

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη διαδικτυακής εφαρμογής στο περιβάλλον Laravel.

The development of a web application using the Laravel framework.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:

ΣΠΥΡΟΣ ΜΕΜΟ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ:

ΛΙΑΡΟΚΑΠΗΣ ΔΗΜΗΤΡΙΟΣ

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

Επιβλέπων καθηγητής: Δημήτριος Λιαροκαπης

Εξεταστική επιτροπή: Φώτιος Βαρτζιώτης

Εξεταστική επιτροπή: Σπυριδούλα Μαργαρίτη

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον Καθηγητή μου για την ανάθεση αυτού του θέματος καθώς και την σωστή καθοδήγησή του προκειμένου να ολοκληρωθεί με επιτυχία η πτυχιακή μου εργασία για το θέμα ανάπτυξη ενός barbershop με το framework της Laravel.

ΠΕΡΙΛΗΨΗ

Η ιστοσελίδα του κουρείου που ανέπτυξα είναι μια σύγχρονη και φιλική προς τον χρήστη πλατφόρμα που σχεδιάστηκε για να είναι, και να προσφέρει μια εύκολη και αποτελεσματική εμπειρία . Η ιστοσελίδα επιτρέπει στους χρήστες να πραγματοποιήσουν όλες τις βασικές λειτουργίες που απαιτούνται σε ένα κουρείο, χωρίς να χρειάζεται να επισκεφθούν την εγκατάσταση με φυσική παρουσία παρα μόνο όταν θα έρθει η ώρα για να κουρευτούν .

Η σχεδίαση είναι καθαρή και εύληπτη, με σαφή κατευθύνσεις και ενδείξεις που βοηθούν τους χρήστες να βρουν ό,τι χρειάζονται γρήγορα και εύκολα. Η διεπαφή είναι αναπτυγμένη με τη χρήση των τεχνικών προτύπων του σήμερα, με στόχο να προσφέρεται μια εμπειρία χρήστη που είναι αποτελεσματική.

Οι χρήστες μπορούν να περιηγηθούν στην ιστοσελίδα και να αγοράσουν προϊόντα σχετικά με το κουρείο. Η πλατφόρμα προσφέρει μια σειρά προϊόντων που μπορούν να αγοραστούν online, με απλή και γρήγορη διαδικασία. Οι πελάτες μπορούν να επιλέξουν τα προϊόντα που τους αρέσουν, να τα προσθέσουν στο καλάθι τους και να τα πληρώσουν με ασφαλή μέθοδο πληρωμής.

Ένα από τα κύρια χαρακτηριστικά της ιστοσελίδας είναι η δυνατότητα να κλείνει ο χρήστης ραντεβού online. Οι χρήστες μπορούν να επιλέξουν την υπηρεσία που τους ενδιαφέρει, να διαλέξουν μια χρονική περίοδο και να καταγράψουν το ραντεβού τους μέσω της ιστοσελίδας. Η διαδικασία είναι απλή και γρήγορη, με σαφή κατευθύνσεις που βοηθούν τους χρήστες να ολοκληρώσουν την κλήση χωρίς προβλήματα.

Η ιστοσελίδα προσφέρει επίσης μια πλήρη διαδικασία εγγραφής και σύνδεσης. Οι χρήστες μπορούν να δημιουργήσουν έναν λογαριασμό με τις πληροφορίες τους και να συνδεθούν στην ιστοσελίδα . Η εγγραφή είναι απλή και γρήγορη, με σαφείς κατευθύνσεις που βοηθούν τους χρήστες να ολοκληρώσουν την διαδικασία χωρίς προβλήματα. Η σύνδεση είναι επίσης ασφαλής και γρήγορη..

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ.....	3
ΠΕΡΙΛΗΨΗ.....	4
ΠΕΡΙΕΧΟΜΕΝΑ.....	5
ΕΙΣΑΓΩΓΗ.....	6
Κεφάλαιο 1 Ανάλυση Απαιτήσεων.....	8-10
Κεφάλαιο 2 – Αρχιτεκτονικός Σχεδιασμός της Εφαρμογής.....	11-18
2.1 Blade templating engine	13-14
2.2 Ρόλος και Σημασία του Mode.....	14-15
2.3 Ρόλος και Σημασία του View.....	15
2.4 Η Υλοποίηση του View στο Laravel – Blade Templating Engine.....	17
2.5 Ο Ελεγκτής (Controller).....	18-19
Κεφάλαιο 3 – Λειτουργική Περιγραφή της Εφαρμογής.....	13-14
Κεφάλαιο 4 – Εμπειρία Χρήσης και Πλοήγηση στην Εφαρμογή.....	15-21
Κεφάλαιο 5 – Τεχνική Υλοποίηση και Εσωτερική Δομή της Εφαρμογής.....	21-24
Κεφάλαιο 6 – Σύνοψη και Συμπεράσματα.....	25-26
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	27

ΕΙΣΑΓΩΓΗ

Στη σύγχρονη εποχή, η τεχνολογία έχει καθιερωθεί ως αναπόσπαστο στοιχείο της λειτουργίας των επιχειρήσεων, ανεξαρτήτως του μεγέθους τους ή του κλάδου στον οποίο δραστηριοποιούνται. Ακόμα και μικρές επιχειρήσεις, όπως τα κουρεία (barbershops), αναγνωρίζουν όλο και περισσότερο τη σημασία της ψηφιακής παρουσίας και της αυτοματοποίησης βασικών λειτουργιών — από τη διαχείριση των ραντεβού και την προβολή των υπηρεσιών έως την εξυπηρέτηση των πελατών. Μέσα από την αξιοποίηση σύγχρονων ψηφιακών εργαλείων, τα κουρεία μπορούν να ενισχύσουν τη λειτουργικότητά τους, να περιορίσουν τα διοικητικά βάρη και να προσφέρουν μια αναβαθμισμένη εμπειρία εξυπηρέτησης.

Η συγκεκριμένη πτυχιακή εργασία εστιάζει στην υλοποίηση μιας διαδραστικής διαδικτυακής εφαρμογής για την υποστήριξη ενός καταστήματος, μέσω της οποίας καθίσταται δυνατή η αποτελεσματική οργάνωση ραντεβού, υπηρεσιών και πελατών. Η πλατφόρμα δίνει τη δυνατότητα στον πελάτη να ενημερωθεί για τις παρεχόμενες υπηρεσίες, να επιλέξει χρονική ζώνη που τον εξυπηρετεί και να προβεί σε ηλεκτρονική κράτηση, εξαλείφοντας την ανάγκη φυσικής παρουσίας για την οργάνωση της επίσκεψής του. Από την πλευρά της επιχείρησης, ο διαχειριστής έχει πρόσβαση σε πίνακα ελέγχου (admin panel) που του επιτρέπει να διαχειρίζεται δυναμικά τα ραντεβού και τις υπηρεσίες, προσφέροντας παράλληλα έλεγχο και εποπτεία στη λειτουργία του καταστήματος.

Η ανάπτυξη της εφαρμογής υλοποιήθηκε με τη χρήση του Laravel, ενός ισχυρού και δημοφιλούς περιβάλλοντος ανάπτυξης (framework) βασισμένου στη γλώσσα PHP. Το Laravel διακρίνεται για τη δομημένη αρχιτεκτονική του, την προσήλωσή του στην ασφάλεια και την υψηλή αποδοτικότητα στην ανάπτυξη εφαρμογών. Υιοθετεί το μοντέλο Model-View-Controller (MVC), το οποίο προάγει την καθαρή διάκριση μεταξύ επιχειρησιακής λογικής, παρουσίασης και διαχείρισης δεδομένων. Χάρη στην αρχιτεκτονική αυτή, διευκολύνεται η συντήρηση, η αναβάθμιση και η επεκτασιμότητα του λογισμικού.

Ενδεικτικά, το Laravel ενσωματώνει εργαλεία και μηχανισμούς όπως:

- Το Eloquent ORM για αλληλεπίδραση με βάσεις δεδομένων σε αντικειμενοστραφή μορφή.
- Το Blade, ένα σύστημα template rendering που συνδυάζει HTML με λογική παρουσίασης.
- Το Artisan, εργαλείο γραμμής εντολών για αυτοματοποίηση διαδικασιών προγραμματισμού.
- Το σύστημα Routing και Middleware για προσαρμοσμένη διαχείριση αιτήσεων και ελέγχων.

Η αξιοποίηση του Laravel παρέχει σταθερή υποδομή για την ανάπτυξη web εφαρμογών, μειώνοντας το απαιτούμενο αναπτυξιακό κόστος και τον χρόνο παράδοσης χωρίς συμβιβασμούς στην απόδοση ή στην ασφάλεια. Επιπλέον, η ενεργή κοινότητα χρηστών και η λεπτομερής τεκμηρίωση υποστηρίζουν σε μεγάλο βαθμό την ανάπτυξη ακόμα και από μεμονωμένους προγραμματιστές ή μικρές ομάδες.

Ο στόχος της πτυχιακής εργασίας είναι διπλός: αφενός να παραδοθεί μια λειτουργική και ολοκληρωμένη εφαρμογή διαχείρισης barbershop, και αφετέρου να αναδειχθεί η χρήση του Laravel ως ένα σύγχρονο εργαλείο που απευθύνεται όχι μόνο σε μεγάλες επιχειρήσεις αλλά και σε μικρομεσαίες μονάδες που επιδιώκουν τον ψηφιακό τους μετασχηματισμό.

Κεφάλαιο 1: Ανάλυση Απαιτήσεων

Η φάση της καταγραφής απαιτήσεων αποτελεί τον θεμελιώδη λίθο στον σχεδιασμό και την υλοποίηση πληροφοριακών συστημάτων. Στην παρούσα εφαρμογή, που απευθύνεται σε σύγχρονη επιχείρηση παροχής υπηρεσιών περιποίησης (barbershop), η αναλυτική αποτύπωση των αναγκών των τελικών χρηστών διασφαλίζει ότι το τελικό προϊόν θα ανταποκρίνεται επακριβώς στις επιθυμητές λειτουργίες και χαρακτηριστικά.

1.1.1.1 Μη Λειτουργικές Απαιτήσεις

Οι μη λειτουργικές απαιτήσεις ορίζουν ποιοτικά χαρακτηριστικά που επηρεάζουν τη συμπεριφορά και τη βιωσιμότητα του συστήματος, χωρίς να αφορούν άμεσα την κύρια λειτουργικότητα:

- **Ασφάλεια:** Ιδιαίτερη έμφαση δίνεται στην προστασία των ευαίσθητων δεδομένων των χρηστών. Η εφαρμογή αξιοποιεί τεχνικές κρυπτογράφησης, έλεγχο εγκυρότητας δεδομένων και μηχανισμούς ταυτοποίησης, προσφέροντας ένα αξιόπιστο περιβάλλον διαχείρισης.
- **Επεκτασιμότητα:** Ο σχεδιασμός της εφαρμογής βασίζεται σε ανοικτή αρχιτεκτονική, επιτρέποντας την εύκολη ενσωμάτωση επιπλέον δυνατοτήτων, όπως διαχειριστικό πάνελ ή ηλεκτρονικές πληρωμές, χωρίς να απαιτείται ριζική αναδόμηση του συστήματος.
- **Απόδοση:** Οι χρόνοι φόρτωσης των σελίδων και η απόκριση στις ενέργειες των χρηστών έχουν βελτιστοποιηθεί, εξασφαλίζοντας ομαλή και αποδοτική χρήση.
- **Χρηστικότητα:** Το περιβάλλον χρήσης έχει σχεδιαστεί ώστε να είναι διαισθητικό και απλό, επιτρέποντας ακόμη και σε άτομα χωρίς ιδιαίτερη εμπειρία την αποτελεσματική πλοήγηση και χρήση της εφαρμογής.
- **Συντηρησιμότητα:** Ο πηγαίος κώδικας έχει οργανωθεί κατά τρόπο αρθρωτό, ακολουθώντας το αρχιτεκτονικό πρότυπο MVC (Model-View-Controller), γεγονός που διευκολύνει τον εντοπισμό, τροποποίηση και επέκταση της λειτουργικότητας στο μέλλον.

Η παραπάνω προσέγγιση επιτρέπει στην εφαρμογή να λειτουργεί όχι μόνο ως ένα εργαλείο καθημερινής εξυπηρέτησης, αλλά και ως θεμέλιο για την ψηφιακή ανάπτυξη και προβολή της επιχείρησης.

1.1.1.2 Λειτουργικές Απαιτήσεις (Αναδιατυπωμένη εκδοχή)

Οι λειτουργικές απαιτήσεις αποτελούν την περιγραφή των βασικών ενεργειών και δυνατοτήτων που πρέπει να υποστηρίξει το σύστημα ώστε να ανταποκρίνεται στις ανάγκες των τελικών χρηστών. Στην προτεινόμενη web εφαρμογή για barbershop, οι λειτουργικές δυνατότητες αναπτύσσονται βάσει των εξής επιμέρους λειτουργιών:

- **Δημιουργία Λογαριασμού Χρήστη (Εγγραφή):** Παρέχεται στον επισκέπτη η δυνατότητα εγγραφής με χρήση email και κωδικού πρόσβασης, διαμορφώνοντας έναν εξατομικευμένο λογαριασμό.

- Είσοδος Χρήστη (Σύνδεση): Οι καταχωρημένοι χρήστες μπορούν να εισέρχονται στην εφαρμογή μέσω της διαδικασίας ταυτοποίησης, αποκτώντας πρόσβαση σε προσωπικές υπηρεσίες, ιστορικό ραντεβού και αγορών.
- Προβολή Διαθέσιμων Υπηρεσιών: Η εφαρμογή παρέχει πλήρη κατάλογο των προσφερόμενων υπηρεσιών (κούρεμα, περιποίηση γενιών, κ.ά.), συνοδευόμενο από σύντομες περιγραφές.
- Εμφάνιση Τιμοκαταλόγου: Οι χρήστες μπορούν να ενημερώνονται για τις τιμές κάθε υπηρεσίας, ώστε να διαμορφώσουν άποψη για το κόστος προτού προβούν σε κράτηση.
- Σύστημα Κράτησης Ραντεβού: Ο χρήστης έχει τη δυνατότητα επιλογής υπηρεσίας, ημερομηνίας και χρονικής ζώνης για την επίσκεψή του. Το σύστημα ελέγχει τη διαθεσιμότητα και καταγράφει την κράτηση.
- Παρακολούθηση και Διαχείριση Ραντεβού: Ο εγγεγραμμένος χρήστης μπορεί να δει τα ενεργά και προηγούμενα ραντεβού του, να τα τροποποιήσει ή να τα ακυρώσει εφόσον το επιθυμεί.
- Διαχείριση Προϊόντων και Αγορών: Η πλατφόρμα περιλαμβάνει δυνατότητα ηλεκτρονικών παραγγελιών προϊόντων περιποίησης. Ο χρήστης μπορεί να προσθέσει προϊόντα στο καλάθι αγορών, να δει το κόστος και να ολοκληρώσει την παραγγελία του.
- Καλάθι Αγορών και Υποβολή Παραγγελίας: Προσφέρεται λειτουργία επεξεργασίας του καλαθιού, υπολογισμού του συνολικού κόστους και υποβολής παραγγελίας με όλα τα απαιτούμενα στοιχεία (όνομα, διεύθυνση, τηλέφωνο κ.ά.).
- Φιλικός προς τον Χρήστη Σχεδιασμός (Responsive Design): Η διεπαφή της εφαρμογής προσαρμόζεται αυτόματα σε οθόνες διαφορετικών συσκευών (υπολογιστές, tablets, κινητά), διασφαλίζοντας ομαλή και άμεση εμπειρία πλοήγησης.
- Επικοινωνία: Μέσω ειδικής φόρμας ή στοιχείων επικοινωνίας, ο χρήστης μπορεί να στείλει ερωτήματα, αιτήματα υποστήριξης ή παρατηρήσεις προς την επιχείρηση.

1.1.1.3 Διαχείριση Χρήστη

Το σύστημα επιτρέπει τη δημιουργία και διαχείριση προσωπικού λογαριασμού. Οι χρήστες μπορούν να εγγραφούν στην πλατφόρμα, να συνδεθούν με ασφάλεια και να αποκτήσουν πρόσβαση σε εξατομικευμένες υπηρεσίες, όπως η παρακολούθηση των ραντεβού τους και η διαχείριση παραγγελιών. Η διαδικασία ταυτοποίησης αξιοποιεί τους μηχανισμούς του Laravel, εξασφαλίζοντας προστασία των προσωπικών δεδομένων.

1.1.1.4 Κρατήσεις και Αγορές

Έμφαση δίνεται στην ενσωμάτωση δύο βασικών πυλώνων: του συστήματος ραντεβού και της λειτουργικότητας e-shop. Ο χρήστης επιλέγει υπηρεσία, διαθέσιμη ημερομηνία και ώρα και υποβάλλει αίτημα κράτησης, το οποίο αποθηκεύεται και διαχειρίζεται με ευκολία.

Ταυτόχρονα, δίνεται η δυνατότητα περιήγησης και αγοράς προϊόντων περιποίησης. Ο χρήστης προσθέτει τα είδη στο καλάθι, ελέγχει το συνολικό ποσό και ολοκληρώνει την

παραγγελία του μέσω φιλικού και εύχρηστου interface. Οι παραγγελίες αποθηκεύονται για επεξεργασία ή ιστορικό αναφοράς.

1.1.1.5 Υποδομή και Σχέσεις Δεδομένων

Η διαχείριση της βάσης δεδομένων επιτυγχάνεται με το εργαλείο Eloquent ORM, το οποίο επιτρέπει αντικειμενοστραφή διαχείριση πινάκων. Σχεδιάζονται πίνακες για χρήστες, ραντεβού, προϊόντα και παραγγελίες με ξεκάθαρες σχέσεις και ενσωματωμένη ακεραιότητα. Η βάση υποστηρίζει πολύπλοκες συσχετίσεις μεταξύ οντοτήτων και παρέχει δυνατότητα για ταχεία ανάκτηση πληροφορίας.

Παράλληλα, εφαρμόζονται τεχνικές προστασίας όπως CSRF Tokens, SQL Injection filters και XSS prevention για την αποτροπή κακόβουλων ενεργειών μέσω των φόρμων εισαγωγής.

1.1.1.6 Δυνατότητα Επέκτασης

Η υπάρχουσα έκδοση της εφαρμογής έχει σχεδιαστεί με προοπτική επέκτασης. Αν και απευθύνεται αποκλειστικά σε τελικούς χρήστες, η προσθήκη ρόλων διαχειριστή ή υπαλλήλων κρίνεται απλή και υλοποιήσιμη. Επιπλέον, μπορεί εύκολα να ενσωματωθεί σύστημα πληρωμών, ειδοποιήσεις ή στατιστικά επισκεψιμότητας, δίνοντας στην εφαρμογή χαρακτήρα ενιαίου πληροφοριακού κόμβου.

1.1.1.7 Γενική Εκτίμηση

Η υλοποίηση αυτής της εφαρμογής θέτει τις βάσεις για τον πλήρη ψηφιακό μετασχηματισμό ενός barbershop, παρέχοντας στην επιχείρηση τη δυνατότητα αυτοματοποίησης καθημερινών διαδικασιών, προβολής υπηρεσιών και αύξησης του πελατολογίου. Η χρήση Laravel συμβάλλει στη σταθερότητα, στην απόδοση και στην ασφάλεια, διασφαλίζοντας ότι η πλατφόρμα μπορεί να λειτουργήσει απρόσκοπτα και να εξελιχθεί περαιτέρω με βάση τις ανάγκες του επιχειρησιακού περιβάλλοντος.

Κεφάλαιο 2 – Αρχιτεκτονικός Σχεδιασμός της Εφαρμογής

Η αρχιτεκτονική σχεδίαση της εφαρμογής που αναπτύχθηκε στοχεύει στη δημιουργία μιας ευέλικτης και αξιόπιστης δομής, ικανής να υποστηρίξει τόσο την παρούσα λειτουργικότητα όσο και μελλοντικές επεκτάσεις. Το πλαίσιο που επιλέχθηκε για την υλοποίηση είναι το Laravel, ένα σύγχρονο και δημοφιλές PHP framework που προσφέρει σταθερές βάσεις για την ανάπτυξη δυναμικών web εφαρμογών.

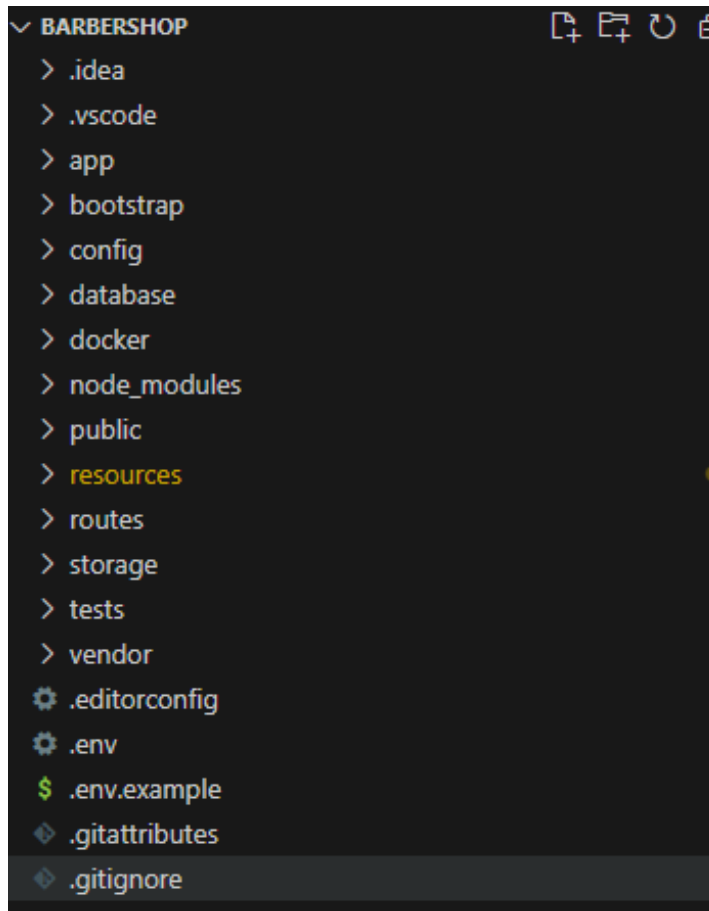
Η υλοποίηση βασίζεται στο δοκιμασμένο αρχιτεκτονικό μοντέλο Model-View-Controller (MVC), το οποίο επιτρέπει την οργάνωση του κώδικα σε σαφώς διαχωρισμένες ενότητες. Αυτή η προσέγγιση διασφαλίζει την ανεξαρτησία της επιχειρησιακής λογικής από το επίπεδο προβολής, διευκολύνοντας την επαναχρησιμοποίηση, τη συντήρηση και την αναβάθμιση του λογισμικού. Συγκεκριμένα:

- Το επίπεδο του Μοντέλου (Model) είναι υπεύθυνο για τη διαχείριση των δομών δεδομένων, την αντιστοίχιση με τους πίνακες της βάσης και την επιχειρησιακή λογική που αφορά τη διαχείριση αυτών των δεδομένων.
- Το επίπεδο Εμφάνισης (View) χειρίζεται την απόδοση της πληροφορίας στον τελικό χρήστη, αξιοποιώντας τη Blade templating engine του Laravel για καθαρή και ευέλικτη σύνθεση διεπαφών.
- Ο Ελεγκτής (Controller) αναλαμβάνει τον συντονισμό της λογικής, δέχεται τις αιτήσεις από τον χρήστη και καθορίζει την κατάλληλη απόκριση ή προβολή, δρώντας ως διαμεσολαβητής μεταξύ των υπολοίπων επιπέδων.

Για την επικοινωνία με τη βάση δεδομένων χρησιμοποιείται το ενσωματωμένο ORM του Laravel, το Eloquent. Μέσω αυτής της προσέγγισης, κάθε αντικείμενο (όπως χρήστης, υπηρεσία, παραγγελία, προϊόν ή ραντεβού) αντιστοιχίζεται σε μοντέλο, το οποίο επιτρέπει την ευέλικτη αλληλεπίδραση με τα δεδομένα και εξασφαλίζει συμβατότητα με αρχές ασφάλειας και ακεραιότητας.

Η βασική δομή αρχείων της εφαρμογής διαμορφώνεται ως εξής:

- routes/web.php: Περιλαμβάνει τον καθορισμό των URL διαδρομών που οδηγούν στις αντίστοιχες ενέργειες των controllers.
- app/Models: Περιέχει τις κλάσεις μοντέλων που εκπροσωπούν τους πίνακες της βάσης.
- app/Http/Controllers: Περιέχει τη λογική των controllers για την επεξεργασία αιτημάτων.
- resources/views: Περιλαμβάνει τα αρχεία Blade που διαμορφώνουν την τελική προβολή των σελίδων.



Η σχεδίαση της βάσης δεδομένων περιλαμβάνει πίνακες όπως: users, services, appointments, products, orders και order_items. Κάθε πίνακας είναι οργανωμένος με βάση τις αρχές κανονικοποίησης και τις βέλτιστες πρακτικές σε επίπεδο foreign keys, indexing και αποδοτικής διάρθρωσης δεδομένων.

Η ροή δεδομένων στο σύστημα είναι δομημένη με τρόπο που να διασφαλίζει την ακεραιότητα και την ασφάλεια των συναλλαγών. Κάθε αίτημα που υποβάλλεται από τον χρήστη (όπως φόρμες ραντεβού ή παραγγελιών) περνά μέσα από ελεγκτές middleware, μηχανισμούς επικύρωσης (validation) και ελέγχους CSRF. Σε περιπτώσεις σφάλματος παρέχονται στοχευμένα και κατανοητά μηνύματα για καθοδήγηση του χρήστη.

Η διαχείριση της κατάστασης του χρήστη γίνεται μέσω session state, όπου ο καθένας μπορεί να δει και να αλληλεπιδρά μόνο με τα δικά του δεδομένα. Η ασφάλεια των δεδομένων είναι θεμελιώδης και ενσωματώνεται σε κάθε επίπεδο, από την αυθεντικοποίηση έως την πρόσβαση στο περιεχόμενο του προφίλ του χρήστη.

Αν και η αρχική έκδοση της εφαρμογής απευθύνεται αποκλειστικά σε τελικούς χρήστες, η αρχιτεκτονική της είναι κατασκευασμένη με ορίζοντα επέκτασης. Είναι έτοιμη να ενσωματώσει διακριτούς ρόλους, όπως διαχειριστές ή υπαλλήλους, και να υποστηρίξει πρόσθετα υποσυστήματα όπως APIs για πληρωμές, συνδέσεις με mobile apps, ειδοποιήσεις και σύστημα αξιολόγησης υπηρεσιών.

Ένα από τα βασικά γνωρίσματα της παρούσας αρχιτεκτονικής είναι η modular λογική: κάθε λειτουργία είναι ανεξάρτητη από τις υπόλοιπες και οργανώνεται γύρω από τις αρχές του loose coupling και high cohesion. Έτσι, μπορεί κανείς να αφαιρέσει ή να προσθέσει νέα τμήματα (όπως πρόγραμμα επιβράβευσης, στατιστικά ή νέα κανάλια παραγγελίας) χωρίς να επηρεάσει αρνητικά την υπάρχουσα δομή.

Συνοψίζοντας, η αρχιτεκτονική του συστήματος αντανακλά σύγχρονες πρακτικές ανάπτυξης λογισμικού, με έμφαση στη σταθερότητα, την επεκτασιμότητα και την ευχρηστία. Η επιλογή του Laravel ως βάση προσφέρει ισχυρή υποστήριξη εργαλείων, τεκμηρίωσης και ασφάλειας, καθιστώντας την εφαρμογή κατάλληλη για παραγωγικό και βιώσιμο περιβάλλον.

2.1 Blade templating engine

Το Blade Templating Engine και ο Ρόλος του στη Διαμόρφωση της Διεπαφής Χρήστη

Στον πυρήνα της διαδικασίας ανάπτυξης web εφαρμογών με το Laravel framework, ιδιαίτερη σημασία αποδίδεται στην αποδοτική και ευέλικτη διαμόρφωση του επιπέδου παρουσίασης (presentation layer). Το Blade, το εγγενές σύστημα templating του Laravel, επιτελεί ακριβώς αυτόν τον ρόλο, προσφέροντας ένα ισχυρό αλλά ταυτόχρονα εύχρηστο εργαλείο για τη δημιουργία δυναμικών ιστοσελίδων.

Σε αντίθεση με παραδοσιακά συστήματα δημιουργίας προβολών, το Blade αξιοποιεί μία συντακτική προσέγγιση που διακρίνεται για την απλότητά της, χωρίς να στερείται δυνατοτήτων. Μέσω ειδικών εκφράσεων και οδηγιών, όπως οι εντολές `@if`, `@foreach` και `@include`, το Blade επιτρέπει την εύκολη ενσωμάτωση λογικών δομών μέσα σε αρχεία HTML, διατηρώντας παράλληλα την αναγνωσιμότητα του κώδικα. Αυτή η διαχωριστική γραμμή μεταξύ λογικής και παρουσίασης διευκολύνει τόσο τη συντήρηση όσο και τη μελλοντική επέκταση του συστήματος.

Σημαντική πτυχή του Blade αποτελεί η δυνατότητα κληρονομικότητας μεταξύ των αρχείων προβολής. Μέσω του μηχανισμού layouts, οι προγραμματιστές έχουν τη δυνατότητα να ορίσουν βασικές δομές για το σύνολο των σελίδων, αποφεύγοντας την επανάληψη κώδικα και ενισχύοντας τη συνέπεια στην αισθητική και στη λειτουργικότητα του ιστότοπου. Ειδικότερα, η χρήση των sections και των yields επιτρέπει την προσαρμογή επιμέρους περιοχών της διεπαφής, ανάλογα με τις ανάγκες κάθε σελίδας.

Πέραν της δομικής οργάνωσης, το Blade παρέχει ενσωματωμένους μηχανισμούς ασφαλείας, όπως η αυτόματη διαφυγή HTML (automatic HTML escaping), που προστατεύει την εφαρμογή από κοινές ευπάθειες, όπως οι επιθέσεις XSS (Cross-Site Scripting). Η συγκεκριμένη λειτουργία εξασφαλίζει ότι το περιεχόμενο που εισάγεται στις προβολές ελέγχεται και αποδίδεται με ασφάλεια, χωρίς να απαιτείται επιπρόσθετη παρέμβαση από τον προγραμματιστή.

Ένα ακόμη χαρακτηριστικό που καθιστά το Blade ιδιαίτερα αποδοτικό είναι η δυνατότητα προ-μεταγλώττισης των αρχείων του σε native PHP κώδικα. Αυτή η διαδικασία επιταχύνει τον χρόνο απόκρισης της εφαρμογής, καθώς μειώνει το φορτίο κατά τη φάση της εκτέλεσης, ενώ παράλληλα συμβάλλει στη βελτιστοποίηση των επιδόσεων.

Συνοψίζοντας, το Blade δεν αποτελεί απλώς ένα μέσο για την ανάπτυξη προβολών, αλλά έναν θεμελιώδη πυλώνα για τη δημιουργία ενός συνεκτικού, ασφαλούς και επεκτάσιμου περιβάλλοντος διεπαφής. Ο συνδυασμός ευχρηστίας, απόδοσης και αυστηρής οργάνωσης που το χαρακτηρίζει καθιστά το Blade ένα εργαλείο στρατηγικής σημασίας για την αποτελεσματική ανάπτυξη σύγχρονων web εφαρμογών.

.2.2 Ρόλος και Σημασία του Model

Το Model δεν αποτελεί απλώς έναν διαμεσολαβητή για την αποθήκευση και ανάκτηση δεδομένων, αλλά ενσωματώνει πλήρως την επιχειρησιακή λογική που διέπει τα δεδομένα της εφαρμογής. Στην πράξη, το Model διαχειρίζεται:

- Τη δομή και τα χαρακτηριστικά των δεδομένων (attributes).
- Τους κανόνες ακεραιότητας και επικύρωσης (validation rules).
- Τις σχέσεις μεταξύ διαφορετικών οντοτήτων.
- Τις λειτουργίες ανάκτησης, δημιουργίας, τροποποίησης και διαγραφής εγγραφών.

Με άλλα λόγια, το Model αποτελεί την «καρδιά» της εφαρμογής, όπου συγκεντρώνονται όλοι οι κανόνες και οι διαδικασίες που αφορούν τα δεδομένα.

X.3 Εργαλεία και Μηχανισμοί του Model στο Laravel

Στο Laravel, το Model αξιοποιεί πλήρως τις δυνατότητες του Eloquent ORM, παρέχοντας μια ισχυρή εργαλειοθήκη για την υλοποίηση λειτουργιών βάσης δεδομένων. Μεταξύ των βασικών δυνατοτήτων του Model συγκαταλέγονται:

- Αναπαράσταση των πινάκων ως κλάσεις (classes) με συγκεκριμένες ιδιότητες.
- Αυτοματοποιημένη αντιστοίχιση πρωτεύοντων κλειδιών (primary keys) και χρονικών στιγμών (timestamps).
- Υποστήριξη σχέσεων μεταξύ πινάκων (hasOne, hasMany, belongsTo, belongsToMany).
- Χρήση scopes για επαναχρησιμοποιούμενα φίλτρα και ερωτήματα.
- Υποστήριξη soft deletes για λογική διαγραφή εγγραφών χωρίς φυσική διαγραφή από τη βάση.

Αυτοί οι μηχανισμοί επιτρέπουν την ανάπτυξη ευέλικτων και επεκτάσιμων εφαρμογών με ελάχιστο κώδικα.

Η δημιουργία ενός Model στο Laravel είναι μια απλή διαδικασία, η οποία ενσωματώνει ισχυρά πρότυπα σχεδίασης. Για παράδειγμα, το Model User αναπαριστά τον πίνακα users και μπορεί να χρησιμοποιηθεί ως εξής:

```
$user = User::find(1);
```

```
$user->name = 'Νέο Όνομα';
```

```
$user->save();
```

Το παραπάνω παράδειγμα καταδεικνύει τη δύναμη της αντικειμενοστραφούς προσέγγισης του Eloquent ORM, όπου ο προγραμματιστής εργάζεται αποκλειστικά με αντικείμενα και ιδιότητες, χωρίς να απαιτείται άμεση σύνταξη SQL εντολών.

Ένα από τα πλέον ισχυρά χαρακτηριστικά του Model στο Laravel είναι η ευκολία με την οποία ορίζονται οι σχέσεις μεταξύ οντοτήτων. Ενδεικτικά:

- Ένας χρήστης μπορεί να έχει πολλά ραντεβού (hasMany relationship).
- Κάθε ραντεβού ανήκει σε έναν συγκεκριμένο χρήστη (belongsTo relationship).
- Τα προϊόντα και οι παραγγελίες μπορούν να συσχετιστούν μέσω many-to-many σχέσεων.

Οι σχέσεις αυτές ορίζονται μέσα στο Model με απλές μεθόδους, επιτρέποντας την άμεση προσπέλαση σχετιζόμενων δεδομένων μέσω εύχρηστων συντακτικών δομών.

Η αξιοποίηση του επιπέδου Model προσφέρει πολλαπλά πλεονεκτήματα:

- Δομημένος και ευανάγνωστος κώδικας.
- Επαναχρησιμοποίηση λογικής μέσω μεθόδων και scopes.
- Αυξημένη ασφάλεια μέσω αυτόματης αποτροπής επιθέσεων SQL Injection.
- Ευελιξία στην επέκταση και τη συντήρηση του συστήματος.

Αυτά τα πλεονεκτήματα καθιστούν το Model αναπόσπαστο τμήμα κάθε σύγχρονης Laravel εφαρμογή

2.3 Ρόλος και Σημασία του View

Το View λειτουργεί ως το σημείο επαφής μεταξύ του χρήστη και του συστήματος. Αποσκοπεί στη μετατροπή των δεδομένων σε κατανοητές και οπτικά ευχάριστες δομές, διευκολύνοντας την πλοήγηση και την αλληλεπίδραση με την εφαρμογή. Ορισμένες από τις βασικές αρμοδιότητες του View περιλαμβάνουν:

- Την απεικόνιση δεδομένων που παρέχονται από το Model.
- Τη μορφοποίηση και διάταξη των πληροφοριών στη σελίδα.
- Τη διαχείριση της αλληλεπίδρασης με τον χρήστη, όπως φόρμες και κουμπιά.
- Την υποστήριξη πολύγλωσσου περιβάλλοντος μέσω μεταφράσεων.

- Τη δυναμική εμφάνιση περιεχομένου ανάλογα με την κατάσταση του χρήστη (π.χ. σύνδεση, ρόλος).

2.4 Η Υλοποίηση του View στο Laravel – Blade Templating Engine

Στο Laravel, το View υλοποιείται με τη χρήση του Blade, ενός ισχυρού και ελαφρού template engine, που έχει σχεδιαστεί αποκλειστικά για εφαρμογές Laravel. Το Blade παρέχει έναν συνδυασμό από HTML και εντολές σε ειδική σύνταξη, επιτρέποντας την εύκολη δημιουργία δυναμικού περιεχομένου χωρίς περίπλοκο κώδικα.

Μερικά από τα βασικά χαρακτηριστικά του Blade περιλαμβάνουν:

- Δυνατότητα ενσωμάτωσης μεταβλητών PHP απευθείας σε HTML.
- Υποστήριξη επαναχρησιμοποιήσιμων templates μέσω της κληρονομικότητας (template inheritance).
- Παροχή εντολών για βρόχους (loops), συνθήκες (if/else) και διαχείριση σφαλμάτων.
- Εύκολη συμπερίληψη επιμέρους views με χρήση του @include.
- Ενσωμάτωση συστήματος slots και components για την καλύτερη οργάνωση της διεπαφής.

Η χρήση του Blade επιτρέπει στον προγραμματιστή να δημιουργεί templates που συνδυάζουν HTML και δυναμικό περιεχόμενο με σαφή και κατανοητό τρόπο. Ένα απλό παράδειγμα Blade view μπορεί να είναι το εξής:

```
@extends('layouts.app')
```

```
@section('content')
```

```
<h1>Καλωσορίσατε, {{ $user->name }}!</h1> <p>Τα πρόσφατα ραντεβού σας:</p>
@foreach ($appointments as $appointment) <div>{{ $appointment->date }} - {{
$appointment->service }}</div> @endforeach @endsection
```

Το παραπάνω απόσπασμα παρουσιάζει τα οφέλη της σύντομης και ευανάγνωστης σύνταξης του Blade, ενώ ταυτόχρονα διατηρεί τον διαχωρισμό ανάμεσα στον κώδικα παρουσίασης και την επιχειρησιακή λογική.

Η χρήση του Blade Templating Engine σε συνδυασμό με το View παρέχει πολλαπλά οφέλη:

- Καθαρός διαχωρισμός ανάμεσα στη λογική της εφαρμογής και την παρουσίαση.
- Βελτιωμένη αναγνωσιμότητα και συντηρησιμότητα του κώδικα.
- Μείωση επαναλαμβανόμενου κώδικα μέσω κληρονομικότητας και includes.
- Ευελιξία στην ανάπτυξη πολύπλοκων διεπαφών χρήστη.
- Ενισχυμένη ταχύτητα ανάπτυξης με λιγότερες πιθανότητες λαθών.

2.5 Ο Ελεγκτής (Controller)

Ο Ελεγκτής (Controller) αποτελεί το ενδιάμεσο επίπεδο του αρχιτεκτονικού προτύπου Model-View-Controller (MVC) και έχει καθοριστικό ρόλο στον συντονισμό της ροής των δεδομένων ανάμεσα στα υπόλοιπα δύο επίπεδα, δηλαδή το Μοντέλο (Model) και την Εμφάνιση (View). Ο κύριος σκοπός του Controller είναι να επεξεργάζεται τα αιτήματα που υποβάλλουν οι χρήστες, να καλεί την αντίστοιχη επιχειρησιακή λογική και να επιστρέφει τα αποτελέσματα προς εμφάνιση.

Οι βασικές λειτουργίες του Controller περιλαμβάνουν:

- Λήψη και επεξεργασία εισερχόμενων αιτημάτων (HTTP requests).
- Εφαρμογή κανόνων επικύρωσης δεδομένων (validation) ώστε να διασφαλιστεί η ορθότητα των εισροών.
- Επικοινωνία με τα μοντέλα για την ανάκτηση, επεξεργασία ή αποθήκευση δεδομένων.
- Επιλογή και απόδοση του κατάλληλου View για την παρουσίαση των δεδομένων στον χρήστη.
- Διαχείριση διαδικασιών πλοήγησης, ανακατευθύνσεων και ειδοποιήσεων στο περιβάλλον χρήστη.
- Έλεγχος δικαιωμάτων πρόσβασης και πιστοποίησης χρηστών.

Στο Laravel, οι Controllers δημιουργούνται ως ξεχωριστές κλάσεις εντός του φακέλου `app/Http/Controllers`. Το Laravel προσφέρει ειδικές μεθόδους (actions) μέσα στους Controllers για την αντιμετώπιση συγκεκριμένων λειτουργιών ή αιτημάτων, διατηρώντας τον κώδικα οργανωμένο και ευανάγνωστο.

Μία βασική δομή Controller στο Laravel έχει ως εξής:

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\User;

class UserController extends Controller
{
    public function index()
    {
        $users = User::all();
        return view('users.index', compact('users'));
    }
}
```

Στο παραπάνω παράδειγμα, ο UserController αντλεί δεδομένα από το μοντέλο User και επιστρέφει το αντίστοιχο View για την προβολή τους.

Η χρήση Controllers προσφέρει σημαντικά πλεονεκτήματα, όπως:

- Ξεκάθαρος διαχωρισμός των αρμοδιοτήτων εντός της εφαρμογής.
- Ευκολία στη συντήρηση και επεκτασιμότητα του κώδικα.
- Απλοποίηση της διαδικασίας ανάπτυξης, καθώς το κάθε Controller εστιάζει σε συγκεκριμένο σύνολο λειτουργιών.
- Ενίσχυση της ασφάλειας μέσω ελέγχου πρόσβασης και επικύρωσης δεδομένων.
- Ευκολότερη ανάλυση και εντοπισμός σφαλμάτων κατά τον έλεγχο της εφαρμογής.

Ο Controller αποτελεί έναν θεμελιώδη πυλώνα του αρχιτεκτονικού προτύπου MVC, καθώς διασφαλίζει την ομαλή και αποτελεσματική ροή δεδομένων μεταξύ των υπόλοιπων επιπέδων. Στο πλαίσιο του Laravel, η αξιοποίηση των Controllers επιτρέπει την οργάνωση του κώδικα με συστηματικό τρόπο, διευκολύνοντας τόσο την ανάπτυξη όσο και την μελλοντική επέκταση της εφαρμογής. Έτσι, η ορθή σχεδίαση και υλοποίηση των Controllers αποτελεί κρίσιμο παράγοντα για την επιτυχία και τη σταθερότητα της εφαρμογής.

Κεφάλαιο 3 – Λειτουργική Περιγραφή της Εφαρμογής

Η διαδικτυακή πλατφόρμα που αναπτύχθηκε για την εξυπηρέτηση ενός σύγχρονου barbershop ενσωματώνει κρίσιμες λειτουργίες που καλύπτουν την πλήρη αλληλεπίδραση του πελάτη με την επιχείρηση. Οι ενότητες της εφαρμογής σχεδιάστηκαν ώστε να παρέχουν ευχρηστία, διαφάνεια και αποτελεσματικότητα, αξιοποιώντας τις τεχνικές δυνατότητες του Laravel framework.

Η λειτουργικότητα χωρίζεται σε πέντε βασικά επίπεδα: διαχείριση χρήστη (εγγραφή και είσοδος), παρουσίαση υπηρεσιών, κράτηση ραντεβού, ηλεκτρονικές παραγγελίες προϊόντων και προβολή προσωπικού προφίλ.

3.1 Δημιουργία και Ταυτοποίηση Χρήστη

Η πρόσβαση στις περισσότερες λειτουργίες απαιτεί την ύπαρξη λογαριασμού. Η διαδικασία εγγραφής επιτρέπει στον χρήστη να καταχωρήσει τα βασικά του στοιχεία, όπως ονοματεπώνυμο, διεύθυνση ηλεκτρονικού ταχυδρομείου και επιθυμητό κωδικό. Οι καταχωρήσεις περνούν από server-side validation, ενώ ο κωδικός αποθηκεύεται με χρήση ασφαλούς αλγορίθμου κρυπτογράφησης (bcrypt).

Μετά την εγγραφή, ο χρήστης έχει τη δυνατότητα σύνδεσης στον λογαριασμό του μέσω της ενσωματωμένης υποδομής authentication του Laravel. Κατά την επιτυχή είσοδο, δημιουργείται session που ενεργοποιεί την προσωποποιημένη εμπειρία περιήγησης.

3.2 Παρουσίαση Υπηρεσιών και Πλοήγηση

Το σύστημα επιτρέπει την ελεύθερη περιήγηση στις παρεχόμενες υπηρεσίες, ακόμα και χωρίς λογαριασμό. Κάθε υπηρεσία συνοδεύεται από σύντομη περιγραφή, χρονική διάρκεια και τιμή. Η παρουσίαση υλοποιείται με δυναμική φόρτωση των δεδομένων από τη βάση, επιτρέποντας προσθήκες και αλλαγές μέσω του backend χωρίς επεμβάσεις στον κώδικα του frontend.

Το γραφικό περιβάλλον υλοποιείται με χρήση Blade templates και CSS για responsive σχεδιασμό. Ο επισκέπτης μπορεί εύκολα να εντοπίσει την επιθυμητή υπηρεσία χάρη στη σαφή τυπογραφία, τις λειτουργικές κατηγοριοποιήσεις και την ξεκάθαρη παρουσίαση των επιλογών.

3.3 Λειτουργία Κράτησης Ραντεβού

Μέσω της δυνατότητας κράτησης ραντεβού, ο πελάτης έχει τη δυνατότητα να διαμορφώσει πλήρως την επίσκεψή του. Μετά την αυθεντικοποίηση, ο χρήστης μπορεί να επιλέξει την επιθυμητή υπηρεσία ή συνδυασμό υπηρεσιών, καθώς και συγκεκριμένη ημερομηνία και ώρα.

Το σύστημα ελέγχει τη διαθεσιμότητα σε πραγματικό χρόνο, αποτρέποντας την καταχώρηση ραντεβού σε ήδη κατειλημμένες ώρες. Το backend επεξεργάζεται την αίτηση μέσω controller, ενώ η εγγραφή του ραντεβού πραγματοποιείται στον πίνακα

appointments της βάσης δεδομένων. Η υποδομή είναι σχεδιασμένη ώστε να μπορεί να υποστηρίξει επιπλέον χαρακτηριστικά στο μέλλον (π.χ. σχολιασμός, επιλογή κουρέα).

3.4 Παραγγελίες Προϊόντων και Καλάθι Αγορών

Επιπλέον των υπηρεσιών, η πλατφόρμα λειτουργεί και ως διαδικτυακό κατάστημα, στο οποίο προβάλλονται προϊόντα περιποίησης (π.χ. έλαια, aftershave, σαμπουάν). Ο χρήστης μπορεί να προσθέσει προϊόντα στο καλάθι του και να προχωρήσει στην παραγγελία.

Η αποθήκευση των επιλογών γίνεται προσωρινά μέσω session και οριστικοποιείται κατά το checkout, όπου δημιουργείται εγγραφή στον πίνακα orders και καταχωρούνται τα αντίστοιχα προϊόντα στον πίνακα order_items. Το backend διασφαλίζει τη συνέπεια των δεδομένων και την ορθή αντιστοίχιση κάθε παραγγελίας με τον πελάτη.

3.5 Προσωπικό Προφίλ και Ιστορικό Συναλλαγών

Η εφαρμογή παρέχει στον χρήστη πρόσβαση σε μια σελίδα προσωπικού προφίλ (dashboard), όπου παρουσιάζονται οι βασικές του πληροφορίες και πλήρες ιστορικό των ραντεβού και των αγορών του. Ο χρήστης μπορεί να διαχειριστεί τις ενέργειες αυτές, όπως ακύρωση ραντεβού ή επισκόπηση παλαιών παραγγελιών.

Η πρόσβαση σε αυτή την ενότητα είναι προστατευμένη με αυθεντικοποίηση, ενώ όλα τα δεδομένα που αφορούν τον πελάτη αποθηκεύονται με ασφαλή τρόπο, σύμφωνα με τα πρότυπα ασφάλειας του Laravel.

Συμπερασματικά, το σύστημα προσφέρει μια πληθώρα λειτουργιών οι οποίες είναι οργανωμένες με συνέπεια και σαφήνεια, δημιουργώντας ένα ολοκληρωμένο ψηφιακό περιβάλλον εξυπηρέτησης πελατών barbershop. Η χρήση των σύγχρονων τεχνολογιών διασφαλίζει τη σταθερότητα, ασφάλεια και επεκτασιμότητα της εφαρμογής.

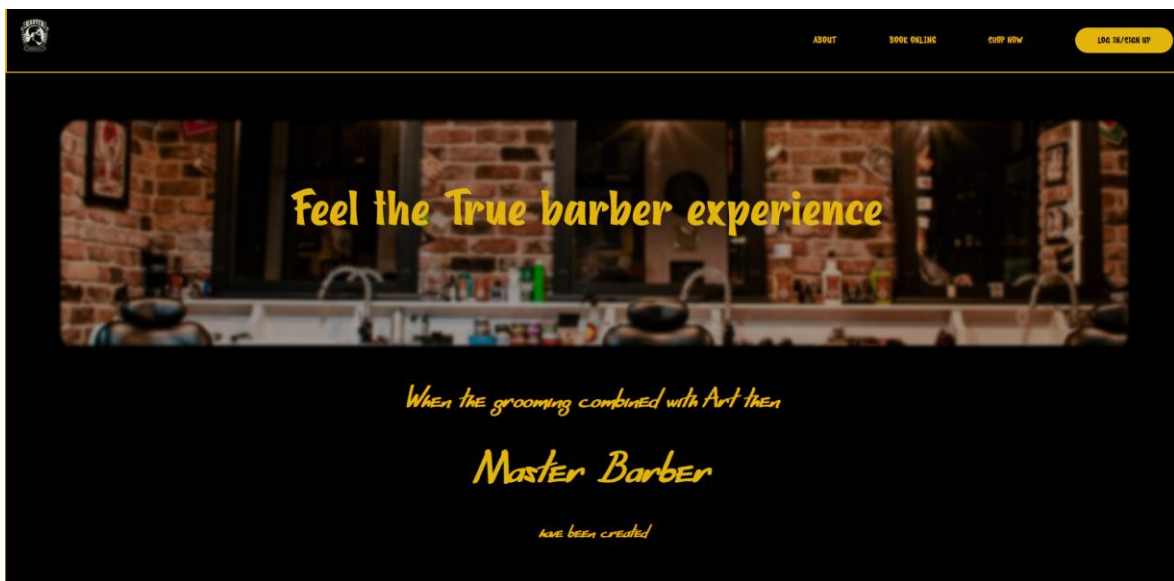
Κεφάλαιο 4 – Εμπειρία Χρήσης και Πλοήγηση στην Εφαρμογή

Η παρούσα ενότητα περιγράφει τη διαδικασία πλοήγησης ενός χρήστη μέσα στην εφαρμογή του barbershop, αναδεικνύοντας τις βασικές λειτουργίες της μέσω παραστατικών στιγμιότυπων. Η πλοήγηση ακολουθεί τη λογική ροή μιας πραγματικής αλληλεπίδρασης χρήστη, ξεκινώντας από την αρχική σελίδα και ολοκληρώνοντας είτε με την κράτηση ραντεβού είτε με την υποβολή ηλεκτρονικής παραγγελίας.

Η διεπαφή έχει σχεδιαστεί με έμφαση στη φιλικότητα προς τον τελικό χρήστη, με σύγχρονα αισθητικά χαρακτηριστικά, ευανάγνωστη τυπογραφία και responsive σχεδίαση που υποστηρίζει πλήρως φορητές και σταθερές συσκευές.

4.1 Αρχική Σελίδα – Home Page

Η αρχική οθόνη αποτελεί την πρώτη επαφή του επισκέπτη με την εφαρμογή. Περιλαμβάνει περιεκτική παρουσίαση της ταυτότητας του barbershop και εικαστικά στοιχεία που αποπνέουν επαγγελματισμό. Η μπάρα πλοήγησης στο άνω μέρος περιλαμβάνει συνδέσμους για τις βασικές ενότητες της εφαρμογής, όπως: «About», «Book Online», «Shop Now» και «Log In / Sign Up».



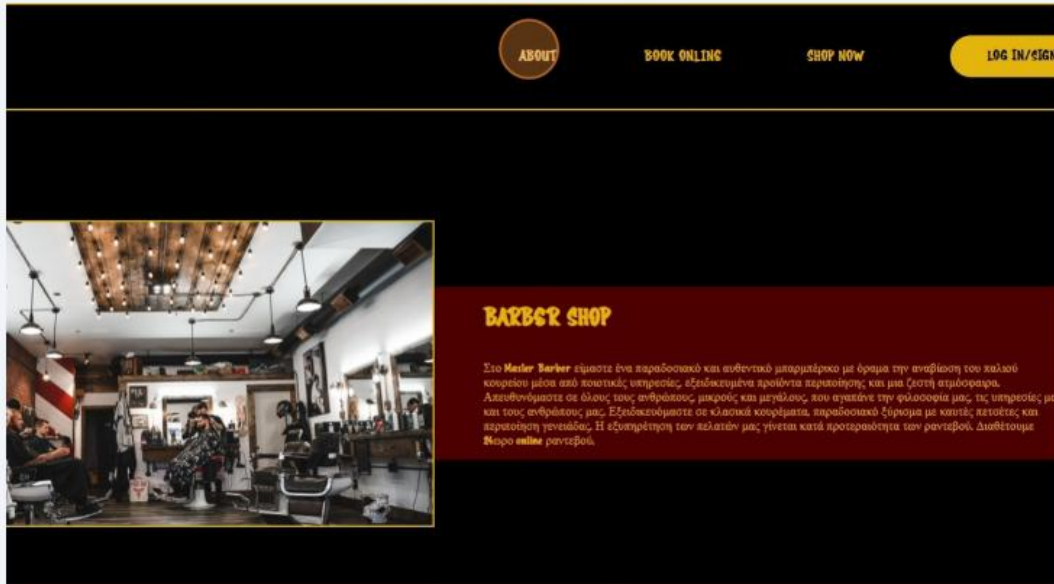
★ Εικόνα 4.1 – Αρχική Σελίδα με κύρια πλοήγηση

4.2 Πληροφορίες για την Επιχείρηση (About Page)”

Η ενότητα «About» προσφέρει πληροφορίες σχετικές με τη φιλοσοφία, το προσωπικό και τις υπηρεσίες του καταστήματος. Πρόκειται για μια στατική αλλά καλαίσθητα σχεδιασμένη σελίδα, που ενισχύει την εμπιστοσύνη του πελάτη παρουσιάζοντας μια προσωπική εικόνα του επαγγελματικού χώρου.

12 Click this button.

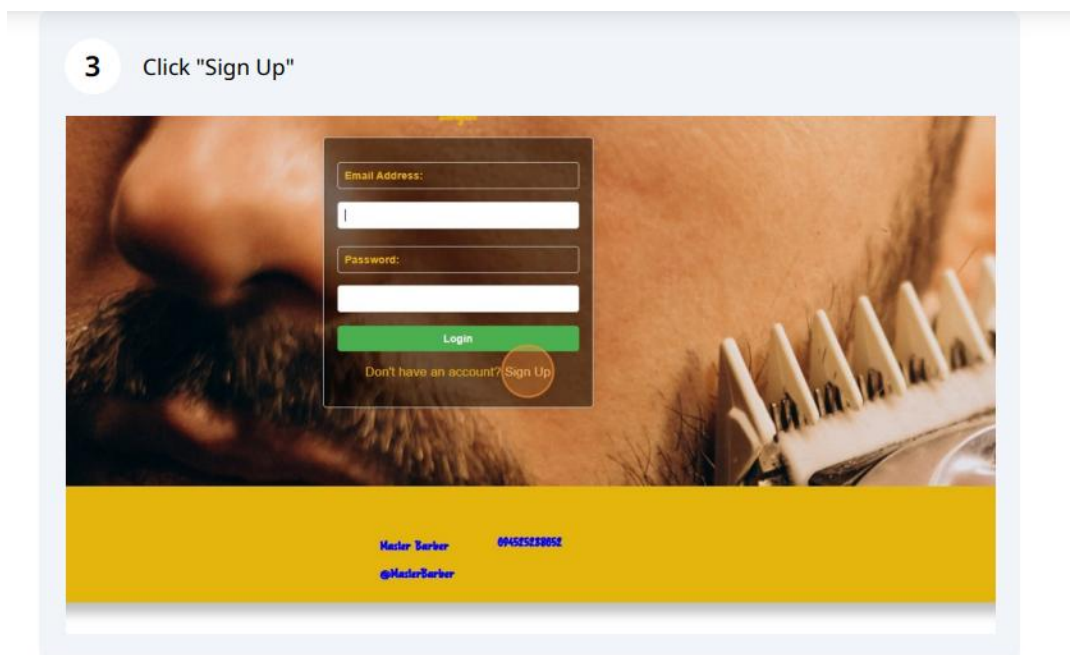
13 Click "ABOUT"



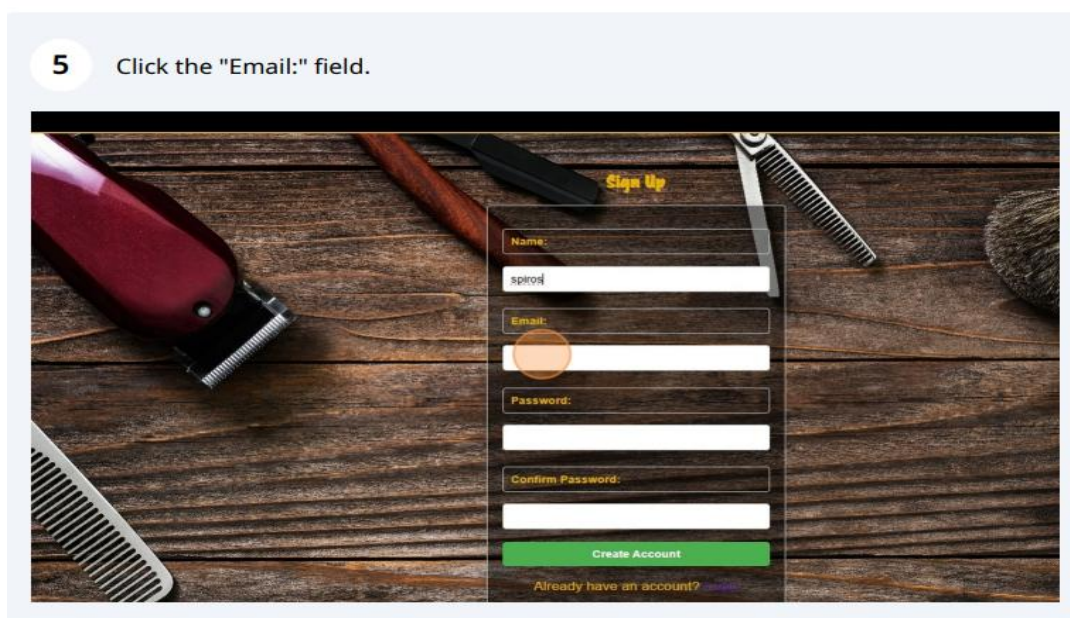
Εικόνα 4.2 – Σελίδα παρουσίασης του κουρείου με γενικές πληροφορίες

4.3 Σελίδες Σύνδεσης και Εγγραφής

Η λειτουργία ταυτοποίησης περιλαμβάνει δύο βασικές σελίδες: αυτή της εγγραφής (Sign Up) και της σύνδεσης (Log In). Ο σχεδιασμός είναι απλός και εργονομικός, με πεδία εισαγωγής email και κωδικού πρόσβασης, καθώς και σύνδεσμο για μετάβαση μεταξύ των δύο φορμών. Η προσβασιμότητα ενισχύεται με αυτόματη ανακατεύθυνση στις φόρμες, όταν ο χρήστης προσπαθεί να πραγματοποιήσει λειτουργία που απαιτεί αυθεντικοποίηση.



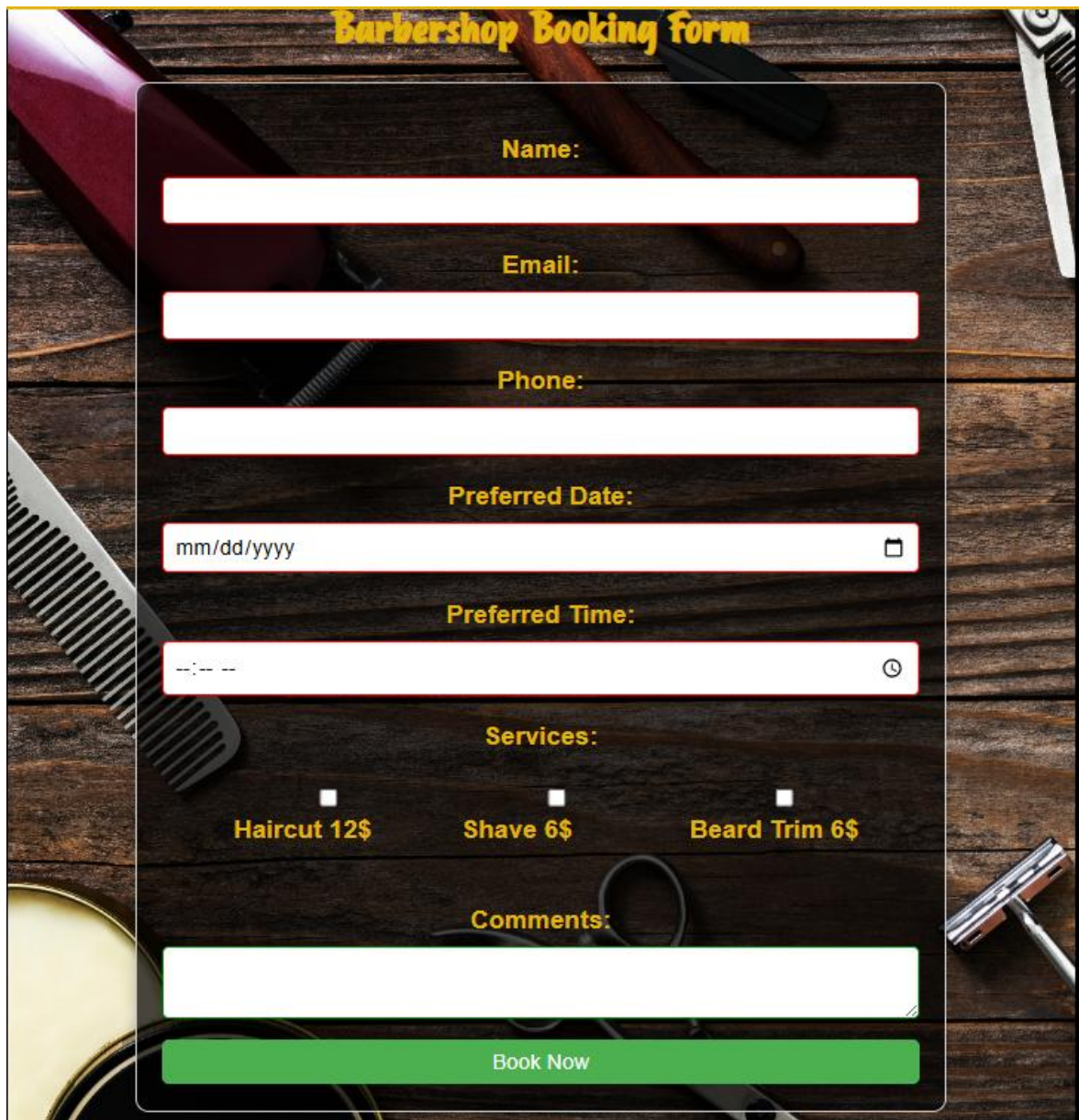
Εικόνα 4.3 – Οθόνη σύνδεσης χρήστη



Εικόνα 4.4 – Οθόνη εγγραφής νέου χρήστη

4.4 Επιλογή Υπηρεσιών και Ραντεβού

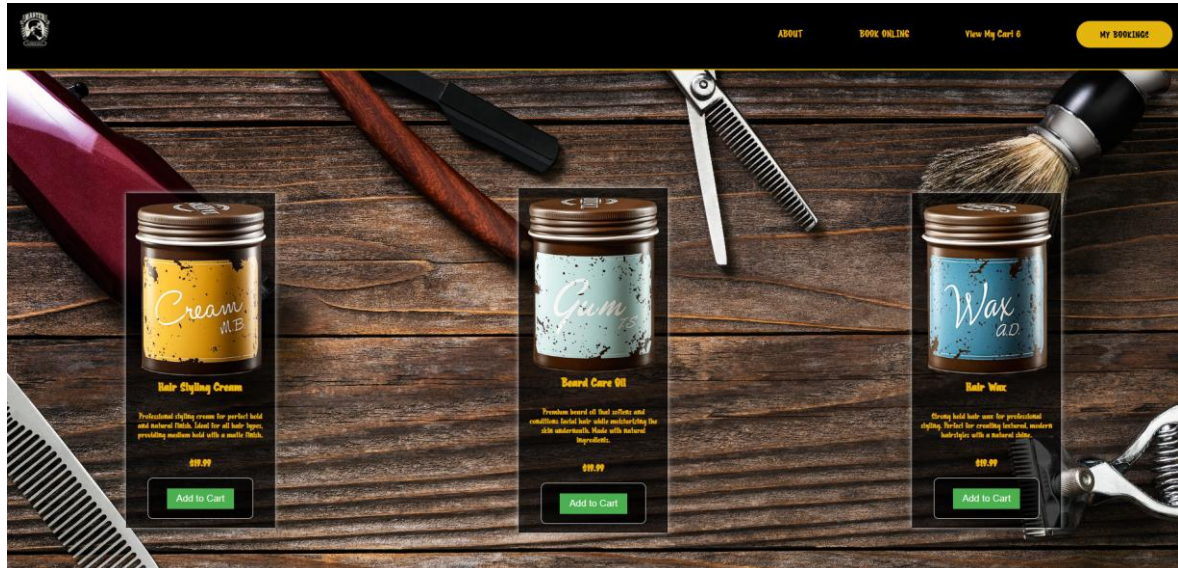
Μέσα από το μενού «Book Online», οι χρήστες μπορούν να επιλέξουν από μια λίστα διαθέσιμων υπηρεσιών περιποίησης. Κάθε καταχώριση περιλαμβάνει περιγραφή, τιμή και εκτιμώμενη διάρκεια. Η φόρμα κράτησης είναι ενσωματωμένη στη σελίδα και περιλαμβάνει πεδία επιλογής ημέρας και ώρας. Όλες οι καταχωρήσεις επαληθεύονται από το backend μέσω validation του Laravel.

A screenshot of a web form titled "Barbershop Booking Form" overlaid on a background image of barbershop tools like a hairbrush, scissors, and a hair dryer. The form contains several input fields: "Name:", "Email:", "Phone:", "Preferred Date:" (with a date picker icon), and "Preferred Time:" (with a time picker icon). Below these is a "Services:" section with three radio button options: "Haircut 12\$", "Shave 6\$", and "Beard Trim 6\$". At the bottom is a "Comments:" text area and a green "Book Now" button.

Εικόνα 4.5 – Ενότητα κράτησης με επιλογή υπηρεσιών και χρονικού διαστήματος

4.5 Ηλεκτρονικό Κατάστημα (Shop)

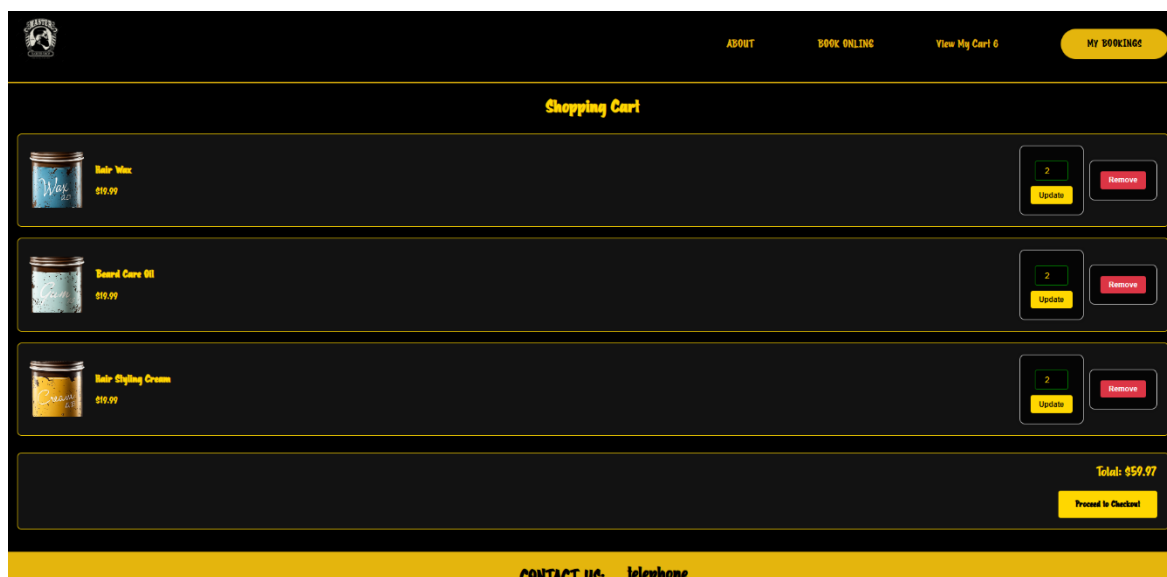
Η σελίδα του ηλεκτρονικού καταστήματος προβάλλει όλα τα προϊόντα που διατίθενται προς πώληση, όπως έλαια, σαμπουάν και είδη περιποίησης. Κάθε προϊόν περιλαμβάνει εικόνα, τιμή και επιλογή ποσότητας. Οι χρήστες μπορούν να προσθέσουν προϊόντα στο καλάθι τους με ένα κλικ.



★ Εικόνα 4.6 – Κατάστημα με δυνατότητα προσθήκης στο καλάθι

4.6 Καλάθι Αγορών (Cart Page)

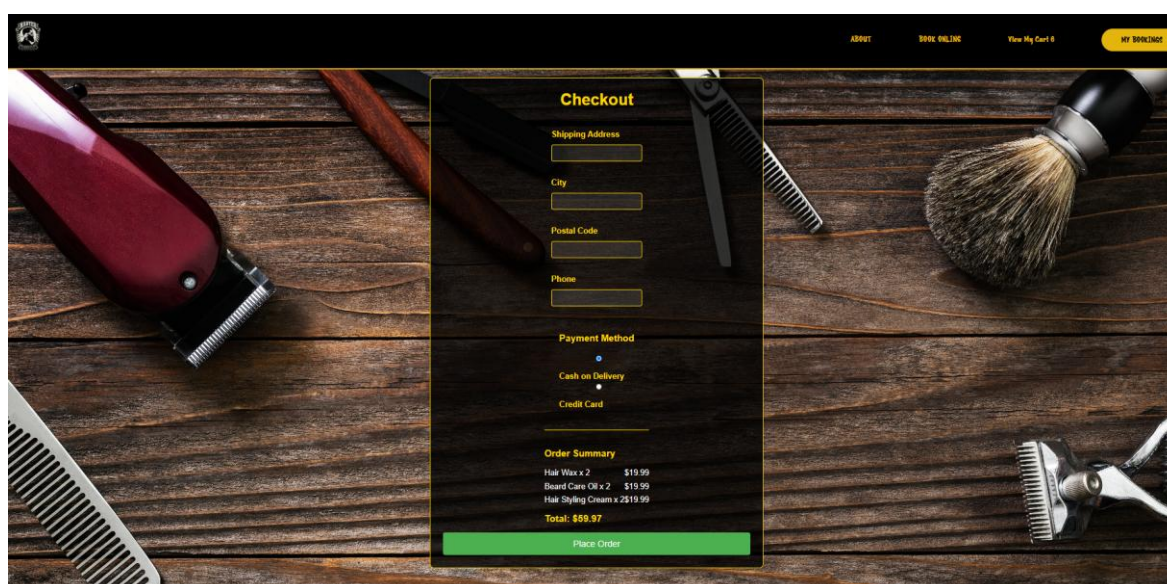
Η σελίδα καλαθιού εμφανίζει συγκεντρωτικά τα επιλεγμένα προϊόντα του χρήστη. Παρέχονται λειτουργίες επεξεργασίας, αφαίρεσης και ενημέρωσης ποσοτήτων, ενώ το σύστημα υπολογίζει αυτόματα το συνολικό κόστος. Ο σχεδιασμός είναι ευανάγνωστος και προσαρμοσμένος για χρήση σε κινητές συσκευές.



✦ Εικόνα 4.7 – Περιεχόμενα καλαθιού αγορών

4.7 Σελίδα Ολοκλήρωσης Παραγγελίας (Checkout)

Η φόρμα checkout καθοδηγεί τον χρήστη για την καταχώριση στοιχείων αποστολής ή πληρωμής. Περιλαμβάνει αναλυτική περίληψη της παραγγελίας, καθώς και το κουμπί επιβεβαίωσης “Place Order”. Η διάταξη των στοιχείων ενισχύει την αίσθηση αξιοπιστίας και επαγγελματισμού, με χρήση αισθητικά συνεπούς σχεδιασμού.



Εικόνα 4.8 – Τελική οθόνη υποβολής παραγγελίας με σύνοψη καλαθιού

Συνολική Εκτίμηση της Εμπειρίας Χρήσης

Η πλοήγηση στην εφαρμογή έχει σχεδιαστεί έτσι ώστε να διευκολύνει τον χρήστη σε κάθε βήμα, από την αρχική πλοήγηση έως την ολοκλήρωση υπηρεσίας ή αγοράς. Η απλότητα του σχεδιασμού, σε συνδυασμό με την ευελιξία και τη λειτουργικότητα, δημιουργούν μια θετική και ομαλή εμπειρία. Η αισθητική της εφαρμογής – με κυρίαρχα χρώματα το μαύρο και το χρυσό – αποπνέει επαγγελματισμό και πολυτέλεια, ενισχύοντας την εικόνα της επιχείρησης στον ψηφιακό χώρο.

Κεφάλαιο 5 – Τεχνική Υλοποίηση και Εσωτερική Δομή της Εφαρμογής

5.1 Εισαγωγή

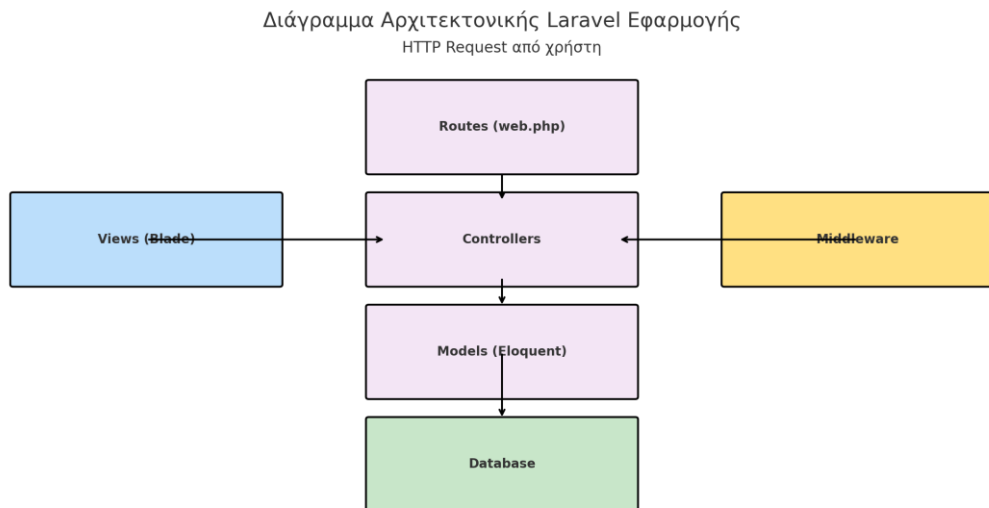
Το παρόν κεφάλαιο αναλύει την τεχνική πλευρά της υλοποίησης της διαδικτυακής πλατφόρμας για barbershop, η οποία αναπτύχθηκε αξιοποιώντας το Laravel framework. Η παρουσίαση εστιάζει στη συνολική αρχιτεκτονική, στη ροή πλοήγησης μεταξύ των διαφόρων λειτουργιών, στη δομή της βάσης δεδομένων και στα τεχνολογικά εργαλεία που χρησιμοποιήθηκαν κατά την ανάπτυξη. Σκοπός είναι να δοθεί μια πλήρης εικόνα του τρόπου λειτουργίας της εφαρμογής πίσω από το γραφικό περιβάλλον, ώστε να καταστεί σαφής η τεχνική υποδομή και η δυνατότητα μελλοντικής συντήρησης ή επέκτασης.

5.2 Αρχιτεκτονική της Laravel Εφαρμογής

Η δομή του συστήματος είναι βασισμένη στο πρότυπο Model-View-Controller (MVC), που ενσωματώνεται πλήρως στον πυρήνα του Laravel. Το πλαίσιο παρέχει προηγμένες δυνατότητες όπως routing, middleware για ασφάλεια, Blade για διαχείριση εμφάνισης, και Eloquent ORM για αλληλεπίδραση με τη βάση δεδομένων.

Η βασική ροή των αιτημάτων υλοποιείται ως εξής:

- Ο χρήστης στέλνει ένα αίτημα μέσω της διεπαφής (browser).
- Η διαδρομή του αιτήματος ορίζεται στο αρχείο routes/web.php.
- Ο αντίστοιχος controller αναλαμβάνει να επεξεργαστεί το αίτημα.
- Ο controller καλεί Eloquent μοντέλα, εάν απαιτείται πρόσβαση σε δεδομένα.
- Η απάντηση διαμορφώνεται ως Blade view και αποστέλλεται στον χρήστη.

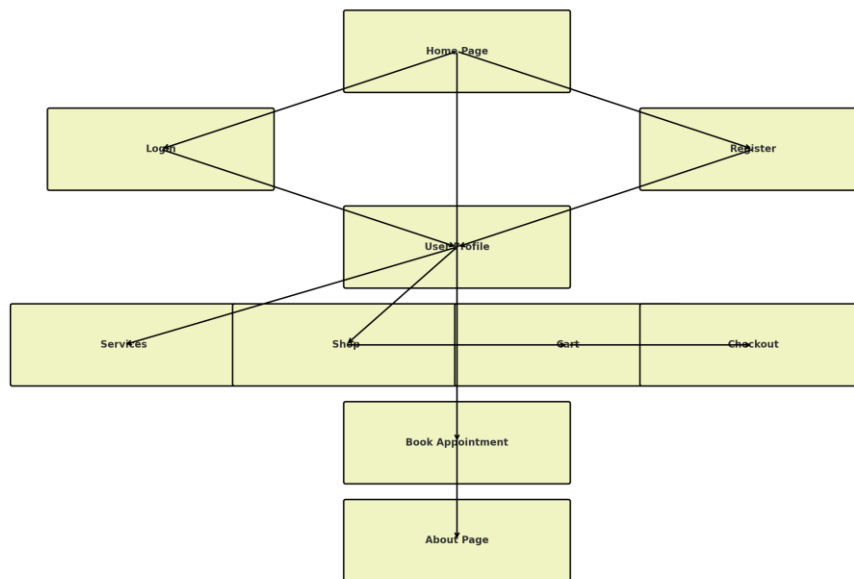


Εικόνα 5.1 – Διάγραμμα Αρχιτεκτονικής Laravel

5.3 Δομή Πλοήγησης Χρήστη – UI Navigation Flow

Η διεπαφή του χρήστη έχει οργανωθεί έτσι ώστε να διευκολύνει την πρόσβαση σε βασικές λειτουργίες, όπως η εγγραφή, η σύνδεση, η κράτηση ραντεβού, η περιήγηση στα διαθέσιμα προϊόντα και η διαχείριση προφίλ. Το παρακάτω διάγραμμα UI Flow αποτυπώνει τις βασικές ροές μετάβασης μεταξύ των οθονών, με βάση την τυπική εμπειρία χρήστη.

User Interface Flow Diagram – Barbershop Web Application



Εικόνα 5.2 – Διάγραμμα Πλοήγησης Οθονών

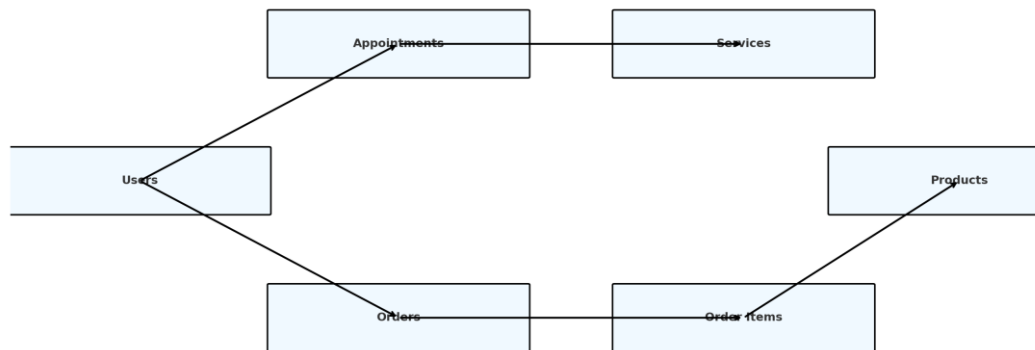
5.4 Δομή και Σχεδίαση Βάσης Δεδομένων

Η βάση δεδομένων σχεδιάστηκε με γνώμονα την κανονικοποίηση και την αποδοτική αποθήκευση δεδομένων. Περιλαμβάνει πίνακες όπως:

- users: αποθηκεύει στοιχεία χρηστών.
- appointments: κρατήσεις ραντεβού.
- services: διαθέσιμες υπηρεσίες.
- products: προϊόντα προς πώληση.
- orders: καταγεγραμμένες παραγγελίες.
- order_items: επιμέρους προϊόντα ανά παραγγελία.

Οι σχέσεις μεταξύ των πινάκων διαμορφώνονται μέσω ξένων κλειδιών (foreign keys), ενώ το Eloquent ORM διαχειρίζεται τις συσχετίσεις σε επίπεδο κώδικα.

ER Diagram - Σχεσιακή Βάση Δεδομένων

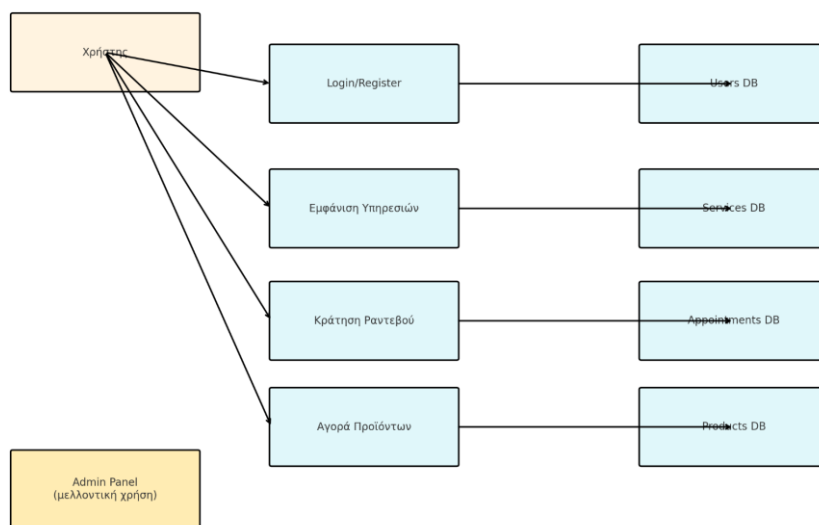


Εικόνα 5.3 – Διάγραμμα Οντοτήτων-Συσχετίσεων (ER)

5.5 Διάγραμμα Ροής Πληροφοριών (DFD)

Το διάγραμμα ροής δεδομένων παρουσιάζει τις βασικές λειτουργικές αλληλεπιδράσεις της εφαρμογής. Ο τελικός χρήστης επικοινωνεί με λειτουργικά modules (όπως login, επιλογή υπηρεσίας, κράτηση ραντεβού, προσθήκη προϊόντων στο καλάθι), τα οποία με τη σειρά τους αλληλεπιδρούν με τα αντίστοιχα δεδομένα που αποθηκεύονται σε μόνιμη μορφή (π.χ. πίνακες βάσης).

Data Flow Diagram (DFD) – Barbershop Web Application



Εικόνα 5.4 – Διάγραμμα Ροής Πληροφορίας

Κεφάλαιο 6 – Σύνοψη και Συμπεράσματα

6.1 Ανακεφαλαίωση Αντικειμένου και Υλοποίησης

Η παρούσα εργασία επικεντρώθηκε στην ανάλυση, σχεδίαση και ανάπτυξη μιας διαδικτυακής εφαρμογής που προορίζεται να υποστηρίξει τις ανάγκες ενός σύγχρονου barbershop. Η εφαρμογή αποσκοπεί στην ενίσχυση της επιχειρησιακής λειτουργίας μέσω ψηφιακών εργαλείων, προσφέροντας στους τελικούς χρήστες τη δυνατότητα ηλεκτρονικών κρατήσεων, αγορών και παρακολούθησης του προσωπικού τους ιστορικού.

Η επιλογή του Laravel ως βασικό τεχνολογικό υπόβαθρο αποτέλεσε στρατηγική απόφαση που διασφάλισε την οργανωμένη ανάπτυξη, την εύκολη διαχείριση και την υψηλή επεκτασιμότητα. Η προσέγγιση MVC, σε συνδυασμό με τις τεχνολογίες MySQL, Blade και HTML/CSS/JavaScript, συνέβαλε στην υλοποίηση ενός ευέλικτου και φιλικού προς τον χρήστη συστήματος.

Η ανάπτυξη της εφαρμογής προχώρησε σύμφωνα με σταδιακά βήματα, τα οποία περιλάμβαναν:

- Τον εντοπισμό και την καταγραφή λειτουργικών και μη λειτουργικών απαιτήσεων.
- Τον καθορισμό της αρχιτεκτονικής υποδομής, ακολουθώντας αρχές modular σχεδίασης.
- Τη δημιουργία μιας βελτιστοποιημένης σχεσιακής βάσης δεδομένων, σύμφωνα με τις αρχές της κανονικοποίησης και της ασφάλειας.
- Την ανάπτυξη πλήρως λειτουργικής διεπαφής, περιλαμβάνοντας εγγραφή/σύνδεση, σύστημα ραντεβού και e-shop.
- Την τεκμηρίωση των λειτουργιών με χρήση διαγραμμάτων αρχιτεκτονικής, ροής δεδομένων και οντοτήτων.

Η υλοποίηση αυτής της εφαρμογής δεν περιορίστηκε απλώς στην τεχνική διάσταση της ανάπτυξης λογισμικού, αλλά συνδύασε και τη μελέτη αναγκών του χρήστη, την ανάλυση διαδικασιών και την προσέγγιση με γνώμονα τη χρηστικότητα.

6.2 Συμπερασματική Εκτίμηση

Η εργασία ανέδειξε τις πολλαπλές δυνατότητες που προσφέρει η σύγχρονη τεχνολογία σε μικρές και μεσαίες επιχειρήσεις. Μέσα από τη δημιουργία μιας ολοκληρωμένης web εφαρμογής αποδείχθηκε ότι η ψηφιοποίηση υπηρεσιών μπορεί να βελτιώσει θεαματικά την οργάνωση, την προσβασιμότητα και την εμπειρία πελάτη.

Το Laravel framework επιβεβαίωσε την καταλληλότητά του, προσφέροντας ένα πλούσιο περιβάλλον ανάπτυξης, σταθερότητα στον χειρισμό αιτημάτων, και υψηλό επίπεδο ασφάλειας. Οι τεχνολογίες PHP, JavaScript και CSS συνεργάστηκαν αρμονικά για τη δημιουργία ενός πλήρως λειτουργικού frontend και backend περιβάλλοντος.

Αξίζει να επισημανθεί ότι η υλοποιημένη πλατφόρμα έχει σχεδιαστεί έτσι ώστε να μπορεί να εμπλουτιστεί στο μέλλον με επιπλέον λειτουργικότητες, όπως ενσωμάτωση ηλεκτρονικών πληρωμών, πίνακα διαχειριστή, σύστημα ειδοποιήσεων ή ακόμα και mobile εφαρμογή. Επιπλέον, η χρήση εργαλείων ανοιχτού λογισμικού κατέστησε την ανάπτυξη οικονομικά προσιτή χωρίς συμβιβασμούς στην ποιότητα.

Συνολικά, η πτυχιακή αυτή εργασία απέδειξε ότι μια καλά σχεδιασμένη και τεχνικά θεμελιωμένη διαδικτυακή εφαρμογή μπορεί να μεταμορφώσει την καθημερινή λειτουργία ενός barbershop. Παράλληλα, αποτέλεσε ένα πολύτιμο εκπαιδευτικό εργαλείο για τον φοιτητή, ενισχύοντας τις δεξιότητές του σε ανάλυση απαιτήσεων, αρχιτεκτονικό σχεδιασμό, ανάπτυξη λογισμικού και τεκμηρίωση εφαρμογών.

Βιβλιογραφία

1. Laracasts. (2023). Laravel: The PHP Framework For Web Artisans. Διαθέσιμο στο: <https://laravel.com/> (Τελευταία πρόσβαση: 10 Ιουνίου 2025).
2. Otwell, T. (2021). Laravel: Up and Running – A Framework for Building Modern PHP Apps. 2η έκδ. Sebastopol: O'Reilly Media.
3. Ullman, L. (2019). PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide. 5η έκδ. Berkeley, CA: Peachpit Press.
4. Welling, L. & Thomson, L. (2017). PHP and MySQL Web Development. 5η έκδ. Addison-Wesley.
5. Freeman, E. & Robson, E. (2020). Head First HTML and CSS. 2η έκδ. Sebastopol: O'Reilly Media.
6. Nielsen, J. (1999). Designing Web Usability: The Practice of Simplicity. Indianapolis: New Riders Publishing.
7. Krug, S. (2014). Don't Make Me Think: A Common Sense Approach to Web Usability. 3η έκδ. New Riders.
8. MySQL Documentation. (2024). MySQL 8.0 Reference Manual. Oracle Corporation. Διαθέσιμο στο: <https://dev.mysql.com/doc/> (Τελευταία πρόσβαση: 12 Ιουνίου 2025).
9. W3Schools. (2025). PHP Tutorial, HTML, CSS, JavaScript. Διαθέσιμο στο: <https://www.w3schools.com/> (Τελευταία πρόσβαση: 10 Ιουνίου 2025).
10. ISO/IEC 25010:2011. (2011). Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models. Διεθνής Οργανισμός Τυποποίησης.