εφαρμογών. Με φιλική προς τον χρήστη σύνταξη, εκτεταμένες βιβλιοθήκες και ισχυρή υποστήριξη της κοινότητας, η PHP παραμένει σχετική και επιδραστική στον συνεχώς εξελισσόμενο τομέα της ανάπτυξης Ιστού.

Παρακάτω παρουσιάζονται τρία απλά παραδείγματα κώδικα PHP:

```
<?php  
echo "Hello, World!";  
?>
```

Αυτό είναι το κλασικό "Hello, World!" πρόγραμμα σε PHP. Χρησιμοποιεί τη δήλωση echo για την έξοδο του καθορισμένου κειμένου.

Βασικός Υπολογισμός:

```
<?php  
$number1 = 10;  
$number2 = 5;  
$sum = $number1 + $number2;  
echo "Το άθροισμα των $number1 και $number2 είναι: $sum";  
?>
```

Αυτό το παράδειγμα δείχνει τη βασική χρήση μεταβλητών και τις αριθμητικές πράξεις. Προσθέτει δύο αριθμούς και εμφανίζει το αποτέλεσμα.

Απλός χειρισμός:

```
<!DOCTYPE html>

<html>

<head>

  <title>Simple Form</title>

</head>

<body>

  <h2>Simple Form</h2>

  <form method="POST" action="">

    <label for="name">Enter your name:</label>

    <input type="text" name="name" id="name">

    <input type="submit" name="submit" value="Submit">

  </form>

  <?php
  if (isset($_POST['submit'])) {
    $name = $_POST['name'];
    echo "Hello, $name!";
  }
  ?>

</body>

</html>
```

Αυτό το παράδειγμα παρουσιάζει μια απλή φόρμα HTML που λαμβάνει ως είσοδο το όνομα ενός χρήστη. Όταν υποβληθεί η φόρμα, ο κώδικας PHP ανακτά το όνομα που εισαγάγατε και εμφανίζει ένα μήνυμα χαιρετισμού.

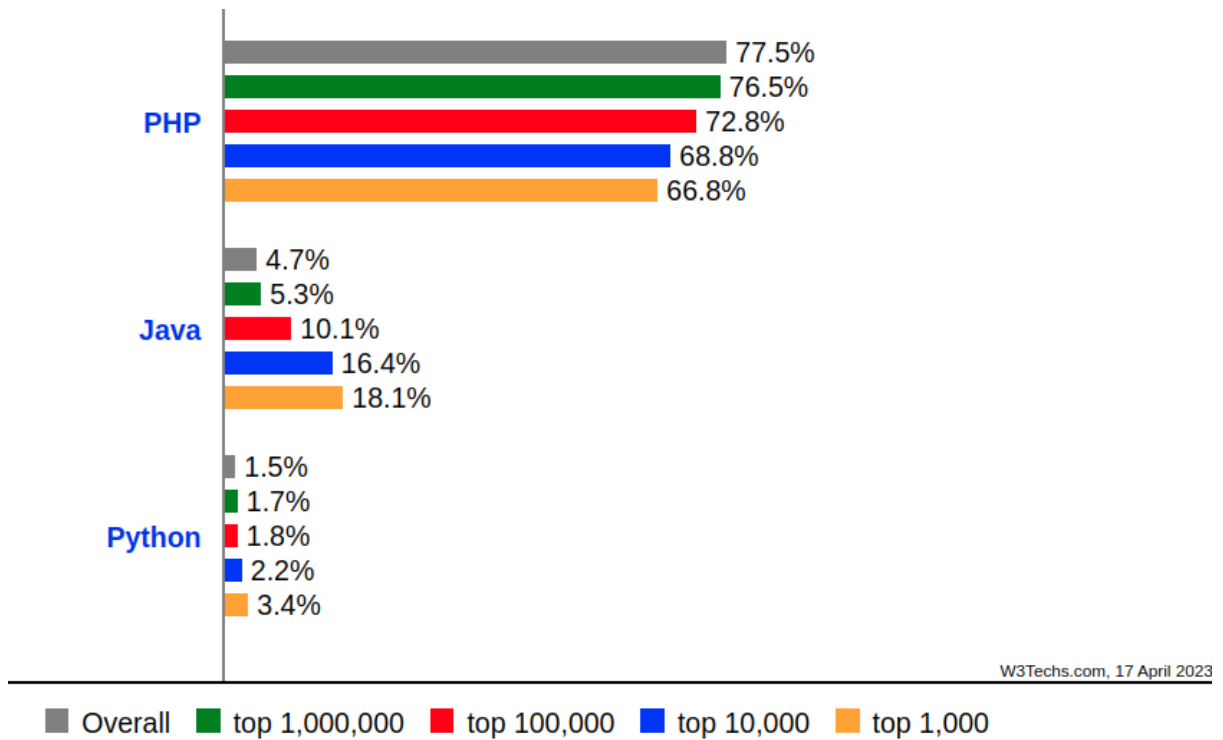
Για να εκτελέσετε κώδικα PHP, χρειάζεστε συνήθως έναν διακομιστή web και έναν διερμηνέα PHP.

Θα χρειαστείτε ένα λογισμικό διακομιστή ιστού όπως το Apache, το Nginx ή το LiteSpeed εγκατεστημένο στον υπολογιστή σας. Αυτοί οι διακομιστές θα χειρίζονται αιτήματα HTTP και θα εξυπηρετούν τα αρχεία σας PHP σε προγράμματα περιήγησης.

Πρέπει να έχετε εγκαταστήσει την PHP στον υπολογιστή σας. Μπορείτε να κάνετε λήψη του προγράμματος εγκατάστασης PHP από τον επίσημο ιστότοπο της PHP (<https://www.php.net/downloads.php>) ή να χρησιμοποιήσετε διαχειριστές πακέτων όπως το apt (στο Ubuntu/Debian) ή το brew (στο macOS) για να το εγκαταστήσετε.

Εάν χρησιμοποιείτε έναν τοπικό διακομιστή, όπως XAMPP ή WAMP, υπάρχει συνήθως ένας κατάλογος που ονομάζεται "htdocs" ή "www" όπου πρέπει να τοποθετήσετε το αρχείο PHP. Για εγκαταστάσεις μεμονωμένων διακομιστών, διαμορφώστε τη ρίζα του εγγράφου του διακομιστή σας ώστε να οδηγεί στον κατάλογο όπου βρίσκεται το αρχείο PHP. Εκκινήστε τον διακομιστή Ιστού σας. Ανοίξτε ένα πρόγραμμα περιήγησης και εισαγάγετε τη διεύθυνση URL στο αρχείο PHP. Εάν το αρχείο PHP σας ονομάζεται index.php και ο διακομιστής σας εκτελείται στον τοπικό σας υπολογιστή, μπορείτε να αποκτήσετε πρόσβαση σε αυτό πληκτρολογώντας κάτι σαν <http://localhost/index.php>. Ο διακομιστής θα εκτελέσει τον κώδικα PHP και θα επιστρέψει το αποτέλεσμα στο πρόγραμμα περιήγησης.

Η ανάπτυξη Ιστού εξελίσσεται ταχέως, με νέες τεχνολογίες και γλώσσες να αναδύονται καθημερινά. Μερικές δοκιμασμένες και δοκιμασμένες γλώσσες έχουν αντέξει στη δοκιμασία του χρόνου όσον αφορά την ανάπτυξη ιστού. Μεταξύ .NET, PHP, Java και Python, η PHP χρησιμοποιείται συνήθως από το 77,5% όλων των ιστότοπων. Αυτό δεν σημαίνει ότι η PHP είναι η καλύτερη για όλα τα έργα ανάπτυξης. [3]



Εικόνα 3.2: Σύγκριση χρήσης PHP με διάφορες γλώσσες

[<https://www.capconnect.com/blog/en/php-8/> :17-04-2023]

Apr 2023	Apr 2022	Change	Programming Language	Ratings	Change
1	1		Python	14.51%	+0.59%
2	2		C	14.41%	+1.71%
3	3		Java	13.23%	+2.41%
4	4		C++	12.96%	+4.68%
5	5		C#	8.21%	+1.39%
6	6		Visual Basic	4.40%	-1.00%
7	7		JavaScript	2.10%	-0.31%
8	9	▲	SQL	1.68%	-0.61%
9	10	▲	PHP	1.36%	-0.28%

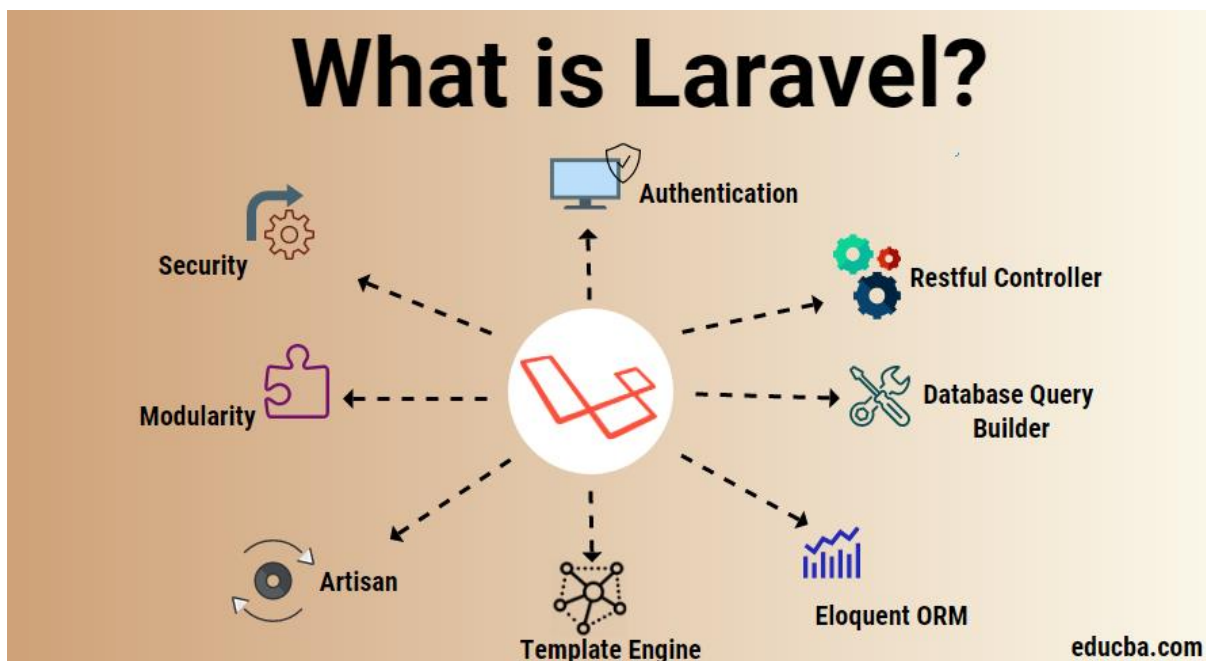
Εικόνα 3.3: Σύγκριση δημοτικότητας PHP με διάφορες γλώσσες

(Περισσότερο δημοφιλείς server-side programming languages)

[<https://www.tiobe.com/tiobe-index/>: 25-08-2023]

3.2 Laravel

Το Laravel, το οποίο συχνά διαφημίζεται ως «το framework PHP για τους τεχνίτες ιστού», είναι ένα ισχυρό και φιλικό προς τους αρχάριους πλαίσιο που απλοποιεί τη διαδικασία δημιουργίας εφαρμογών Ιστού. Ακόμα κι αν κάποιος εντελώς αρχάριος στην ανάπτυξη ιστού, η Laravel μπορεί να χρησιμεύσει ως ένα εξαιρετικό σημείο εκκίνησης. Αυτό το framework παρέχει τα απαραίτητα εργαλεία και τη δομή για τη δημιουργία ισχυρών και διατηρήσιμων εφαρμογών ιστού χωρίς να εμβαθύνει σε πολύπλοκες τεχνικές λεπτομέρειες. [4]



Εικόνα 3.4: Framework Laravel

[<https://www.educba.com/what-is-laravel/>]

Στον πυρήνα του, η Laravel δίνει έμφαση σε μια καθαρή και εκφραστική σύνταξη που κάνει την κωδικοποίηση παιχνιδάκι για αρχάριους. Το πλαίσιο ακολουθεί το αρχιτεκτονικό μοτίβο Model-View-Controller (MVC), που σημαίνει ότι ο κώδικας είναι οργανωμένος σε τρία κύρια στοιχεία: μοντέλα (για αλληλεπίδραση με βάση δεδομένων), προβολές (για διεπαφή χρήστη) και ελεγκτές (για το χειρισμό της λογικής εφαρμογής). Αυτός ο διαχωρισμός απλοποιεί την ανάπτυξη, καθιστώντας ευκολότερη τη διαχείριση και την τροποποίηση διαφορετικών τμημάτων της εφαρμογής.

Ένα από τα ξεχωριστά χαρακτηριστικά του Laravel είναι η ενσωματωμένη υποστήριξη διαχείρισης βάσεων δεδομένων μέσω ενός συστήματος αντικειμενικής-σχεσιακής χαρτογράφησης (ORM) που ονομάζεται Eloquent. Αυτό εξαλείφει την ανάγκη για ακατέργαστα ερωτήματα SQL και επιτρέπει να αλληλεπιδράτε με τη βάση δεδομένων σας χρησιμοποιώντας διαισθητική σύνταξη PHP. Για αρχάριους,

αυτό μεταφράζεται σε λιγότερο χρόνο για πολύπλοκες αλληλεπιδράσεις με βάση δεδομένων και περισσότερο χρόνο εστίασης στην πραγματική λειτουργικότητα της εφαρμογής.

Το Laravel συνοδεύεται επίσης από μια ευρεία γκάμα προκατασκευασμένων λειτουργιών και βιβλιοθηκών, γνωστών ως "πακέτα". Αυτά τα πακέτα καλύπτουν τα πάντα, από τον έλεγχο ταυτότητας και την ασφάλεια έως τα API και τη διαχείριση χρηστών. Ως αρχάριος, αυτό σημαίνει προηγμένες λειτουργίες στην εφαρμογή χωρίς να χρειάζεται εκτεταμένος κώδικας από την αρχή. Με την απλή προσθήκη και διαμόρφωση αυτών των πακέτων, μπορεί ο προγραμματιστής να εξοικονομήσει χρόνο και προσπάθεια κατά τη δημιουργία μιας εφαρμογής πλούσιας σε δυνατότητες.

Εκτός από τα τεχνικά χαρακτηριστικά του, η Laravel διαθέτει μια ισχυρή και φιλική κοινότητα. Αυτό σημαίνει ότι ως αρχάριος, παρέχεται πρόσβαση σε πληθώρα εκπαιδευτικών προγραμμάτων, τεκμηρίωσης και φόρουμ όπου έμπειροι προγραμματιστές είναι πρόθυμοι να βοηθήσουν.

Η Laravel προσφέρει ένα εξαιρετικό σημείο εισόδου για αρχάριους στην ανάπτυξη Ιστού. Με την καθαρή σύνταξη, την αρχιτεκτονική MVC, το Eloquent ORM, τα προκατασκευασμένα πακέτα και την υποστηρικτική κοινότητα, κάποιος μπορεί να κάνει με σιγουριά τα πρώτα του βήματα στη δημιουργία διαδικτυακών εφαρμογών. Καθώς αποκτάει εμπειρία και εξειδίκευση, θα διαπιστώσει ότι οι δυνατότητες της Laravel μπορούν να κλιμακωθούν για να ανταποκριθούν στις απαιτήσεις πιο περίπλοκων έργων, καθιστώντας την πολύτιμη δεξιότητα που πρέπει να κατακτηθεί τόσο για νεοφερμένους όσο και για έμπειρους προγραμματιστές.

Το Laravel, ένα δημοφιλές πλαίσιο PHP, έχει κερδίσει τεράστια αναγνώριση και υιοθέτηση στην κοινότητα ανάπτυξης Ιστού. Τα μοναδικά χαρακτηριστικά και οι δυνατότητές του το καθιστούν μια προτιμώμενη επιλογή για τη δημιουργία ισχυρών και επεκτάσιμων διαδικτυακών εφαρμογών.

Η Laravel υπερηφανεύεται για την κομψή και εκφραστική σύνταξη της, καθιστώντας την εξαιρετικά φιλική προς τους προγραμματιστές. Το πλαίσιο ακολουθεί το αρχιτεκτονικό μοτίβο Model-View-Controller (MVC), διευκολύνοντας τον σαφή διαχωρισμό των ανησυχιών και βελτιώνοντας τη δυνατότητα συντήρησης του κώδικα. Η διαισθητική σύνταξή του επιταχύνει την ανάπτυξη, επιτρέποντας στους προγραμματιστές να επικεντρωθούν στις δυνατότητες κατασκευής αντί να εμπλέκονται σε πολύπλοκες δομές κώδικα.

Το Laravel περιλαμβάνει μια ισχυρή διεπαφή γραμμής εντολών που ονομάζεται Artisan. Βελτιώνει επαναλαμβανόμενες εργασίες, όπως μετεγκαταστάσεις βάσεων δεδομένων, δημιουργία μοντέλων και πολλά άλλα. Το Artisan εξουσιοδοτεί τους προγραμματιστές να αυτοματοποιούν διάφορες εργασίες ανάπτυξης, ενισχύοντας την παραγωγικότητα και διατηρώντας τη συνέπεια μεταξύ των έργων.

Το ενσωματωμένο σύστημα μετεγκατάστασης βάσεων δεδομένων της Laravel απλοποιεί τη διαχείριση σχημάτων βάσης δεδομένων, επιτρέποντας στους προγραμματιστές να ορίζουν και να τροποποιούν δομές βάσης δεδομένων χρησιμοποιώντας κώδικα. Επιπλέον, το Eloquent ORM (Object-Relational

Mapping) απλοποιεί τις λειτουργίες της βάσης δεδομένων παρέχοντας έναν διαισθητικό τρόπο αλληλεπίδρασης με βάσεις δεδομένων χρησιμοποιώντας κλάσεις PHP, μειώνοντας την ανάγκη για πολύπλοκα ερωτήματα SQL.

Η επιλογή της Laravel για τα έργα ανάπτυξης Ιστού σας μπορεί να οδηγήσει σε βελτιωμένη παραγωγικότητα, δυνατότητα συντήρησης και δυνατότητα δημιουργίας εφαρμογών πλούσιες σε χαρακτηριστικά. Η κομψή σύνταξη, το πλούσιο οικοσύστημα πακέτων, τα ισχυρά εργαλεία γραμμής εντολών, οι αποτελεσματικές δυνατότητες διαχείρισης βάσεων δεδομένων και η ισχυρή υποστήριξη της κοινότητας συνδυάζονται για να κάνουν τη Laravel μια συναρπαστική επιλογή για σύγχρονες προσπάθειες ανάπτυξης ιστού. Είτε δημιουργείτε μια μικρής κλίμακας εφαρμογή είτε μια πολύπλοκη διαδικτυακή πλατφόρμα, η Laravel παρέχει τα εργαλεία και τη δομή που είναι απαραίτητα για τον εξορθολογισμό της ανάπτυξης και τη δημιουργία λύσεων υψηλής ποιότητας, επεκτάσιμων.

Σύγκριση με άλλα frameworks PHP

Η σύγκριση του Laravel με άλλα πλαίσια PHP περιλαμβάνει την αξιολόγηση διαφόρων παραγόντων όπως χαρακτηριστικά, απόδοση, υποστήριξη κοινότητας, ευκολία χρήσης, επεκτασιμότητα και πολλά άλλα.

Symfony:

Το Symfony είναι ένα καθιερωμένο πλαίσιο PHP, γνωστό για την ευελιξία του.

Παρέχει ένα σύνολο αποσυνδεδεμένων και επαναχρησιμοποιήσιμων εξαρτημάτων που μπορούν να χρησιμοποιηθούν ανεξάρτητα.

Η καμπύλη εκμάθησης της Symfony μπορεί να είναι πιο απότομη σε σύγκριση με τη Laravel λόγω της πολύπλοκης αρχιτεκτονικής της.

Πηγή: Επίσημη Ιστοσελίδα Symfony [5]

CodeIgniter:

Το CodeIgniter είναι ελαφρύ και εύκολο στην εκμάθηση, καθιστώντας το κατάλληλο για μικρού έως μεσαίου μεγέθους έργα.

Δεν διαθέτει πολλές ενσωματωμένες δυνατότητες που προσφέρει η Laravel, αλλά αυτό μπορεί να είναι πλεονέκτημα για έργα που απαιτούν μινιμαλιστική προσέγγιση.

Το CodeIgniter δεν ακολουθεί το μοτίβο MVC τόσο αυστηρά όσο η Laravel.

Πηγή: Επίσημος ιστότοπος CodeIgniter [6]

Yii:

Το Yii είναι γνωστό για την υψηλή απόδοση και τη στιβαρότητά του.

Έρχεται με ένα σύνολο εργαλείων που απλοποιούν εργασίες όπως ο έλεγχος ταυτότητας, η προσωρινή αποθήκευση και η ανάπτυξη RESTful API.

Το Yii 2 εισήγαγε σύγχρονα χαρακτηριστικά και βελτιωμένη απόδοση σε σχέση με τον προκάτοχό του, το Yii 1.

Πηγή: Yii Framework Επίσημος Ιστότοπος [7]

Zend Framework:

Το Zend Framework, γνωστό πλέον ως Laminas, εστιάζει στην παροχή μιας συλλογής επαγγελματικών πακέτων PHP.

Είναι κατάλληλο για εφαρμογές μεγάλης κλίμακας όπου τα εξαρτήματα μπορούν να χρησιμοποιηθούν μεμονωμένα.

Το πλαίσιο είναι γνωστό για την ευελιξία του, αλλά μπορεί να απαιτεί περισσότερες ρυθμίσεις σε σύγκριση με το Laravel.

Πηγή: Ιστοσελίδα Laminas Project [8]

Phalcon:

Το Phalcon είναι μοναδικό καθώς υλοποιείται ως επέκταση C, με αποτέλεσμα εξαιρετική απόδοση.

Η αρχιτεκτονική του στοχεύει στην ελαχιστοποίηση των γενικών εξόδων και στη μεγιστοποίηση της απόδοσης.

Το Phalcon μπορεί να είναι μια καλή επιλογή για εφαρμογές κρίσιμες για την απόδοση.

Πηγή: Επίσημος Ιστότοπος Phalcon [9]

Χρήση

Ένας οδηγός φιλικός για αρχάριους για το πώς να ξεκινήσετε με το Laravel:

1. Ρύθμιση του αναπτυξιακού περιβάλλοντος:

Για να ξεκινήσετε με τη Laravel, χρειάζεστε ένα περιβάλλον ανάπτυξης. Ξεκινήστε εγκαθιστώντας έναν τοπικό διακομιστή ανάπτυξης όπως ο XAMPP ή ο Laravel Valet. Αυτά τα εργαλεία σας βοηθούν να ρυθμίσετε έναν διακομιστή για την εκτέλεση των εφαρμογών Laravel σας. Μόλις εγκατασταθεί, βεβαιωθείτε ότι έχετε εγκαταστήσει το PHP και το Composer (ένας διαχειριστής εξαρτήσεων PHP). Το Composer είναι ζωτικής σημασίας για την εγκατάσταση του Laravel και των εξαρτήσεών του.

2. Εγκατάσταση Laravel:

Με το περιβάλλον ανάπτυξής σας στη θέση του μπορείτε να εγκαταστήσετε το Laravel. Ανοίξτε ένα τερματικό και εκτελέστε την ακόλουθη εντολή:

```
composer global require laravel/installer
```

Αυτό εγκαθιστά το πρόγραμμα εγκατάστασης Laravel παγκοσμίως στο σύστημά σας. Μετά την εγκατάσταση, μπορείτε να δημιουργήσετε ένα νέο έργο Laravel εκτελώντας:

```
laravel new grammateia
```

Αυτή η εντολή θα κατεβάσει και θα εγκαταστήσει μια νέα εγκατάσταση Laravel.

3. Διερεύνηση της δομής του έργου:

Μόλις δημιουργηθεί το έργο σας, μεταβείτε στον κατάλόγό του χρησιμοποιώντας το τερματικό. Η Laravel ακολουθεί το μοτίβο MVC (Model-View-Controller). Θα βρείτε τη λογική της κύριας εφαρμογής στο φάκελο "app", τις προβολές στο φάκελο "resources/views" και τις διαδρομές στο φάκελο "routes". Ο "δημόσιος" φάκελος χρησιμεύει ως σημείο εισόδου για την εφαρμογή σας.

4. Δημιουργία Διαδρομών και Προβολών:

Το σύστημα δρομολόγησης της Laravel είναι φιλικό προς το χρήστη. Ανοίξτε το αρχείο "routes/web.php" για να ορίσετε τις διαδρομές σας. Κάθε διαδρομή αντιστοιχίζεται σε μια συγκεκριμένη διεύθυνση URL και μια ενέργεια ελεγκτή. Δημιουργήστε αντίστοιχα αρχεία προβολής

στο φάκελο "πόροι/προβολές" για απόδοση περιεχομένου HTML. Μπορείτε να χρησιμοποιήσετε το Blade, τη μηχανή προτύπων της Laravel, για να προσθέσετε δυναμικό περιεχόμενο στις προβολές σας.

5. Εργασία με ελεγκτές και μοντέλα:

Οι ελεγκτές χειρίζονται τη λογική πίσω από την αίτησή σας. Δημιουργήστε ελεγκτές χρησιμοποιώντας την εντολή artisan:

```
php artisan make:controller GrammateiaController
```

Αυτή η εντολή δημιουργεί ένα νέο αρχείο ελεγκτή στον κατάλογο "app/Http/Controllers". Τα μοντέλα, από την άλλη πλευρά, αντιπροσωπεύουν τις δομές δεδομένων της εφαρμογής σας. Μπορείτε να δημιουργήσετε ένα μοντέλο χρησιμοποιώντας την εντολή:

```
php artisan make:model GrammateiaModel
```

Τα μοντέλα συνήθως αποθηκεύονται στον κατάλογο "app". Θυμηθείτε να ορίσετε σχέσεις μεταξύ μοντέλων και να ρυθμίσετε τις συνδέσεις βάσης δεδομένων στο αρχείο "config/database.php".

3.3 MySQL

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ανοιχτού κώδικα (RDBMS) που έχει αποκτήσει ευρεία δημοτικότητα για την αξιοπιστία, την απόδοση και την ευκολία χρήσης του. Αναπτύχθηκε από την Oracle Corporation, η MySQL είναι ένας βασικός παίκτης στον κόσμο της βάσης δεδομένων, ενδυναμώνοντας διάφορες εφαρμογές, από εφαρμογές web μικρής κλίμακας έως συστήματα μεγάλων επιχειρήσεων. Τα ισχυρά χαρακτηριστικά, η επεκτασιμότητα και η ανάπτυξη με γνώμονα την κοινότητα το καθιστούν ιδανική επιλογή για πολλούς προγραμματιστές και οργανισμούς. [10]

Η MySQL προσφέρει μια σειρά από δυνατότητες που καλύπτουν τόσο τις βασικές όσο και τις προηγμένες ανάγκες βάσης δεδομένων. Υποστηρίζει διάφορους τύπους δεδομένων, που κυμαίνονται από ακέραιους έως συμβολοσειρές, ημερομηνίες και πολλά άλλα, επιτρέποντας στους προγραμματιστές να αντιπροσωπεύουν διάφορους τύπους πληροφοριών. Η συμμόρφωση ACID (Atomicity, Consistency, Isolation, Durability) διασφαλίζει την ακεραιότητα και την αξιοπιστία των δεδομένων, καθιστώντας τα κατάλληλα για κρίσιμες εφαρμογές. Οι δυνατότητες βελτιστοποίησης απόδοσης της MySQL, συμπεριλαμβανομένης της δημιουργίας ευρετηρίου, της προσωρινής

αποθήκευσης και της βελτιστοποίησης ερωτημάτων, ενισχύουν την αποτελεσματικότητα των λειτουργιών της βάσης δεδομένων.

Η MySQL χρησιμοποιεί την SQL, μια τυπική γλώσσα προγραμματισμού για τη διαχείριση και το χειρισμό σχεσιακών βάσεων δεδομένων. Η SQL επιτρέπει στους χρήστες να δημιουργούν, να τροποποιούν και να ανακτούν δεδομένα από βάσεις δεδομένων. Με τη MySQL, μπορείτε να εκτελέσετε σύνθετα ερωτήματα, να συγκεντρώσετε δεδομένα, να πραγματοποιήσετε συνδέσεις σε πολλούς πίνακες και ακόμη και να διαχειριστείτε την πρόσβαση και τα δικαιώματα των χρηστών. Αυτό το καθιστά ένα ευέλικτο εργαλείο για το χειρισμό διαφόρων εργασιών που σχετίζονται με δεδομένα. Η MySQL δίνει μεγάλη έμφαση στην ασφάλεια των δεδομένων. Παρέχει μηχανισμούς ελέγχου ταυτότητας και εξουσιοδότησης για τον έλεγχο της πρόσβασης των χρηστών στη βάση δεδομένων. Στους χρήστες μπορούν να εκχωρηθούν συγκεκριμένα προνόμια, όπως πρόσβαση μόνο για ανάγνωση ή ανάγνωση-εγγραφή, για να διασφαλιστεί η εμπιστευτικότητα και η ακεραιότητα των δεδομένων. Επιπλέον, η MySQL υποστηρίζει κρυπτογράφηση για δεδομένα σε μεταφορά και σε κατάσταση ηρεμίας, προστατεύοντας ευαίσθητες πληροφορίες από πιθανές παραβιάσεις.

Η MySQL προσφέρει επιλογές κλιμάκωσης τόσο κάθετες όσο και οριζόντιες. Η κάθετη επεκτασιμότητα περιλαμβάνει την αναβάθμιση πόρων υλικού σε έναν μόνο διακομιστή, ενώ η οριζόντια επεκτασιμότητα συνεπάγεται την κατανομή του φόρτου εργασίας σε πολλούς διακομιστές. Αυτή η ευελιξία επιτρέπει στην MySQL να δέχεται αυξανόμενες ποσότητες δεδομένων και αυξημένη επισκεψιμότητα χρηστών. Με τη βελτιστοποίηση των ερωτημάτων και τη διαμόρφωση μηχανισμών προσωρινής αποθήκευσης, οι προγραμματιστές μπορούν να βελτιώσουν περαιτέρω την απόδοση των βάσεων δεδομένων MySQL.

Ένα από τα δυνατά σημεία της MySQL είναι η μεγάλη και ενεργή κοινότητα προγραμματιστών, διαχειριστών και χρηστών. Η κοινότητα συμβάλλει στη συνεχή ανάπτυξη της MySQL παρέχοντας σχόλια, αναφέροντας σφάλματα και αναπτύσσοντας πρόσθετα και επεκτάσεις. Αυτό το οικοσύστημα υποστήριξης διασφαλίζει ότι το σύστημα βάσης δεδομένων παραμένει ενημερωμένο με τις τελευταίες τάσεις και απαιτήσεις, ενώ προσφέρει επίσης πληθώρα πόρων, όπως τεκμηρίωση, φόρουμ και σεμινάρια.

Βρίσκει εφαρμογή σε διάφορους τομείς, συμπεριλαμβανομένου του ηλεκτρονικού εμπορίου, των συστημάτων διαχείρισης περιεχομένου, της αποθήκευσης δεδομένων και άλλων. Η ευελιξία και η επεκτασιμότητα του το καθιστούν κατάλληλο για νεοφυείς επιχειρήσεις που θέλουν να δημιουργήσουν μια ισχυρή βάση για τις εφαρμογές τους, καθώς και για καθιερωμένες επιχειρήσεις που αναζητούν μια ισχυρή λύση βάσης δεδομένων. Οι δυνατότητες ενσωμάτωσης της MySQL με γλώσσες προγραμματισμού όπως η PHP, η Python και η Java συμβάλλουν επίσης στη δημοτικότητά της μεταξύ των προγραμματιστών παγκοσμίως. Αποτελεί ένα ισχυρό και ευέλικτο RDBMS που συνεχίζει να διαδραματίζει σημαντικό ρόλο στον κόσμο της διαχείρισης δεδομένων. Το πλούσιο σύνολο

δυνατοτήτων του, η τήρηση των προτύπων και η ζωντανή υποστήριξη της κοινότητας το καθιστούν εξαιρετική επιλογή για έργα όλων των μεγεθών, είτε είστε μεμονωμένος προγραμματιστής που δημιουργεί μια εφαρμογή ιστού είτε αρχιτέκτονας επιχειρήσεων που σχεδιάζει μια πολύπλοκη υποδομή δεδομένων.

Η χρήση της MySQL με PHP είναι μια κοινή πρακτική στην ανάπτυξη ιστού και υπάρχουν διάφοροι λόγοι για τους οποίους χρησιμοποιούνται συχνά μαζί. Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) που σας επιτρέπει να αποθηκεύετε και να ανακτάτε δεδομένα με δομημένο τρόπο. Η PHP μπορεί να αλληλεπιδράσει με βάσεις δεδομένων MySQL για να εκτελέσει λειτουργίες όπως η αποθήκευση πληροφοριών χρήστη, λεπτομερειών προϊόντων και διάφορων άλλων τύπων δεδομένων που χρειάζονται οι ιστότοποι και οι εφαρμογές. Οι ιστότοποι απαιτούν συχνά μηχανισμούς ελέγχου ταυτότητας και εξουσιοδότησης χρήστη. Η PHP μπορεί να χειριστεί τις συνδέσεις χρηστών, την εγγραφή και τη διαχείριση περιόδων σύνδεσης, ενώ η MySQL μπορεί να αποθηκεύει τα διαπιστευτήρια χρήστη με ασφάλεια και να διαχειρίζεται τους ρόλους και τα δικαιώματα χρήστη. Η MySQL και η PHP χρησιμοποιούνται συχνά σε πλατφόρμες ηλεκτρονικού εμπορίου και σε διάφορες διαδικτυακές εφαρμογές όπου τα δεδομένα πρέπει να αποθηκεύονται, να ανακτώνται και να χειρίζονται σε πραγματικό χρόνο. Για παράδειγμα, τα καλάθια αγορών, η επεξεργασία παραγγελιών και η διαχείριση αποθέματος μπορούν να εφαρμοστούν αποτελεσματικά χρησιμοποιώντας αυτόν τον συνδυασμό. Πολλά συστήματα διαχείρισης περιεχομένου, όπως το WordPress και το Joomla, έχουν δημιουργηθεί με χρήση PHP και MySQL. Αυτά τα συστήματα βασίζονται στη βάση δεδομένων για την αποθήκευση περιεχομένου, δεδομένων χρήστη, προτύπων και διαμορφώσεων. Η MySQL έχει σχεδιαστεί για να χειρίζεται μεγάλες ποσότητες δεδομένων αποτελεσματικά. Όταν συνδυάζεται με την PHP, μπορείτε να δημιουργήσετε επεκτάσιμες εφαρμογές που μπορούν να διαχειριστούν σημαντικά φορτία χρηστών και όγκους δεδομένων.

Τόσο η MySQL όσο και η PHP είναι τεχνολογίες ανοιχτού κώδικα με μια μεγάλη και ενεργή κοινότητα προγραμματιστών. Αυτό σημαίνει ότι μπορείτε να βρείτε εκτενή τεκμηρίωση, σεμινάρια και πόρους που θα σας βοηθήσουν να μάθετε, να αντιμετωπίσετε προβλήματα και να βελτιστοποιήσετε τις εφαρμογές σας. Η χρήση τεχνολογιών ανοιχτού κώδικα όπως η MySQL και η PHP μπορεί να μειώσει το κόστος που σχετίζεται με τα τέλη αδειοδότησης. Αυτό το καθιστά ελκυστική επιλογή, ειδικά για νεοφυείς επιχειρήσεις και μικρές επιχειρήσεις. Η MySQL υποστηρίζει διάφορους τύπους δεδομένων και σας επιτρέπει να ορίζετε σχέσεις μεταξύ πινάκων, επιτρέποντάς σας να σχεδιάζετε πολύπλοκες βάσεις δεδομένων. Η ευελιξία της PHP σας επιτρέπει να δημιουργήσετε προσαρμοσμένη λογική και ροές εργασίας γύρω από αυτά τα δεδομένα. Η MySQL και η PHP μπορούν να ενσωματωθούν με άλλες τεχνολογίες και υπηρεσίες. Για παράδειγμα, μπορείτε να χρησιμοποιήσετε πλαίσια JavaScript στο frontend για να δημιουργήσετε δυναμικές διεπαφές που επικοινωνούν με την PHP στο backend, το οποίο με τη σειρά του αλληλεπιδρά με τη βάση δεδομένων MySQL. Η χρήση της MySQL με PHP παρέχει μια ισχυρή στοίβα για τη δημιουργία δυναμικών εφαρμογών ιστού που βασίζονται σε

δεδομένα. Ενώ υπάρχουν άλλες διαθέσιμες επιλογές, αυτός ο συνδυασμός είναι μια δημοφιλής επιλογή για πολλά χρόνια λόγω της αποτελεσματικότητας, της ευελιξίας και της υποστήριξης της κοινότητας.

MariaDB

Η MariaDB είναι ένα ανοιχτού κώδικα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων που έχει κερδίσει δημοτικότητα ως διχάλα της MySQL. Αναπτύχθηκε από τους αρχικούς δημιουργούς της MySQL αφού προέκυψαν ανησυχίες σχετικά με την κατεύθυνση που πήρε η Oracle, η μητρική εταιρεία της MySQL, με τις ιδιότητες προσφορές της. [11]

Η MariaDB συχνά υπερηφανεύεται για καλύτερη απόδοση σε σύγκριση με τη MySQL. Περιλαμβάνει διάφορες μηχανές αποθήκευσης όπως Aria, TokuDB και άλλες, οι οποίες μπορούν να βελτιώσουν την ταχύτητα και να βελτιστοποιήσουν τη χρήση των πόρων. Επίσης, διατηρεί ισχυρή δέσμευση στις αρχές ανοιχτού κώδικα και ενθαρρύνει τις συνεισφορές της κοινότητας. Αυτό έχει ως αποτέλεσμα πιο γρήγορη ανάπτυξη, τακτικές ενημερώσεις και πληθώρα προσθηκών και επεκτάσεων που δημιουργούνται από τον χρήστη. Η MariaDB έχει σχεδιαστεί για να είναι εξαιρετικά συμβατό με τη MySQL. Οι περισσότερες εφαρμογές και εργαλεία που λειτουργούν με MySQL μπορούν να χρησιμοποιηθούν με την MariaDB χωρίς σημαντικές τροποποιήσεις. Αυτή η συμβατότητα απλοποιεί τις διαδικασίες μετεγκατάστασης.

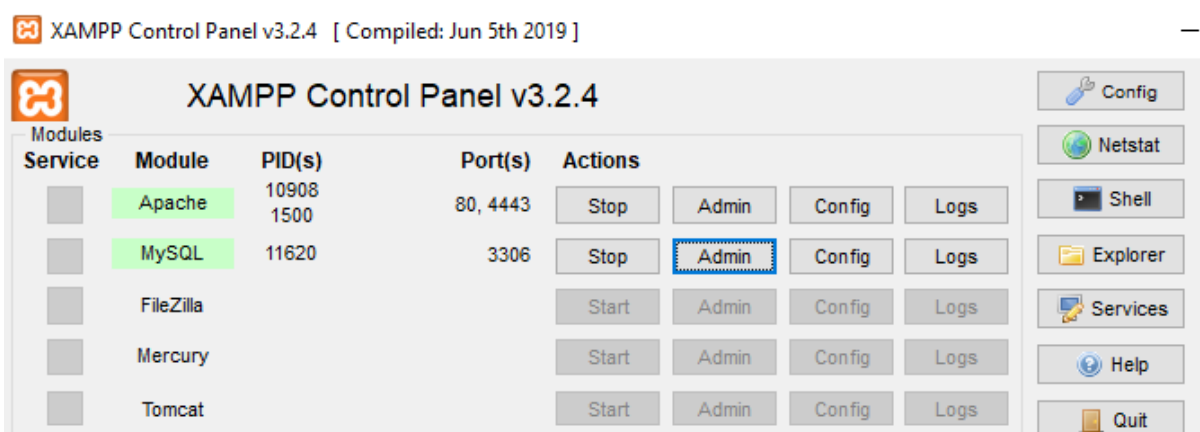
Η ανάπτυξη της MariaDB είναι διαφανής και καθοδηγείται από την κοινότητα. Οι αποφάσεις λαμβάνονται συλλογικά και οι χάρτες πορείας είναι ανοιχτοί σε δημόσια συζήτηση. Αυτό έρχεται σε αντίθεση με τη MySQL, όπου η ανάπτυξη μπορεί να επηρεαστεί από μια εταιρική ατζέντα. Επιπρόσθετα προσφέρει μια ποικιλία μηχανών αποθήκευσης, συμπεριλαμβανομένων των προεπιλεγμένων Aria, InnoDB (η προεπιλογή και στη MySQL) και άλλων όπως το TokuDB και το ColumnStore. Αυτή η ποικιλομορφία επιτρέπει στους χρήστες να βελτιστοποιούν τη βάση δεδομένων τους για συγκεκριμένες περιπτώσεις χρήσης. Η MariaDB συχνά εφαρμόζει βελτιώσεις ασφαλείας πιο γρήγορα από τη MySQL. Δυνατότητες όπως η κρυπτογράφηση δεδομένων σε κατάσταση ηρεμίας και οι προσθήκες για έλεγχο ταυτότητας το καθιστούν ελκυστική επιλογή για χρήστες που έχουν επίγνωση της ασφάλειας. Η MariaDB συνήθως εκδίδει ενημερώσεις και διορθώνει σφάλματα πιο συχνά, διασφαλίζοντας ότι οι χρήστες έχουν πρόσβαση στις πιο πρόσφατες δυνατότητες και βελτιώσεις. Περιλαμβάνει αρκετές βελτιώσεις που σχετίζονται με τις επιδόσεις, όπως η μηχανή αποθήκευσης Aria για ταχύτερα σύνθετα ερωτήματα και η παράλληλη αναπαραγωγή για καλύτερη επεκτασιμότητα. Ενθαρρύνει την ανάπτυξη πρόσθετων, επιτρέποντας στους χρήστες να προσαρμόζουν τις βάσεις δεδομένων τους σύμφωνα με τις ανάγκες τους. Αυτό το επίπεδο επεκτασιμότητας μπορεί να ωφελήσει πολύ τις επιχειρήσεις με συγκεκριμένες απαιτήσεις. Από τα πιο σημαντικά χαρακτηριστικά είναι ότι παρέχει ένα δυναμικό χώρο αποθήκευσης στηλών για το χειρισμό μεγάλων ποσοτήτων δεδομένων και υποστηρίζει επίσης έναν τύπο δεδομένων JSON. Αυτό επιτρέπει στους χρήστες να εργάζονται με

δομημένα και ημι-δομημένα δεδομένα πιο αποτελεσματικά. Παρέχει καλύτερα και πιο διαφανή στατιστικά και διαγνωστικά, διευκολύνοντας τον εντοπισμό και την αντιμετώπιση των σημείων συμφόρησης απόδοσης.

Η επιλογή μεταξύ MariaDB και MySQL εξαρτάται από τις συγκεκριμένες ανάγκες και τις προτεραιότητές σας. Η εστίαση της MariaDB στην απόδοση, τις αξίες ανοιχτού κώδικα, τη συμβατότητα και την ανάπτυξη με γνώμονα την κοινότητα το καθιστούν μια συναρπαστική επιλογή για πολλούς χρήστες που εκτιμούν αυτές τις πτυχές σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων.

3.4 Xampp

Το XAMPP (που σημαίνει Cross-Platform, Apache, MySQL, PHP και Perl) είναι ένα δημοφιλές πακέτο λογισμικού ανοιχτού κώδικα που παρέχει ένα τοπικό περιβάλλον ανάπτυξης για τη δημιουργία εφαρμογών Ιστού. Έχει σχεδιαστεί για να απλοποιεί τη διαδικασία ρύθμισης και διαμόρφωσης των απαραίτητων στοιχείων για την εκτέλεση ενός διακομιστή web και ενός διακομιστή βάσης δεδομένων στον τοπικό σας υπολογιστή. [12] Το XAMPP σας επιτρέπει να ρυθμίσετε ένα περιβάλλον τοπικού διακομιστή στον υπολογιστή σας, το οποίο προσομοιώνει έναν πραγματικό διακομιστή web. Αυτό είναι ιδιαίτερα χρήσιμο για την ανάπτυξη και τη δοκιμή εφαρμογών Ιστού πριν από την ανάπτυξή τους σε έναν ζωντανό διακομιστή. Βοηθά τους προγραμματιστές να εντοπίσουν και να διορθώσουν προβλήματα νωρίς στη διαδικασία ανάπτυξης. Το XAMPP συνοδεύεται από ένα απλό πρόγραμμα εγκατάστασης που συνδυάζει Apache (έναν διακομιστή ιστού), MySQL (ένα σύστημα διαχείρισης βάσης δεδομένων), PHP (γλώσσα δέσμης ενεργειών από την πλευρά του διακομιστή) και άλλα εργαλεία. Αυτό καθιστά τη διαδικασία εγκατάστασης πολύ πιο εύκολη σε σύγκριση με τη μη αυτόματη ρύθμιση παραμέτρων αυτών των στοιχείων ξεχωριστά.



Εικόνα 3.5: XAMPP control panel

Το XAMPP είναι διαθέσιμο για Windows, macOS, Linux και άλλες πλατφόρμες. Αυτή η υποστήριξη πολλαπλών πλατφορμών την καθιστά προσβάσιμη σε προγραμματιστές ανεξάρτητα από το λειτουργικό τους σύστημα. Επίσης, αντί να εγκαταστήσετε κάθε στοιχείο ξεχωριστά, το XAMPP παρέχει ένα ενιαίο πακέτο που περιλαμβάνει όλα όσα χρειάζεστε. Αυτό μειώνει την πολυπλοκότητα της εγκατάστασης και διασφαλίζει τη συμβατότητα μεταξύ των στοιχείων. Ακόμη, χρησιμοποιείται συχνά για την ανάπτυξη και τη δοκιμή εφαρμογών ιστού τοπικά. Μπορείτε να εργαστείτε στον κώδικά σας, να δοκιμάσετε τις αλληλεπιδράσεις της βάσης δεδομένων και να κάνετε προεπισκόπηση του ιστότοπού σας πριν τον αναπτύξετε σε έναν ζωντανό διακομιστή.

Πολύ σημαντικό είναι ότι με το XAMPP, δεν χρειάζεστε σύνδεση στο διαδίκτυο για να αναπτύξετε και να δοκιμάσετε τις εφαρμογές σας. Αυτό είναι ιδιαίτερα χρήσιμο όταν εργάζεστε σε περιβάλλοντα με περιορισμένη ή καθόλου πρόσβαση στο Διαδίκτυο. Το XAMPP μπορεί να χρησιμοποιηθεί ως εκπαιδευτικό εργαλείο για την εκμάθηση της ανάπτυξης ιστοσελίδων, καθώς παρέχει ένα ελεγχόμενο περιβάλλον για πειραματισμούς με διάφορες τεχνολογίες Ιστού χωρίς να επηρεάζονται ζωντανοί ιστότοποι. Ενώ το XAMPP δεσμεύει στοιχεία που χρησιμοποιούνται συνήθως, μπορείτε ακόμα να προσαρμόσετε τη διαμόρφωσή του για να προσθέσετε ή να αφαιρέσετε λειτουργίες σύμφωνα με τις απαιτήσεις του έργου σας. Επιτρέπει να δημιουργείτε γρήγορα πρωτότυπες εφαρμογές web χωρίς την ανάγκη για ζωντανό διακομιστή. Αυτό το περιβάλλον ταχείας ανάπτυξης μπορεί να σας βοηθήσει να επαναλάβετε τις ιδέες σας πιο γρήγορα. Είναι σημαντικό να σημειωθεί ότι το XAMPP έχει σχεδιαστεί για τοπική ανάπτυξη και δοκιμή. Δεν συνιστάται η χρήση του XAMPP ως περιβάλλον παραγωγής, καθώς ενδέχεται να μην έχει το ίδιο επίπεδο μέτρων ασφαλείας με έναν επαγγελματικά διαχειριζόμενο διακομιστή web.

Το XAMPP παρέχει έναν βολικό τρόπο για να ρυθμίσετε ένα τοπικό περιβάλλον ανάπτυξης ιστού που περιλαμβάνει βασικά στοιχεία όπως διακομιστή web, διακομιστή βάσης δεδομένων και γλώσσες δέσμης ενεργειών. Είναι ένα χρήσιμο εργαλείο για προγραμματιστές που θέλουν να εργαστούν σε εφαρμογές web πριν τις αναπτύξουν σε έναν ζωντανό διακομιστή.

3.4.1 Apache Server

Ο Apache, είναι ένα ευρέως χρησιμοποιούμενο λογισμικό διακομιστή web ανοιχτού κώδικα που αναπτύχθηκε και διατηρείται από το Ίδρυμα Λογισμικού Apache. Είναι ένα από τα πιο δημοφιλή λογισμικά διακομιστών ιστού στο διαδίκτυο, που χρησιμεύει ως βάση για πολλούς ιστότοπους και εφαρμογές Ιστού.[13] Ο Apache HTTP Server είναι ικανός να χειρίζεται διάφορα πρωτόκολλα, συμπεριλαμβανομένων των HTTP, HTTPS και άλλων. Είναι εξαιρετικά προσαρμόσιμο μέσω αρχείων διαμόρφωσης, υποστηρίζει πολλαπλές μονάδες για επέκταση της λειτουργικότητάς του και είναι

διαθέσιμο για διάφορα λειτουργικά συστήματα. Το Apache είναι ένα λογισμικό ανοιχτού κώδικα, που σημαίνει ότι μπορείτε να το χρησιμοποιήσετε χωρίς τέλη αδειοδότησης. Αυτό το καθιστά μια οικονομικά αποδοτική επιλογή για άτομα και οργανισμούς. Είναι διαθέσιμο για διάφορα λειτουργικά συστήματα, όπως Linux, Windows, macOS και άλλα. Αυτή η συμβατότητα μεταξύ πλατφορμών σας επιτρέπει να αναπτύσσετε τις εφαρμογές Ιστού σας σε διαφορετικά συστήματα. Έχει μακρά ιστορία και ισχυρή φήμη για σταθερότητα και αξιοπιστία. Έχει δοκιμαστεί εκτενώς και χρησιμοποιείται σε ποικίλα περιβάλλοντα, καθιστώντας το κατάλληλο για κρίσιμες εφαρμογές.

Το Apache προσφέρει ένα ευρύ φάσμα επιλογών διαμόρφωσης και λειτουργικών μονάδων που σας επιτρέπουν να προσαρμόσετε τη συμπεριφορά του για να ταιριάζει στις συγκεκριμένες ανάγκες σας. Μπορείτε να ενεργοποιήσετε ή να απενεργοποιήσετε τις λειτουργίες όπως απαιτείται, καθιστώντας τις προσαρμόσιμες σε διάφορες περιπτώσεις χρήσης. Η αρθρωτή αρχιτεκτονική του Apache σας επιτρέπει να επεκτείνετε τη λειτουργικότητά του προσθέτοντας διάφορα modules. Αυτό σας δίνει τη δυνατότητα να ενσωματώσετε λειτουργίες όπως έλεγχος ταυτότητας, επανεγγραφή URL, συμπίεση, προσωρινή αποθήκευση και πολλά άλλα. Παρέχει πολλές δυνατότητες και επιλογές ασφαλείας για την προστασία των εφαρμογών Ιστού σας. Υποστηρίζει κρυπτογράφηση SSL/TLS, σας επιτρέπει να ελέγχετε την πρόσβαση σε πόρους με βάση διευθύνσεις IP ή άλλα κριτήρια και προσφέρει εργαλεία για τον μετριασμό κοινών απειλών ασφαλείας. Με τις κατάλληλες ρυθμίσεις παραμέτρων και ενότητες, το Apache μπορεί να βελτιστοποιηθεί για καλύτερη απόδοση. Μπορείτε να ενεργοποιήσετε μηχανισμούς προσωρινής αποθήκευσης, συμπίεση και εξισορρόπηση φορτίου για να διασφαλίσετε την αποτελεσματική απόδοση των εφαρμογών Ιστού σας. Έχει μια μεγάλη και ενεργή κοινότητα χρηστών και προγραμματιστών. Αυτό σημαίνει ότι μπορείτε να βρείτε πολλούς πόρους, σεμινάρια και τεκμηρίωση που θα σας βοηθήσουν να ρυθμίσετε, να διαμορφώσετε και να αντιμετωπίσετε τα προβλήματα του διακομιστή σας. Επίσης μπορεί να λειτουργήσει ως διακομιστής αντίστροφου διακομιστή μεσολάβησης, διανέμοντας εισερχόμενα αιτήματα σε πολλούς διακομιστές υποστήριξης. Αυτό είναι χρήσιμο για την εξισορρόπηση φορτίου και τη βελτίωση της συνολικής διαθεσιμότητας και απόδοσης των εφαρμογών σας.

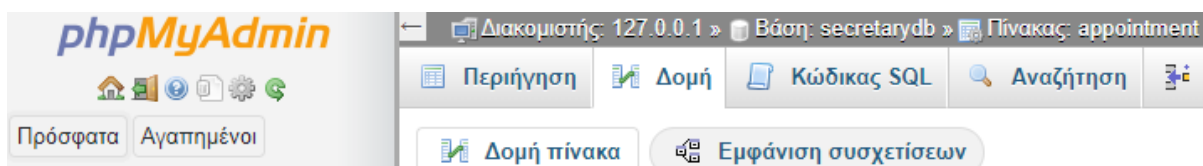
Ενώ το Apache είναι κατάλληλο για ιστοσελίδες μικρού έως μεσαίου μεγέθους, μπορεί επίσης να ρυθμιστεί για μεγαλύτερα, πιο περίπλοκα περιβάλλοντα. Χρησιμοποιώντας λειτουργίες όπως η εξισορρόπηση φορτίου και η ομαδοποίηση, μπορείτε να κλιμακώσετε τις εφαρμογές σας ανάλογα με τις ανάγκες. Ακόμη, πολλά δημοφιλή συστήματα διαχείρισης περιεχομένου, όπως το WordPress, το Drupal και το Joomla, έχουν σχεδιαστεί για να λειτουργούν καλά με το Apache. Η διαμόρφωση του διακομιστή μπορεί να προσαρμοστεί ώστε να ανταποκρίνεται στις συγκεκριμένες απαιτήσεις αυτών των συστημάτων.

Ενώ ο Apache έχει πολλά πλεονεκτήματα, αξίζει να σημειωθεί ότι υπάρχουν και άλλες διαθέσιμες επιλογές διακομιστή ιστού, όπως το Nginx. Η επιλογή του διακομιστή web εξαρτάται από τη συγκεκριμένη περίπτωση χρήσης, την εξοικείωση με το λογισμικό και τις δυνατότητες που χρειάζεστε.

3.4.2 phpmysqladmin

Το PHPMysqlAdmin είναι μια ευρέως χρησιμοποιούμενη διαδικτυακή εφαρμογή γραμμένη σε PHP, σχεδιασμένη για τη διαχείριση και τη διαχείριση βάσεων δεδομένων MySQL και MariaDB μέσω μιας φιλικής προς το χρήστη γραφικής διεπαφής. Επιτρέπει στους χρήστες να αλληλεπιδρούν με βάσεις δεδομένων χωρίς την ανάγκη περίπλοκων λειτουργιών γραμμής εντολών, καθιστώντας το ιδιαίτερα χρήσιμο για άτομα που μπορεί να μην είναι καλά γνώστες της διαχείρισης βάσεων δεδομένων ή της αναζήτησης SQL. [14]

Το PHPMysqlAdmin προσφέρει διάφορες δυνατότητες, όπως δημιουργία βάσης δεδομένων, τροποποίηση δομών πινάκων, εισαγωγή και επεξεργασία δεδομένων, εκτέλεση ερωτήματος SQL και έλεγχος πρόσβασης χρήστη. Παρουσιάζει μια οπτικά αισθητική διεπαφή που εμφανίζει τη δομή της βάσης δεδομένων, επιτρέποντας στους χρήστες να πλοηγούνται εύκολα σε πίνακες, προβολές, ευρετήρια και σχέσεις. Αυτό το εργαλείο διευκολύνει τη διαχείριση βάσεων δεδομένων, προσφέροντας μια βολική πλατφόρμα για την εκτέλεση εργασιών όπως δημιουργία αντιγράφων ασφαλείας δεδομένων, επαναφορά και διαχείριση αδειών χρήστη.



Εικόνα 3.6: PHPMysqlAdmin web panel

Απλοποιεί τη διαδικασία εργασίας με βάσεις δεδομένων παρέχοντας μια φιλική προς το χρήστη διεπαφή front-end, μειώνοντας την πολυπλοκότητα που σχετίζεται με τις παραδοσιακές μεθόδους διαχείρισης βάσεων δεδομένων.

Κεφάλαιο 4ο: Το σύστημα Ηλεκτρονικής Γραμματειακής Υποστήριξης

4.1 Εισαγωγή στο σύστημα

Ο χρήστης θα πρέπει να συνδεθεί στο σύστημα αφού πατήσει το κουμπί σύνδεση ή να πατήσει το κουμπί εγγραφή για να δημιουργήσει λογαριασμό, όπως φαίνεται στην παρακάτω Εικόνα 4.1.



Ηλεκτρονική Γραμματειακή Υποστήριξη

Κλείστε ή διαχειριστείτε τα ραντεβού σας online

υπο την υποστήριξη του

Διεθνές Πανεπιστημίου Ελλάδος

Σύνδεση

ή

Εγγραφή

Εικόνα 4.1: Το σύστημα Ηλεκτρονικής Γραμματειακής Υποστήριξης - Αρχική

Αρχικά ο χρήστης αφού πατήσει το κουμπί σύνδεση πρέπει να βάλει το email και τον κωδικό του όπως φαίνεται στην παρακάτω Εικόνα 4.2.

Ηλεκτρονική Γραμματειακή Υποστήριξη

Μπορείτε να συνδεθείτε ως πελάτης (ή γραμματέας ή ιδιοκτήτης-επαγγελματίας)

Email

Κωδικός

Σύνδεση

Εικόνα 4.2: Εισαγωγή των στοιχείων πρόσβασης

Ο χρήστης αφού πατήσει το κουμπί εγγραφή πρέπει να βάλει το email και τον κωδικό του όπως φαίνεται στην παρακάτω Εικόνα 4.3 για να λάβει email επιβεβαίωσης.

Εγγραφή

Ηλεκτρονική Γραμματειακή Υποστήριξη

Θα σας στείλουμε ένα email και εσείς θα πατήσετε το link που θα περιέχει για να επιβεβαιώσουμε την εγγραφή σας.
Στη συνέχεια μπορείτε να κάνετε σύνδεση.

Email

Κωδικός

Εγγραφή

Εικόνα 4.3: Εισαγωγή στοιχείων χρήστη για εγγραφή

4.1.1 Ιδιοκτήτης

Αρχικά θα περιγραφεί ο ρόλος του Ιδιοκτήτη όπου μπορεί να είναι ένας Γιατρός, ένας Δικηγόρος και άλλα παρόμοια επαγγέλματα που χρειάζονται σύστημα για ραντεβού.

Η πρώτη σελίδα που βλέπει ο ιδιοκτήτης παρουσιάζεται στην Εικόνα 4.4.

Είναι η σελίδα με τα ραντεβού σε συγκεκριμένες ημερομηνίες. Βλέπει ένα συγκεκριμένο εύρος ώρας που έχει οριστεί κεντρικά από το σύστημα.

Πατώντας το κουμπί ‘Τα ραντεβού μου’

Με διάφορα χρώματα απεικονίζεται το είδος του κάθε ραντεβού, το οποίο περιγράφεται στις παρακάτω Εικόνες.



Τα ραντεβού μου

21-08-2023

- 10:00
- 10:15
- 10:30
- 10:45
- 11:00
- 11:15
- 11:30
- 11:45
- 12:00
- 12:15
- 12:30
- 12:45

22-08-2023

- 10:00
- 10:15
- 10:30
- 10:45
- 11:00
- 11:15
- 11:30
- 11:45
- 12:00
- 12:15
- 12:30
- 12:45

23-08-2023

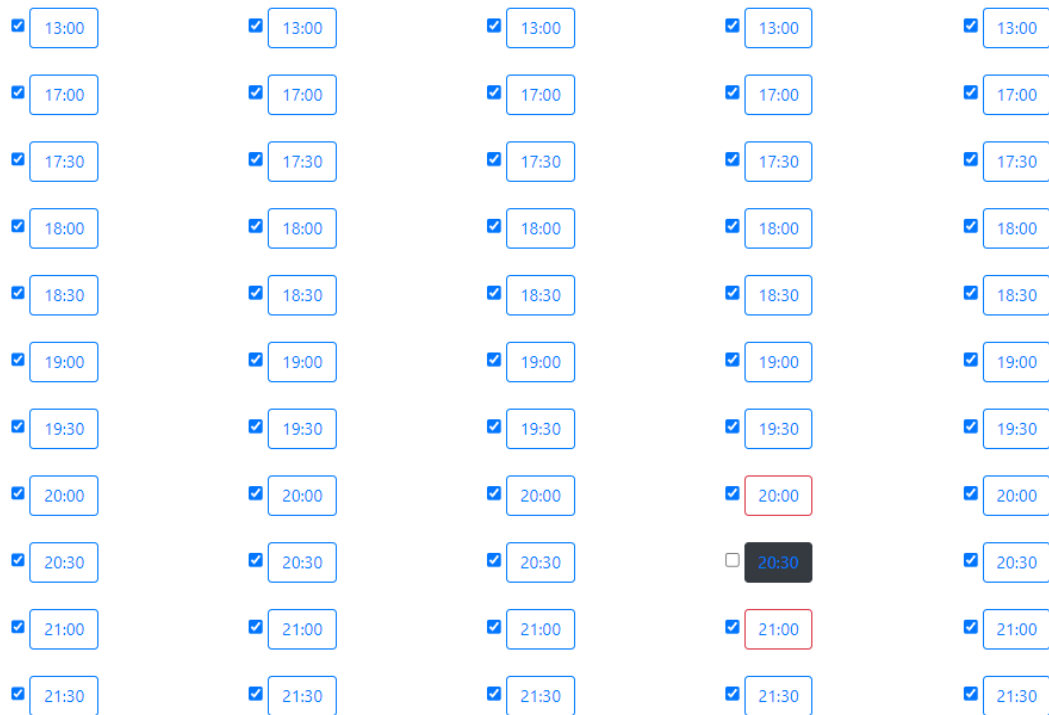
- 10:00
- 10:15
- 10:30
- 10:45
- 11:00
- 11:15
- 11:30
- 11:45
- 12:00
- 12:15
- 12:30
- 12:45

24-08-2023

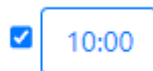
- 10:00
- 10:15
- 10:30
- 10:45
- 11:00
- 11:15
- 11:30
- 11:45
- 12:00
- 12:15
- 12:30
- 12:45

25-08-2023

- 10:00
- 10:15
- 10:30
- 10:45
- 11:00
- 11:15
- 11:30
- 11:45
- 12:00
- 12:15
- 12:30
- 12:45



Εικόνα 4.4: Κεντρική σελίδα με τα ραντεβού σε συγκεκριμένες ημερομηνίες



Εικόνα 4.5: Μπλε απεικόνιση – Ελεύθερο ραντεβού



Ηλεκτρονική
Γραμματειακή
Υποστήριξη

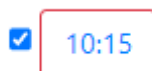
Έξοδος

[doctor2@gmail.com]

Μπορείτε να ορίσετε ένα νέο ραντεβού σε έναν χρήστη

Προσθήκη και Επιβεβαίωση

Εικόνα 4.6: Ελεύθερο ραντεβού για δέσμευση ενός ραντεβού από τον ιδιοκτήτη



Εικόνα 4.7: Κόκκινη απεικόνιση – Δεσμευμένο ραντεβού



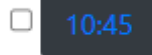
Το ραντεβού ανήκει στον χρήστη: [Πελατοεπώνυμο3 Πελατόνομα3](#)
Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.
Μπορείτε να το επιβεβαιώσετε

Επιβεβαίωση

Αν θέλετε μπορείτε να το ακυρώσετε

Ακύρωσε το ραντεβού

Εικόνα 4.8: Από Κόκκινη απεικόνιση – ο ιδιοκτήτης μπορεί να επιβεβαιώσει ή να ακυρώσει ένα δεσμευμένο ραντεβού από τον χρήστη



Εικόνα 4.9: Μαύρη απεικόνιση – Μη ενεργοποιημένο ραντεβού

Στην Εικόνα 4.9 με τη μαύρη απεικόνιση χαρακτηρίζεται το Μη ενεργοποιημένο ραντεβού το οποίο μπορεί να ενεργοποιηθεί ή όχι από το διπλανό στα αριστερά check box.



Εικόνα 4.10: Πράσινη απεικόνιση – Επιβεβαιωμένο ραντεβού – Ο χρήστης μπορεί να προσέλθει στο ραντεβού του



Το ραντεβού ανήκει στον χρήστη: **Πελατοεπώνυμο3 Πελατόνομα3**
Το ραντεβού έχει επιβεβαιωθεί.
Αν θέλετε μπορείτε να το ακυρώσετε

Ακύρωσε το ραντεβού

Εικόνα 4.11: Από Πράσινο κουμπί – λεπτομέρειες του ραντεβού

Όταν ο ιδιοκτήτης πατήσει το Πράσινο κουμπί βλέπει το τις λεπτομέρειες του ραντεβού και μπορεί να το ακυρώσει ή να δει ο χρήστης πόσα άλλα ραντεβού έχει με αυτόν τον ιδιοκτήτη, όπως παρουσιάζεται στην Εικόνα 4.12.



21-08-2023 10:15

Πελατοεπώνυμο3 Πελατόνομα3

Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.

Η ημερομηνία του ραντεβού είναι παλαιότερη.

23-08-2023 10:30

Πελατοεπώνυμο3 Πελατόνομα3

Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.

Η ημερομηνία του ραντεβού είναι παλαιότερη.

24-08-2023 21:00

Πελατοεπώνυμο3 Πελατόνομα3

Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.

Εικόνα 4.12: Πόσα ραντεβού έχει ένας χρήστης με ιδιοκτήτη



21-08-2023 10:15	Πελατοεπώνυμο3 Πελατόνομα3	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
22-08-2023 10:00	Πελατοεπώνυμο2 Πελατόνομα2	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
23-08-2023 10:30	Πελατοεπώνυμο3 Πελατόνομα3	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
24-08-2023 20:00	Πελατοεπώνυμο1 Πελατόνομα1	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
24-08-2023 21:00	Πελατοεπώνυμο3 Πελατόνομα3	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	

Εικόνα 4.13: Πόσα ραντεβού έχει ο ιδιοκτήτης

Ο ιδιοκτήτης μπορεί να δει τα ενεργοποιημένα ραντεβού επιβεβαιωμένα ή όχι του σε μια λίστα όπως φαίνεται στην Εικόνα 4.13.

4.1.2 Χρήστης

Μια από τις καλύτερες λειτουργίες του συστήματος είναι η σύνδεση με το προσωπικό site του κάθε ιδιοκτήτη.

Τρόπος Χρήσης από έναν άλλο ιστότοπο:

ο ιδιοκτήτης μπορεί να βάλει αυτό το link στην ιστοσελίδα του

<http://localhost/secretary/appointment/8A2B1F>

με το 8A2B1F να είναι το χαρακτηριστικό id του κάθε ιδιοκτήτη για να μπορεί να προβάλει στη συνέχεια τα ραντεβού αυτού του γιατρού η δικηγόρου κτλ.



Ραντεβού στον Γιατροεπώνυμο2 Γιατρόνομα2

Τα ραντεβού μου

24-08-2023

25-08-2023

10:00

10:15

10:30

10:45

11:00

11:15

11:30

21:00

11:45

21:30

12:00

12:15

12:30

12:45

13:00

Εικόνα 4.14: Ιστοσελίδα με τα ραντεβού του 8A2B1F που βλέπει ο χρήστης.

Αφού ο χρήστης εισήχθησε στην ιστοσελίδα στις 24-08-23, 20:40 τότε προβάλλονται τα μεταγενέστερα ραντεβού.

Όταν ο χρήστης (αφού είναι συνδεδεμένος) μπει στο συγκεκριμένο λινκ

<http://localhost/secretary/appointment/>

τότε προβάλλονται οι σελίδες που παρουσιάζονται στις Εικόνες 4.15 και 4.16.



Τα ραντεβού μου

Εικόνα 4.15: Εισαγωγή



Ηλεκτρονική Γραμματειακή Υποστήριξη

Έξοδος

[customer3@gmail.com]

21-08-2023	10:15	Γιατροπέωνμο2 Γιατρόνομα2	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
23-08-2023	10:30	Γιατροπέωνμο2 Γιατρόνομα2	Το ραντεβού έχει επιβεβαιωθεί.	Η ημερομηνία του ραντεβού είναι παλαιότερη.
24-08-2023	21:00	Γιατροπέωνμο2 Γιατρόνομα2	Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.	

Εικόνα 4.16: Εισαγωγή

Ο χρήστης μπορεί να πατήσει πάνω στο λινκ του γιατρού και να μπει στη λίστα με τα ραντεβού του, όπως φαίνεται στην Εικόνα 4.17.



Ηλεκτρονική Γραμματειακή Υποστήριξη

Έξοδος

[customer3@gmail.com]

Ραντεβού στον Γιατροπέωνμο2 Γιατρόνομα2

Τα ραντεβού μου

21-08-2023	22-08-2023	23-08-2023	24-08-2023	25-08-2023
10:00	10:00	10:00	10:45	10:00
10:15	10:15	10:15	11:00	10:15
10:30	10:30	10:30	11:15	10:30
11:30	10:45	10:45	11:30	10:45
11:45	11:00	11:00	11:45	11:00
12:00	11:15	11:15	12:00	11:15
12:15	11:30	11:30	12:15	11:30
12:30	11:45	11:45	12:30	11:45
12:45	12:00	12:00	12:45	12:00
13:00	12:15	12:15	13:00	12:15
17:00	12:30	12:30	17:00	12:30
17:30	12:45	12:45	17:30	12:45
18:00	13:00	13:00	18:00	13:00

Εικόνα 4.17: Εισαγωγή

Στην Εικόνα 4.17 παρουσιάζονται όλα τα ραντεβού του γιατρού αλλά για λόγους παρουσίασης επιλέχθηκε προγενέστερη ημερομηνία από τις 21-08-23 για να προβληθούν όλα.

21-08-2023

10:00

Εικόνα 4.18: Στη μπλε απεικόνιση – Ελεύθερο ραντεβού για τον χρήστη



Ηλεκτρονική Γραμματειακή Υποστήριξη

Έξοδος

[customer3@gmail.com]

Μπορείτε να κλείσετε το ραντεβού (Θα περιμένετε στη συνέχεια να επιβεβαιωθεί από τον υπεύθυνο)

Κλείσε ραντεβού

Εικόνα 4.19: Ο χρήστης μπορεί να ζητήσει ραντεβού

Όπως φαίνεται στην Εικόνα 4.19 ο χρήστης μπορεί να πατήσει σε ένα μπλε κουμπί-ελεύθερο και να ζητήσει ραντεβού. Στη συνέχεια θα περιμένει επιβεβαίωση από τον ιδιοκτήτη ή γραμματεία του.

10:15

Εικόνα 4.20: Μαύρη απεικόνιση – Μη επιβεβαιωμένο αλλά και μη ελεύθερο ραντεβου



Ηλεκτρονική Γραμματειακή Υποστήριξη

Έξοδος

[customer3@gmail.com]

Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.










Εικόνα 4.21: Αν ο χρήστης πατήσει στο μαύρο κουμπί θα προβληθεί μήνυμα μη επιβεβαίωσης του ραντεβού

10:00

Εικόνα 4.22: Κόκκινη απεικόνιση – δεσμευμένο ραντεβού

4.1.3 Γραμματεία

Ο/Η γραμματέας αποκτάει ρόλο ιδιοκτήτη όταν του δίνεται πρόσβαση από τη βάση δεδομένων, όπως φαίνεται στην Εικόνα 23.

				id	secretaryid	ownerid
<input type="checkbox"/>	 Επεξεργασία	 Αντιγραφή	 Διαγραφή	1	4	7
<input type="checkbox"/>	 Επεξεργασία	 Αντιγραφή	 Διαγραφή	2	4	8
<input type="checkbox"/>	 Επεξεργασία	 Αντιγραφή	 Διαγραφή	3	4	9

Εικόνα 4.23: Πίνακας αντιστοίχισης του id του ιδιοκτήτη με του γραμματέα.

Σε έναν ειδικό Πίνακα στη βάση - αντιστοίχισης του id του ιδιοκτήτη με του γραμματέα πραγματοποιείται η μεταξύ τους σύνδεση. Για παράδειγμα ο γραμματέας με το id 4 (secretary1) είναι γραμματέας των ιδιοκτητών με id 7,8,9 (doctor1,doctor2,lwyer1), όπως φαίνεται στην Εικόνα 4.24.



Ηλεκτρονική Γραμματειακή Υποστήριξη

Έξοδος

[secretary1@gmail.com]

Γιατροεπώνυμο1 Γιατρόνομα1

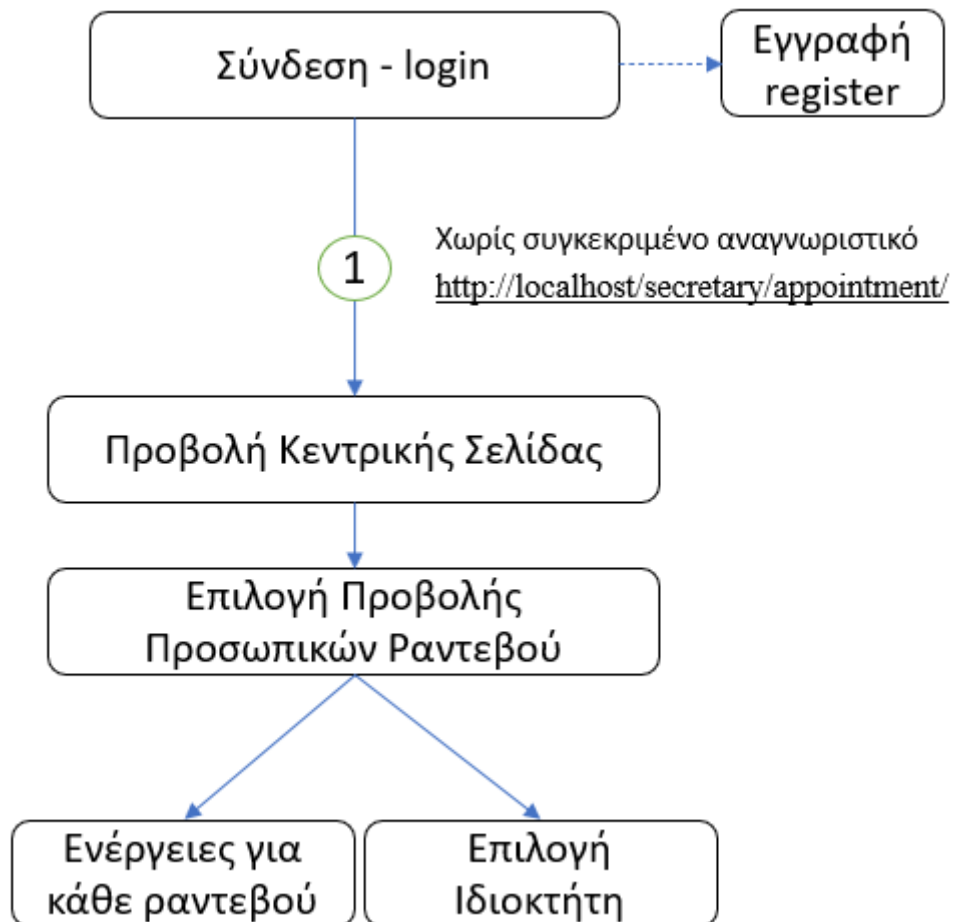
Γιατροεπώνυμο2 Γιατρόνομα2

Δικηγοροεπώνυμο1 Δικηγορόνομα1

Εικόνα 4.24: Ποιες ιδιοκτήτες έχει αναλάβει για εξυπηρέτηση ο γραμματέας με id 4

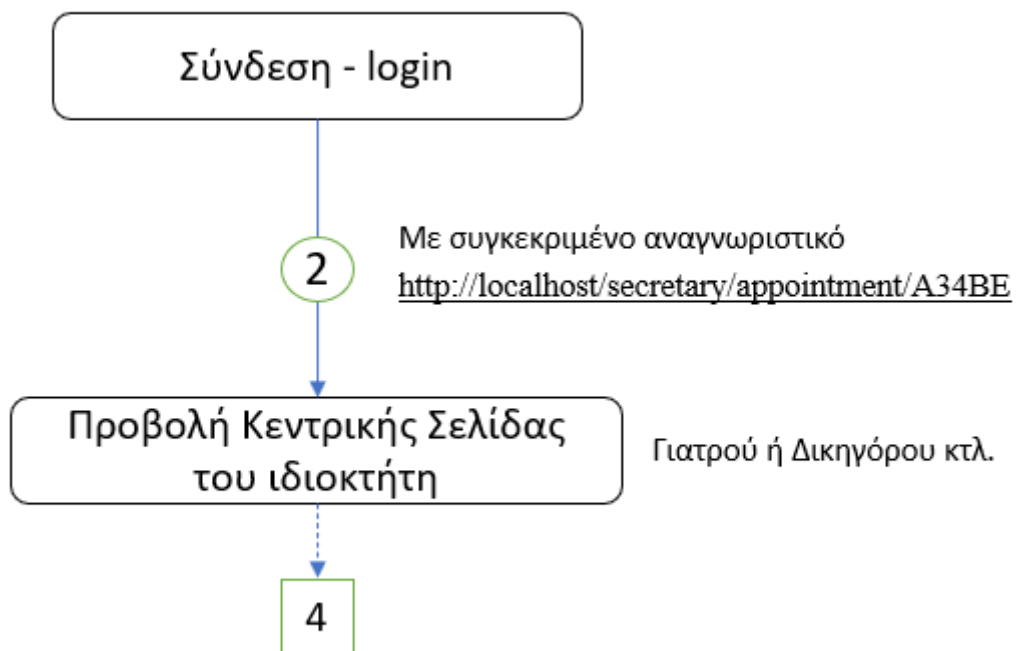
4.1.4 Διαγράμματα ροής

Σε αυτό το υποκεφάλαιο παρουσιάζονται τα διαγράμματα ροής για τον χρήστη και τον ιδιοκτήτη.



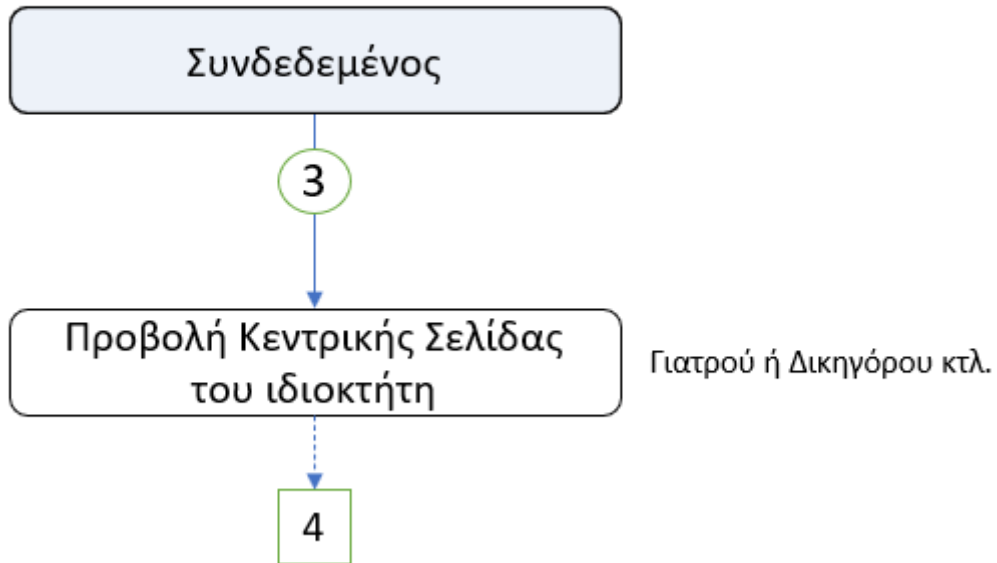
Εικόνα 4.25: Διάγραμμα ροής χρήστη: για την είσοδο, εγγραφή, προβολή ραντεβού και ενέργειες.

Χωρίς συγκεκριμένο αναγνωριστικό



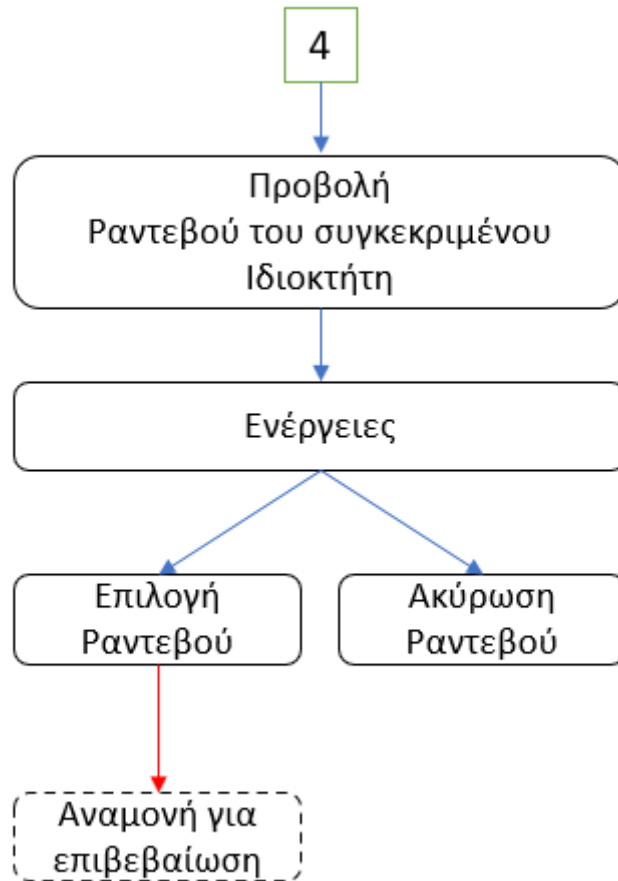
Εικόνα 4.26: Διάγραμμα ροής χρήστη: για την είσοδο και προβολής κεντρικής σελίδας του ιδιοκτήτη με χρήση συγκεκριμένου αναγνωριστικού

Με συγκεκριμένο αναγνωριστικό
<http://localhost/secretary/appointment/A34BE>

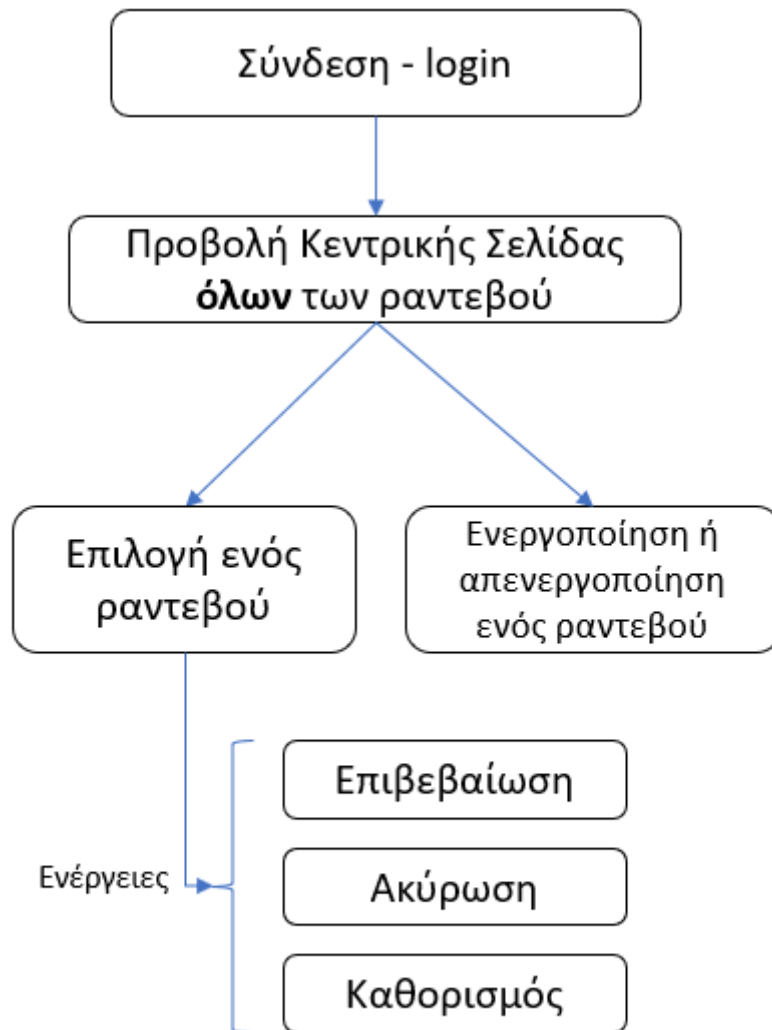


Εικόνα 4.27: Διάγραμμα ροής χρήστη: για την είσοδο και προβολής κεντρικής σελίδας του ιδιοκτήτη με χρήση συγκεκριμένου αναγνωριστικού

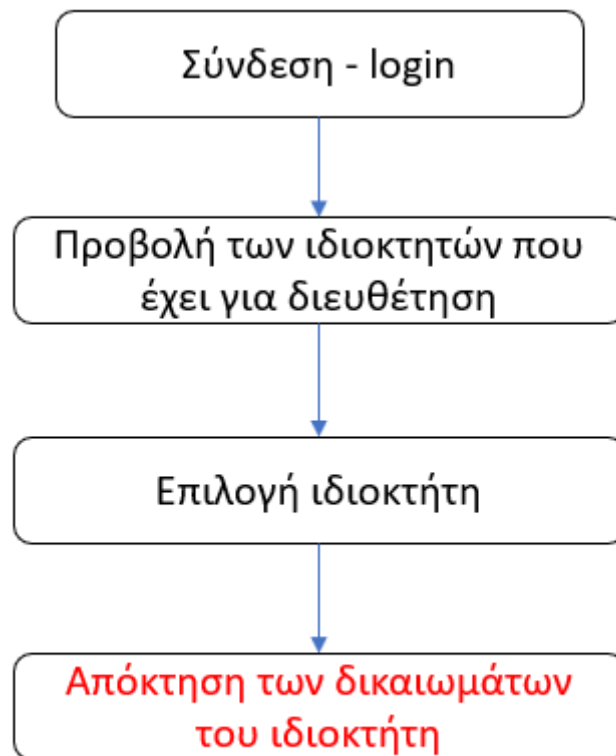
Και οι δυο ροές που παρουσιάζονται στις Εικόνες 4.26 και 4.27 καταλήγουν στο Σημείο 4 που φαίνεται στην Εικόνα 4.28.



Εικόνα 4.28: Διάγραμμα ροής για προβολή των ραντεβού του συγκεκριμένου ιδιοκτήτη. Ενέργειες για τα ραντεβού.



Εικόνα 4.29: Διάγραμμα ροής για τον ιδιοκτήτη. Προβολή των ραντεβού του συγκεκριμένου ιδιοκτήτη. Ενέργειες για τα ραντεβού.



Εικόνα 4.30: Διάγραμμα ροής την/τον γραμματέα. Προβολή των ιδιοκτητών, επιλογή.

Στην Εικόνα 30 παρουσιάζεται το διάγραμμα ροής την/τον γραμματέα και η προβολή των ιδιοκτητών που έχει αναλάβει να προσφέρει τις υπηρεσίες του.

4.2 Η βάση δεδομένων

Πίνακας	Ενέργεια	Εγγραφές	Τύπος	Σύνθεση	Μέγεθος	Περίσσεια
appointment	Περιήγηση, Δομή, Αναζήτηση, Προσθήκη, Αδειασμα, Διαγραφή	115	InnoDB	utf8mb4_general_ci	16,0 KB	-
secowner	Περιήγηση, Δομή, Αναζήτηση, Προσθήκη, Αδειασμα, Διαγραφή	2	InnoDB	utf8mb4_general_ci	32,0 KB	-
user	Περιήγηση, Δομή, Αναζήτηση, Προσθήκη, Αδειασμα, Διαγραφή	10	InnoDB	utf8mb4_general_ci	32,0 KB	-
3 πίνακες	Σύνολο	127	InnoDB	utf8mb4_general_ci	80,0 KB	0 B

Εικόνα 4.31: Η βάση δεδομένων με τους πίνακες που χρησιμοποιήθηκαν

Η βάση κατασκευάστηκε με MySQL MariaDB και στο Εικόνα 4.24 παρουσιάζεται το πλάνο με τους πίνακες που χρησιμοποιήθηκαν στο έργο.

Στις επόμενες Εικόνες παρουσιάζονται οι πίνακες, τα πεδία και τα περιεχόμενα που περιλαμβάνει ο καθένας.

Οι πίνακες όλοι διαθέτουν τα πεδία `id` που είναι το κεντρικό κλειδί και είναι αύξων αριθμός,

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα
<input type="checkbox"/>	1 <code>id</code>	<code>int(10)</code>		UNSIGNED	Όχι	Καμία		AUTO_INCREMENT
<input type="checkbox"/>	2 <code>guid</code>	<code>varchar(10)</code>	<code>utf8mb4_general_ci</code>		Όχι	A3B5E		
<input type="checkbox"/>	3 <code>active</code>	<code>tinyint(3)</code>		UNSIGNED	Όχι	1		
<input type="checkbox"/>	4 <code>kind</code>	<code>smallint(5)</code>		UNSIGNED	Όχι	0	0:customer,1:secr,2:owner	
<input type="checkbox"/>	5 <code>email</code>	<code>varchar(100)</code>	<code>utf8mb4_general_ci</code>		Όχι	Καμία		
<input type="checkbox"/>	6 <code>password</code>	<code>varchar(100)</code>	<code>utf8mb4_general_ci</code>		Όχι	Καμία		
<input type="checkbox"/>	7 <code>firstname</code>	<code>varchar(100)</code>	<code>utf8mb4_general_ci</code>		Όχι	Καμία		
<input type="checkbox"/>	8 <code>lastname</code>	<code>varchar(100)</code>	<code>utf8mb4_general_ci</code>		Όχι	Καμία		
<input type="checkbox"/>	9 <code>city</code>	<code>varchar(30)</code>	<code>utf8mb4_general_ci</code>		Ναι	NULL		
<input type="checkbox"/>	10 <code>tel</code>	<code>varchar(20)</code>	<code>utf8mb4_general_ci</code>		Ναι	NULL		
<input type="checkbox"/>	11 <code>lang</code>	<code>varchar(2)</code>	<code>utf8mb4_general_ci</code>		Όχι	el		
<input type="checkbox"/>	12 <code>approved</code>	<code>tinyint(3)</code>		UNSIGNED	Όχι	0		
<input type="checkbox"/>	13 <code>created_at</code>	<code>timestamp</code>			Όχι	<code>current_timestamp()</code>		
<input type="checkbox"/>	14 <code>updated_at</code>	<code>timestamp</code>			Όχι	<code>current_timestamp()</code>		ON UPDATE CURRENT_TIMESTAMP()

Εικόνα 4.32: Η δομή του πίνακα `user` – οι χρήστες

	<code>id</code>	<code>guid</code>	<code>active</code>	<code>kind</code>	<code>email</code>	<code>password</code>	<code>firstname</code>	<code>lastname</code>	<code>city</code>	<code>tel</code>	<code>lang</code>	<code>approved</code>	<code>created_at</code>	<code>updated_at</code>
<input type="checkbox"/>	1	1A3B5E	1	0	customer1@gmail.com	1234	Πελατόνομα1	Πελατοεπίπνομα1	Θεσσαλονίκη	2310123456	el	1	2023-08-13 12:30:12	2023-08-13 14:06:10
<input type="checkbox"/>	2	2A4B5C	1	0	customer2@gmail.com	1234	Πελατόνομα2	Πελατοεπίπνομα2	Θεσσαλονίκη	2310123457	el	1	2023-08-13 12:31:04	2023-08-13 14:06:22
<input type="checkbox"/>	3	3B3C3E	1	0	customer3@gmail.com	1234	Πελατόνομα3	Πελατοεπίπνομα3	Θεσσαλονίκη	2310123459	el	1	2023-08-13 12:31:31	2023-08-13 14:06:37
<input type="checkbox"/>	4	4D345B	1	1	secretary1@gmail.com	1234	Γραμματείανομα1	Γραμματείαπύνομα1	Θεσσαλονίκη	2310876543	el	1	2023-08-13 12:32:46	2023-08-14 13:40:42
<input type="checkbox"/>	5	53B5EA	1	1	secretary2@gmail.com	1234	Γραμματείανομα2	Γραμματείαπύνομα2	Θεσσαλονίκη	2310876544	el	1	2023-08-13 12:33:02	2023-08-14 13:40:45
<input type="checkbox"/>	6	643B5E	1	1	secretary3@gmail.com	1234	Γραμματείανομα3	Γραμματείαπύνομα3	Θεσσαλονίκη	2310876545	el	1	2023-08-13 12:33:21	2023-08-14 13:40:48
<input type="checkbox"/>	7	7D7B5D	1	2	doctor1@gmail.com	1234	Γιατρόνομα1	Γιατροεπίπνομα1	Θεσσαλονίκη	2310334455	el	1	2023-08-13 12:34:28	2023-08-14 13:40:27
<input type="checkbox"/>	8	8A2B1F	1	2	doctor2@gmail.com	1234	Γιατρόνομα2	Γιατροεπίπνομα2	Θεσσαλονίκη	2310334456	el	1	2023-08-13 12:34:43	2023-08-14 13:40:32
<input type="checkbox"/>	9	9B6D4B	1	2	hyer1@gmail.com	1234	Διηγορόνομα1	Διηγοροεπίπνομα1	Θεσσαλονίκη	2310334457	el	1	2023-08-13 12:36:21	2023-08-14 13:40:36
<input type="checkbox"/>	10	10AA32B	1	2	hyer2@gmail.com	1234	Διηγορόνομα2	Διηγοροεπίπνομα2	Θεσσαλονίκη	2310334458	el	1	2023-08-13 12:36:36	2023-08-14 13:40:39

Εικόνα 4.33: Το περιεχόμενο των χρηστών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	1 <code>id</code>	<code>int(11)</code>			Όχι	Καμία		AUTO_INCREMENT	Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	2 <code>active</code>	<code>tinyint(4)</code>			Όχι	1			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	3 <code>enabled</code>	<code>tinyint(4)</code>			Όχι	1			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	4 <code>booked</code>	<code>int(4)</code>		UNSIGNED	Όχι	0			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	5 <code>confirmed</code>	<code>int(11)</code>			Όχι	0			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	6 <code>ownerid</code>	<code>int(11)</code>			Όχι	Καμία			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	7 <code>meraora</code>	<code>datetime</code>			Όχι	<code>current_timestamp()</code>			Αλλαγή Διαγραφή Περισσότερα

Εικόνα 4.34: Η δομή για τον πίνακα `appointment`

<input type="checkbox"/>				id	active	enabled	booked	confirmed	ownerid	meraora
<input type="checkbox"/>				1	1	1	0	0	8	2023-08-21 10:00:00
<input type="checkbox"/>				2	1	1	3	0	8	2023-08-21 10:15:00
<input type="checkbox"/>				3	1	1	0	0	8	2023-08-21 10:30:00
<input type="checkbox"/>				4	1	0	0	0	8	2023-08-21 10:45:00
<input type="checkbox"/>				5	1	0	0	0	8	2023-08-21 11:00:00
<input type="checkbox"/>				6	1	0	0	0	8	2023-08-21 11:15:00
<input type="checkbox"/>				7	1	1	0	0	8	2023-08-21 11:30:00
<input type="checkbox"/>				8	1	1	0	0	8	2023-08-21 11:45:00
<input type="checkbox"/>				9	1	1	0	0	8	2023-08-21 12:00:00
<input type="checkbox"/>				10	1	1	0	0	8	2023-08-21 12:15:00
<input type="checkbox"/>				11	1	1	0	0	8	2023-08-21 12:30:00
<input type="checkbox"/>				12	1	1	0	0	8	2023-08-21 12:45:00
<input type="checkbox"/>				13	1	1	0	0	8	2023-08-21 13:00:00
<input type="checkbox"/>				14	1	1	0	0	8	2023-08-21 17:00:00
<input type="checkbox"/>				15	1	1	0	0	8	2023-08-21 17:30:00
<input type="checkbox"/>				16	1	1	0	0	8	2023-08-21 18:00:00
<input type="checkbox"/>				17	1	1	0	0	8	2023-08-21 18:30:00
<input type="checkbox"/>				18	1	1	0	0	8	2023-08-21 19:00:00
<input type="checkbox"/>				19	1	1	0	0	8	2023-08-21 19:30:00
<input type="checkbox"/>				20	1	1	0	0	8	2023-08-21 20:00:00
<input type="checkbox"/>				21	1	1	0	0	8	2023-08-21 20:30:00
<input type="checkbox"/>				22	1	1	0	0	8	2023-08-21 21:00:00
<input type="checkbox"/>				23	1	1	0	0	8	2023-08-21 21:30:00
<input type="checkbox"/>				24	1	1	2	0	8	2023-08-22 10:00:00
<input type="checkbox"/>				25	1	1	0	0	8	2023-08-22 10:15:00
<input type="checkbox"/>				26	1	1	0	0	8	2023-08-22 10:30:00
<input type="checkbox"/>				27	1	1	0	0	8	2023-08-22 10:45:00

Εικόνα 4.35: Το περιεχόμενο του πίνακα appointment – περιέχει τα ραντεβού

Active: αν είναι ενεργό

Enabled: αν είναι ενεργοποιημένο από τον ιδιοκτήτη (checkbox)

Booked: 0: αν είναι διαθέσιμο, αλλιώς έχει το id του user που το δέσμευσε

Confirmed: 0 όχι, 1 ναι

Ownerid: σε ποιον ιδιοκτήτη ανήκει

Meraora: για ποια ώρα και μέρα αφορά

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία		AUTO_INCREMENT	Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	2 secretaryid	int(10)		UNSIGNED	Όχι	Καμία			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	3 ownerid	int(10)		UNSIGNED	Όχι	Καμία			Αλλαγή Διαγραφή Περισσότερα

Εικόνα 4.36: Πίνακας secowner με αντιστοίχιση του ιδιοκτήτη με τον/την γραμματέα.

		id	secretaryid	ownerid
<input type="checkbox"/>	Επεξεργασία Αντιγραφή Διαγραφή	1	5	7
<input type="checkbox"/>	Επεξεργασία Αντιγραφή Διαγραφή	2	5	8

Εικόνα 4.37: Περιεχόμενα-δείγμα του πίνακα secowner

Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης

Σε αυτήν την εργασία πραγματοποιήθηκε η μελέτη, η σχεδίαση και η υλοποίηση ενός συστήματος ηλεκτρονικής γραμματειακής υποστήριξης για ραντεβού για ειδικότητες όπως ο γιατρός και δικηγόρος.

Μελετήθηκαν παρόμοια συστήματα εκτός και εντός του Ελληνικού χώρου. Επίσης, παρουσιάστηκαν τα εργαλεία που απαιτήθηκαν για την υλοποίηση του έργου.

Το σύστημα που υλοποιήθηκε διαθέτει διαχειριστικό για τον ιδιοκτήτη (γιατρός και δικηγόρος) και για τη γραμματεία του. Διαθέτει ιστοχώρο για να διευθετεί ο πελάτης τα ραντεβού του, να βλέπει τα διαθέσιμα και υπόλοιπα ραντεβού του. Ένα από τα σημαντικότερα χαρακτηριστικά είναι η δυνατότητα σύνδεσης του συστήματος με το website του ιδιοκτήτη-γιατρού. Το σύστημα έχει υλοποιηθεί με το Laravel, PHP και Mysql δίνοντας τη δυνατότητα μέσω αυτής της υλοποίησης την εκμάθηση ειδικών αλγοριθμικών τεχνικών, τεχνολογιών και frameworks και βελτίωση προγραμματιστικών δεξιοτήτων για web εφαρμογές.

Μια από τις προτάσεις βελτίωσης είναι η προσθήκη email στους χρήστες που εμπλέκονται σε μια διευθέτηση ενός ραντεβού, όπως είναι η επιβεβαίωση ή η ακύρωση. Επίσης θα μπορούσε να υπάρχει επιβεβαίωση εγγραφής ενός χρήστη.

Μια άλλη πρόταση βελτίωσης είναι η καλύτερη διευθέτηση των ημερομηνιών μέσω ενός καλύτερου σε παρουσιαστικού και λειτουργιών calendar.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <https://www.healthit.gov/faq/what-electronic-health-record-ehr>
- [2] <https://www.php.net/>
- [3] <https://www.valuecoders.com/blog/app-development/net-vs-php-vs-java-vs-python-which-language-is-better/>
- [4] <https://laravel.com/>
- [5] <https://symfony.com/>
- [6] <https://codeigniter.com/>
- [7] <https://www.yiiframework.com/>
- [8] <https://framework.zend.com/>
- [9] <https://phalcon.io/en-us>
- [10] <https://www.mysql.com/>
- [11] <https://mariadb.org/>
- [12] <https://www.apachefriends.org/download.html>
- [13] <https://httpd.apache.org/>
- [14] <https://www.phpmyadmin.net/>

ΠΑΡΑΡΤΗΜΑ Α

Στο παράρτημα αυτό αναφέρονται τα βασικά κομμάτια του κώδικα που χρησιμοποιήθηκε.

MainController.php

```
namespace App\Http\Controllers;

use Illuminate\Support\Facades\File;
use Illuminate\Http\Request;
use Illuminate\Http\UploadedFile;

use Validator;
use Illuminate\Support\Facades\Log;
use Illuminate\Support\Facades\DB;

class MainController extends Controller
{
    function myowners(Request $request)
    {
        if (session('loggedin') == 0)
            return redirect('/');

        $user = session('loggeduser');
        $user_id = $user->id;
        $user_kind = $user->kind;

        if($user_kind == 1) //secretary
        {
            $owners = DB::select('select * from secowner where secretaryid=?', [$user_id]);
            if($owners != null)
            {
                for ($i = 0; $i < count($owners); $i++) {
                    $user = DB::select('select * from user where active = 1 and id=?', [$owners[$i]->ownerid]);
                    $owners[$i]->user = $user[0];
                }

                return view('myowners', ['owners' => $owners, 'user_kind' => $user_kind, 'user_id' => $user_id]);
            }
        }
        else{
            return view('myowners', ['owners' => null, 'user_kind' => $user_kind, 'user_id' => $user_id]);
        }
    }
}
```

```

}

}

function appointment(Request $request, $id = null)
{
    //Το καλούμε απο το site του γιατρού κτλ
    //Sid -> είναι το id του γιατρού κτλ
    //http://localhost/secretary/appoinment?id=8A2B1F

    //Αυτός ο πελάτης θέλει να διευθετήσει ραντεβού με τον γιατρό: 8A2B1F

    if (session('loggedin') == 0)
        return redirect('/'); //TODO: να αλλά πρέπει να θυμάται το 8A2B1F

    $user = session('loggeduser');
    $user_id = $user->id;
    $user_kind = $user->kind;
    //Log::info(json_encode($user));

    //Αν έχει id valid να δείξω τα ραντεβου-ώρες του γιατρού κτλ
    //Log::info($id);
    $owner = null;
    $query = "select * from user where kind = 2 and active = 1 and approved = 1 and guid="" . $id . """;
    $owner = DB::select($query);
    //Log::info($owner);
    //Log::info(json_encode($owner));

    if ($user_kind == 2 && $id == null) {
        //Log::info(json_encode($user_id));
        $owner = DB::select('select * from user where active = 1 and id=?', [$user_id]);
    }

    if ($owner != null) {

        //Να δείξω διαθέσιμα ραντεβού για αυτόν τον γιατρό
        if ($user_kind == 0) {
            $appointment = DB::select('select * from appointment where active = 1 and enabled = 1 and ownerid=? order by meraora ASC',
[$owner[0]->id]);
        } else {

```

```

    $appointment = DB::select('select * from appointment where active = 1 and ownerid=? order by meraora ASC', [$owner[0]->id]);
}

$lar = array();
$sk = 0;
$sp = 0;

for ($i = 0; $i < count($appointment); $i++) {
    $meraora = new \DateTime($appointment[$i]->meraora);

    $date = $meraora->format('d-m-Y');
    $time = $meraora->format('H:i');

    $appointment[$i]->date = $date;
    $appointment[$i]->time = $time;

    $now = new \DateTime("20-08-2023 21:00");
    //$now = new \DateTime();
    if ($now > $meraora) $appointment[$i]->expired = 1;
    else $appointment[$i]->expired = 0;

    if ($i == 0) $dateold = $date;

    if ($date == $dateold) {
        $lar[$sk][$sp++] = $appointment[$i];
    } else {
        $sk++;
        $sp = 0;
        $i--;
    }
    $dateold = $date;
}

//Log::info(json_encode($lar[0]));
//Log::info(json_encode($lar[1]));

//Log::info(json_encode($lar));

return view('appointment', ['ar' => $lar, 'user_kind' => $user_kind, 'user_id' => $user_id, 'now' => $now, 'owner' => $owner[0]]);
} else {
    //Δεν αντιστοιχεί σε γιατρό κτλ και θα δείξω μόνο προηγούμενα ραντεβού - όλα

```

```

if ($user_kind == 0) //client
{
    return view('appointment', ['user_kind' => $user_kind]);
}
}

function myappointments(Request $request, $id = null)
{
    if (session('loggedin') == 0)
        return redirect('/'); //TODO: ναι αλλά πρέπει να θυμάται το 8A2B1F

    $user = session('loggeduser');
    $user_id = $user->id;
    $user_kind = $user->kind;

    $siamowner_getlist = false;
    if ($user_kind == 2) //owner
    {
        if ($id != null) {
            $userclient = DB::select('select * from user where active = 1 and id=?', [$id]);
            if ($userclient != null) {
                //Είμαι owner και θα δω μόνο του client τα ραντεβού που σχετίζονται με μένα
                $ownerid = $user_id;
                $clientid = $userclient[0]->id;
                $siamowner_getlist = true;
            }
        }
        if ($id == null) {
            $ownerid = $user_id;
            $siamowner_getlist = true;
        }
    }

    if ($user_kind == 0 || $siamowner_getlist == true) //client
    {
        if ($siamowner_getlist == true)
        {
            //Είμαι owner και θα δω μόνο του client τα ραντεβού που σχετίζονται με μένα
            //$appointments = DB::select('select * from appointment where active = 1 and enabled = 1 and booked=? and ownerid=? order by
            meraora ASC', [$user_id,$user->id]);

```

```

if($id != null)
{
    $appointments = DB::select('select * from appointment where active = 1 and enabled = 1 and booked=? and ownerid=? order by
meraora ASC', [$clientid,$ownerid]);
}
else
{
    //Owner: Όλα τα ραντεβού μου
    $appointments = DB::select('select * from appointment where active = 1 and enabled = 1 and booked!=0 and ownerid=? order by
meraora ASC', [$ownerid]);
    //Log::info(json_encode($appointments));
}
}
else
{
    $appointments = DB::select('select * from appointment where active = 1 and enabled = 1 and booked=? order by meraora ASC',
[$user_id]);
}
//Log::info(json_encode($appointments));

for ($i = 0; $i < count($appointments); $i++) {

    if ($iamowner_getlist == true)
    {
        $user = DB::select('select * from user where active = 1 and id=?', [$appointments[$i]->booked]);
    }
    else{
        $user = DB::select('select * from user where kind = 2 and active = 1 and id=?', [$appointments[$i]->ownerid]);
    }

    if ($user[0] != null) {
        $appointments[$i]->user = $user[0];

        $meraora = new \DateTime($appointments[$i]->meraora);

        $date = $meraora->format('d-m-Y');
        $time = $meraora->format('H:i');

        $appointments[$i]->date = $date;
        $appointments[$i]->time = $time;

        $now = new \DateTime();

```

```

        if ($now > $meraora) $appointments[$i]->expired = 1;
        else $appointments[$i]->expired = 0;
    } else {
        array_splice($appointments, $i, 1);
    }
}

return view('myappointments', ['appointments' => $appointments, 'user_kind' => $user_kind, 'user_id' =>
$user_id, 'iamowner_getlist' => $iamowner_getlist]);
}
}

function bookappointment(Request $request, $id = null)
{
    //Το καλούμε απο το site του appointment
    // $id -> είναι το id του ραντεβού
    //http://localhost/secretary/bookappointment?id=23

    if (session('loggedin') == 0)
        return redirect('/');

    $user = session('loggeduser');
    $user_id = $user->id;
    $user_kind = $user->kind;

    //Ελεγχος αν αυτό το ραντεβού
    //active=1, enabled=1, booked != 0
    //Αν booked != 0 τότε να ελεγχθεί αν το booked είναι ίδιο με το user_id του.
    //Αν ναι τότε είναι δικό του ραντεβού. Και μπορεί να το ακυρώσει μόνο αν το έχει εγκρίνει ή όχι ο γιατρός.

    //Αν όχι μπορεί να πατήσει το κουμπί για Κλείσιμο Ραντεβού.
    // Μπορεί να κλείσει ενα ραντεβού μόνο αν δεν έχει κλεισμένο σε μεταγενέστερο χρόνο ενα άλλο στον ίδιο γιατρό(κτλ)

    //Ο γιατρός βλέπει τα ραντεβού και λαμβάνει μήνυμα για να πατήσει επιβεβαίωση του ραντεβού. Αλλιώς το ραντεβού δεν ισχύει.
    //Μπορεί να το ξεβεβαιώσει.

    // if($id != null)
    $appointment = DB::select('select * from appointment where active = 1 and enabled = 1 and id=?', [$id]);

    if ($appointment != null) {
        if ($user_kind != 220) //client

```

```

{
    $userclient = null;
    if ($appointment[0]->booked != 0)
        $userclient = DB::select('select * from user where active = 1 and id=?', [$appointment[0]->booked]);

    if ($userclient == null)
        $userclient[0] = null;

    //Log::info(json_encode($appointment[0]));
    //Log::info(json_encode($userclient));

    return view('bookappointment', ['appointment' => $appointment[0], 'user_kind' => $user_kind, 'user_id' => $user_id, 'userclient' =>
$userclient[0]]);
}
} else {
    //Δεν αντιστοιχεί σε ραντεβού
    if ($user_kind == 0) //client
    {
        return view('bookappointment', ['user_kind' => $user_kind]);
    }
}
}

function actionsforappointment(Request $request)
{
    Log::info("actionsforappointment");
    if (session('loggedin') == 0)
        return redirect('/');

    $user = session('loggeduser');
    $user_id = $user->id;
    $user_kind = $user->kind;

    $sch = $request->input('ch');
    $sid = $request->input('id');

    if ($sch == 1) //Ζητάω ραντεβού
    {
        if ($user_kind == 0) {
            $appointment = DB::select('select * from appointment where active = 1 and enabled = 1 and id=? and booked!=? and booked=0',
[$sid, $user_id]);

```

```

//Log::info(json_encode($appointment));
if (count($appointment) > 0) {
    $queryState = DB::table('appointment')
        ->where('active', 1)
        ->where('id', $id)
        ->update(['booked' => $user_id]);

    if ($queryState) {
        // the query succeed
        $msg = 'Αποθηκεύτηκε';
        return response()->json(array('msg' => $msg, 'status' => true));
    } else {
        // the query failed
        $msg = 'Κάποιο πρόβλημα παρουσιάστηκε - Ξαναπροσπαθήστε!';
        return response()->json(array('msg' => $msg, 'status' => false));
    }
}
}

if ($sch == 2) //Ακυρώνω ραντεβού
{
    if ($user_kind == 0) {
        $appointment = DB::select('select * from appointment where active = 1 and enabled = 1 and id=? and booked=?', [$id, $user_id]);
        //Log::info(json_encode($appointment));
        if (count($appointment) > 0) {
            $queryState = DB::table('appointment')
                ->where('active', 1)
                ->where('id', $id)
                ->update(['booked' => 0]);

            if ($queryState) {
                // the query succeed
                $msg = 'Ακυρώθηκε';
                return response()->json(array('msg' => $msg, 'status' => true));
            } else {
                // the query failed
                $msg = 'Κάποιο πρόβλημα παρουσιάστηκε - Ξαναπροσπαθήστε!';
                return response()->json(array('msg' => $msg, 'status' => false));
            }
        }
    }
}
}

```

```
}  
}  
}
```

myappointments.blade.php

```
<!DOCTYPE html>  
<html>  
  
<head>  
  <title>My Appointments</title>  
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>  
  <link rel="canonical" href="https://getbootstrap.com/docs/4.0/examples/album/">  
  <!-- Bootstrap core CSS -->  
  <link href="https://getbootstrap.com/docs/4.0/dist/css/bootstrap.min.css" rel="stylesheet">  
  <link href="https://getbootstrap.com/docs/4.0/examples/album/album.css" rel="stylesheet">  
</head>  
  
<body>  
  <div class="container">  
  
    @include('header')  
  
    @if($user_kind!=220)  
  
      @if(isset($appointments))  
        <br/>  
        @for($i = 0; $i < count($appointments); $i++)  
          @if($appointments[$i]->confirmed == 1 && $appointments[$i]->booked == $user_id)  
  
            <span style="margin-left: 10px;"> {{ $appointments[$i]->date }} </span>  
            <span style="margin-left: 20px;"> {{ $appointments[$i]->time }} </span>  
            <span style="margin-left: 20px; font-weight: bold;"><a href="{ { config('app.url') } /appointment' . $appointments[$i]->user-  
>guid } }" >{{ $appointments[$i]->user->lastname.' ' . $appointments[$i]->user->firstname }} </a> </span>  
            <span style="margin-left: 20px;"> Το ραντεβού έχει επιβεβαιωθεί.</span>  
            @if($appointments[$i]->expired==0)  
              <span style="margin-left: 0px;">Αν θέλετε μπορείτε να το ακυρώσετε.</span>  
              <span style="margin-left: 20px;"><a style="font-size: 10px;" onclick="afa(2,'{ { $appointments[$i]->id } }'" class="btn btn-  
outline-danger my-2">Ακύρωσε το ραντεβού</a></span>  
            @endif  
  
            @if($appointments[$i]->expired==1)
```

```

        <span style="margin-left: 20px; font-size:10px; font-color:rgb(85, 11, 11);">Η ημερομηνία του ραντεβού είναι
παλαιότερη.</span>
        @endif
        <br/>
        @endif

        @if($appointments[$i]->confirmed == 0 && $appointments[$i]->booked == $user_id)
            <span style="margin-left: 10px;"> {{ $appointments[$i]->date }} </span>
            <span style="margin-left: 20px;"> {{ $appointments[$i]->time }} </span>
            <span style="margin-left: 20px; font-weight: bold;"> <a href="{{ config('app.url') . '/appointment/' . $appointments[$i]->user-
>guid }}">{{ $appointments[$i]->user->lastname.' '.$appointments[$i]->user->firstname }} </a> </span>
            <span style="margin-left: 20px;"> Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα. </span>

            @if($appointments[$i]->expired==1)
                <span style="margin-left: 20px; font-size:10px; font-color:rgb(85, 11, 11);">Η ημερομηνία του ραντεβού είναι
παλαιότερη.</span>
                @endif
                <br/>
            @endif

            @if($iamowner_getlist)
                <br/>
                <span style="margin-left: 20px;"> <a href="{{ config('app.url') . '/bookappointment/' . $appointments[$i]->id }}" class="btn btn-
outline-primary">{{ $appointments[$i]->date.' '.$appointments[$i]->time }}</a> </span>
                <span style="margin-left: 20px; font-weight: bold;"> <a href="{{ config('app.url') . '/myappointments/' . $appointments[$i]-
>user->id }}">{{ $appointments[$i]->user->lastname.' '.$appointments[$i]->user->firstname }} </a> </span>
                <span style="margin-left: 20px;"> Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα. </span>

                @if($appointments[$i]->expired==1)
                    <span style="margin-left: 20px; font-size:10px; font-color:rgb(85, 11, 11);">Η ημερομηνία του ραντεβού είναι
παλαιότερη.</span>
                    @endif
                    <br/>
                @endif

            @endif
        @endif

        @endif
    </br></br>
</div>
</script>

```

```
function afa(ch,id) {
if (confirm("Θέλεις να προχωρήσεις;")) {
var token = "{{ csrf_token() }}";

$.ajax({
type: 'post',
url: '{{ config('app.url') }}/afa',
data: {
_token: token,
ch: ch,
id: id
},
success: function(results) {
if(results.status){
alert(results.msg);
location.reload(true);
return false;
}
else{
alert(results.msg);
return false;
}
},
error: function( response ) {
alert("error: "+response.responseText+" (" +response.statusText+"");
return false;
}
});
}
}

</script>

</body>

</html>
```

bookappointment.blade.php

```
<!DOCTYPE html>
<html>
```

```

<head>
  <title>Main - Appointment</title>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
  <link rel="canonical" href="https://getbootstrap.com/docs/4.0/examples/album/">
  <!-- Bootstrap core CSS -->
  <link href="https://getbootstrap.com/docs/4.0/dist/css/bootstrap.min.css" rel="stylesheet">
  <link href="https://getbootstrap.com/docs/4.0/examples/album/album.css" rel="stylesheet">
</head>

<body>
  <div class="container">

    @include('header')

    @if($user_kind==0)

      @if(isset($appointment))

        @if($appointment->confirmed == 1 && $appointment->booked == $user_id)
          Το ραντεβού έχει επιβεβαιωθεί.
        <br/>
          Αν θέλετε μπορείτε να το ακυρώσετε
        <br/>
          <a style="font-size: 10px;" onclick="afa(2,{{ $appointment->id}})" class="btn btn-outline-danger my-2">Ακύρωσε το ραντεβού</a>
        @endif

        @if($appointment->confirmed == 0 && $appointment->booked == $user_id)
          Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα.
        @endif

        @if($appointment->confirmed == 0 && $appointment->booked != $user_id && $appointment->booked == 0)
          Μπορείτε να κλείσετε το ραντεβού (Θα περιμένετε στη συνέχεια να επιβεβαιωθεί από τον υπεύθυνο)
          <a style="font-size: 10px;" onclick="afa(1,{{ $appointment->id}})" class="btn btn-outline-success my-2">Κλείσε ραντεβού</a>
        @endif

        @if($appointment->confirmed == 0 && $appointment->booked != $user_id && $appointment->booked != 0)
          Το ραντεβού αυτό έχει κλειστεί από άλλον πελάτη.
        @endif
      @endif
    @endif
  </div>

```

```

@endif

@endif

@if($user_kind==2)

    @if(isset($appointment))

        @if($appointment->booked != 0)
            Το ραντεβού ανήκει στον χρήστη:
            @if(isset($userclient))
                <a href="{ {config('app.url')}/myappointments/'.$appointment->booked}" class="">{{ $userclient->lastname.' '.$userclient->firstname}}</a>
            @endif
        </br>

        @if($appointment->confirmed == 1)
            Το ραντεβού έχει επιβεβαιωθεί.
        </br>
            Αν θέλετε μπορείτε να το ακυρώσετε
        </br>
            <a style="font-size: 10px;" onclick="afa(2, '{{ $appointment->id}})" class="btn btn-outline-danger my-2">Ακύρωσε το ραντεβού</a>
        @endif

        {{!-- @if($appointment->confirmed == 0) --}}
        @if($appointment->confirmed == 0 && $appointment->booked != 0)
            Το ραντεβού δεν έχει επιβεβαιωθεί ακόμα. <br/>
            Μπορείτε να το επιβεβαιώσετε <br/>
            <a style="font-size: 10px;" onclick="afa(1, '{{ $appointment->id}})" class="btn btn-outline-success my-2">Επιβεβαίωση</a>
        </br>
            Αν θέλετε μπορείτε να το ακυρώσετε
        </br>
            <a style="font-size: 10px;" onclick="afa(2, '{{ $appointment->id}})" class="btn btn-outline-danger my-2">Ακύρωσε το ραντεβού</a>
        @endif

    @endif

    @else
        Μπορείτε να ορίσετε ένα νέο ραντεβού σε έναν χρήστη<br/>
        <input type="text" class="form-control" id="theuser" aria-describedby="theuser" placeholder="Χρήστης">
        <a style="font-size: 10px;" onclick="afa(3, '{{ $appointment->id}})" class="btn btn-outline-success my-2">Προσθήκη και Επιβεβαίωση</a>
    @endif

```

```

        <br/>
    @endif
@endif

</br></br>
</div>

<script>
    function afa(ch,id) {
        if (confirm("Θέλεις να προχωρήσεις;")) {
            var token = "{{ csrf_token() }}";

            $.ajax({
                type: 'post',
                url: '{{config('app.url')}}/afa',
                data: {
                    _token: token,
                    ch: ch,
                    id: id
                },
                success: function(results) {
                    if(results.status){
                        alert(results.msg);
                        location.reload(true);
                        return false;
                    }
                    else{
                        alert(results.msg);
                        return false;
                    }
                },
                error: function( response ) {
                    alert("error: "+response.responseText+" (" +response.statusText+"");
                    return false;
                }
            });
        }
    }
</script>

```

```
</body>
```

```
</html>
```