



Τμήμα Ηλεκτρολόγων Μηχανικών και
Μηχανικών Υπολογιστών

Πτυχιακή Εργασία

**«Σχεδιασμός και ανάπτυξη συστήματος
παρακολούθησης φορέων και θέσεων
πρακτικής άσκησης»**

Γκουρβέλος Σπυρίδων

Αριθμός Μητρώου 2995

Επιβλέπων καθηγητής: Χριστοδούλου Σωτήριος

Επιτροπή αξιολόγησης: Τζήμας Ιωάννης

Παρασκευάς Μιχαήλ

Πάτρα, 2022

ΠΕΡΙΛΗΨΗ

Τις τελευταίες δεκαετίες, ιδιαίτερα όμως τα τελευταία χρόνια, το Διαδίκτυο έχει καταστεί ένα απαραίτητο εργαλείο και αναπόσπαστο κομμάτι της καθημερινότητάς μας. Η χρήση του, αδιαμφισβήτητα, προσφέρει ποικίλα οφέλη σε όσους είναι πρόθυμοι να το χρησιμοποιήσουν με κατάλληλο τρόπο. Είναι κοινώς αποδεκτό το γεγονός ότι το Διαδίκτυο αποτελεί ένα πολύτιμο εργαλείο για την άντληση πληροφοριών καθιστώντας το ίσως την ισχυρότερη πηγή πληροφορίας στον κόσμο. Αντικείμενο της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη μιας διαδικτυακής εφαρμογής καίριας σημασίας για τους φοιτητές του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου, καθώς θα τους επιτρέψει να έχουν μια σφαιρική εικόνα τόσο για τους φορείς στους οποίους δύνανται να πραγματοποιήσουν την πρακτική τους άσκηση, όσο και για τα αντικείμενα ενασχόλησης σε τέτοιες θέσεις. Στόχος του συστήματος παρακολούθησης φορέων και θέσεων πρακτικής άσκησης, είναι να λειτουργήσει ως μία χρήσιμη πηγή πληροφορίας για τους φοιτητές του τμήματος, κάνοντας πιο εύκολη την έρευνά τους για την εύρεση μίας επερχόμενης θέσης πρακτικής άσκησης, μια διαδικασία που μπορεί να αποδειχθεί αρκετά χρονοβόρα και επίπονη χωρίς τη χρήση κατάλληλων μέσων. Για την υλοποίηση του συστήματος έχουν χρησιμοποιηθεί τεχνολογίες αιχμής όπως το PHP web framework Laravel, η JavaScript βιβλιοθήκη React JS, αλλά και η βάση δεδομένων MariaDB. Τέλος για την εμφάνιση των φορέων πρακτικής άσκησης σε χάρτη γίνεται χρήση της JavaScript βιβλιοθήκης Leaflet, αντλώντας γεωγραφικά δεδομένα από το OpenStreetMap project.

Λέξεις κλειδιά: Πρακτική Άσκηση, React JS, Laravel, PHP, Bootstrap, MySQL, MariaDB, Leaflet, OpenStreetMap

ABSTRACT

In recent decades, but especially in recent years, the Internet has become an indispensable tool and integral part of our daily lives. Its use undoubtedly offers a variety of benefits to those who are willing to use it in an appropriate manner. It is widely accepted that the Internet is a valuable tool for obtaining information, making it perhaps the most powerful source of information in the world. The objective of this thesis is the development of a web application of key importance for the students of the Department of Electrical and Computer Engineering of the University of Peloponnese, as it will allow them to have a comprehensive overview of both the institutions where they can carry out their internship as well as the subjects of employment in such positions. The aim of the system for monitoring institutions and internships is to serve as a useful source of information for the students of the department, making it easier for them to search for an upcoming internship, a process that can prove to be quite time-consuming and laborious without the use of appropriate tools. For the implementation of the system, cutting-edge technologies such as the PHP web framework Laravel, the JavaScript library React JS, and the MariaDB database have been utilized. Last but not least, the Leaflet JavaScript library is used to display the internship institutions on a map, pulling geographic data from the OpenStreetMap project.

Keywords: Internship, React JS, Laravel, PHP, Bootstrap, MySQL, MariaDB, Leaflet, OpenStreetMap

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά, θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή της πτυχιακής μου εργασίας, κ. Χριστοδούλου Σωτήριο, για την καθοδήγηση που μου προσέφερε, την εποικοδομητική κριτική του και το αμέριστο ενδιαφέρον του καθ' όλη την διάρκεια της εκπόνησης αυτής της εργασίας αλλά και συνολικά της φοιτητικής μου πορείας.

Επίσης, θα ήθελα να ευχαριστήσω όλους τους καθηγητές του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου για τη συμβολή τους στην τεχνολογική μου πορεία στα χρόνια της φοίτησής μου στο τμήμα.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου, τους συμφοιτητές και τους φίλους μου για την στήριξή τους.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ.....	1
ABSTRACT.....	2
ΕΥΧΑΡΙΣΤΙΕΣ.....	3
ΠΕΡΙΕΧΟΜΕΝΑ	4
ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ	6
ΕΙΣΑΓΩΓΗ.....	7
ΚΕΦΑΛΑΙΟ 1 - ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ	8
1.1 Τεχνολογίες Back End	8
1.1.1 PHP	8
1.1.2 Laravel	10
1.1.3 MySQL / MariaDB.....	12
1.1.4 REST API	13
1.2 Τεχνολογίες Front End	14
1.2.1 HTML	14
1.2.2 CSS.....	16
1.2.2.1 Sass.....	17
1.2.3 JavaScript	18
1.2.4 Bootstrap.....	21
1.2.5 React	22
1.3 Βιβλιοθήκες.....	24
1.3.1 Axios.....	24
1.3.2 Syncfusion Grid	25
1.3.3 Leaflet	26
1.3.4 Turnstone	27
1.3.5 React Hot Toast.....	28
ΚΕΦΑΛΑΙΟ 2 - ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ.....	29
2.1 Βάση Δεδομένων	29
2.1.1 Πίνακες τμημάτων	30
2.1.2 Πίνακες φορέων.....	31
2.1.3 Πίνακας πρακτικής άσκησης.....	32
2.2 Κατηγορίες Χρηστών.....	33
2.3 Εφαρμογή τεχνολογιών	34
ΚΕΦΑΛΑΙΟ 3 - ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ / ΣΕΝΑΡΙΑ ΧΡΗΣΗΣ	35
3.1 Επισκέπτης - Φοιτητής.....	35

3.1.1 Πλοήγηση στην εφαρμογή	35
3.1.2 Αναζήτηση φορέα ή θέσης πρακτικής άσκησης.....	36
3.1.3 Προβολή λεπτομερειών φορέα και θέσεων πρακτικής άσκησης	38
3.2 Διαχειριστής - Γραμματεία / Γραφείο πρακτικής άσκησης	39
3.2.1 Πλοήγηση στην εφαρμογή	39
3.2.2 Διαχείριση φορέων	40
3.2.3 Προσθήκη/Επεξεργασία φορέα	41
3.2.4 Διαχείριση θέσεων πρακτικής άσκησης	44
3.2.5 Προσθήκη/Επεξεργασία θέσης πρακτικής άσκησης.....	45
ΕΠΙΛΟΓΟΣ	46
ΒΙΒΛΙΟΓΡΑΦΙΑ / ΔΙΚΤΥΟΓΡΑΦΙΑ	47

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1 - Παράδειγμα ενσωμάτωσης PHP σε αρχείο HTML.....	9
Εικόνα 2 - Απεικόνιση των αλληλεπιδράσεων στο μοντέλο MVC.....	10
Εικόνα 3 - Παράδειγμα χρήσης του Eloquent για άντληση δεδομένων από βάση δεδομένων	11
Εικόνα 4 - Στιγμιότυπο της γραμμής εντολών της MySQL	12
Εικόνα 5 - Απεικόνιση τρόπου λειτουργίας ενός REST API	13
Εικόνα 6 - Βασικές ετικέτες HTML και η περιγραφή τους.....	14
Εικόνα 7 - Παράδειγμα περιεχομένου ενός αρχείου HTML.....	15
Εικόνα 8 - Ανατομία κανόνα CSS	16
Εικόνα 9 - Παράδειγμα σελίδας HTML με (αριστερά) και χωρίς (δεξιά) μορφοποίηση CSS	16
Εικόνα 10 - Παράδειγμα χρήσης μεταβλητών σε αρχείο SASS	17
Εικόνα 11 - Παράδειγμα ενσωμάτωσης κώδικα JavaScript στο Head μίας σελίδας HTML	19
Εικόνα 12 - Παράδειγμα ενσωμάτωσης κώδικα JavaScript στο Body μίας σελίδας HTML	19
Εικόνα 13 - Παράδειγμα ενσωμάτωσης κώδικα JavaScript σε σελίδα HTML από εξωτερικό αρχείο .	20
Εικόνα 14 - Παράδειγμα χρήσης Bootstrap.....	21
Εικόνα 15 - Παράδειγμα περιεχομένου ενός αρχείου React	22
Εικόνα 16 - Παράδειγμα εκτέλεσης αιτήματος GET με χρήση της βιβλιοθήκης Axios	24
Εικόνα 17 - Παράδειγμα απεικόνισης δεδομένων με τη βιβλιοθήκη Syncfusion Grid.....	25
Εικόνα 18 - Παράδειγμα δημιουργίας διαδραστικού χάρτη με τη βιβλιοθήκη Leaflet.....	26
Εικόνα 19 - Παράδειγμα αναζήτησης εγγραφών με χρήση του React component Turnstone	27
Εικόνα 20 - Παράδειγμα δημιουργίας ειδοποίησης με χρήση του component React-Hot-Toast.....	28
Εικόνα 21 - Διάγραμμα οντοτήτων και συσχετίσεων της βάσης δεδομένων	29
Εικόνα 22 - Η αρχική σελίδα της εφαρμογής από την οπτική του επισκέπτη	35
Εικόνα 23 - Η σελίδα της εφαρμογής «Χάρτης Φορέων»	36
Εικόνα 24 - Αναζήτηση φορέων και θέσεων πρακτικής άσκησης.....	36
Εικόνα 25 - Αναδυόμενο παράθυρο με πληροφορίες για το φορέα	37
Εικόνα 26 - Σελίδα φορέα με πληροφορίες και θέσεις πρακτικής άσκησης	38
Εικόνα 27 - Η σελίδα σύνδεσης για τους διαχειριστές.....	39
Εικόνα 28 - Η αρχική σελίδα της εφαρμογής από την οπτική του διαχειριστή	39
Εικόνα 29 - Η σελίδα διαχείρισης φορέων της εφαρμογής	40
Εικόνα 30 - Φόρμα επεξεργασίας φορέα	41
Εικόνα 31 - Έλεγχος για την ύπαρξη φορέα στην εφαρμογή	41
Εικόνα 32 - Εύρεση συντεταγμένων κατά την εισαγωγή διεύθυνσης φορέα	42
Εικόνα 33 - Εμφάνιση τοποθεσίας φορέα στο χάρτη	42
Εικόνα 34 - Αναζήτηση εναλλακτικής τοποθεσίας φορέα στο χάρτη	43
Εικόνα 35 - Μήνυμα τύπου «toast» μετά την αποθήκευση του φορέα	43
Εικόνα 36 - Η σελίδα διαχείρισης θέσεων πρακτικής άσκησης της εφαρμογής	44
Εικόνα 37 - Προβολή περιεχομένου μεγάλων πεδίων των πινάκων	44
Εικόνα 38 - Φόρμα επεξεργασίας θέσης πρακτικής άσκησης	45
Εικόνα 39 - Μήνυμα τύπου «toast» μετά την αποθήκευση θέσης πρακτικής άσκησης.....	45

ΕΙΣΑΓΩΓΗ

Η παρούσα πτυχιακή εργασία πραγματοποιείται στο πλαίσιο της φοίτησής μου στο τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου και έχει ως σκοπό να καλύψει ένα κενό στις παρεχόμενες υπηρεσίες του τμήματος προς τους φοιτητές, αυτό της εύκολης αναζήτησης πληροφοριών για φορείς στους οποίους μπορούν να πραγματοποιήσουν την πρακτική τους άσκηση.

Η σχεδίαση της εφαρμογής έχει γίνει με χρήση τεχνολογιών αιχμής όπως το PHP web framework Laravel, η JavaScript βιβλιοθήκη React JS, αλλά και η βάση δεδομένων MariaDB, τεχνολογίες που χρησιμοποιούνται κατά κόρον για την ανάπτυξη σύγχρονων εφαρμογών και υπηρεσιών διαδικτύου.

Η εφαρμογή έχει ως στόχο να παρέχει ένα πρακτικό περιβάλλον μέσω του οποίου οι φοιτητές θα μπορούν εύκολα να βρουν πληροφορίες για φορείς στους οποίους οι συμφοιτητές τους έχουν πραγματοποιήσει την πρακτική τους άσκηση, να αναζητήσουν φορείς και θέσεις πρακτικής άσκησης γύρω από το αντικείμενο που τους ενδιαφέρει, αλλά και να δουν λεπτομερείς πληροφορίες για όλες τις πρακτικές ασκήσεις που έχουν πραγματοποιήσει φοιτητές του τμήματος, κάτι αρκετά χρήσιμο για τη δική τους αναζήτηση θέσεων πρακτικής άσκησης.

Την ίδια στιγμή, έχει δημιουργηθεί ένα εύκολο στη χρήση περιβάλλον μέσα από το οποίο η γραμματεία του τμήματος ή το γραφείο πρακτικής άσκησης του ιδρύματος θα μπορεί να εισάγει πληροφορίες για φορείς και θέσεις πρακτικής άσκησης που έχουν πραγματοποιήσει οι φοιτητές.

Η εργασία αποτελείται από τρία βασικά μέρη. Στο πρώτο μέρος, παρουσιάζονται όλα τα εργαλεία που χρησιμοποιήθηκαν για τη σχεδίαση και ανάπτυξη του συστήματος. Στο δεύτερο μέρος, γίνεται αναφορά στον τρόπο υλοποίησης της εφαρμογής, ενώ στο τρίτο μέρος γίνεται η παρουσίασή της μέσα από κάποιες βασικές περιπτώσεις χρήσης και συνοδευόμενη από στιγμιότυπα και επεξήγηση των πρακτικών που ακολουθήθηκαν, αλλά και των αποφάσεων που πάρθηκαν κατά την ανάπτυξή της, με γνώμονα την εύκολη και ευχάριστη χρήση της από τους τελικούς χρήστες.

ΚΕΦΑΛΑΙΟ 1 - ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ

Στο κεφάλαιο αυτό, θα παρουσιαστούν οι τεχνολογίες που χρησιμοποιήθηκαν κατά την ανάπτυξη του συστήματος παρακολούθησης φορέων και θέσεων πρακτικής άσκησης. Αρχικά θα γίνει παρουσίαση των τεχνολογιών back-end (κεφάλαιο 1.1), ακολουθούμενες από τις τεχνολογίες front-end (κεφάλαιο 1.2) και θα ολοκληρωθεί με μια σύντομη αναφορά στις εξωτερικές βιβλιοθήκες JavaScript που χρησιμοποιήθηκαν κατά την ανάπτυξη του συστήματος (κεφάλαιο 1.3).

1.1 Τεχνολογίες Back End

1.1.1 PHP

Η PHP (αρκτικόλεξο του όρου PHP Hypertext Preprocessor), είναι μια γλώσσα προγραμματισμού ανοιχτού κώδικα που χρησιμοποιείται ευρέως για τη δημιουργία δικτυακών εφαρμογών. Αυτό που διαφοροποιεί την PHP από άλλες γλώσσες scripting όπως η JavaScript που εκτελείται στην πλευρά του πελάτη (client-side), είναι ότι ο κώδικάς της εκτελείται στον διακομιστή (server-side).

Δημιουργήθηκε το φθινόπωρο του 1994 από τον Rasmus Lerdorf και οι πρώτες μη κυκλοφορημένες εκδόσεις χρησιμοποιούνταν στην προσωπική του σελίδα για να παρακολουθεί ποιος έβλεπε το βιογραφικό του. Η πρώτη έκδοση που χρησιμοποιήθηκε από άλλους ήταν διαθέσιμη κάποια στιγμή στις αρχές του 1995 και ήταν γνωστή ως Personal Home Page Tools. Αποτελούνταν από μια πολύ απλοϊκή μηχανή ανάλυσης που καταλάβαινε μόνο μερικές ειδικές μακροεντολές και έναν αριθμό από βοηθητικά προγράμματα που ήταν σε ευρεία χρήση στις ιστοσελίδες εκείνης της εποχής.

Ο κώδικας της ανανεώθηκε πλήρως το 1996 και η έκδοση PHP/FI 2.0 ήταν γεγονός. Το FI προήλθε από ένα άλλο πρόγραμμα που είχε γράψει ο Rasmus, το οποίο ερμήνευε δεδομένα από φόρμες html. Αυτή η δεύτερης γενιάς υλοποίηση άρχισε να εξελίσσει πραγματικά την PHP από μια σουίτα εργαλείων σε μια αυτοτελή γλώσσα προγραμματισμού. Περιελάμβανε ενσωματωμένη υποστήριξη για βάσεις δεδομένων DBM, mSQL και Postgres95, cookies, υποστήριξη συναρτήσεων που ορίζονται από τον χρήστη και πολλά άλλα.

Η έκδοση 3.0 της PHP ήταν η πρώτη έκδοση που μοιάζει πολύ με την PHP όπως τη βλέπουμε σήμερα. Διαπιστώνοντας ότι η PHP/FI 2.0 εξακολουθούσε να είναι αναποτελεσματική και ότι δεν διέθετε τα χαρακτηριστικά που χρειαζόνταν για να αναπτύξουν μια εφαρμογή ηλεκτρονικού εμπορίου, ο Andi Gutmans και ο Zeev Suraski από το Τελ Αβίβ του Ισραήλ ξεκίνησαν μια ακόμη πλήρη επανεγγραφή του αναλυτή της PHP/FI το 1997. Αυτή η εντελώς νέα γλώσσα κυκλοφόρησε σε συνεργασία με τον Rasmus υπό νέα ονομασία. Μετονομάστηκε απλά σε 'PHP', με την έννοια να γίνεται ένα αναδρομικό ακρωνύμιο - PHP: Hypertext Preprocessor.

Ένα από τα μεγαλύτερα πλεονεκτήματα της PHP 3.0 ήταν τα ισχυρά χαρακτηριστικά επεκτασιμότητάς της. Εκτός από την παροχή στους τελικούς χρήστες μιας ώριμης διεπαφής για πολλαπλές βάσεις δεδομένων, πρωτόκολλα και APIs, η ευκολία επέκτασης της ίδιας της γλώσσας προσέλκυσε δεκάδες προγραμματιστές. Άλλα βασικά χαρακτηριστικά που εισήχθησαν στην PHP 3.0 περιλάμβαναν την υποστήριξη αντικειμενοστραφούς προγραμματισμού και ένα πολύ πιο ισχυρό και συνεπές συντακτικό της γλώσσας.

Τη στιγμή που γράφεται η παρούσα εργασία, η PHP βρίσκεται στην έκδοση 8.1. Σε σχέση με τις προηγούμενες εκδόσεις φέρνει πολλά νέα χαρακτηριστικά και βελτιστοποιήσεις όπως υποστήριξη χαρακτηριστικών και ονομαστικών ορισμάτων, αυστηρότερο σύστημα τύπων και υποστήριξη

μεταγλωττιστή Just-In-Time (JIT). Σύμφωνα με την ιστοσελίδα W3Techs, η οποία παρέχει πληροφορίες σχετικά με τη χρήση διαφόρων τύπων τεχνολογιών στο διαδίκτυο, η PHP χρησιμοποιείται από περίπου το 77% όλων των ιστοσελίδων των οποίων γνωρίζουμε τη γλώσσα προγραμματισμού από την πλευρά του διακομιστή.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      PHP Example
    </title>
  </head>
  <body>
    <?php
      echo "Hi, I'm a PHP script!";
    ?>
  </body>
</html>
```

ΕΙΚΟΝΑ 1 - ΠΑΡΑΔΕΙΓΜΑ ΕΝΣΩΜΑΤΩΣΗΣ PHP ΣΕ ΑΡΧΕΙΟ HTML

Καθώς η PHP είναι μια server-side scripting γλώσσα προγραμματισμού, έχει τη δυνατότητα να συλλέγει δεδομένα, να παράγει δυναμικό περιεχόμενο αλλά και να επεξεργάζεται sessions και cookies. Υπάρχουν τρεις διαφορετικές περιπτώσεις στις οποίες χρησιμοποιείται ένα PHP script:

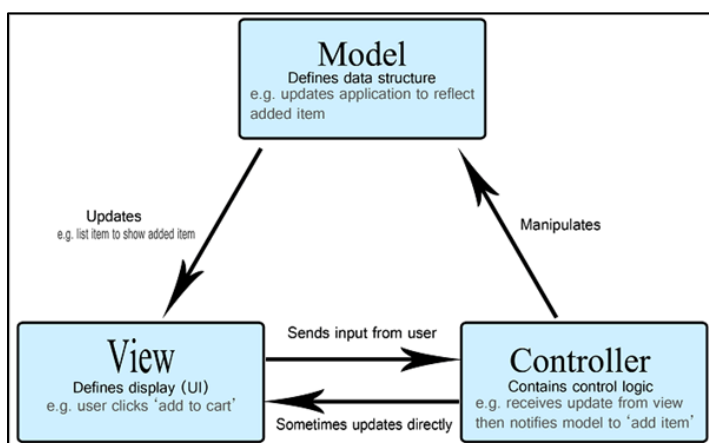
1. **Server-side scripting:** Αυτό είναι το πιο παραδοσιακό και κύριο πεδίο εφαρμογής της PHP και χρειάζεται τρία πράγματα για να λειτουργήσει: τον αναλυτή PHP (CGI ή server module), έναν διακομιστή ιστού (web server) και ένα πρόγραμμα περιήγησης ιστού (web browser). Αρχικά πρέπει να ξεκινήσει ο διακομιστής ιστού και να γίνει σύνδεση με μία εγκατάσταση PHP. Για να εμφανιστεί η έξοδος του προγράμματος PHP αρκεί μόνο η πρόσβαση σε ένα πρόγραμμα περιήγησης ιστού, που θα προβάλλει τη σελίδα PHP μέσω του διακομιστή.
2. **Command line scripting:** Για να εκτελεστεί ένα PHP script, δεν απαιτείται κανένας διακομιστής ή πρόγραμμα περιήγησης αλλά μόνο ο αναλυτής της PHP. Αυτός ο τρόπος χρήσης είναι ιδανικός για scripts που εκτελούνται τακτικά με τη χρήση cronjobs (σε Linux) ή με χρήση του Task Scheduler (σε Windows).
3. **Ανάπτυξη εφαρμογών με γραφικό περιβάλλον χρήστη (GUI):** Η PHP δεν είναι ίσως η καλύτερη γλώσσα για τη δημιουργία μιας εφαρμογής με γραφικό περιβάλλον χρήστη, ωστόσο προσφέρει προηγμένα χαρακτηριστικά για client-side εφαρμογές, αλλά και δυνατότητα ανάπτυξης διαπλατφορμικών εφαρμογών.

1.1.2 Laravel

Το Laravel είναι ένα δωρεάν, ανοικτού κώδικα PHP web framework το οποίο δημιουργήθηκε από τον Taylor Otwell το 2011 και προορίζεται για την ανάπτυξη διαδικτυακών εφαρμογών που ακολουθούν το αρχιτεκτονικό πρότυπο MVC (model-view-controller) και βασίζονται στο Symfony.

Ένα web framework είναι μια υποστηρικτική δομή λογισμικού που έχει σχεδιαστεί για να βοηθάει στην ανάπτυξη εφαρμογών διαδικτύου, περιλαμβάνοντας υπηρεσίες, πόρους και διεπαφές προγραμματισμού εφαρμογών (APIs). Τα web frameworks παρέχουν έναν τυποποιημένο τρόπο για τη δημιουργία και την ανάπτυξη εφαρμογών διαδικτύου στοχεύοντας στην αυτοματοποίηση των πιο συχνών διαδικασιών κατά την ανάπτυξη. Για παράδειγμα, πολλά web frameworks παρέχουν βιβλιοθήκες για την πρόσβαση σε βάσεις δεδομένων, πλαίσια διαμόρφωσης προτύπων και διαχείρισης sessions και συχνά προωθούν την επαναχρησιμοποίηση κώδικα.

Το Laravel όπως και πολλά άλλα frameworks ακολουθεί το αρχιτεκτονικό πρότυπο MVC (Model - View - Controller) για να διαχωρίσει το μοντέλο δεδομένων σε επιχειρηματικούς κανόνες (τον «ελεγκτή - controller») και τη διεπαφή χρήστη (την «προβολή - view»). Αυτό θεωρείται γενικά καλή πρακτική, καθώς αρθρώνει τον κώδικα, προωθεί την επαναχρησιμοποίηση κώδικα και επιτρέπει την υλοποίηση πολλαπλών διεπαφών. Στις διαδικτυακές εφαρμογές, αυτό επιτρέπει την παρουσίαση διαφορετικών προβολών, για παράδειγμα, την εξυπηρέτηση διαφορετικών ιστοσελίδων για προγράμματα περιήγησης για κινητά τηλέφωνα έναντι προγραμμάτων περιήγησης για υπολογιστές, ή την παροχή διεπαφών για υπηρεσίες διαδικτύου με δυνατότητα μηχανικής ανάγνωσης.



ΕΙΚΟΝΑ 2 - ΑΠΕΙΚΟΝΙΣΗ ΤΩΝ ΑΛΛΗΛΕΠΙΔΡΑΣΕΩΝ ΣΤΟ ΜΟΝΤΕΛΟ MVC

Οι αρμοδιότητες των τμημάτων του μοντέλου MVC στο Laravel είναι οι εξής:

1. **Model - Μοντέλο:** Είναι μια κλάση που αναπαριστά ένα πίνακα βάσης δεδομένων και τη σχέση μεταξύ άλλων μοντέλων στην εφαρμογή. Μέσα σε μία κλάση μοντέλου είναι δυνατόν να οριστούν σχέσεις SQL (Structured Query Language) όπως «ένα προς πολλά», «ένα προς ένα» και «πολλά προς πολλά».
2. **View - Προβολή:** Απεικονίζει τα δεδομένα που λαμβάνονται από το μοντέλο μέσω του ελεγκτή (controller) ή της διαδρομής (route). Σε μία απλή εφαρμογή, η λογική για την παροχή δεδομένων στην προβολή μπορεί να γίνει και μέσα από τα routes.
3. **Controller - Ελεγκτής:** Υπάρχουν δύο τύποι ελεγκτών που ονομάζονται «standard» και «resource» για να χειρίζονται τα εισερχόμενα αιτήματα HTTP (Hypertext transfer protocol) και να στέλνουν την κατάλληλη απάντηση. Ο standard controller χειρίζεται τα εισερχόμενα

αιτήματα από τις προβολές (views). Από την άλλη πλευρά, ο resource controller χειρίζεται τα αιτήματα HTTP που προέρχονται από κάποιο API.

Ένα από τα σημαντικά πλεονεκτήματα του Laravel σε σχέση με άλλα PHP frameworks, είναι ότι προσφέρει ολοκληρωμένες λύσεις σε μία σειρά από τομείς. Αρχικά στον τομέα της πιστοποίησης ταυτότητας των χρηστών (authentication) παρέχει session-based ταυτοποίηση για προγραμματισμό εφαρμογών διαδικτύου αλλά και token-based ταυτοποίηση για χρήση σε APIs, καλύπτοντας τις βασικές ανάγκες μιας εφαρμογής για την πιστοποίηση χρηστών χωρίς τη χρήση εξωτερικών εργαλείων και βιβλιοθηκών. Ακόμα παρέχει σημαντική ευκολία στην αλληλεπίδραση με βάσεις δεδομένων λόγω του Eloquent, ένα εργαλείο ORM (Object-Oriented Mapper) που επιτρέπει την εύκολη αλληλεπίδραση με αντικείμενα βάσεων δεδομένων χρησιμοποιώντας εκφραστική σύνταξη.

```
public function getLastFlights()
{
    return DB::table('flights')
        ->select('name')
        ->whereColumn('destination', 'ATH')
        ->orderBy('arrived_at', 'desc')
        ->limit(10)
        ->get();
}
```

ΕΙΚΟΝΑ 3 - ΠΑΡΑΔΕΙΓΜΑ ΧΡΗΣΗΣ ΤΟΥ ELOQUENT ΓΙΑ ΑΝΤΛΗΣΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Μια ακόμα σημαντική ευκολία του Laravel είναι ότι ενσωματώνει αρκετά χαρακτηριστικά και δυνατότητες σχετικά με την ασφάλεια των εφαρμογών. Ενδεικτικά περιλαμβάνει προστασία από SQL Injection, προηγμένη προστασία από ευπάθειες CSRF και XSS, hashing των κωδικών πρόσβασης και προστασία των cookies.

Το web framework Laravel τη στιγμή συγγραφής της παρούσας εργασίας βρίσκεται στην έκδοση 9. Σύμφωνα με έρευνα της εταιρείας JetBrains που πραγματοποιήθηκε το 2021, από τους ερωτηθέντες προγραμματιστές, το 67% απάντησε ότι χρησιμοποιεί συστηματικά το Laravel, βάζοντάς το στην κορυφή των προτιμήσεων μεταξύ των PHP frameworks.

1.1.3 MySQL / MariaDB

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) ανοικτού κώδικα. Το όνομά της είναι ένας συνδυασμός του «My», του ονόματος της κόρης του συνιδρυτή Michael Widenius και του «SQL», της συντομογραφίας της Structured Query Language.

Μια σχεσιακή βάση δεδομένων οργανώνει τα δεδομένα σε έναν ή περισσότερους πίνακες δεδομένων, στους οποίους τα δεδομένα μπορούν να συσχετίζονται μεταξύ τους και οι σχέσεις αυτές βοηθούν στη δόμηση των δεδομένων. Η SQL είναι μια γλώσσα που χρησιμοποιούν οι προγραμματιστές για να δημιουργούν, να τροποποιούν και να εξάγουν δεδομένα από σχεσιακές βάσεις δεδομένων, καθώς και για να ελέγχουν την πρόσβαση των χρηστών σε αυτές.

Η MySQL είναι ελεύθερο λογισμικό ανοικτού κώδικα σύμφωνα με τους όρους της Γενικής Άδειας Δημόσιας Χρήσης GNU και διατίθεται επίσης με διάφορες ιδιόκτητες άδειες χρήσης. Η MySQL ανήκε και χρηματοδοτούταν από τη σουηδική εταιρεία MySQL AB, η οποία εξαγοράστηκε από τη Sun Microsystems (σήμερα Oracle Corporation). Το 2010, όταν η Oracle εξαγόρασε τη Sun, ο Widenius δημιούργησε ένα fork του ανοικτού κώδικα της MySQL και δημιούργησε τη MariaDB με το όνομα να προέρχεται από το όνομα της μικρότερης κόρης του.

Η MariaDB έχει σκοπό να διατηρεί υψηλή συμβατότητα με τη MySQL, με ακριβή ταύτιση με τα API και τις εντολές της MySQL, επιτρέποντάς της σε πολλές περιπτώσεις να λειτουργεί ως αντικαταστάτης της. Ωστόσο, τα νέα της χαρακτηριστικά αποκλίνουν καθώς περιλαμβάνει νέες μηχανές αποθήκευσης όπως η Aria, η ColumnStore και η MyRocks οι οποίες δεν υποστηρίζονται στη MySQL.

Η MySQL διαθέτει αυτόνομους clients που επιτρέπουν στους χρήστες να αλληλεπιδρούν απευθείας με μια βάση δεδομένων MySQL χρησιμοποιώντας SQL, αλλά πιο συχνά, η MySQL χρησιμοποιείται μαζί με άλλα προγράμματα για την υλοποίηση εφαρμογών που χρειάζονται δυνατότητες σχεσιακής βάσης δεδομένων. Η MySQL αποτελεί συστατικό στοιχείο της γνωστής στοίβας λογισμικού εφαρμογών διαδικτύου LAMP, η οποία είναι ακρωνύμιο των Linux, Apache, MySQL, Perl/PHP/Python. Η MySQL χρησιμοποιείται από πολλές διαδικτυακές εφαρμογές που βασίζονται σε βάσεις δεδομένων, όπως το Drupal, το Joomla, το phpBB και το WordPress. Χρησιμοποιείται επίσης και από πολλές δημοφιλείς ιστοσελίδες, όπως το Facebook, το Flickr, το Twitter και το YouTube.

```
PS C:\Program Files\MySQL\MySQL Workbench 8.0 CE> .\mysql.exe
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.5.5-10.4.22-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW tables;
+-----+
| Tables_in_alumni_internships_db |
+-----+
```

ΕΙΚΟΝΑ 4 - ΣΤΙΓΜΙΟΤΥΠΟ ΤΗΣ ΓΡΑΜΜΗΣ ΕΝΤΟΛΩΝ ΤΗΣ MYSQL

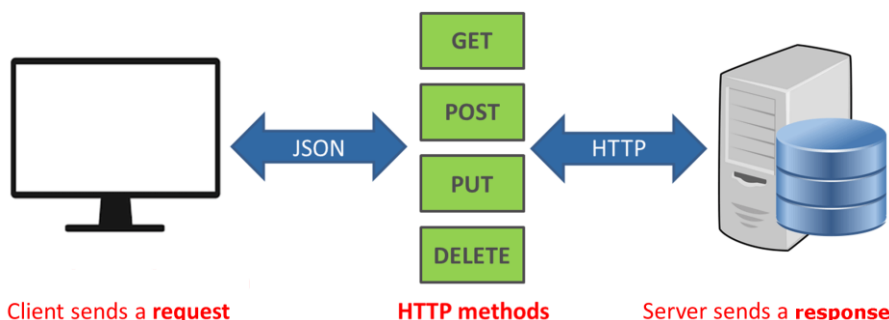
1.1.4 REST API

Ένα REST API (επίσης γνωστό ως RESTful API) είναι μια διεπαφή προγραμματισμού εφαρμογών (API) που συμμορφώνεται με τους περιορισμούς του αρχιτεκτονικού στυλ REST και επιτρέπει την αλληλεπίδραση με RESTful υπηρεσίες ιστού. Το REST σημαίνει Representational State Transfer (μεταφορά κατάστασης αναπαράστασης) και δημιουργήθηκε από τον επιστήμονα πληροφορικής Roy Fielding ο οποίος το όρισε κατά τη διδακτορική του διατριβή το 2000.

Ένα API είναι ένα σύνολο ορισμών και πρωτοκόλλων για τη δημιουργία και την ενσωμάτωση λογισμικού εφαρμογών. Ορισμένες φορές αναφέρεται ως σύμβαση μεταξύ ενός παρόχου πληροφοριών και ενός χρήστη πληροφοριών, η οποία καθορίζει το περιεχόμενο που απαιτείται από τον καταναλωτή (η κλήση) και το περιεχόμενο που απαιτείται από τον παραγωγό (η απάντηση).

Το REST είναι ένα σύνολο αρχιτεκτονικών περιορισμών και όχι ένα πρωτόκολλο ή ένα πρότυπο. Οι προγραμματιστές API μπορούν να υλοποιήσουν το REST με διάφορους τρόπους. Όταν ένα αίτημα πελάτη πραγματοποιείται μέσω ενός RESTful API, μεταφέρει μια αναπαράσταση της κατάστασης του πόρου στον αιτούντα ή στο τελικό σημείο. Αυτές οι πληροφορίες ή η αναπαράσταση παραδίδονται σε μία από τις διάφορες μορφές μέσω HTTP: JSON (Javascript Object Notation), HTML, XML, Python, PHP ή απλό κείμενο. Το JSON είναι η πιο δημοφιλής μορφή αρχείου που χρησιμοποιείται επειδή, παρά το όνομά του, είναι γλωσσικά αδιάφορο, καθώς και αναγνώσιμο τόσο από ανθρώπους όσο και από μηχανές.

Το πιο συνηθισμένο πρωτόκολλο για αιτήματα (requests) και απαντήσεις (responses) είναι το HTTP, το οποίο παρέχει μεθόδους όπως OPTIONS, GET, POST, PUT, PATCH και DELETE. Με τη χρήση ενός stateless πρωτοκόλλου και τυποποιημένων λειτουργιών, τα RESTful συστήματα στοχεύουν στη γρήγορη απόδοση, την αξιοπιστία και τη δυνατότητα ανάπτυξης μέσω της επαναχρησιμοποίησης στοιχείων που μπορούν να διαχειριστούν και να ενημερωθούν χωρίς να επηρεάζουν το σύστημα στο σύνολό του, ακόμη και κατά τη διάρκεια της λειτουργίας του.



ΕΙΚΟΝΑ 5 - ΑΠΕΙΚΟΝΙΣΗ ΤΡΟΠΟΥ ΛΕΙΤΟΥΡΓΙΑΣ ΕΝΟΣ REST API

1.2 Τεχνολογίες Front End

1.2.1 HTML

Η HTML (HyperText Markup Language, Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, τα στοιχεία της οποίας είναι τα βασικά δομικά στοιχεία των ιστοσελίδων. Ο όρος «υπερκειμένο» αναφέρεται στους συνδέσμους που συνδέουν τις ιστοσελίδες μεταξύ τους, είτε εντός ενός δικτυακού τόπου είτε μεταξύ δικτυακών τόπων. Οι σύνδεσμοι είναι μια θεμελιώδης πτυχή του Διαδικτύου.

Το 1980, ο φυσικός Τιμ Μπέρνερς Λι, ο οποίος εργαζόταν στο CERN, επινόησε το ENQUIRE, ένα σύστημα χρήσης και διαμοιρασμού εγγράφων για τους ερευνητές του CERN, και κατασκεύασε ένα πρωτότυπό του. Αργότερα, το 1989, πρότεινε ένα σύστημα βασισμένο στο διαδίκτυο, το οποίο θα χρησιμοποιούσε υπερκειμένο. Το σύστημα αυτό δεν υιοθετήθηκε ποτέ επίσημα από το CERN. Η πρώτη δημόσια διαθέσιμη περιγραφή της HTML ήταν ένα έγγραφο με το όνομα «Ετικέτες HTML», το οποίο πρωτοαναφέρθηκε στο Διαδίκτυο από τον Μπέρνερς Λι στα τέλη του 1991. Περιέγραφε τα 20 στοιχεία τα οποία αποτελούσαν τον αρχικό και σχετικά απλό σχεδιασμό της HTML. Εκτός από την ετικέτα υπερσυνδέσμου, οι υπόλοιπες ήταν έντονα επηρεασμένες από την SGMLguid, μια μορφή δημιουργίας τεκμηρίωσης, φτιαγμένη στο CERN και βασισμένη στην SGML.

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περιλαμβάνονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ετικέτα	Περιγραφή
<html> </html>	Ορίζει την αρχή και το τέλος μίας HTML σελίδας.
<head> </head>	Ορίζει το τμήμα εκείνο της ιστοσελίδας που περιέχει πληροφορίες σχετικά με το html έγγραφο. Περιέχει στοιχεία όπως τον τίτλο και τους συνδέσμους για τα css και js αρχεία.
<body> </body>	Ορίζει το περιεχόμενο της HTML σελίδας.
<title> </title>	Ορίζει τον τίτλο της HTML σελίδας.
<p> </p>	Ορίζει μια παράγραφο
 	Ορίζει την αλλαγή γραμμής
	Εμφανίζει μία εικόνα
	Ορίζει έναν υπερσύνδεσμο

ΕΙΚΟΝΑ 6 - ΒΑΣΙΚΕΣ ΕΤΙΚΕΤΕΣ HTML ΚΑΙ Η ΠΕΡΙΓΡΑΦΗ ΤΟΥΣ

Ο σκοπός ενός web browser είναι να διαβάσει τα έγγραφα HTML και να τα συνθέσει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να παρουσιάσει το περιεχόμενο της σελίδας. Οι browsers μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS (Cascading Style Sheets) για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Hello HTML
    </title>
  </head>
  <body>
    <p>
      Hello world
    </p>
    
    <a href="https://www.ece.uop.gr">
      Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
    </a>
  </body>
</html>
```

ΕΙΚΟΝΑ 7 - ΠΑΡΑΔΕΙΓΜΑ ΠΕΡΙΕΧΟΜΕΝΟΥ ΕΝΟΣ ΑΡΧΕΙΟΥ HTML

Η HTML5 είναι η πιο πρόσφατη προδιαγραφή της γλώσσας HTML και αποτελεί μια σημαντική διαφοροποίηση από τις προηγούμενες πρακτικές σήμανσης. Ο σκοπός των ριζικών αλλαγών στη γλώσσα ήταν να τυποποιηθούν οι πολλοί νέοι τρόποι με τους οποίους τη χρησιμοποιούσαν οι προγραμματιστές, καθώς και να προωθηθεί ένα ενιαίο σύνολο βέλτιστων πρακτικών όσον αφορά την ανάπτυξη ιστοσελίδων.

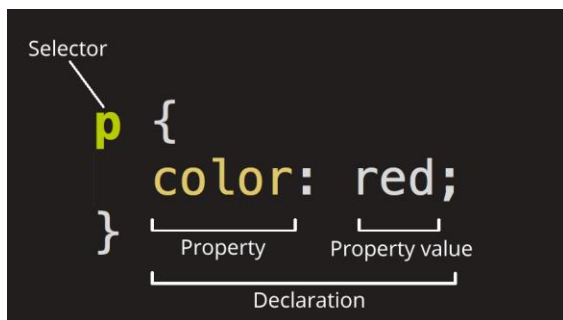
Οι περισσότερες από τις επιμέρους αλλαγές είναι αποτέλεσμα ευρύτερων στόχων στο σχεδιασμό της γλώσσας. Αυτοί οι στόχοι περιλαμβάνουν κυρίως:

- Ενθάρρυνση της σημασιολογικής (ουσιαστικής) σήμανσης
- Διαχωρισμός του σχεδιασμού από το περιεχόμενο
- Προώθηση της προσβασιμότητας και του ευέλικτου σχεδιασμού (responsive design)
- Μείωση της επικάλυψης μεταξύ HTML, CSS και JavaScript
- Υποστήριξη πλούσιων εμπειριών πολυμέσων με ταυτόχρονη εξάλειψη της ανάγκης για πρόσθετα όπως το Flash ή τη Java

1.2.2 CSS

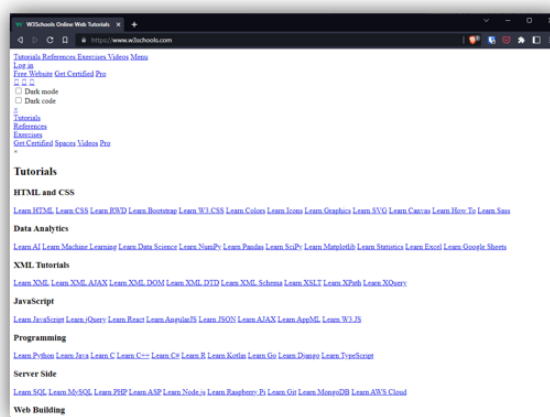
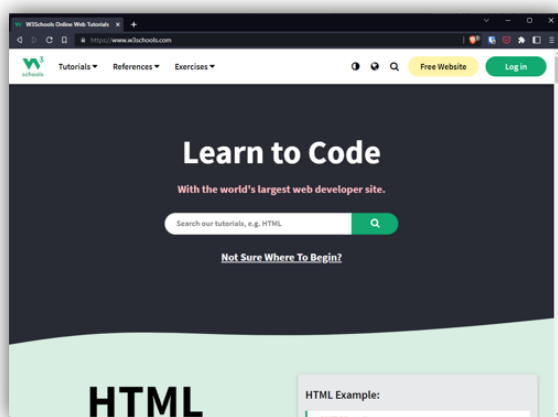
Η CSS (Cascading Style Sheets - διαδοχικά φύλλα ύφους) είναι μια γλώσσα η οποία περιέχει κανόνες για τον έλεγχο της εμφάνισης ενός εγγράφου γραμμένου σε μια γλώσσα σήμανσης, όπως η HTML. Αποτελεί ακρογωνιαίο λίθο του Παγκόσμιου Ιστού, μαζί με την HTML και τη JavaScript και έχει σχεδιαστεί για να επιτρέπει το διαχωρισμό της παρουσίασης και του περιεχομένου, συμπεριλαμβανομένης της διάταξης, των χρωμάτων και των γραμματοσειρών.

Ένας «κανόνας» CSS αποτελείται από 2 βασικά μέρη: Τον επιλογέα (selector) ο οποίος αφορά το τί θα μορφοποιηθεί και μία ή περισσότερες δηλώσεις (declarations) οι οποίες αφορούν το πώς αυτό θα μορφοποιηθεί.



ΕΙΚΟΝΑ 8 - ΑΝΑΤΟΜΙΑ ΚΑΝΟΝΑ CSS

Στο παράδειγμα της εικόνας 8, βλέπουμε ότι ο επιλογέας είναι το στοιχείο « <p> » δηλαδή μία παράγραφος. Υπάρχει μόνο μία δήλωση, η «color: red». Το αριστερά κομμάτι (ιδιότητα) δείχνει την ιδιότητα του HTML στοιχείου που θέλουμε να μορφοποιήσουμε (color) και στο δεξιά κομμάτι την τιμή που θέλουμε να δώσουμε σε αυτή την ιδιότητα (red). Στο παραπάνω παράδειγμα, εάν εφαρμοστεί ο κανόνας CSS, το κείμενο που βρίσκεται μέσα στην παράγραφο, θα αποκτήσει κόκκινο χρώμα.



ΕΙΚΟΝΑ 9 - ΠΑΡΑΔΕΙΓΜΑ ΣΕΛΙΔΑΣ HTML ΜΕ (ΑΡΙΣΤΕΡΑ) ΚΑΙ ΧΩΡΙΣ (ΔΕΞΙΑ) ΜΟΡΦΟΠΟΙΗΣΗ CSS

1.2.2.1 Sass

Η Sass (Syntactically Awesome Style Sheets) είναι μια preprocessor scripting γλώσσα που διερμηνεύεται ή μεταγλωττίζεται σε Cascading Style Sheets (CSS). Το SassScript είναι η ίδια η scripting γλώσσα.

Η Sass σχεδιάστηκε αρχικά από τον Hampton Catlin και αναπτύχθηκε από τη Natalie Weizenbaum. Μετά τις αρχικές εκδόσεις της, ο Weizenbaum και ο Chris Eppstein συνέχισαν να επεκτείνουν το Sass με το SassScript, μια γλώσσα σεναρίων που χρησιμοποιείται στα αρχεία Sass.

Μια γλώσσα preprocessor για CSS μπορεί να βοηθήσει στο πρόβλημα πολυπλοκότητας που έχει η CSS σε περιπτώσεις απαιτητικής και σύνθετης μορφοποίησης περιεχομένου. Αυτό το επιτυγχάνει με μία σειρά από χαρακτηριστικά που δεν υπάρχουν στη CSS, όπως μεταβλητές, εμφωλευμένους κανόνες, mixins, εισαγωγές, κληρονομικότητα και ενσωματωμένες συναρτήσεις.

```
/* define variables for the primary colors */
$primary_1: #a2b9bc;
$primary_2: #b2ad7f;
$primary_3: #878f99;

/* use the variables */
.main-header {
  background-color: $primary_1;
}

.menu-left {
  background-color: $primary_2;
}

.menu-right {
  background-color: $primary_3;
}
```

ΕΙΚΟΝΑ 10 - ΠΑΡΑΔΕΙΓΜΑ ΧΡΗΣΗΣ ΜΕΤΑΒΛΗΤΩΝ ΣΕ ΑΡΧΕΙΟ SASS

1.2.3 JavaScript

Η JavaScript είναι μια γλώσσα προγραμματισμού που αποτελεί μια από τις βασικές τεχνολογίες του Παγκόσμιου Ιστού, μαζί με την HTML και την CSS. Χρησιμοποιείται κυρίως για να γίνει μία ιστοσελίδα διαδραστική. Ενώ η HTML και η CSS είναι γλώσσες που παρέχουν δομή και μορφοποίηση στις ιστοσελίδες, η JavaScript παρέχει στις ιστοσελίδες διαδραστικά στοιχεία βελτιώνοντας έτσι δραστικά την εμπειρία χρήσης τους. Μερικά παραδείγματα χρήσης JavaScript στην καθημερινή ζωή των περισσότερων χρηστών του διαδικτύου περιλαμβάνουν την αυτόματη συμπλήρωση στο πλαίσιο αναζήτησης στο Google, ένα βίντεο που ενσωματώνεται σε κάποια ειδησεογραφική ιστοσελίδα ή την ανανέωση της ροής σε μία σελίδα κοινωνικής δικτύωσης.

Σύμφωνα με την ιστοσελίδα W3Techs, η οποία παρέχει πληροφορίες σχετικά με τη χρήση διαφόρων τύπων τεχνολογιών στο διαδίκτυο, η JavaScript χρησιμοποιείται από περίπου το 98% των ιστοσελίδων στην πλευρά του πελάτη (client), ενσωματώνοντας συχνά βιβλιοθήκες τρίτων. Όλα τα μεγάλα προγράμματα περιήγησης ιστού διαθέτουν ειδική μηχανή JavaScript για την εκτέλεση του κώδικα στις συσκευές των χρηστών.

Η JavaScript δημιουργήθηκε στην Netscape Communications από τον Brendan Eich το 1995. Η Netscape και ο Eich σχεδίασαν τη JavaScript ως μια scripting γλώσσα για χρήση με το πρόγραμμα περιήγησης ιστού της εταιρείας, το Netscape Navigator. Αρχικά ήταν γνωστή ως LiveScript αλλά η Netscape άλλαξε το όνομα σε JavaScript, ώστε να μπορέσει να την παρουσιάσει ως συμπλήρωμα της γλώσσας Java, προϊόν του συνεργάτη της, της Sun Microsystems. Εκτός από κάποιες επιφανειακές συντακτικές ομοιότητες, όμως, η JavaScript δεν έχει καμία σχέση με τη γλώσσα προγραμματισμού Java.

Μετά την κυκλοφορία της, όλο και περισσότερα προγράμματα περιήγησης άρχισαν να την υποστηρίζουν. Παρόλα αυτά, για μεγάλο μέρος της ιστορίας της η JavaScript δεν θεωρούνταν ως μία σοβαρή γλώσσα προγραμματισμού. Οι πρώτες εκδόσεις της υπέφεραν από σημαντικά προβλήματα απόδοσης και ασφάλειας, αλλά οι προγραμματιστές δεν είχαν εναλλακτικές λύσεις. Αν ήθελαν να εκτελούν κώδικα σε προγράμματα περιήγησης, η JavaScript ήταν μονόδρομος.

Το 2008, η δημιουργία του Chrome V8, μιας ανοικτού κώδικα μηχανής JavaScript υψηλής απόδοσης από τη Google, αποτέλεσε ένα κρίσιμο σημείο καμπής για τη JavaScript. Η επακόλουθη διάδοση των γρήγορων μηχανών JavaScript κατέστησε δυνατή για τους προγραμματιστές τη δημιουργία εξελιγμένων εφαρμογών βασισμένων στο πρόγραμμα περιήγησης με επιδόσεις που ανταγωνίζονταν τις εφαρμογές γραφείου και κινητών τηλεφώνων.

Λίγο αργότερα, ο Ryan Dahl κυκλοφόρησε ένα ανοικτού κώδικα διαπλατφορμικό περιβάλλον με την ονομασία Node.js. Το Node.js παρείχε έναν τρόπο εκτέλεσης κώδικα JavaScript έξω από τα προγράμματα περιήγησης, το οποίο οδήγησε άμεσα στη σημερινή δημοτικότητα της JavaScript. Σήμερα η JavaScript μπορεί να χρησιμοποιηθεί για τη συγγραφή όλων των ειδών εφαρμογών, συμπεριλαμβανομένων εφαρμογών περιήγησης, διακομιστή, κινητών, αλλά και εφαρμογών για υπολογιστές. Οι περισσότερες μεγάλες σημερινές εταιρείες τεχνολογίας όπως η Meta, το Twitter, το Netflix και η Google, χρησιμοποιούν JavaScript στα προϊόντα τους.

Τον Νοέμβριο του 1996, η Netscape ανακοίνωσε ότι είχε υποβάλει τη γλώσσα JavaScript στο Ecma International (μια οργάνωση τυποποίησης των γλωσσών προγραμματισμού) για εξέταση ως βιομηχανικό πρότυπο, και ως αποτέλεσμα αυτής της ενέργειας το 1997 προέκυψε η τυποποιημένη ονομασία της JavaScript, ECMAScript. Ως εκ τούτου, οι εκδόσεις της JavaScript ονομάζονταν ES1 έως ES6 μέχρι το 2015, ως συντομογραφία του ECMAScript 1 έως 6. Από το 2016 και μετά οι εκδόσεις είναι ανά χρόνο, πχ ECMAScript 2016, ECMAScript 2017 και ECMAScript 2022.

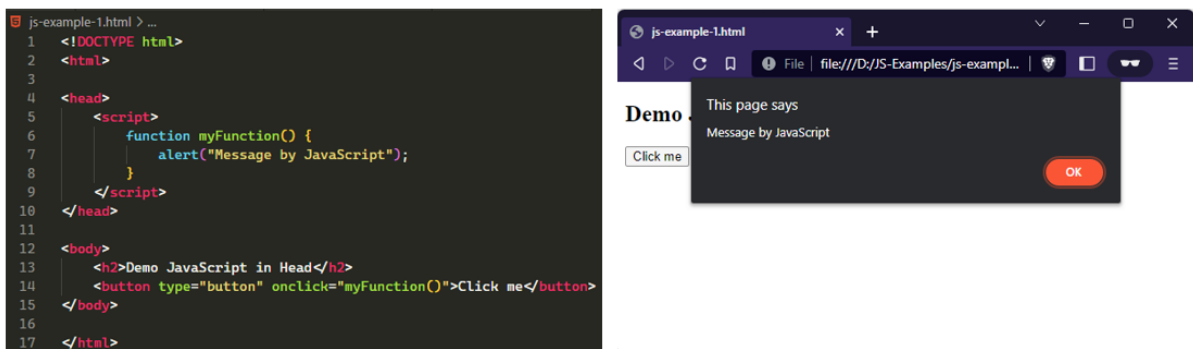
Η σύγχρονη JavaScript είναι μια «ασφαλής» γλώσσα προγραμματισμού. Δεν παρέχει πρόσβαση χαμηλού επιπέδου στη μνήμη ή στη CPU καθώς αρχικά δημιουργήθηκε για προγράμματα περιήγησης τα οποία δεν απαιτούν κάτι τέτοιο. Οι δυνατότητες της JavaScript εξαρτώνται σε μεγάλο βαθμό από το περιβάλλον στο οποίο εκτελείται. Για παράδειγμα, το Node.js υποστηρίζει λειτουργίες που επιτρέπουν στη JavaScript να διαβάζει/γράφει αρχεία, να εκτελεί αιτήματα δικτύου κλπ.

Εντός ενός προγράμματος περιήγησης ωστόσο, η JavaScript μπορεί να κάνει οτιδήποτε σχετίζεται με τον χειρισμό ιστοσελίδων, την αλληλεπίδραση με τον χρήστη και την επικοινωνία με εξυπηρετητές ιστού (webservers). Μερικά παραδείγματα δυνατοτήτων της JavaScript εντός του προγράμματος περιήγησης είναι τα εξής:

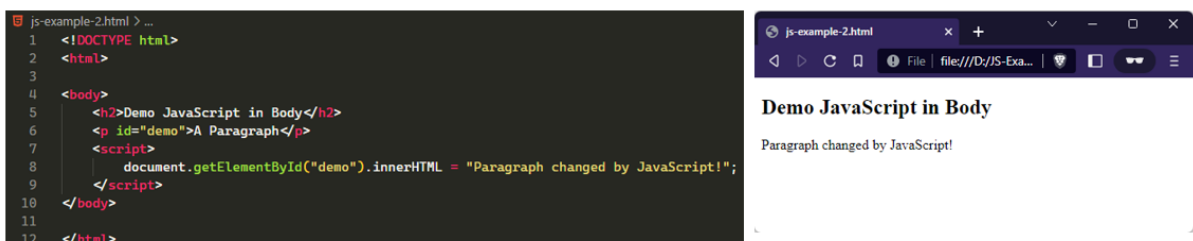
- Προσθήκη και αφαίρεση στοιχείων HTML στη σελίδα, αλλαγή του τρέχοντος περιεχομένου, αλλά και τροποποίηση της μορφοποίησής του.
- Αντίδραση σε ενέργειες του χρήστη, εκτέλεση κώδικα μετά από ένα «κλικ» του ποντικιού, κινήσεις του δείκτη του ποντικιού ή πατήματα πλήκτρων.
- Αποστολή αιτημάτων σε απομακρυσμένους διακομιστές, κατέβαση και ανέβαση αρχείων.
- Λήψη και ορισμός των cookies, προβολή μηνυμάτων ή απεύθυνση ερωτημάτων προς το χρήστη μέσω αναδυόμενων παραθύρων.
- Αποθήκευση δεδομένων στο πρόγραμμα περιήγησης (local storage).

Για να ενσωματωθεί κώδικας JavaScript σε μία HTML σελίδα, χρησιμοποιείται η HTML ετικέτα «script». Αυτό μπορεί να γίνει με τρεις βασικούς τρόπους:

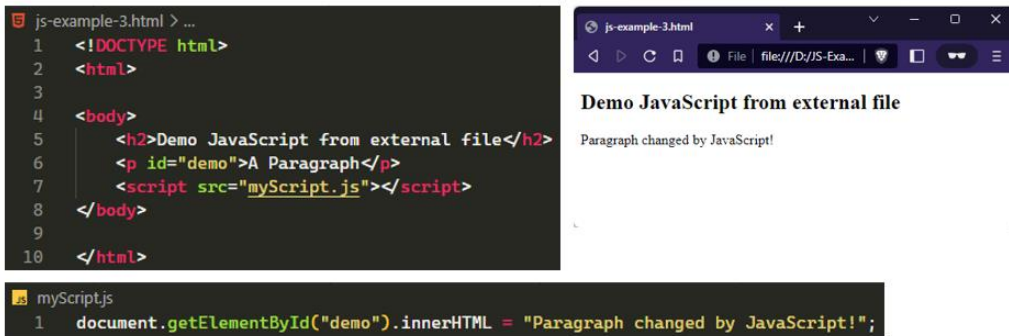
1. Με δήλωση μίας συνάρτησης στο head της σελίδας και κλήση αυτής από ένα στοιχείο HTML όπως ένα κουμπί (Εικόνα 11).
2. Με χρήση κώδικα JavaScript απευθείας μέσα στο body μίας σελίδας (Εικόνα 12).
3. Με κλήση εξωτερικού αρχείου JavaScript (Εικόνα 13). Στην περίπτωση αυτή, μπορούμε να βάλουμε τη διαδρομή του αρχείου ή να βάλουμε μια διεύθυνση URL αν το αρχείο βρίσκεται σε κάποια εξωτερική ιστοσελίδα.



ΕΙΚΟΝΑ 11 - ΠΑΡΑΔΕΙΓΜΑ ΕΝΣΩΜΑΤΩΣΗΣ ΚΩΔΙΚΑ JAVASCRIPT ΣΤΟ HEAD ΜΙΑΣ ΣΕΛΙΔΑΣ HTML



ΕΙΚΟΝΑ 12 - ΠΑΡΑΔΕΙΓΜΑ ΕΝΣΩΜΑΤΩΣΗΣ ΚΩΔΙΚΑ JAVASCRIPT ΣΤΟ BODY ΜΙΑΣ ΣΕΛΙΔΑΣ HTML



ΕΙΚΟΝΑ 13 - ΠΑΡΑΔΕΙΓΜΑ ΕΝΣΩΜΑΤΩΣΗΣ ΚΩΔΙΚΑ JAVASCRIPT ΣΕ ΣΕΛΙΔΑ HTML ΑΠΟ ΕΞΩΤΕΡΙΚΟ ΑΡΧΕΙΟ

Η χρήση της JavaScript συχνά συνοδεύεται και από χρήση βιβλιοθηκών και web frameworks τα οποία επιτρέπουν σημαντική εξοικονόμηση χρόνου και προσπάθειας από την πλευρά των προγραμματιστών, καθώς αφορούν κώδικα και λειτουργίες που χρησιμοποιούνται συχνά και έχουν τη δυνατότητα επαναχρησιμοποίησης, με στόχο την εκτέλεση μιας συγκεκριμένης εργασίας.

Η μακράν πιο δημοφιλής βιβλιοθήκη αυτή τη στιγμή είναι η jQuery, καθώς χρησιμοποιείται από πάνω από το 75% των ιστοσελίδων παγκοσμίως. Μια ακόμα αρκετά δημοφιλής βιβλιοθήκη είναι η React, την οποία δημιούργησε το Facebook για χρήση στη δική του ιστοσελίδα, ωστόσο αργότερα την κυκλοφόρησε ως βιβλιοθήκη ανοιχτού κώδικα. Παρομοίως, το framework Angular που δημιουργήθηκε από την Google για τις ιστοσελίδες της, συμπεριλαμβανομένου του YouTube και του Gmail, είναι τώρα ένα project ανοιχτού κώδικα που χρησιμοποιείται και από πολλές άλλες εταιρείες.

Στον αντίποδα, ο όρος «Vanilla JS» έχει επινοηθεί για τις ιστοσελίδες που δεν χρησιμοποιούν βιβλιοθήκες ή frameworks, αλλά βασίζονται εξ ολοκλήρου στην πρότυπη λειτουργικότητα της JavaScript.

1.2.4 Bootstrap

Το Bootstrap είναι μια βιβλιοθήκη HTML, CSS & JS που επικεντρώνεται στην απλοποίηση της ανάπτυξης ιστοσελίδων εφαρμόζοντας μια σειρά από προκαθορισμένες ιδιότητες στα στοιχεία της σελίδας όπως το χρώμα, το μέγεθος, τη γραμματοσειρά και τη διάταξή τους. Δημιουργήθηκε το 2011 από τους προγραμματιστές του Twitter τους Mark Otto και Jacob Thornton και αποτελεί ένα από τα πλέον πιο δημοφιλή Web Design Frameworks.

Μόλις προστεθεί σε μια σελίδα, το Bootstrap παρέχει βασικούς ορισμούς στυλ για όλα τα στοιχεία HTML. Το αποτέλεσμα είναι μια ομοιόμορφη εμφάνιση για τον πεζό λόγο, τους πίνακες και τα στοιχεία φόρμας σε όλους τους φυλλομετρητές ιστού. Επιπλέον, οι προγραμματιστές μπορούν να επωφεληθούν από τις κλάσεις CSS που ορίζονται στο Bootstrap για να προσαρμόσουν περαιτέρω την εμφάνιση του περιεχομένου τους.

Το Bootstrap διαθέτει επίσης αρκετά στοιχεία JavaScript τα οποία δεν απαιτούν άλλες βιβλιοθήκες όπως η jQuery. Παρέχει πρόσθετα στοιχεία διεπαφής χρήστη, όπως πλαίσια διαλόγου, tooltips, μπάρες προόδου, drop-downs και carousels. Κάθε στοιχείο του Bootstrap αποτελείται από μια δομή HTML, δηλώσεις CSS και σε ορισμένες περιπτώσεις από συνοδευτικό κώδικα JavaScript.

The image shows a code editor on the left with HTML code for a Bootstrap page. The code includes the following sections:

- Page Header:** A container with a page header.
- Alerts:** Two alert boxes, one for success (green) and one for info (blue).
- Columns:** A row of three columns, each containing placeholder text.
- Buttons:** A row of five buttons: Basic, Default, Primary, Warning, and Link.

On the right, the rendered output of the code is shown. It features a white background with a light gray border. The 'Alerts' section shows two boxes: a green one for success and a blue one for info. The 'Columns' section shows three columns of text. The 'Buttons' section shows five buttons with different styles and colors.

ΕΙΚΟΝΑ 14 - ΠΑΡΑΔΕΙΓΜΑ ΧΡΗΣΗΣ BOOTSTRAP

1.2.5 React

Η React (γνωστή και ως ReactJS) είναι μια δωρεάν και ανοικτού κώδικα βιβλιοθήκη JavaScript για τη δημιουργία διεπαφών χρήστη που βασίζονται σε στοιχεία UI. Συντηρείται από τη Meta (πρώην Facebook) και μια κοινότητα μεμονωμένων προγραμματιστών και εταιρειών.

Δημιουργήθηκε από τον Jordan Walke, έναν μηχανικό λογισμικού στη Facebook, ο οποίος κυκλοφόρησε ένα πρώιμο πρωτότυπο της React με την ονομασία «FaxJS», το οποίο ήταν επηρεασμένο από την XHP, μια βιβλιοθήκη συστατικών HTML για την PHP. Χρησιμοποιήθηκε για πρώτη φορά στη ροή ειδήσεων του Facebook το 2011 και αργότερα στο Instagram το 2012. Το 2013 η βιβλιοθήκη έγινε ανοικτού κώδικα κατά τη διάρκεια του συνεδρίου JSConf US.

Το 2015, ανακοινώθηκε στο Facebook React Conf το React Native, ένα UI Software Framework το οποίο επιτρέπει την εγγενή ανάπτυξη εφαρμογών Android, iOS και UWP με χρήση της React.

Στις 18 Απριλίου 2017, το Facebook ανακοίνωσε το React Fiber, ένα νέο σύνολο εσωτερικών αλγορίθμων για την απόδοση, σε αντίθεση με τον παλιό αλγόριθμο απόδοσης του React, το Stack. Το React Fiber επρόκειτο να αποτελέσει τη βάση για κάθε μελλοντική βελτίωση και ανάπτυξη χαρακτηριστικών της βιβλιοθήκης React.

Η τελευταία έκδοση της React κατά τη συγγραφή της παρούσας εργασίας είναι η React 18, ή οποία παρουσιάστηκε το Μάρτιο του 2022 και φέρνει μία σειρά από καινούριες λειτουργίες.

```
import React from 'react';
import ReactDOM from 'react-dom/client';

const Greeting = () => {
  return (
    <div className="hello-world">
      <h1>Hello, world!</h1>
    </div>
  );
};

const App = () => {
  return <Greeting />;
};

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

ΕΙΚΟΝΑ 15 - ΠΑΡΑΔΕΙΓΜΑ ΠΕΡΙΕΧΟΜΕΝΟΥ ΕΝΟΣ ΑΡΧΕΙΟΥ REACT

Η φιλοσοφία της React διαφέρει αρκετά από την τυπική αρχιτεκτονική μίας ιστοσελίδας. Συνήθως, όταν ζητάμε μια ιστοσελίδα πληκτρολογώντας τη διεύθυνση URL της στο πρόγραμμα περιήγησης ιστού μας, το πρόγραμμα περιήγησής στέλνει ένα αίτημα για την εν λόγω ιστοσελίδα, η οποία στην συνέχεια θα απεικονιστεί στην οθόνη μας. Εάν κάνουμε κλικ σε έναν σύνδεσμο σε αυτή την ιστοσελίδα για να μεταβούμε σε μια άλλη σελίδα του ιστότοπου, αποστέλλεται ένα νέο αίτημα στον διακομιστή για να λάβει τη νέα σελίδα.

Αυτό το μοτίβο φόρτωσης ερώτησης-απάντησης μεταξύ του προγράμματος περιήγησης και του διακομιστή συνεχίζεται για κάθε νέα σελίδα ή πόρο ενός ιστότοπου στον οποίο προσπαθούμε να

αποκτήσουμε πρόσβαση. Αυτή η τυπική προσέγγιση για τη φόρτωση ιστότοπων λειτουργεί, ωστόσο σε έναν ιστότοπο που βασίζεται σε δεδομένα, η φόρτωση ολόκληρης της ιστοσελίδας είναι περιττή και δημιουργεί κακή εμπειρία χρήστη. Επιπλέον, όταν τα δεδομένα αλλάζουν σε μια παραδοσιακή εφαρμογή JavaScript, απαιτείται χειροκίνητος χειρισμός του DOM (Document Object Model) για να αντικατοπτρίζονται αυτές οι αλλαγές. Πρέπει να προσδιοριστούν ποια δεδομένα άλλαξαν και να ενημερωθεί το DOM για να αντικατοπτρίσει αυτές τις αλλαγές, με αποτέλεσμα την επαναφόρτωση ολόκληρης της σελίδας.

Η React ακολουθεί μια διαφορετική προσέγγιση, επιτρέποντας τη δημιουργία εφαρμογών μίας σελίδας (Single Page Application, SPA). Μια εφαρμογή SPA φορτώνει μόνο ένα έγγραφο HTML μαζί με την απαιτούμενη CSS και JavaScript κατά το πρώτο αίτημα. Από εκεί και έπειτα, ενημερώνει μόνο το συγκεκριμένο τμήμα της ιστοσελίδας που χρειάζεται αλλαγή, χρησιμοποιώντας JavaScript.

Αυτό το μοτίβο είναι γνωστό ως δρομολόγηση από την πλευρά του χρήστη, όπου κάνοντας κλικ ο χρήστης σε εσωτερικούς συνδέσμους μέσα στην εφαρμογή, δεν προκύπτει ολόκληρη ανανέωση της σελίδας, αλλά μόνο αλλαγή στη γραμμή διευθύνσεων του προγράμματος περιήγησης. Παράλληλα, η React παρεμβάλλει το αίτημα του χρήστη και αλλάζει μόνο τα τμήματα της σελίδας που χρειάζονται αλλαγή δίνοντας την αίσθηση αλλαγής σελίδας. Αυτή η προσέγγιση έχει ως αποτέλεσμα καλύτερες επιδόσεις και μια πιο δυναμική εμπειρία χρήστη.

Η παραπάνω ενέργεια είναι δυνατή λόγω της χρήσης ενός εικονικού DOM, το οποίο είναι ένα αντίγραφο του πραγματικού DOM. Το εικονικό DOM της React επαναφορτώνεται αμέσως για να αντικατοπτρίζει οποιαδήποτε αλλαγή κάθε φορά που υπάρχει τροποποίηση στην κατάσταση των δεδομένων. Έπειτα, η React συγκρίνει το εικονικό DOM με το πραγματικό DOM για να καταλάβει τι ακριβώς έχει αλλάξει και υπολογίζει τον λιγότερο δαπανηρό τρόπο για να επιδιορθώσει το πραγματικό DOM με αυτή την ενημέρωση, χωρίς να το εκτελέσει. Ως αποτέλεσμα, τα components και τα στοιχεία UI της React αντικατοπτρίζουν πολύ γρήγορα τις αλλαγές, αφού δεν χρειάζεται να επαναφορτώνεται ολόκληρη η ιστοσελίδα κάθε φορά που κάτι αλλάζει σε αυτήν.

1.3 Βιβλιοθήκες

Στο υπό-κεφάλαιο αυτό θα γίνει μια σύντομη αναφορά στις κυριότερες εξωτερικές βιβλιοθήκες που χρησιμοποιήθηκαν στην εφαρμογή React του συστήματος παρακολούθησης φορέων και θέσεων πρακτικής άσκησης.

1.3.1 Axios

Η Axios είναι μια πολύ δημοφιλής βιβλιοθήκη JavaScript που μπορεί να χρησιμοποιηθεί για την εκτέλεση αιτημάτων HTTP, όπως GET, POST και DELETE. Έχει σχεδιαστεί με τέτοιο τρόπο ώστε να μπορεί να εκτελεστεί τόσο στο περιβάλλον ενός προγράμματος περιήγησης όσο και σε περιβάλλον Node.js χωρίς κάποια αλλαγή στον κώδικά της.

Ένα από τα βασικά της πλεονεκτήματα σε σχέση με τον προκαθορισμένο τρόπο εκτέλεσης αιτημάτων HTTP στη JavaScript είναι η χρήση αντικειμένων «Promise». Ένα αντικείμενο Promise αναπαριστά την ενδεχόμενη ολοκλήρωση (ή αποτυχία) μιας ασύγχρονης λειτουργίας και την παραγόμενη τιμή της. Στην περίπτωση των HTTP requests αυτό είναι αρκετά σημαντικό καθώς επιτρέπει την πραγματοποίηση ασύγχρονων αιτημάτων.

Μερικά ακόμα πλεονεκτήματά της περιλαμβάνουν την προστασία από επιθέσεις CSRF, τη δυνατότητα ακύρωσης αιτημάτων, τροποποίησης των δεδομένων των αιτημάτων αλλά και των απαντήσεων και τον αυτόματο μετασχηματισμό των απαντήσεων σε μορφή JSON (JavaScript Object Notation).

```
const axios = require('axios').default;

// Make a request for a user with a given ID
axios.get('/user', {
  params: {
    ID: 12345
  }
})
  .then(function (response) {
    console.log(response);
  })
  .catch(function (error) {
    console.log(error);
  })
  .finally(function () {
    // always executed
  });
```

ΕΙΚΟΝΑ 16 - ΠΑΡΑΔΕΙΓΜΑ ΕΚΤΕΛΕΣΗΣ ΑΙΤΗΜΑΤΟΣ GET ΜΕ ΧΡΗΣΗ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ AXIOS

1.3.2 Syncfusion Grid

Το Syncfusion Grid είναι υποσύνολο μίας μεγάλης βιβλιοθήκης αποτελούμενης από πληθώρα στοιχείων ελέγχου διεπαφής χρήστη (UI controls) τα οποία έχουν δημιουργηθεί με στόχο το να είναι ελαφριά, ευέλικτα και φιλικά προς το χρήστη. Η βιβλιοθήκη αυτή έχει αναπτυχθεί σε γλώσσα JavaScript και δεν έχει εξωτερικές εξαρτήσεις. Ακόμα περιλαμβάνει πλήρη υποστήριξη για τα frameworks Angular, React, Vue, ASP.NET MVC και ASP.NET Core.

Το στοιχείο ελέγχου Syncfusion Grid χρησιμοποιείται για την εμφάνιση και τον χειρισμό δεδομένων σε μορφή πίνακα με επιλογές διαμόρφωσης για τον έλεγχο του τρόπου παρουσίασης και χειρισμού των δεδομένων. Αντλεί δεδομένα από μία πηγή δεδομένων όπως αντικείμενα JSON, OData web services ή DataManager και τα εναποθέτει στον πίνακα. Επίσης εμφανίζει επικεφαλίδα στήλης για την αναγνώριση του πεδίου καθώς και υποστήριξη για ομαδοποίηση των εγγραφών. Τα πιο σημαντικά χαρακτηριστικά του περιλαμβάνουν σελιδοποίηση, ταξινόμηση, φιλτράρισμα, αναζήτηση και ομαδοποίηση των εγγραφών.

OrderID	EmployeeID	CustomerID	ShipCountry	Freight
10248	5	VINET	France	32.38
10249	6	TOMSP	Germany	11.61
10250	4	HANAR	Brazil	65.83
10251	3	VICTE	France	41.34
10252	4	SUPRD	Belgium	51.3
10253	3	HANAR	Brazil	58.17
10254	5	CHOPS	Switzerland	22.98
10255	9	RICSU	Switzerland	148.33

1 of 25 pages (200 items)

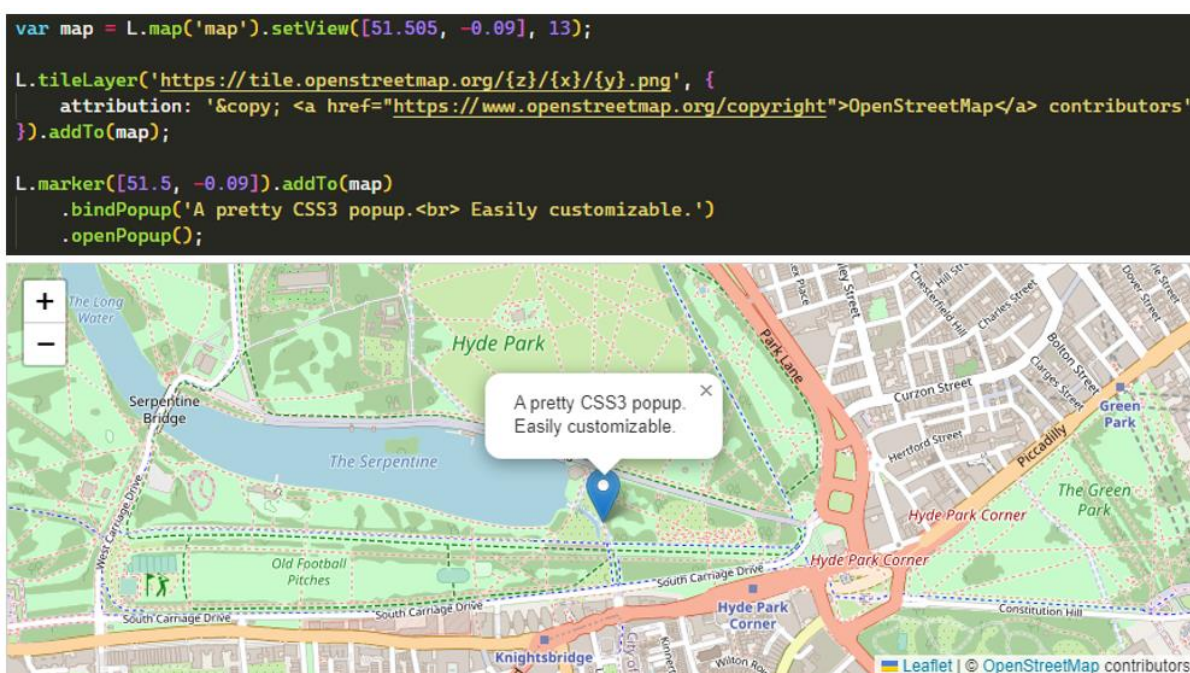
ΕΙΚΟΝΑ 17 - ΠΑΡΑΔΕΙΓΜΑ ΑΠΕΙΚΟΝΙΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΗ ΒΙΒΛΙΟΘΗΚΗ SYNCFUSION GRID

1.3.3 Leaflet

Το Leaflet είναι μια από τις πιο γνωστές βιβλιοθήκες JavaScript για διαδραστικούς χάρτες. Διαθέτει όλες τις δυνατότητες χαρτογράφησης που χρειάζονται οι περισσότεροι προγραμματιστές και είναι λογισμικό ανοιχτού κώδικα.

Δημιουργήθηκε το 2011 από τον Volodymyr Agafonkin με γνώμονα την απλότητα, την απόδοση και τη χρηστικότητα και υποστηρίζει τις περισσότερες πλατφόρμες κινητών και υπολογιστών. Επίσης έχει ένα εύχρηστο και καλά τεκμηριωμένο API. Μεταξύ των χρηστών της βιβλιοθήκης είναι και οι εφαρμογές των FourSquare, Pinterest και Flickr.

Το Leaflet επιτρέπει στους προγραμματιστές χωρίς υπόβαθρο GIS να εμφανίζουν πολύ εύκολα χάρτες που φιλοξενούνται σε δημόσιους εξυπηρετητές, με προαιρετικές επικαλύψεις (overlays). Μπορεί να φορτώσει δεδομένα χαρακτηριστικών από αρχεία GeoJSON, να τα διαμορφώσει και να δημιουργήσει διαδραστικά επίπεδα, όπως για παράδειγμα δείκτες με αναδυόμενα παράθυρα όταν πατάει κάποιος σε αυτούς. Επίσης οι λειτουργίες του μπορούν να επεκταθούν με χρήση εξωτερικών επεκτάσεων.



ΕΙΚΟΝΑ 18 - ΠΑΡΑΔΕΙΓΜΑ ΔΗΜΙΟΥΡΓΙΑΣ ΔΙΑΔΡΑΣΤΙΚΟΥ ΧΑΡΤΗ ΜΕ ΤΗ ΒΙΒΛΙΟΘΗΚΗ LEAFLET

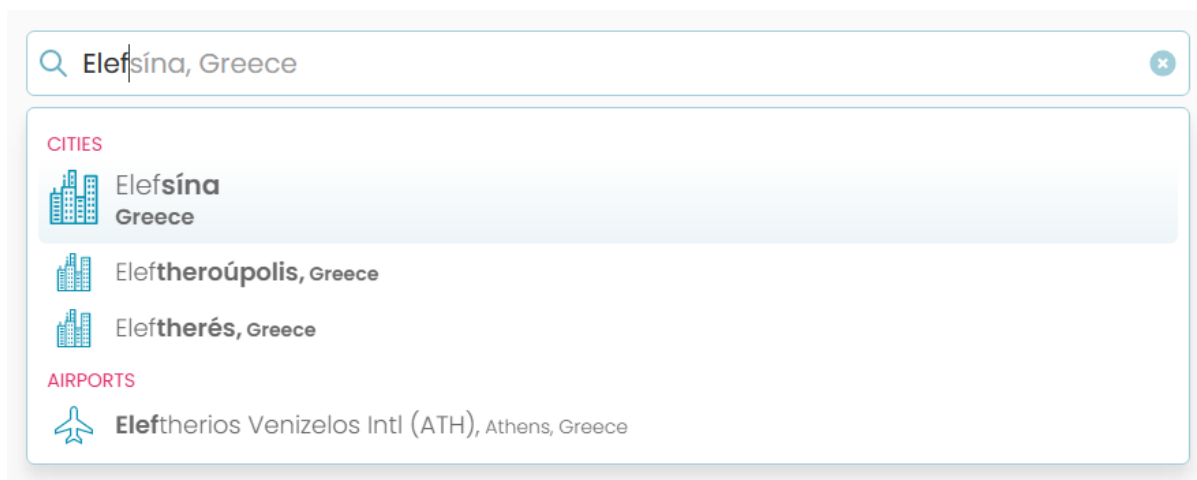
1.3.4 Turnstone

Το Turnstone είναι μια βιβλιοθήκη/component για React που επιτρέπει στον προγραμματιστή να βάλει ένα πλήρως παραμετροποιήσιμο πεδίο αναζήτησης με δυνατότητα αυτόματης συμπλήρωσης στην εφαρμογή του.

Μία σημαντική δυνατότητά του είναι αυτή της ενσωμάτωσης περισσότερων από μία πηγών άντλησης δεδομένων, ώστε ο χρήστης να πραγματοποιεί αναζήτηση για πολλά είδη αποτελεσμάτων παράλληλα. Ακόμα ο προγραμματιστής μπορεί να επιλέξει τον αριθμό αποτελεσμάτων που θα εμφανιστούν αλλά και την αναλογία τους σε περίπτωση χρήσης πολλαπλών πηγών δεδομένων.

Στον τομέα της οπτικής παραμετροποίησης, το Turnstone δίνει πολλές επιλογές, καθώς επιτρέπει τη μορφοποίηση του UI component τόσο με χρήση CSS Modules, αλλά και με το γνωστό CSS Framework, Tailwind. Επίσης στα αποτελέσματα μπορούν να προστεθούν και εικονίδια όπως στο παράδειγμα της εικόνας 19.

Τέλος η βιβλιοθήκη είναι ανοιχτού κώδικα και διαθέτει ένα αρκετά καλογραμμένο documentation εξηγώντας όλες τις πιθανές επιλογές που έχει ένας προγραμματιστής για την ενσωμάτωσή της σε μια διαδικτυακή εφαρμογή.



ΕΙΚΟΝΑ 19 - ΠΑΡΑΔΕΙΓΜΑ ΑΝΑΖΗΤΗΣΗΣ ΕΓΓΡΑΦΩΝ ΜΕ ΧΡΗΣΗ ΤΟΥ REACT COMPONENT TURNSTONE

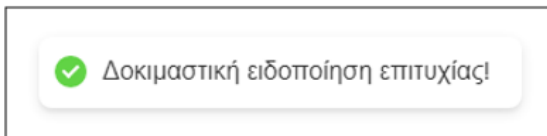
1.3.5 React Hot Toast

Το React Hot Toast αποτελεί μία αρκετά προσαρμόσιμη βιβλιοθήκη/React component που προσθέτει όμορφες ειδοποιήσεις σε εφαρμογές React.

Οι ειδοποιήσεις toast είναι στοιχεία που μοιάζουν με modal και εμφανίζουν πληροφορίες ή οποιοδήποτε μήνυμα στους χρήστες, ως αποτέλεσμα κάποιας ενέργειας. Οι ειδοποιήσεις toast χρησιμοποιούνται συνήθως για την εμφάνιση:

- Ειδοποιήσεων ή προσαρμοσμένων μηνυμάτων με πληροφορίες για το χρήστη.
- Μηνυμάτων επιτυχίας κατά την υποβολή μίας φόρμας.
- Μηνυμάτων σφάλματος σε περίπτωση αποτυχίας αιτημάτων API.

Το React Hot Toast επιτρέπει την εμφάνιση ειδοποιήσεων toast σε οποιοδήποτε σημείο της οθόνης θελήσει ο προγραμματιστής, ενώ οι ειδοποιήσεις είναι πλήρως προσαρμόσιμες, με δυνατότητα δημιουργίας θεμάτων προσθέτοντας μορφοποίηση CSS. Το περιεχόμενο των ειδοποιήσεων μπορεί να περιέχει απλά μηνύματα μιας γραμμής, μηνύματα πολλαπλών γραμμών ή και ολόκληρα components, καθώς επιτρέπει τον καθορισμό του περιεχομένου με χρήση JSX. Επιπλέον ο προγραμματιστής μπορεί να επιλέξει αν οι ειδοποιήσεις θα εξαφανίζονται μετά από κάποιο χρονικό διάστημα αλλά και να το ορίσει.



```
import toast, { Toaster } from 'react-hot-toast';

const notify = () => toast.success('Δοκιμαστική ειδοποίηση επιτυχίας!');

const App = () => {
  return (
    <div>
      <button onClick={notify}>Make me a toast</button>
      <Toaster />
    </div>
  );
};
```

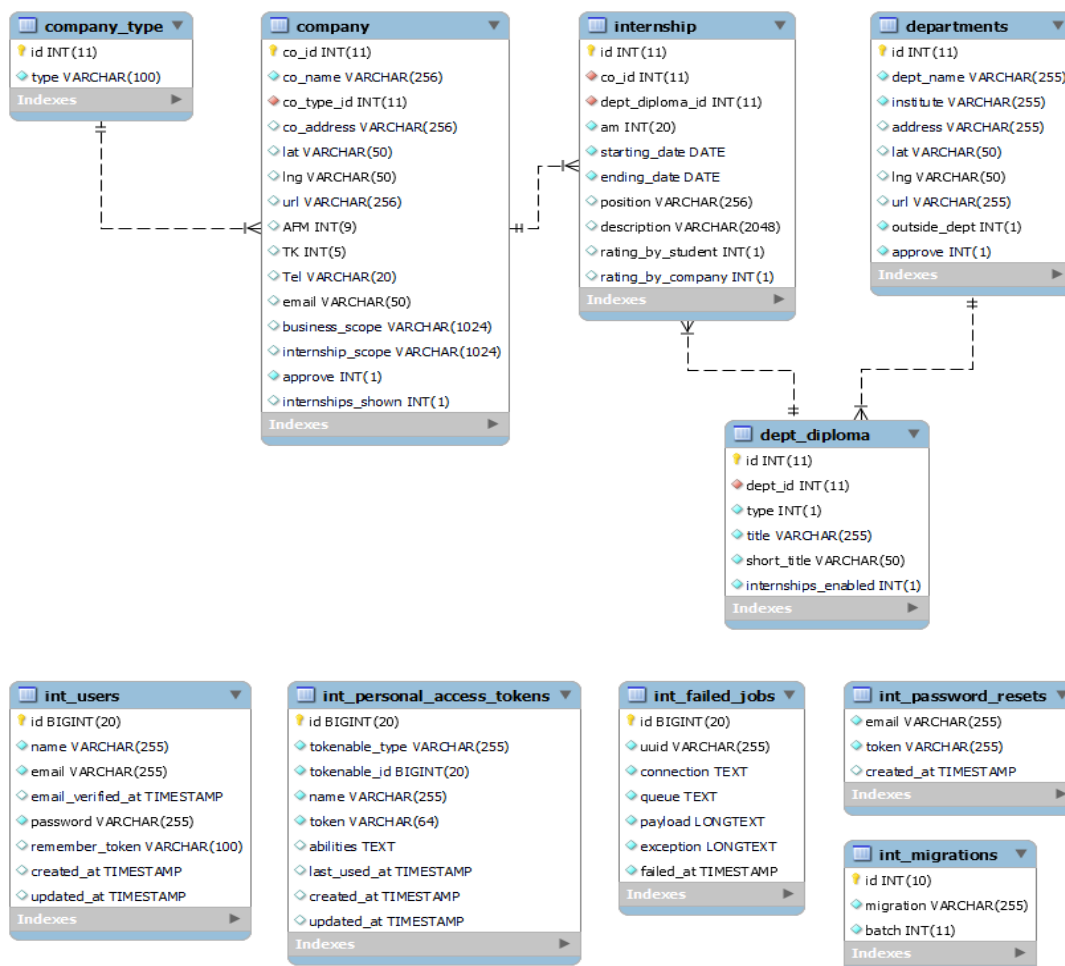
ΕΙΚΟΝΑ 20 - ΠΑΡΑΔΕΙΓΜΑ ΔΗΜΙΟΥΡΓΙΑΣ ΕΙΔΟΠΟΙΗΣΗΣ ΜΕ ΧΡΗΣΗ ΤΟΥ COMPONENT REACT-HOT-TOAST

ΚΕΦΑΛΑΙΟ 2 - ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ

2.1 Βάση Δεδομένων

Η βάση δεδομένων της εφαρμογής αποτελείται από 5 βασικούς πίνακες και 5 πίνακες που είναι σημαντικοί για το back-end της εφαρμογής, ωστόσο δεν χρειάζονται κάποια ξεχωριστή ανάλυση. Στους τελευταίους περιλαμβάνονται ο «int_migrations», ο οποίος αποτελεί ένα version control για τις αλλαγές στη δομή της βάσης δεδομένων, ο «int_users» ο οποίος περιέχει τους χρήστες της εφαρμογής, ο «int_personal_access_tokens» στον οποίο αποθηκεύονται API tokens, ο «int_password_resets» στον οποίο αποθηκεύονται τα tokens που δημιουργούνται όταν ένας χρήστης ζητά επαναφορά κωδικού πρόσβασης και τέλος ο «int_failed_jobs» όπου και καταγράφονται οι αποτυχημένες εκτελέσεις διεργασιών. Το πρόθεμα «int» στους παραπάνω πίνακες αναφέρεται στη λέξη internship (πρακτική άσκηση), καθώς τη βάση δεδομένων θα τη χρησιμοποιεί και μια ξεχωριστή εφαρμογή που αφορά την εργασιακή πορεία των φοιτητών μετά την αποφοίτησή τους (alumni).

Στην εικόνα 21 απεικονίζεται το διάγραμμα οντοτήτων και συσχετίσεων της βάσης δεδομένων (Entity Relationship Diagram). Στα υπό-κεφάλαια που ακολουθούν θα γίνει παρουσίαση των βασικών πινάκων της εφαρμογής, ξεκινώντας με τους πίνακες που σχετίζονται με τα τμήματα που αναθέτουν την πρακτική άσκηση, ακολουθούμενους από τους πίνακες που σχετίζονται με τους φορείς στους οποίους εκτελείται η πρακτική άσκηση και τέλος αυτούς που σχετίζονται με την ίδια την πρακτική άσκηση.



ΕΙΚΟΝΑ 21 - ΔΙΑΓΡΑΜΜΑ ΟΝΤΟΤΗΤΩΝ ΚΑΙ ΣΥΣΧΕΤΙΣΕΩΝ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

2.1.1 Πίνακες τμημάτων

Ο πρώτος πίνακας που σχετίζεται με τα τμήματα ονομάζεται «departments» και χρησιμοποιείται για τον καθορισμό των τμημάτων που θα συμμετέχουν στην εφαρμογή. Η καταχώρηση τμημάτων γίνεται μόνο από τους διαχειριστές της βάσης δεδομένων. Τα πεδία του πίνακα είναι τα εξής:

- **id:** αποτελεί το πρωτεύον κλειδί του πίνακα και για κάθε νέα εγγραφή αυξάνεται κατά μία μονάδα.
- **dept_name:** περιλαμβάνει το όνομα του τμήματος.
- **institute:** περιλαμβάνει το όνομα του ακαδημαϊκού ιδρύματος.
- **address:** περιλαμβάνει τη διεύθυνση του τμήματος.
- **lat:** περιλαμβάνει το γεωγραφικό μήκος από τις συντεταγμένες του τμήματος για εύκολη εμφάνιση σε χάρτη.
- **lng:** περιλαμβάνει το γεωγραφικό πλάτος από τις συντεταγμένες του τμήματος για εύκολη εμφάνιση σε χάρτη.
- **url:** περιλαμβάνει τη διεύθυνση URL για την ιστοσελίδα του τμήματος.
- **outside_dept:** τιμή boolean που περιγράφει αν το τμήμα είναι εσωτερικό σε σχέση με το ίδρυμα που εγκαθιστά την εφαρμογή ή όχι.
- **approve:** τιμή boolean για την έγκριση ή όχι του τμήματος. Η τιμή αυτή χρησιμοποιείται για να εμφανίζεται ή όχι το τμήμα σε ολόκληρη την εφαρμογή.

Ο επόμενος πίνακας ονομάζεται «dept_diploma» και περιέχει καταχωρήσεις για κάθε πρόγραμμα σπουδών/τίτλο πτυχίου στο οποίο μπορεί να ανήκουν οι φοιτητές που κάνουν πρακτική άσκηση. Ο πίνακας αυτός συμπληρώνεται επίσης από τους διαχειριστές της βάσης δεδομένων. Τα πεδία του πίνακα έχουν ως εξής:

- **id:** αποτελεί το πρωτεύον κλειδί του πίνακα και για κάθε νέα εγγραφή αυξάνεται κατά μία μονάδα.
- **dept_id:** αποτελεί ξένο κλειδί που κάνει αναφορά στον πίνακα «departments» και δηλώνει το id του τμήματος στο οποίο υπάγεται το πρόγραμμα σπουδών.
- **type:** μπορεί να πάρει είτε την τιμή 1 που σημαίνει ότι αφορά βασικό πτυχίο, είτε την τιμή 2 που σημαίνει ότι αφορά κάποιο πρόγραμμα μεταπτυχιακού ή την τιμή 3 αν αφορά κάποιο διδακτορικό.
- **title:** περιλαμβάνει τον τίτλο του προγράμματος σπουδών.
- **short_title:** περιλαμβάνει μια συντομογραφία του τίτλου προγράμματος σπουδών.
- **internships_enabled:** τιμή boolean που χρησιμοποιείται για την εμφάνιση ή μη του προγράμματος σπουδών στην εφαρμογή καθώς ο πίνακας αυτός θα χρησιμοποιείται και από την εξωτερική εφαρμογή για την εργασιακή πορεία των φοιτητών μετά την αποφοίτησή τους (alumni).

2.1.2 Πίνακες φορέων

Οι πίνακες που σχετίζονται με τους φορείς στους οποίους πραγματοποιούνται οι πρακτικές ασκήσεις είναι δύο. Ο πρώτος ονομάζεται «company_type» και σε αυτόν αποθηκεύονται οι τύποι φορέων. Τα πεδία του πίνακα είναι τα εξής:

- **id:** αποτελεί το πρωτεύον κλειδί του πίνακα και για κάθε νέα εγγραφή αυξάνεται κατά μία μονάδα.
- **type:** περιλαμβάνει το όνομα του τύπου φορέα, για παράδειγμα «Δημόσιο», «Ιδιωτικός Τομέας» ή «Τράπεζες».

Ο δεύτερος πίνακας που σχετίζεται με τους φορείς υποδοχής φοιτητών για την πραγματοποίηση πρακτικής άσκησης ονομάζεται «company». Είναι ένας από τους βασικούς πίνακες με τον οποίο θα αλληλεπιδρά η εφαρμογή και περιέχει τα στοιχεία όλων των φορέων που εμφανίζονται σε αυτήν. Τα πεδία του έχουν ως εξής:

- **co_id:** αποτελεί το πρωτεύον κλειδί του πίνακα και για κάθε νέα εγγραφή αυξάνεται κατά μία μονάδα.
- **co_name:** περιέχει το όνομα του φορέα.
- **co_type_id:** αποτελεί ξένο κλειδί που κάνει αναφορά στον πίνακα «company_type» και δηλώνει το id του τύπου φορέα. Κάθε φορέας υποχρεωτικά αντιστοιχίζεται με ένα τύπο φορέα.
- **co_address:** περιλαμβάνει τη διεύθυνση του φορέα.
- **lat:** περιλαμβάνει το γεωγραφικό μήκος από τις συντεταγμένες του φορέα για να είναι δυνατή η εμφάνισή του στο χάρτη.
- **lng:** περιλαμβάνει το γεωγραφικό πλάτος από τις συντεταγμένες του φορέα για να είναι δυνατή η εμφάνισή του στο χάρτη.
- **url:** περιλαμβάνει τη διεύθυνση URL για την ιστοσελίδα του φορέα.
- **AFM:** περιλαμβάνει τον Αριθμό Φορολογικού Μητρώου (ΑΦΜ) του φορέα.
- **TK:** περιλαμβάνει τον Ταχυδρομικό Κώδικα της διεύθυνσης του φορέα.
- **Tel:** περιλαμβάνει τον τηλεφωνικό αριθμό του φορέα.
- **email:** περιλαμβάνει τη διεύθυνση ηλεκτρονικού ταχυδρομείου του φορέα.
- **business_scope:** περιλαμβάνει τα αντικείμενα της επιχείρησης του φορέα σε ελεύθερο κείμενο.
- **internship_scope:** περιλαμβάνει τα αντικείμενα των πρακτικών ασκήσεων που αναλαμβάνει ο φορέας σε ελεύθερο κείμενο.
- **approve:** τιμή boolean για την έγκριση ή όχι του φορέα. Την τιμή αυτή δύναται να την αλλάξει ο διαχειριστής της εφαρμογής αφού επιβεβαιώσει ότι τα στοιχεία του φορέα είναι ορθά.
- **internships_shown:** τιμή boolean που χρησιμοποιείται για την εμφάνιση ή μη του φορέα στην εφαρμογή καθώς ο πίνακας αυτός θα χρησιμοποιείται και από την εξωτερική εφαρμογή για την εργασιακή πορεία των φοιτητών μετά την αποφοίτησή τους (alumni).

2.1.3 Πίνακας πρακτικής άσκησης

Κλείνοντας την περιγραφή των πινάκων της βάσης δεδομένων της εφαρμογής, οφείλουμε να αναφερθούμε σε ίσως το σημαντικότερο πίνακα όσον αφορά την εφαρμογή. Ο πίνακας αυτός ονομάζεται «internship» και είναι ο πίνακας στον οποίο αποθηκεύονται όλες οι πρακτικές ασκήσεις που έχουν πραγματοποιηθεί από φοιτητές του τμήματος. Ως εκ τούτου, συσχετίζει τον φοιτητή με το φορέα που πραγματοποιήθηκε η πρακτική άσκηση και το πρόγραμμα σπουδών το οποίο παρακολουθεί, ώστε να αναφερθεί στις λεπτομέρειες της θέσης της πρακτικής άσκησης του. Τα πεδία του πίνακα έχουν ως εξής:

- **id:** αποτελεί το πρωτεύον κλειδί του πίνακα και για κάθε νέα εγγραφή αυξάνεται κατά μία μονάδα.
- **co_id:** αποτελεί ξένο κλειδί που κάνει αναφορά στον πίνακα «company» και δηλώνει το id του φορέα.
- **dept_diploma_id:** αποτελεί ξένο κλειδί που κάνει αναφορά στον πίνακα «dept_diploma» και δηλώνει το id του προγράμματος σπουδών που παρακολουθεί ο φοιτητής.
- **am:** περιέχει τον Αριθμό Μητρώου που έχει ο φοιτητής στο ίδρυμα όπου φοιτά.
- **starting_date:** περιέχει την ημερομηνία έναρξης της πρακτικής άσκησης.
- **ending_date:** περιέχει την ημερομηνία λήξης της πρακτικής άσκησης.
- **position:** περιέχει τον τίτλο της θέσης πρακτικής άσκησης.
- **description:** περιέχει μία περιγραφή της θέσης πρακτικής άσκησης.
- **rating_by_student:** περιέχει την αξιολόγηση του φορέα από τον φοιτητή σε μία κλίμακα από το 1 έως το 5 για την εμπειρία του στη θέση πρακτικής άσκησης. Τα δεδομένα από το πεδίο αυτό δύναται να χρησιμοποιηθούν στο μέλλον για να εμφανιστούν στατιστικά στην εφαρμογή για τη γνώμη των φοιτητών για το συγκεκριμένο φορέα, όσον αφορά την πρακτική τους άσκηση.
- **rating_by_company:** περιέχει την αξιολόγηση του φοιτητή από τον φορέα σε μία κλίμακα από το 1 έως το 5 για την εμπειρία που είχαν κατά τη διάρκεια της πρακτικής άσκησης. Τα δεδομένα από το πεδίο αυτό δύναται να χρησιμοποιηθούν στο μέλλον για να εμφανιστούν στατιστικά στην εφαρμογή για τη γνώμη των φορέων για τους φοιτητές του τμήματος που έχουν πραγματοποιήσει την πρακτική τους άσκηση σε αυτούς.

2.2 Κατηγορίες Χρηστών

Η εφαρμογή παρακολούθησης φορέων και θέσεων πρακτικής άσκησης απευθύνεται σε δύο είδη χρηστών, τους διαχειριστές του συστήματος, οι οποίοι θα είναι είτε το γραφείο πρακτικής άσκησης είτε η γραμματεία του τμήματος και τους επισκέπτες της εφαρμογής, που θα είναι οι φοιτητές του τμήματος αλλά και όποιοι άλλοι θέλουν να πληροφορηθούν για τις πρακτικές ασκήσεις των φοιτητών του τμήματος.

Οι διαχειριστές έχουν συγκεκριμένη διεύθυνση URL μέσα από την οποία θα μπορούν να συνδεθούν για να έχουν πρόσβαση στις σελίδες και λειτουργίες που αφορούν τους διαχειριστές, ενώ οι επισκέπτες μπορούν να περιηγηθούν στην εφαρμογή χωρίς να απαιτείται κάποια σύνδεση.

Για να μπει κάποιος στο διαχειριστικό περιβάλλον της εφαρμογής, αρκεί να πληκτρολογήσει τη διεύθυνση <https://praktiki.ece.uop.gr/admin/login>. Το πρώτο πράγμα που αντικρίζει ο διαχειριστής είναι η φόρμα εισόδου στην πλατφόρμα (login). Εκεί καταχωρεί τα στοιχεία σύνδεσής του, δηλαδή το email και τον κωδικό πρόσβασής του. Εφόσον τα στοιχεία τα οποία θα καταχωρήσει είναι αληθή, θα εισέλθει στην διαχειριστική πλατφόρμα. Σε αντίθετη περίπτωση, θα λάβει μήνυμα σφάλματος. Μετά τη σύνδεσή του στην πλατφόρμα, δύο νέες εγγραφές θα εμφανιστούν στην μπάρα περιήγησης (navigation bar) της σελίδας. Αυτές είναι η «Διαχείριση Φορέων» και «Διαχείριση Πρακτικών» και όπως παραπέμπει και η ονομασία τους, επιτρέπουν στο διαχειριστή να διαχειριστεί τους φορείς που συμμετέχουν στην εφαρμογή αλλά και τις θέσεις πρακτικής άσκησης των φοιτητών του τμήματος.

Το περιβάλλον των επισκεπτών, που κατά βάση θα είναι φοιτητές που ενδιαφέρονται να μάθουν για τις πρακτικές ασκήσεις άλλων φοιτητών του τμήματος, είναι αυτό που εμφανίζεται όταν κάποιος μπαίνει στην εφαρμογή, πληκτρολογώντας τη διεύθυνση <https://praktiki.ece.uop.gr>. Στη σελίδα αυτή μπορεί να δει τι αφορά η συγκεκριμένη εφαρμογή αλλά και το πως να τη χρησιμοποιήσει. Η βασική σελίδα στην οποία οι οδηγίες παραπέμπουν είναι αυτή που φιλοξενεί ένα χάρτη ο οποίος περιέχει όλους τους φορείς στους οποίους έχουν πραγματοποιηθεί πρακτικές ασκήσεις από φοιτητές του τμήματος.

2.3 Εφαρμογή τεχνολογιών

Ο σχεδιασμός της εφαρμογής έχει γίνει βάση των τελευταίων τεχνολογιών ανάπτυξης εφαρμογών διαδικτύου αλλά και βάσεων δεδομένων. Στην παρούσα εργασία έχουν χρησιμοποιηθεί ονομαστικά οι τεχνολογίες HTML, CSS, JavaScript, React JS, Bootstrap, Laravel (PHP) και MariaDB (MySQL).

Η επιλογή για τη χρήση της Laravel για το backend ήταν αρκετά εύκολη καθώς παρέχει έτοιμες ολοκληρωμένες λύσεις για μία σειρά από τμήματα της εφαρμογής, τα οποία θα παρουσίαζαν σημαντική δυσκολία αν η υλοποίησή τους έπρεπε να γίνει από την αρχή. Σε αυτά περιλαμβάνονται η λύση πιστοποίησης ταυτότητας των χρηστών, Sanctum, η οποία έχει χρησιμοποιηθεί για την πιστοποίηση χρηστών με χρήση cookies και το εργαλείο ORM (Object-Oriented Mapper) που επιτρέπει την εύκολη και απρόσκοπτη αλληλεπίδραση με βάσεις δεδομένων, όπως η MariaDB που έχει χρησιμοποιηθεί στην εφαρμογή. Ακόμα ενσωματώνει αρκετά χαρακτηριστικά ασφαλείας, όπως προστασία από SQL injection και από ευπάθειες CSRF και XSS.

Στο κομμάτι του frontend, η χρήση της React JS έγινε για αρκετούς λόγους. Σε αυτούς περιλαμβάνεται η ταχύτητά της σε σχέση με άλλα JavaScript frameworks, η χρήση components που επιτρέπουν την επαναχρησιμοποίηση κώδικα σε αρκετά σημεία της εφαρμογής και η δυνατότητα εύκολης προσαρμογής της εφαρμογής για χρήση σε συσκευές με μικρές οθόνες (κινητά, τάμπλετ). Ένας ακόμα σημαντικός παράγοντας ήταν η μεγάλη δημοτικότητα της, η οποία έχει ως αποτέλεσμα την ύπαρξη πολλών πρόσθετων βιβλιοθηκών για κάθε ανάγκη αλλά και η προύπηρεσία του συγγραφέα της εργασίας στη χρήση της React για ανάπτυξη εφαρμογών.

Σημαντικές προσθήκες στο κομμάτι του frontend ήταν αυτές της βιβλιοθήκης Bootstrap η οποία περιέχει μια σειρά από προκαθορισμένα χαρακτηριστικά για πληθώρα στοιχείων της σελίδας, όπως κουμπιά, παράθυρα διαλόγου (modals) αλλά και στοιχεία διάταξης, καθώς και αρκετές ακόμα βιβλιοθήκες που στάθηκαν χρήσιμες σε μια σειρά από λειτουργίες της εφαρμογής. Σε αυτές περιλαμβάνονται η βιβλιοθήκη Axios για τη δημιουργία ασύγχρονων αιτημάτων προς το backend, η Syncfusion Grid που χρησιμοποιήθηκε για την εμφάνιση πινάκων μέσα στην εφαρμογή, η Leaflet, που αποτελεί μία από τις πιο γνωστές βιβλιοθήκες για εμφάνιση διαδραστικών χαρτών και έχει ρυθμιστεί για να αντλεί γεωγραφικά δεδομένα από το OpenStreetMap project, η Turnstone, μια βιβλιοθήκη για React που χρησιμοποιήθηκε για τη λειτουργικότητα της αναζήτησης φορέων και θέσεων πρακτικής άσκησης και τέλος η βιβλιοθήκη React Hot Toast ή οποία επέτρεψε τη δημιουργία ειδοποιήσεων και προσαρμοσμένων μηνυμάτων προς το χρήστη για διάφορες ενέργειες της εφαρμογής.

ΚΕΦΑΛΑΙΟ 3 - ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ / ΣΕΝΑΡΙΑ ΧΡΗΣΗΣ

Στο κεφάλαιο αυτό, θα παρουσιαστεί η εφαρμογή παρακολούθησης φορέων και θέσεων πρακτικής άσκησης που υλοποιήθηκε στα πλαίσια της παρούσας πτυχιακής εργασίας, μέσα από κάποια σενάρια χρήσης της, τόσο από την οπτική ενός επισκέπτη - φοιτητή, όσο και από την οπτική ενός διαχειριστή - γραμματεία / γραφείο πρακτικής άσκησης.

3.1 Επισκέπτης - Φοιτητής

3.1.1 Πλοήγηση στην εφαρμογή

Η εφαρμογή από την πλευρά του επισκέπτη - φοιτητή είναι σχεδιασμένη με τέτοιο τρόπο, ώστε η περιήγηση σε αυτή να μην απαιτεί κάποια σύνδεση ή εγγραφή. Στην αρχική σελίδα της εφαρμογής, ο χρήστης μπορεί να δει πληροφορίες για την εφαρμογή και το πως να τη χρησιμοποιήσει.

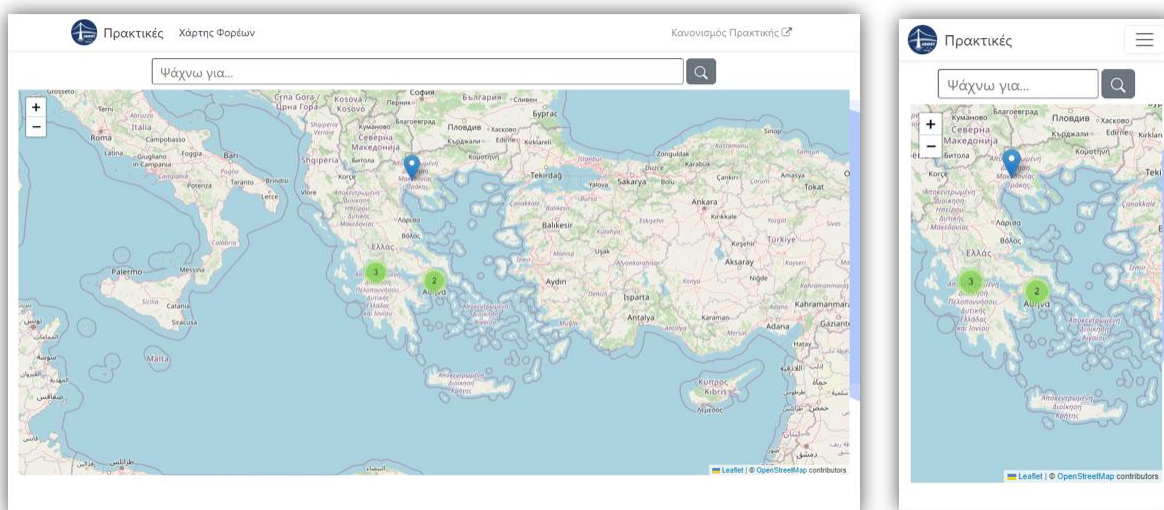
Στο αριστερά μέρος της μπάρας περιήγησης (navigation bar) της σελίδας, ο χρήστης μπορεί να δει το λογότυπο της εφαρμογής μέσω του οποίου μπορεί ανά πάσα στιγμή να επιστρέψει στην αρχική σελίδα της εφαρμογής. Δίπλα σε αυτό βρίσκεται υπερσύνδεσμος για τη βασικότερη σελίδα της εφαρμογής, αυτή του χάρτη φορέων. Τέλος στο δεξιά μέρος της μπάρας περιήγησης, βρίσκεται ένας υπερσύνδεσμος που θα ανακατευθύνει τον επισκέπτη στην ιστοσελίδα του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών και συγκεκριμένα στην ειδική σελίδα που φιλοξενεί πληροφορίες για την πρακτική άσκηση αλλά και τον κανονισμό της, όπως αυτός έχει οριστεί από το τμήμα.



ΕΙΚΟΝΑ 22 - Η ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΑΠΟ ΤΗΝ ΟΠΤΙΚΗ ΤΟΥ ΕΠΙΣΚΕΠΤΗ

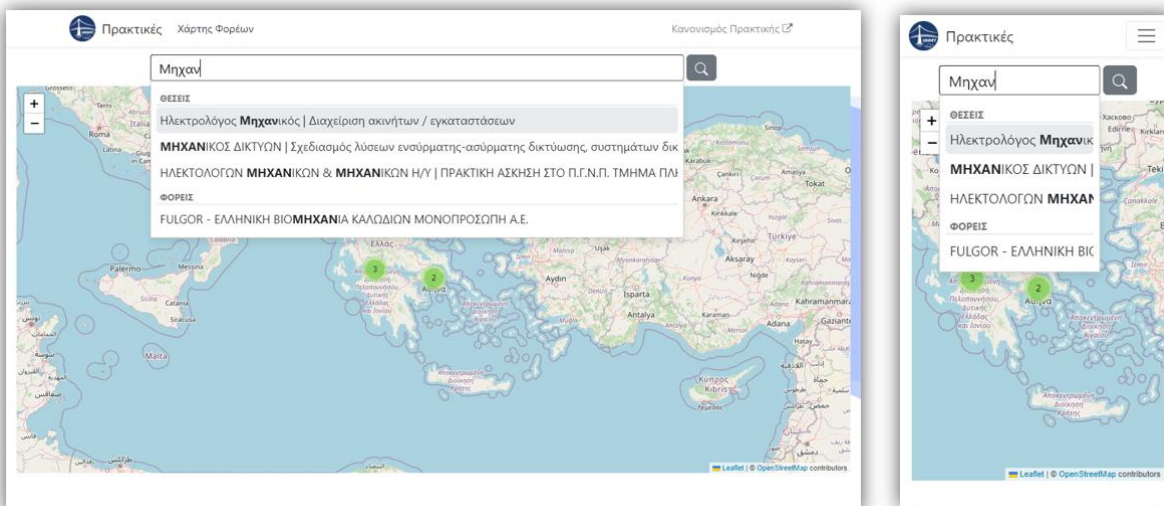
3.1.2 Αναζήτηση φορέα ή θέσης πρακτικής άσκησης

Όταν ο χρήστης της εφαρμογής πατήσει στον υπερσύνδεσμο «Χάρτης Φορέων» ή στο αντίστοιχο κουμπί που βρίσκεται στην αρχική σελίδα, θα ανακατευθυνθεί σε μία σελίδα στην οποία κυρίαρχη θέση παίζει ένας χάρτης. Στον χάρτη αυτό απεικονίζονται όλοι οι φορείς στους οποίους έχουν πραγματοποιηθεί πρακτικές ασκήσεις από τους φοιτητές του τμήματος. Ο χρήστης μπορεί να εκτελέσει ενέργειες πάνω στο χάρτη όπως μεγέθυνση, σμίκρυνση και μετακίνηση. Κάθε φορέας εμφανίζεται με ένα δείκτη χρώματος μπλε. Αν υπάρχουν πολλοί φορείς σε μία συγκεκριμένη περιοχή, αυτοί θα ομαδοποιηθούν αυτόματα και στο χάρτη θα εμφανίζεται ένας πράσινος κύκλος μέσα στον οποίο αναγράφεται ο αριθμός των ομαδοποιημένων φορέων. Πατώντας πάνω στον κύκλο αυτό, ο χάρτης αυτόματα μεγεθύνεται ώστε να περιέχει όλους τους δείκτες της εν λόγω κατηγοριοποίησης.



ΕΙΚΟΝΑ 23 - Η ΣΕΛΙΔΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ «ΧΑΡΤΗΣ ΦΟΡΕΩΝ»

Μια χρήσιμη λειτουργία για το χρήστη της εφαρμογής είναι αυτή της αναζήτησης φορέων και θέσεων πρακτικής άσκησης. Αυτό γίνεται μέσα από τη μπάρα αναζήτησης που βρίσκεται ακριβώς πάνω από το χάρτη. Γράφοντας σε αυτήν, ο χρήστης θα παρατηρήσει ότι υπάρχει αυτόματη συμπλήρωση σε αυτήν, εμφανίζοντας αποτελέσματα που αντιστοιχούν στους όρους που έχει πληκτρολογήσει.

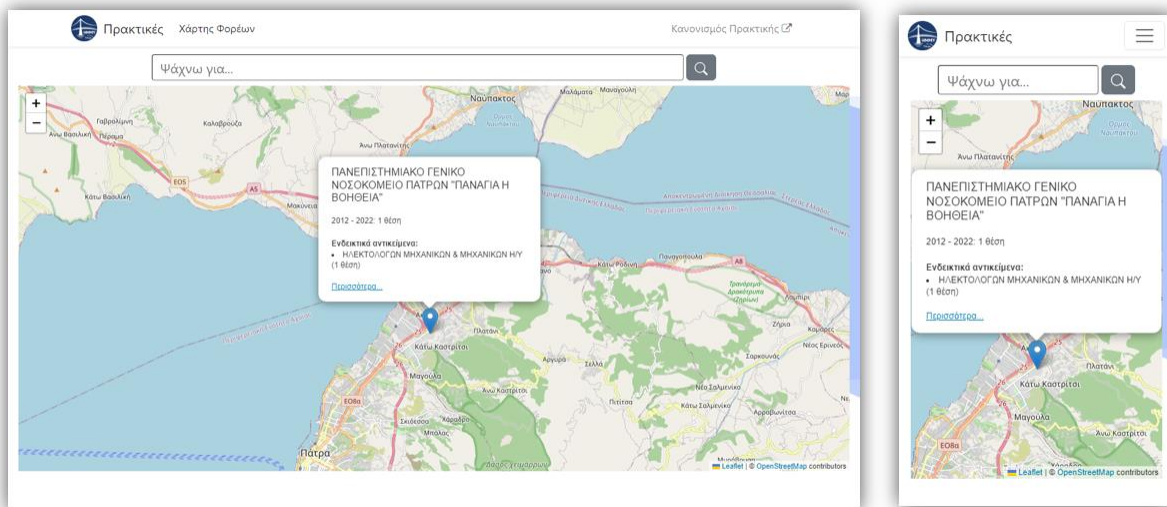


ΕΙΚΟΝΑ 24 - ΑΝΑΖΗΤΗΣΗ ΦΟΡΕΩΝ ΚΑΙ ΘΕΣΕΩΝ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ

Όπως φαίνεται και στην εικόνα 24, τα αποτελέσματα αναζήτησης περιέχουν τόσο θέσεις πρακτικής άσκησης, όσο και φορείς. Στους φορείς η αναζήτηση περιλαμβάνει το όνομα του φορέα και τα αντικείμενα φορέα και πρακτικής, αν τα εν λόγω πεδία είναι συμπληρωμένα. Για την περίπτωση των θέσεων πρακτικής άσκησης η αναζήτηση περιλαμβάνει τον τίτλο της θέσης αλλά και την περιγραφή της.

Πατώντας ο χρήστης το κουμπί της αναζήτησης ή το πλήκτρο «enter», η αναζήτηση θα εκτελεστεί για τους όρους τους οποίους έχει πληκτρολογήσει, και στο χάρτη θα μείνουν μόνο οι δείκτες των φορέων που αντιστοιχούν στα αποτελέσματα. Αντίστοιχα, αν επιλέξει κάποιο από τα αποτελέσματα που εμφανίζονται αυτόματα κατά την πληκτρολόγηση, στον χάρτη θα παραμείνει μόνο ο φορέας που σχετίζεται με το συγκεκριμένο αποτέλεσμα.

Όταν ο χρήστης πατήσει σε κάποιο από τους δείκτες, ένα μικρό αναδυόμενο παράθυρο θα εμφανιστεί, με πληροφορίες για τον φορέα τον οποίο αντιπροσωπεύει. Στο παράθυρο αυτό, αναγράφεται η ονομασία του φορέα, ο αριθμός των θέσεων πρακτικής άσκησης οι οποίες έχουν πραγματοποιηθεί σε αυτόν τα τελευταία 10 χρόνια και μέχρι τρία ενδεικτικά αντικείμενα πρακτικών, μαζί με τον αριθμό θέσεων του κάθε αντικειμένου.

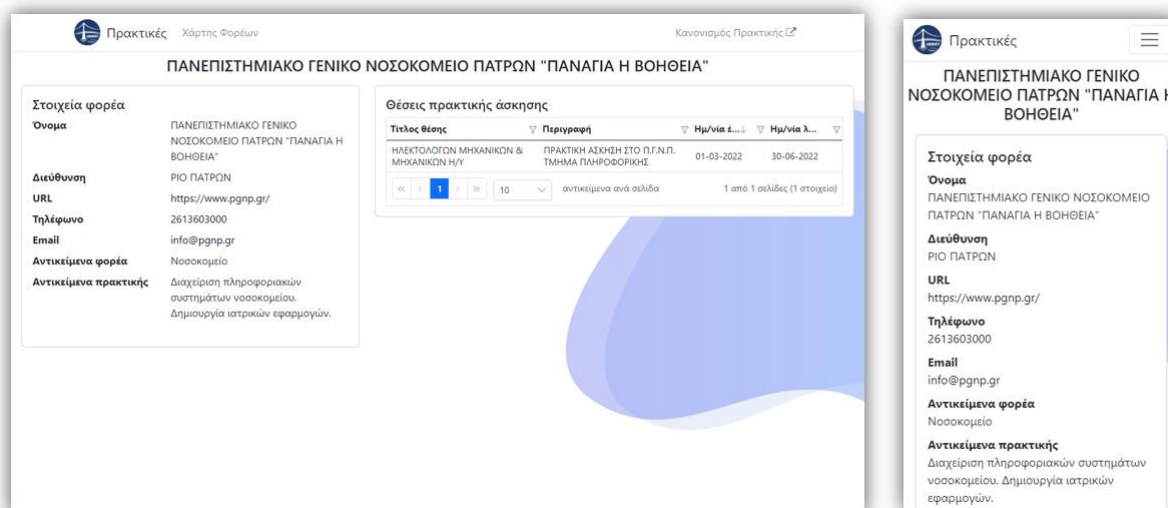


ΕΙΚΟΝΑ 25 - ΑΝΑΔΥΟΜΕΝΟ ΠΑΡΑΘΥΡΟ ΜΕ ΠΛΗΡΟΦΟΡΙΕΣ ΓΙΑ ΤΟ ΦΟΡΕΑ

Στο κάτω μέρος του παραθύρου υπάρχει ένας υπερσύνδεσμος, ο οποίος, όταν πατηθεί, θα ανοίξει σε μία καινούρια καρτέλα μια σελίδα με τα λεπτομερή στοιχεία του φορέα αλλά και τις θέσεις πρακτικής άσκησης σε αυτόν.

3.1.3 Προβολή λεπτομερειών φορέα και θέσεων πρακτικής άσκησης

Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, για κάθε φορέα υπάρχει μία ξεχωριστή σελίδα η οποία παρουσιάζει λεπτομερώς τα στοιχεία του αλλά και τις θέσεις πρακτικής άσκησης σε αυτόν.



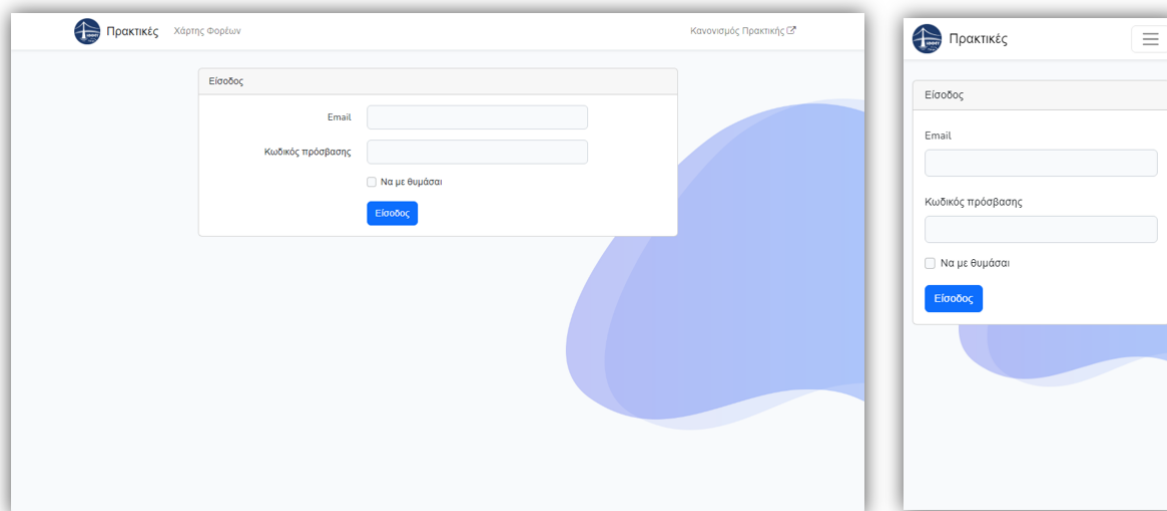
ΕΙΚΟΝΑ 26 - ΣΕΛΙΔΑ ΦΟΡΕΑ ΜΕ ΠΛΗΡΟΦΟΡΙΕΣ ΚΑΙ ΘΕΣΕΙΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ

Όπως φαίνεται και στην εικόνα 26, η σελίδα αυτή είναι χωρισμένη σε δύο τμήματα. Στο αριστερά τμήμα βρίσκονται τα λεπτομερή στοιχεία του φορέα. Αυτά περιλαμβάνουν, το όνομα, τη διεύθυνση, το σύνδεσμο URL για την ιστοσελίδα του, το τηλέφωνο επικοινωνίας, το email, τα αντικείμενα φορέα και τα αντικείμενα πρακτικών ασκήσεων του. Στο δεξιά τμήμα βρίσκεται ένας πίνακας ο οποίος περιλαμβάνει τις θέσεις πρακτικής άσκησης που έχουν πραγματοποιηθεί στον φορέα, αναγράφοντας τον τίτλο της θέσης, την περιγραφή της αλλά και την ημερομηνία έναρξης και λήξης της πρακτικής άσκησης. Ο πίνακας εμφανίζει από προεπιλογή τις 10 πιο πρόσφατες θέσεις πρακτικής άσκησης ταξινομημένες ανά φθίνουσα ημερομηνία έναρξης, αλλά δίνει δυνατότητα στο χρήστη μέσω σελιδοποίησης να δει και παλιότερες θέσεις ή να επιλέξει να εμφανιστούν περισσότερα αποτελέσματα στην ίδια σελίδα. Ο χρήστης μπορεί ακόμα να φιλτράρει και να ταξινομήσει τα στοιχεία του πίνακα όπως επιθυμεί.

3.2 Διαχειριστής - Γραμματεία / Γραφείο πρακτικής άσκησης

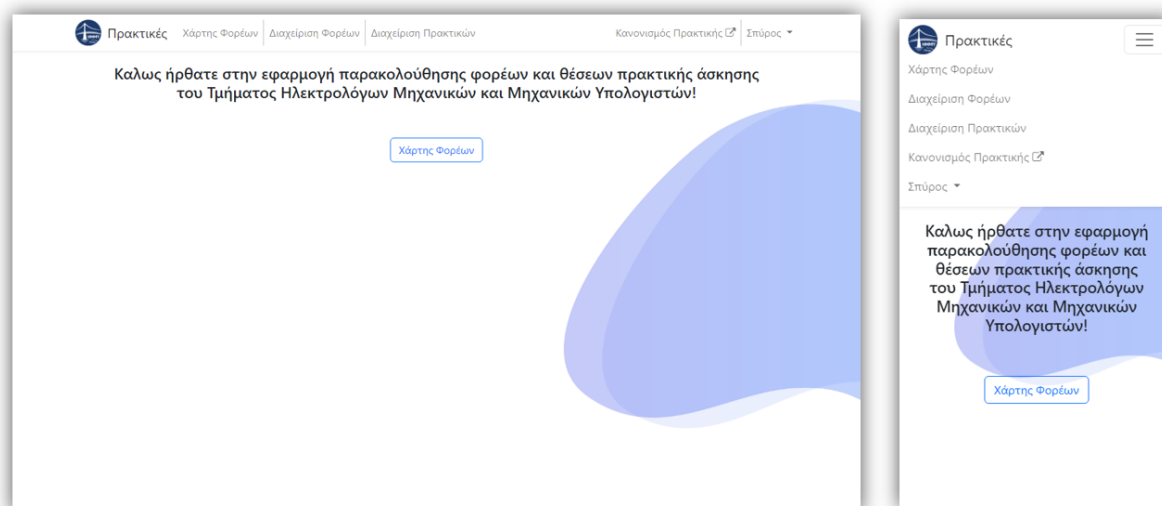
3.2.1 Πλοήγηση στην εφαρμογή

Η εφαρμογή από την πλευρά του διαχειριστή, ο οποίος μπορεί να είναι η γραμματεία του τμήματος ή το γραφείο πρακτικής άσκησης του ιδρύματος, περιέχει μερικές σελίδες στις οποίες οι απλοί επισκέπτες δεν έχουν πρόσβαση. Αυτό επιτυγχάνεται ζητώντας από τους διαχειριστές να κάνουν σύνδεση στην εφαρμογή. Πηγαίνοντας στη διεύθυνση <https://praktiki.ece.uop.gr/admin/login> βρίσκουν μπροστά τους μία σελίδα σύνδεσης, στην οποία θα βάλουν το email και τον κωδικό τους.



ΕΙΚΟΝΑ 27 - Η ΣΕΛΙΔΑ ΣΥΝΔΕΣΗΣ ΓΙΑ ΤΟΥΣ ΔΙΑΧΕΙΡΙΣΤΕΣ

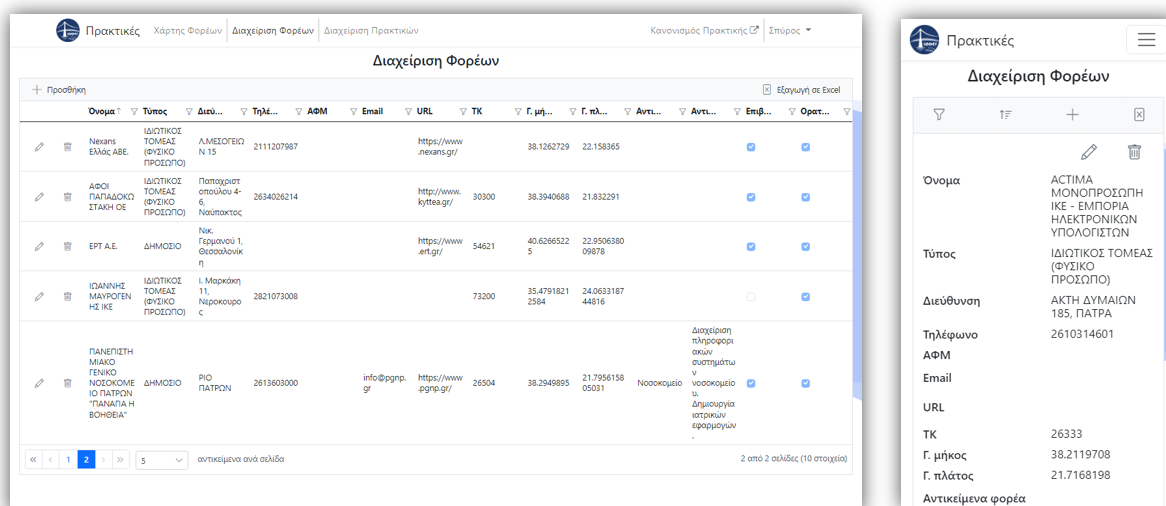
Μόλις το κάνουν αυτό, θα ανακατευθυνθούν στην αρχική σελίδα, όπου δυο καινούριες επιλογές θα έχουν εμφανιστεί στη μπάρα περιήγησης (navigation bar), μαζί με αυτές που είναι εμφανίσιμες και στους επισκέπτες της εφαρμογής. Οι επιλογές αυτές είναι οι «Διαχείριση Φορέων» και «Διαχείριση Πρακτικών». Επίσης πατώντας στο όνομα χρήστη που εμφανίζεται στο τέλος της μπάρας περιήγησης, μπορούν να κάνουν αποσύνδεση από το λογαριασμό τους.



ΕΙΚΟΝΑ 28 - Η ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΑΠΟ ΤΗΝ ΟΠΤΙΚΗ ΤΟΥ ΔΙΑΧΕΙΡΙΣΤΗ

3.2.2 Διαχείριση φορέων

Πηγαίνοντας στη σελίδα διαχείρισης φορέων, ο διαχειριστής μπορεί να δει σε μορφή πίνακα τους φορείς που συμμετέχουν στην εφαρμογή.



ΕΙΚΟΝΑ 29 - Η ΣΕΛΙΔΑ ΔΙΑΧΕΙΡΙΣΗΣ ΦΟΡΕΩΝ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Στον πίνακα γίνεται χρήση σελιδοποίησης, οπότε εμφανίζονται από προεπιλογή 10 αποτελέσματα ταξινομημένα βάση του ονόματος του φορέα. Στο κάτω μέρος της οθόνης ο διαχειριστής μπορεί εύκολα να περιηγηθεί στις σελίδες αποτελεσμάτων αλλά και να αλλάξει τον αριθμό των αποτελεσμάτων ανά σελίδα. Ακόμα του δίνεται η δυνατότητα να ταξινομήσει τα αποτελέσματα όπως αυτός επιθυμεί, πατώντας στο όνομα της στήλης βάση της οποίας θέλει να γίνει η ταξινόμηση. Δίπλα από το όνομα κάθε στήλης βρίσκεται ένα σύμβολο φίλτρου, το οποίο όταν πατηθεί εμφανίζει ένα μενού με επιλογές για φιλτράρισμα των αποτελεσμάτων, προσαρμοσμένο στον τύπο της στήλης. Για παράδειγμα, στην περίπτωση ενός πεδίου κειμένου, ο διαχειριστής μπορεί να εφαρμόσει φίλτρα όπως «ξεκινάει με», «τελειώνει με», «περιέχει», «ισούται» και «δεν ισούται». Αντίστοιχα αν το πεδίο είναι αριθμητικό, οι επιλογές που του δίνονται αφορούν τη σύγκριση μεταξύ αριθμών. Με αυτό τον τρόπο, ο διαχειριστής μπορεί να εμφανίσει μόνο τα αποτελέσματα που τον ενδιαφέρουν να δει. Μια ακόμα χρήσιμη δυνατότητα είναι αυτή της εξαγωγής των αποτελεσμάτων σε αρχείο Microsoft Excel, η οποία βρίσκεται στο πάνω δεξιά μέρος του πίνακα. Μέσω αυτής της δυνατότητας, ο διαχειριστής θα μπορέσει να κατεβάσει όλες τις εγγραφές του πίνακα ή όσες έχει επιλέξει μέσω των φίλτρων που έχει εφαρμόσει στη σελίδα.

Τέλος στο αριστερά μέρος κάθε εγγραφής του πίνακα, υπάρχουν δυο κουμπιά, τα οποία δίνουν τη δυνατότητα επεξεργασίας και διαγραφής της εγγραφής.

3.2.3 Προσθήκη/Επεξεργασία φορέα

Η προσθήκη και η επεξεργασία ενός φορέα από το διαχειριστή είναι αρκετά εύκολη. Και για τις δύο λειτουργίες χρησιμοποιείται μία κοινή φόρμα. Η φόρμα αυτή εμφανίζεται πατώντας το κουμπί «Προσθήκη» στο πάνω μέρος του πίνακα ή το εικονίδιο επεξεργασίας στα αριστερά κάθε εγγραφής.

Επεξεργασία του φορέα ΠΑΝΕΠΙΣΤΗΜΙΑΚΟ ΓΕΝΙΚΟ ΝΟΣΟΚΟΜΕΙΟ ΠΑΤΡΩΝ "ΠΑΝΑΓΙΑ Η ΒΟΗΘΕΙΑ"

Όνομα
ΠΑΝΕΠΙΣΤΗΜΙΑΚΟ ΓΕΝΙΚΟ ΝΟΣΟΚΟΜΕΙΟ ΠΑΤΡΩΝ "ΠΑΝΑΓΙΑ Η ΒΟΗΘΕΙΑ"

Τύπος
ΔΗΜΟΣΙΟ

Διεύθυνση
ΡΙΟ ΠΑΤΡΩΝ

Τηλέφωνο
2613603000

ΑΦΜ

Email
info@pgnp.gr

URL
https://www.pgnp.gr/

TK
26504

Γ. μήκος
38.2949895

Αποθήκευση Ακύρωση

ΕΙΚΟΝΑ 30 - ΦΟΡΜΑ ΕΠΕΞΕΡΓΑΣΙΑΣ ΦΟΡΕΑ

Στην εικόνα 30, μπορούμε να δούμε την προαναφερθείσα φόρμα τη στιγμή της επεξεργασίας ενός φορέα. Ο διαχειριστής καλείται να συμπληρώσει όσα περισσότερα πεδία μπορεί βάση των δεδομένων που έχει για τον φορέα. Για κάθε πεδίο της φόρμας υπάρχει επικύρωση στοιχείων, ώστε να αποφευχθεί η λάθος καταχώρηση στοιχείων στην εφαρμογή.

Ακόμα υπάρχουν κάποιες δυνατότητες που σκοπό έχουν να βοηθήσουν το διαχειριστή στην εύκολη και γρήγορη καταχώρηση νέων φορέων. Η πρώτη δυνατότητα αφορά το όνομα του φορέα. Κατά την πληκτρολόγηση, ο διαχειριστής μπορεί να ενημερωθεί σε πραγματικό χρόνο για το αν υπάρχει ήδη κάποια καταχώρηση για το φορέα που προσπαθεί να εισάγει (Εικόνα 31). Έτσι μπορούν να αποφευχθούν τυχόν διπλές καταχωρήσεις φορέων.

Όνομα
Νοσοκομείο Πατρών

Προσοχή! Υπάρχει ήδη ο φορέας "ΠΑΝΕΠΙΣΤΗΜΙΑΚΟ ΓΕΝΙΚΟ ΝΟΣΟΚΟΜΕΙΟ ΠΑΤΡΩΝ "ΠΑΝΑΓΙΑ Η ΒΟΗΘΕΙΑ"!'

Όνομα
Νοσοκομείο Πατρών Άγιος Ανδρέας

Δεν βρέθηκαν φορείς με το ίδιο όνομα, μπορείτε να προχωρήσετε σε καταχώρηση.

ΕΙΚΟΝΑ 31 - ΈΛΕΓΧΟΣ ΓΙΑ ΤΗΝ ΥΠΑΡΞΗ ΦΟΡΕΑ ΣΤΗΝ ΕΦΑΡΜΟΓΗ

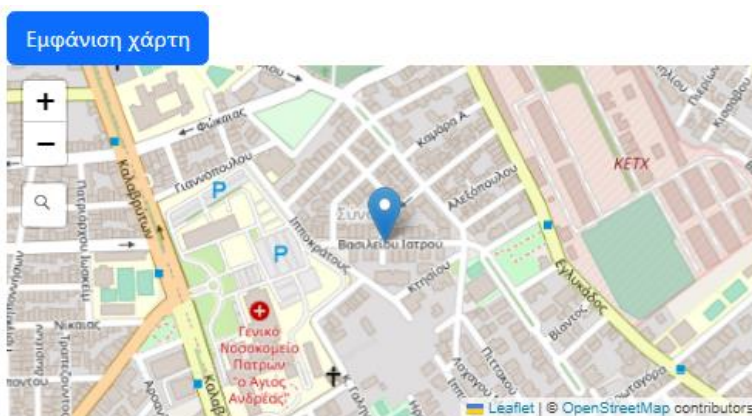
Η επόμενη ευκολία εντοπίζεται στο πεδίο της διεύθυνσης του φορέα. Όταν ο διαχειριστής πληκτρολογήσει μία διεύθυνση, η διεύθυνση αυτή αναζητείται σε ένα εξωτερικό σύστημα αναζήτησης διευθύνσεων, από το οποίο αντλούνται το γεωγραφικό μήκος και πλάτος της διεύθυνσης, τα οποία στη συνέχεια συμπληρώνονται αυτόματα στα αντίστοιχα πεδία της φόρμας με σκοπό την εμφάνιση του φορέα στον κεντρικό χάρτη της εφαρμογής. Στην περίπτωση που η

εξωτερική υπηρεσία δεν επιστρέφει μόνο ένα αποτέλεσμα, γίνεται επιλογή του αποτελέσματος που ταιριάζει περισσότερο στη διεύθυνση που έδωσε ο διαχειριστής και εμφανίζεται αντίστοιχο μήνυμα ώστε να ξέρει ο χρήστης ότι τα πεδία γεωγραφικού μήκους και πλάτους ενδέχεται να μην είναι σωστά (Εικόνα 32).

<p>Διεύθυνση</p> <p>Βασιλείου Ιατρού 17-19, Πάτρα</p> <p>Τα πεδία γεωγραφικού μήκους και πλάτους συμπληρώθηκαν αυτόματα βάση της διεύθυνσης.</p> <p>Τηλέφωνο</p> <p>ΑΦΜ</p> <p>Email</p> <p>URL</p> <p>TK</p> <p>Γ. μήκος</p> <p>38.2354377</p> <p>Γ. πλάτος</p> <p>21.7498793</p>	<p>Διεύθυνση</p> <p>Ακτή Δυμαίων 185, Πάτρα</p> <p>Εντοπίστηκαν 7 εγγραφές για τη συγκεκριμένη διεύθυνση. Τα πεδία γεωγραφικού μήκους και πλάτους συμπληρώθηκαν αυτόματα, ωστόσο ενδέχεται να μην είναι ακριβή.</p> <p>Τηλέφωνο</p> <p>ΑΦΜ</p> <p>Email</p> <p>URL</p> <p>TK</p> <p>Γ. μήκος</p> <p>38.2119708</p> <p>Γ. πλάτος</p> <p>21.7168198</p>
--	---

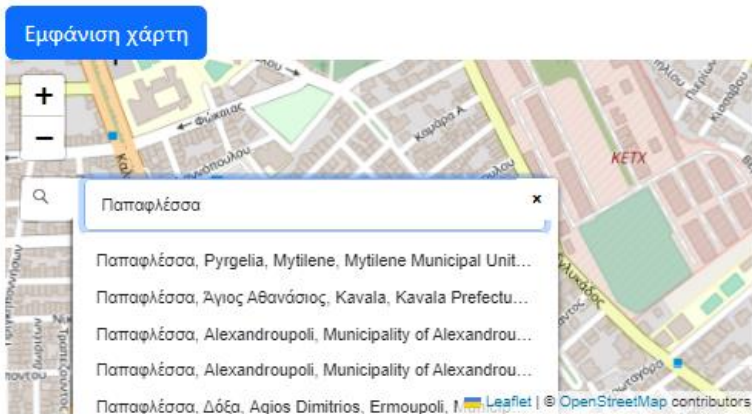
ΕΙΚΟΝΑ 32 - ΕΥΡΕΣΗ ΣΥΝΤΕΤΑΓΜΕΝΩΝ ΚΑΤΑ ΤΗΝ ΕΙΣΑΓΩΓΗ ΔΙΕΥΘΥΝΣΗΣ ΦΟΡΕΑ

Την περίπτωση της μη ορθής αυτόματης συμπλήρωσης των πεδίων γεωγραφικού μήκους και πλάτους έρχεται να ικανοποιήσει η επόμενη και τελευταία ευκολία όσον αφορά τη φόρμα προσθήκης/επεξεργασίας φορέων. Στο κάτω μέρος της φόρμας υπάρχει ένα κουμπί με την ονομασία «Εμφάνιση χάρτη» το οποίο εμφανίζει ένα διαδραστικό χάρτη με δείκτη στο σημείο που υποδεικνύουν τα πεδία γεωγραφικού μήκους και πλάτους (Εικόνα 33).



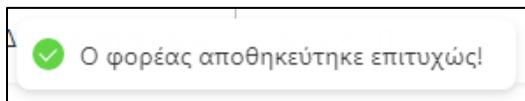
ΕΙΚΟΝΑ 33 - ΕΜΦΑΝΙΣΗ ΤΟΠΟΘΕΣΙΑΣ ΦΟΡΕΑ ΣΤΟ ΧΑΡΤΗ

Στο χάρτη αυτό ο διαχειριστής μπορεί να σύρει τον δείκτη με το ποντίκι του στο σημείο που γνωρίζει ότι βρίσκεται η τοποθεσία του φορέα και αυτόματα τα πεδία γεωγραφικού μήκους και πλάτους θα αλλάξουν για να αντιπροσωπεύουν τη νέα τοποθεσία. Επίσης ο διαχειριστής μπορεί να αναζητήσει για διευθύνσεις και τοποθεσίες απευθείας στο χάρτη πατώντας το μεγεθυντικό φακό στο αριστερά μέρος του χάρτη (Εικόνα 34). Κατά την επιλογή ενός εκ των αποτελεσμάτων αναζήτησης, τα πεδία γεωγραφικού μήκους και πλάτους θα συμπληρωθούν και πάλι.



ΕΙΚΟΝΑ 34 - ΑΝΑΖΗΤΗΣΗ ΕΝΑΛΛΑΚΤΙΚΗΣ ΤΟΠΟΘΕΣΙΑΣ ΦΟΡΕΑ ΣΤΟ ΧΑΡΤΗ

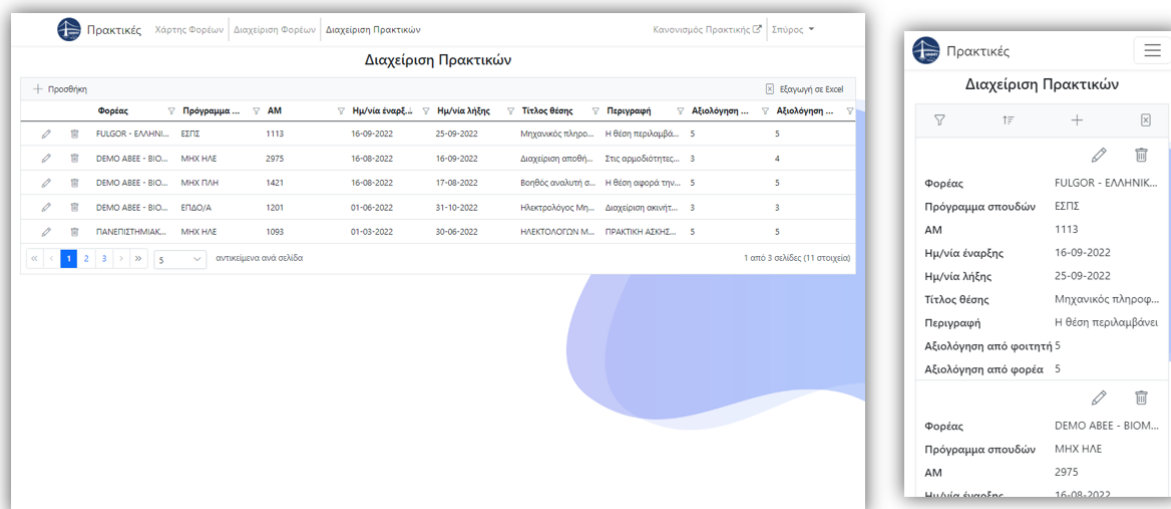
Τέλος, αφού συμπληρώσει τη φόρμα ο διαχειριστής, πατώντας το κουμπί «Αποθήκευση» στο κάτω μέρος της θα μπορέσει να αποθηκεύσει τις αλλαγές του. Μετά την ενέργεια αυτή, ένα μήνυμα επιτυχίας ή σφάλματος τύπου «toast» θα εμφανιστεί στην κορυφή της οθόνης του για να τον ενημερώσει αν η αποθήκευση ήταν επιτυχής (Εικόνα 35).



ΕΙΚΟΝΑ 35 - ΜΗΝΥΜΑ ΤΥΠΟΥ «TOAST» ΜΕΤΑ ΤΗΝ ΑΠΟΘΗΚΕΥΣΗ ΤΟΥ ΦΟΡΕΑ

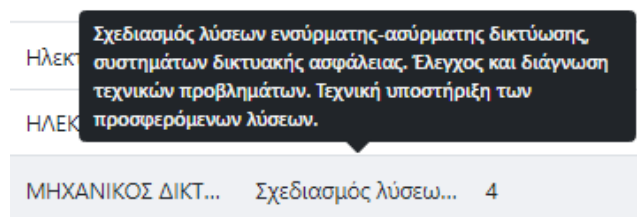
3.2.4 Διαχείριση θέσεων πρακτικής άσκησης

Όπως και με τη σελίδα διαχείρισης φορέων, πηγαίνοντας στη σελίδα διαχείρισης πρακτικών, ο διαχειριστής μπορεί να δει σε μορφή πίνακα όλες τις θέσεις πρακτικής άσκησης που έχουν καταχωρηθεί στην εφαρμογή.



ΕΙΚΟΝΑ 36 - Η ΣΕΛΙΔΑ ΔΙΑΧΕΙΡΙΣΗΣ ΘΕΣΕΩΝ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Ακριβώς όπως και στον πίνακα διαχείρισης φορέων, γίνεται χρήση σελιδοποίησης, ωστόσο μια διαφορά είναι ότι οι καταχωρήσεις στον πίνακα δεν εκτείνονται σε περισσότερες από μια γραμμές, λόγω του ότι οι περιγραφές των θέσεων πρακτικής άσκησης δύνανται να είναι αρκετά μεγάλες. Ο τρόπος με τον οποίο κάποιος μπορεί να διαβάσει ολόκληρα τα πεδία είναι περνώντας με το ποντίκι από πάνω τους (Εικόνα 37).



ΕΙΚΟΝΑ 37 - ΠΡΟΒΟΛΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΜΕΓΑΛΩΝ ΠΕΔΙΩΝ ΤΩΝ ΠΙΝΑΚΩΝ

Οι υπόλοιπες λειτουργίες της σελίδας παραμένουν ίδιες σε σχέση με τη σελίδα διαχείρισης φορέων, όπως η δυνατότητα εξαγωγής των εγγράφων σε αρχείο Microsoft Excel, η δυνατότητα ταξινόμησης και εφαρμογής φίλτρων στις εγγραφές του πίνακα, αλλά και η επεξεργασία και διαγραφή των εγγράφων.

3.2.5 Προσθήκη/Επεξεργασία θέσης πρακτικής άσκησης

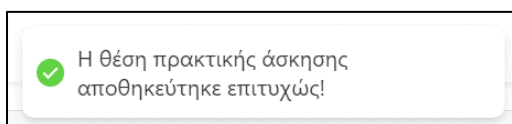
Αντίστοιχη με τη φόρμα προσθήκης/επεξεργασίας φορέα είναι και αυτή της προσθήκης/επεξεργασίας θέσης πρακτικής άσκησης. Μέσα από αυτήν ο διαχειριστής μπορεί να δηλώσει καινούριες θέσεις πρακτικής άσκησης ή να επεξεργαστεί υπάρχουσες.

The screenshot shows a web application interface for managing practical training positions. A modal window titled 'Επεξεργασία θέσης πρακτικής άσκησης (AM φοιτητή: 1093)' is open, displaying a form with the following fields: Φορέας (Institution), Πρόγραμμα σπουδών (Study Program), AM, Ημ/νια έναρξης (Start Date), Ημ/νια λήξης (End Date), Θέση (Position), and Περιγραφή (Description). The 'Save' button is labeled 'Αποθήκευση'. In the background, a table lists existing positions with columns for 'Αξιολόγηση...' and 'Αξιολόγηση...'. The table has 11 rows and 2 columns.

ΕΙΚΟΝΑ 38 - ΦΟΡΜΑ ΕΠΕΞΕΡΓΑΣΙΑΣ ΘΕΣΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ

Στο πεδίο του φορέα στον οποίο πραγματοποιήθηκε η πρακτική άσκηση, εμφανίζεται μία αναδυόμενη λίστα με όλους τους φορείς, ωστόσο δίνεται δυνατότητα στο διαχειριστή να βρει πιο εύκολα το φορέα που ψάχνει κάνοντας αναζήτηση πληκτρολογώντας το όνομα του. Στη συνέχεια καλείται να επιλέξει το πρόγραμμα σπουδών και να εισάγει τον αριθμό μητρώου του φοιτητή. Ακολούθως επιλέγει τις ημερομηνίες έναρξης και λήξης της πρακτικής μέσα από ένα, εύκολο στη χρήση, στοιχείο ημερολογίου. Τα επόμενα πεδία αφορούν τον τίτλο της θέσης πρακτικής άσκησης αλλά και την περιγραφή της και τέλος τη βαθμολογία αξιολόγησης του φοιτητή αλλά και του φορέα σε κλίμακα από το 1 έως το 5. Σε όλα τα παραπάνω πεδία υπάρχει επικύρωση στοιχείων, ώστε να αποφευχθούν λάθη κατά την καταχώρηση στοιχείων στην εφαρμογή.

Τέλος, αφού συμπληρώσει όλα τα απαραίτητα πεδία, ο διαχειριστής μπορεί να αποθηκεύσει την καταχώρηση πατώντας «Αποθήκευση». Ένα μήνυμα «toast» θα τον ενημερώσει για το αν η αποθήκευση της εγγραφής ήταν επιτυχής ή όχι (Εικόνα 39).



ΕΙΚΟΝΑ 39 - ΜΗΝΥΜΑ ΤΥΠΟΥ «TOAST» ΜΕΤΑ ΤΗΝ ΑΠΟΘΗΚΕΥΣΗ ΘΕΣΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ

ΕΠΙΛΟΓΟΣ

Η παρούσα εργασία στόχευε στην ανάλυση των εργαλείων κατασκευής αλλά και την παρουσίαση του συστήματος παρακολούθησης φορέων και θέσεων πρακτικής άσκησης, ένα σύστημα που πρόκειται να γίνει μέρος των παρεχόμενων ηλεκτρονικών υπηρεσιών του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου προς τους φοιτητές.

Με την ολοκλήρωση της εργασίας και αφού η εφαρμογή δοκιμάστηκε διεξοδικά από πειραματικούς χρήστες, καταλήξαμε στο συμπέρασμα ότι ανταποκρίνεται με επιτυχία στις ανάγκες ενός φοιτητή του τμήματος που προσπαθεί να βρει θέσεις πρακτικής άσκησης οι οποίες ταιριάζουν στα ενδιαφέροντά του, μέσα από ένα εύχρηστο και αποτελεσματικό περιβάλλον που είναι προσβάσιμο από οποιαδήποτε συσκευή μπορεί να έχει στην κατοχή του. Ιδιαίτερη μνεία δόθηκε και στο διαχειριστικό κομμάτι της εφαρμογής, με τη δημιουργία ενός εύχρηστου περιβάλλοντος διαχείρισης φορέων και θέσεων πρακτικής άσκησης για τη γραμματεία του τμήματος ή το γραφείο πρακτικής άσκησης του ακαδημαϊκού ιδρύματος, αλλά και την παροχή εργαλείων προς τους διαχειριστές ώστε να διευκολύνεται η εισαγωγή δεδομένων, αποφεύγοντας τυχόν λάθη.

Το συγκεκριμένο σύστημα παρακολούθησης φορέων και θέσεων πρακτικής άσκησης σχεδιάστηκε, κατασκευάστηκε και παρουσιάστηκε στα πλαίσια της εκπόνησης της ακαδημαϊκής πτυχιακής μου εργασίας και σε ορισμένο χρονικό διάστημα. Ως εκ τούτου, υπάρχουν αρκετές προοπτικές για επέκτασή του στο μέλλον, με σκοπό να ικανοποιήσει και άλλες ανάγκες του φοιτητή που αναζητά πληροφορίες για πρακτικές ασκήσεις, όπως η ανάλυση των αξιολογήσεων των φορέων από τους φοιτητές ή η ύπαρξη σχολίων/συμπερασμάτων του φοιτητή, τα οποία θα συνοψίζουν την εμπειρία του κατά τη διάρκεια της πρακτικής του άσκησης.

Τέλος μια ιδιαίτερα χρήσιμη επέκταση του συστήματος θα ήταν αυτή της διασύνδεσης με άλλες ηλεκτρονικές υπηρεσίες του τμήματος, όπως μιας εφαρμογής παρατηρητηρίου αποφοίτων, κάτι που θα διευκόλυνε σημαντικά το φόρτο εργασίας των διαχειριστών των δύο συστημάτων, καθώς θα μπορούσε να έχει μία κοινή βάση δεδομένων με φορείς. Ακόμα θα ήταν δυνατή η διασύνδεση των φοιτητών, ώστε κάποιος να μπορεί να βρει εύκολα αν ο φοιτητής που πραγματοποίησε κάποια πρακτική άσκηση προσλήφθηκε στη συνέχεια από τον ίδιο φορέα ή ποια ήταν η επαγγελματική του πορεία. Λαμβάνοντας υπόψιν τέτοιες πιθανές μελλοντικές επεκτάσεις, η εφαρμογή σχεδιάστηκε με τρόπο που καθιστά εύκολη την υλοποίησή τους, εφόσον αυτό είναι επιθυμητό.

ΒΙΒΛΙΟΓΡΑΦΙΑ / ΔΙΚΤΥΟΓΡΑΦΙΑ

CSS. Ανάκτηση από Wikipedia: <https://en.wikipedia.org/wiki/CSS>

CSS: *Cascading Style Sheets*. Ανάκτηση από MDN Web Docs: <https://developer.mozilla.org/en-US/docs/Web/CSS>

HTML5: *What's New in The Latest Version of HTML?* Ανάκτηση από <https://html.com/html5>

JavaScript. Ανάκτηση από Wikipedia: <https://en.wikipedia.org/wiki/JavaScript>

JavaScript. Ανάκτηση από MDN Web Docs: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

Laravel Documentation. Ανάκτηση από <https://laravel.com/docs/8.x>

Launch School. *Introduction to Programming with JavaScript*.

MariaDB. Ανάκτηση από Wikipedia: <https://en.wikipedia.org/wiki/MariaDB>

MySQL. Ανάκτηση από Wikipedia: <https://en.wikipedia.org/wiki/MySQL>

PHP Documentation. Ανάκτηση από <https://www.php.net/docs.php>

PHP Programming - *The State of Developer Ecosystem in 2021 Infographic*. (2021). Ανάκτηση από JetBrains: <https://www.jetbrains.com/lp/devecosystem-2021/php>

React (JavaScript library). Ανάκτηση από Wikipedia: [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library))

Sass (stylesheet language). Ανάκτηση από Wikipedia: [https://en.wikipedia.org/wiki/Sass_\(stylesheet_language\)](https://en.wikipedia.org/wiki/Sass_(stylesheet_language))

Saunier, R. (2014). *Getting Started with Laravel 4*. Packt Publishing.

Stauffer, M. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps - 2nd Edition*. O'Reilly Media.

Usage Statistics and Market Share of Server-side Programming Languages for Websites. (2022, October). Ανάκτηση από W3Techs: https://w3techs.com/technologies/overview/programming_language

Usage statistics of JavaScript as client-side programming language on websites. (2022, October). Ανάκτηση από W3Techs: <https://w3techs.com/technologies/details/cp-javascript>

What is a REST API? Ανάκτηση από RedHat: <https://www.redhat.com/en/topics/api/what-is-a-rest-api>

Wieruch, R. (2022). *The Road to React - 2022 Edition*.