

Πανεπιστήμιο Δυτικής Μακεδονίας
Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

Σχεδίαση και Υλοποίηση Πληροφοριακού
Συστήματος Ανοιχτού Κώδικα για Διαχείριση
Εκκίνησης Υπολογιστών μέσω Δικτύου

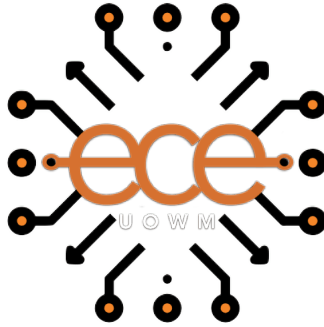
Χρήστος-Γεώργιος Καραμολέγκος

Επιβλέπων Καθηγητής: Δρ. Μηνάς Δασυγένης

<https://arch.ece.uowm.gr/mdasyg/>

Εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων

12 Σεπτεμβρίου 2023



University of Western Macedonia
Department of Electrical & Computer Engineering

Design and Implementation of an Open Source
Platform for Managing Computer Booting over a
Network

Christos-Georgios Karamolegkos

Supervisor: Dr. Minas Dasygenis

<https://arch.ece.uowm.gr/mdasyg/>

Laboratory of Digital Systems and Computer Architecture

September 12, 2023

Ευχαριστίες

Με την παρούσα διπλωματική εργασία ολοκληρώνονται οι σπουδές μου στο προπτυχιακό πρόγραμμα σπουδών του τμήματος Μηχανικών Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Δυτικής Μακεδονίας.

Στις σπουδές μου ήταν καθοριστική η συμβολή των καθηγητών μου, στους οποίους οφείλω να εκφράσω τις ειλικρινείς μου ευχαριστίες, για την πλούσια μεταβίβαση γνώσεων στα γνωστικά αντικείμενα που παρακολούθησα.

Ιδιαίτερα επιθυμώ να ευχαριστήσω τον καθηγητή μου και επιβλέποντα της παρούσας διπλωματικής εργασίας, κο Μηνά Δασυγένη, για την εμπιστοσύνη που μου έδειξε εξ' αρχής, αναθέτοντάς μου το συγκεκριμένο θέμα και για την επιστημονική και συμβουλευτική καθοδήγηση που μου προσέφερε σε όλα τα στάδια εκπόνησης της εργασίας με τις εύστοχες και πολύ εποικοδομητικές παρατηρήσεις του.

Τέλος, οφείλω να ευχαριστήσω τους συναδέλφους, τους φίλους και την οικογένειά μου, για την ανεκτίμητη στήριξη, τη συμπαράσταση και την κατανόησή τους, τόσο κατά την εκπόνηση της παρούσας διπλωματικής εργασίας, όσο και καθ' όλη τη διάρκεια των σπουδών μου.

Χρήστος Καραμολέγκος

Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο “Σχεδίαση και Υλοποίηση Πληροφοριακού Συστήματος Ανοιχτού Κώδικα για Διαχείριση Εκκίνησης Υπολογιστών μέσω Δικτύου” καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Μηνά Δασυγένη αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Περίληψη

Η δικτυακή εκκίνηση, εν συντομία netboot, είναι η διαδικασία εκκίνησης ενός υπολογιστή από ένα δίκτυο σε αντίθεση με μια τοπική συσκευή αποθήκευσης. Αυτή η τεχνική χρησιμοποιείται συχνά από δρομολογητές, σταθμούς εργασίας χωρίς δίσκο και υπολογιστές με κεντρική διαχείριση, όπως αυτοί που βρίσκονται σε επιχειρήσεις, δημόσιες βιβλιοθήκες και εκπαιδευτικά ιδρύματα. Η εκκίνηση μέσω δικτύου επιτρέπει την κεντρική διαχείριση της αποθήκευσης δίσκων, η οποία μπορεί δυνητικά να οδηγήσει σε μείωση του κόστους κεφαλαίου και συντήρησης. Μπορεί επίσης να είναι χρήσιμη σε υπολογιστικές συστάδες, όπου οι κόμβοι μπορεί να μην διαθέτουν τοπικούς δίσκους.

Σκοπός της παρούσας διπλωματικής εργασίας είναι να βοηθήσει τους διαχειριστές εργαστηρίου στο Εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας στη διαχείριση των διαθέσιμων μπλοκ iPXE και των μενού εκκίνησης, καθώς και του χρονοδιαγράμματος κατά το οποίο επιτρέπεται η δικτυακή εκκίνηση των υπολογιστών του εργαστηρίου.

Η διπλωματική εργασία περιγράφει τον σχεδιασμό και την ανάπτυξη μιας διαδικτυακής πλατφόρμας διαχείρισης για τη δικτυακή εκκίνηση υπολογιστών. Στόχος της πλατφόρμας είναι να βοηθήσει τους διαχειριστές στην ομαδοποίηση των υπολογιστών του εργαστηρίου και στην αποτελεσματική διαχείριση του χρονοδιαγράμματος εκκίνησης κάθε ομάδας. Οι διαχειριστές δημιουργούν καταχωρήσεις μενού εκκίνησης στην πλατφόρμα. Αυτές οι καταχωρήσεις αποτελούνται από επαναχρησιμοποιήσιμα μπλοκ iPXE τα οποία οι ομάδες υπολογιστών μπορούν να παρέχουν στα μέλη τους κατά την εκκίνησή τους. Τελικά, σε κάθε υπολογιστή που εκκινείται, παρέχεται ένα δυναμικά παραγόμενο μενού εκκίνησης που είναι προσαρμοσμένο στο ενεργό πρόγραμμα για την ομάδα της οποίας μέλος είναι ο υπολογιστής και της τρέχουσας ημερομηνίας και ώρας. Τέλος, η πλατφόρμα παρέχει αρχεία καταγραφής συμβάντων και διευκολύνει τους διαχειριστές στην παρακολούθηση των ενεργών συστημάτων υπολογιστών στο εργαστήριο.

Η πλατφόρμα δημιουργήθηκε χρησιμοποιώντας τεχνολογία λογισμικού αιχμής, καινοτόμες μεθόδους προγραμματισμού και γλώσσες προγραμματισμού ανοικτού κώδικα. Για τον προγραμματισμό του back-end της πλατφόρμας και του REST API χρησιμοποιήθηκε PHP με την υποστήριξη του CodeIgniter 4 Framework. Το front-end, από την άλλη πλευρά, βασίστηκε σε HTML, CSS, JavaScript και Bootstrap. Όσον αφορά τις τεχνικές προγραμματισμού που χρησιμοποιήθηκαν, το back-end αναπτύχθηκε ακολουθώντας τα πρότυπα αντικειμενοστραφούς προγραμματισμού του σχεδιασμού MVC. Το front-end, με τη σειρά του, ακολούθησε τις αρχές του Responsive design. Η πλατφόρμα αποθηκεύει τα δεδομένα της σε μια σχεσιακή βάση δεδομένων MySQL και οι διεπαφές του REST API τεκμηριώθηκαν και απεικονίστηκαν με τη χρήση του OpenAPI Swagger. Τα εργαλεία ανάπτυξης που χρησιμοποιήθηκαν ήταν το JetBrains PhpStorm και το Axosoft GitKraken.

Λέξεις κλειδιά: DHCP, BOOTP, TFTP, IPXE, NFS, HTTPFS2 / FTPS, ISCSI, PHP, CODEIGNITER, MYSQL, BOOTSTRAP

Abstract

Network booting, abbreviated as netboot, is the process of starting up a computer from a network rather than a local storage device. This technique is often used by routers, diskless workstations, and centrally managed computers, such as those found in businesses, public libraries, and educational institutions. Network booting enables centralized management of disk storage, which can potentially result in reduced capital and maintenance costs. It can also be beneficial in clustered computing environments, where local disks may be absent on nodes.

The purpose of this diploma thesis is to assist laboratory administrators at the University of Western Macedonia's Laboratory of Digital Systems and Computer Architecture in managing the available iPXE blocks and boot menus, as well as the time schedule allowed for lab computers' network booting.

The thesis describes the design and development of a web-based administration platform for computer network booting. The platform aims to assist administrators in grouping the laboratory's computers and effectively manage each group's booting schedule. Administrators create boot menu entries in the platform. These entries are comprised of reusable iPXE blocks that computer groups can provide to their members at boot time. Ultimately, each computer booted is provided with a dynamically generated boot menu that is tailored to the active schedule for the computer's group membership and the current date and time. Finally, the platform provides usage logs and facilitates administrators' monitoring of active computer systems in the lab.

The platform was created using cutting-edge software technology, innovative programming methods, and open-source programming languages. The programming for the platform's back-end and REST API was done in PHP, supported by the CodeIgniter 4 Framework. The front-end, on the other hand, was developed in HTML, CSS, JavaScript, and Bootstrap. Regarding the used programming techniques, the back-end was implemented following the Object-Oriented Programming standards of the MVC design. The front-end, in turn, adhered to Responsive design principles. The platform stores its data in a MySQL relational database and its REST API endpoints were documented and visualized using OpenAPI Swagger. JetBrains PhpStorm and Axosoft GitKraken were the utilized development tools.

Keywords: DHCP, BOOTP, TFTP, IPXE, NFS, HTTPFS2 / FTPS, ISCSI, PHP, CODEIGNITER, MYSQL, BOOTSTRAP

Περιεχόμενα

1	Εισαγωγή	1
1.1	Ορισμός του Προβλήματος	1
1.2	Κίνητρα και Στόχοι Υλοποίησης	1
1.3	Περπτώσεις Παρόμοιων Εργαλείων	2
1.3.1	FOG Project	2
1.3.2	CCboot	3
1.3.3	EigenBoot	3
1.4	Σύνοψη Διπλωματικής Εργασίας	3
1.5	Σύνοψη Κεφαλαίου 1	4
2	Θεωρητικό Υπόβαθρο	5
2.1	Διαδίκτυο	5
2.1.1	Παγκόσμιος Ιστός	5
2.1.2	Εφαρμογή Ιστού	6
2.2	Τεχνικές Προγραμματισμού	7
2.2.1	Αντικειμενοστρεφής Προγραμματισμός	7
2.2.2	Αρχιτεκτονική MVC	7
2.2.3	RESTful API	9
2.3	Γλώσσες Προγραμματισμού Ιστού	10
2.3.1	HTML	10
2.3.2	CSS	11
2.3.3	JavaScript & Ajax	12
2.3.4	PHP	12
2.3.5	SQL	13
2.4	Πλαίσια και Βιβλιοθήκες Προγραμματισμού Ιστού	14
2.4.1	Bootstrap	14
2.4.2	CodeIgniter	15
2.4.3	Swagger - OpenAPI	15
2.5	Τεχνολογίες Ανάπτυξης Λογισμικού	16
2.5.1	Jetbrains PhpStorm	16
2.5.2	Git - GitHub	16
2.6	Σύνοψη Κεφαλαίου 2	17

3	Δομή & Οργάνωση Συστήματος	19
3.1	Ανάλυση Απαιτήσεων	19
3.1.1	Λειτουργικές Απαιτήσεις	19
3.1.2	Μη-Λειτουργικές Απαιτήσεις	20
3.2	Περιπτώσεις Χρήσης	20
3.2.1	Βασικές Λειτουργίες Χρηστών	23
3.2.2	Διαχείριση Εργαστηρίων	27
3.2.3	Διαχείριση Ομάδων Υπολογιστών	29
3.2.4	Διαχείριση Υπολογιστών	31
3.2.5	Διαχείριση Ονοματισμένων Block Εντολών Τύπου iPXE	35
3.2.6	Διαχείριση Μενού Εκκίνησης	37
3.2.7	Διαχείριση Χρονοδιαγραμμάτων Μενού Εκκίνησης	39
3.2.8	Διαχείριση Αρχείων Καταγραφής Συμβάντων	41
3.3	Σχεδιασμός Βάσης Δεδομένων	42
3.4	Σύνοψη Κεφαλαίου 3	47
4	Ανάπτυξη Διαδικτυακής Εφαρμογής iBoot	49
4.1	Διεπαφή Χρήστη	49
4.1.1	Εγγραφή & Σύνδεση στο Σύστημα	49
4.1.2	Αρχική Σελίδα	52
4.1.3	Μενού Πλοήγησης	52
4.1.4	Υπολογιστές	53
4.1.5	Ομάδες Υπολογιστών	55
4.1.6	Εργαστήρια	56
4.1.7	Block εντολών τύπου iPXE	57
4.1.8	Μενού Εκκίνησης	58
4.1.9	Χρονοδιαγράμματα	60
4.1.10	Χρήστες	62
4.1.11	Συμβάντα	63
4.1.12	Ρύθμιση Εκκίνησης Υπολογιστών	64
4.1.13	Προδιαγραφή API	64
4.2	Επεξήγηση Αρχείων	65
4.2.1	Βασική Δομή CodeIgniter 4	65
4.2.2	Δομή Αρχείων iBoot	66
4.3	Ασφάλεια Συστήματος	71
4.3.1	Honeypots	71
4.3.2	CSRF	72
4.3.3	JWT	73
4.3.4	Αντιμετώπιση SQL Injections	75

4.4	Σύνοψη Κεφαλαίου 4	76
5	Αξιολόγηση Συστήματος	77
5.1	Δοκιμή Πλατφόρμας	77
5.1.1	Οι δοκιμές στην ανάπτυξη λογισμικού	77
5.1.2	Οφέλη των δοκιμών λογισμικού	77
5.1.3	Τύποι δοκιμών λογισμικού	78
5.2	Πλάνο Ελέγχου Ορθής Λειτουργίας	78
5.2.1	Βασικές Λειτουργίες Χρηστών	79
5.2.2	Διαχείριση Εργαστηρίων	80
5.2.3	Διαχείριση Ομάδων Υπολογιστών	80
5.2.4	Διαχείριση Υπολογιστών	80
5.2.5	Διαχείριση Ονοματισμένων Block Εντολών Τύπου iPXE	81
5.2.6	Διαχείριση Μενού Εκκίνησης	81
5.2.7	Διαχείριση Χρονοδιαγραμμάτων	82
5.2.8	Διαχείριση Αρχείων Καταγραφής Συμβάντων	82
5.3	Μελέτη Κλιμάκωσης	82
5.4	Σύνοψη Κεφαλαίου 5	83
6	Επίλογος	85
6.1	Ανακεφαλαίωση Διπλωματικής Εργασίας	85
6.2	Μοντέλο Ανάλυσης S.W.O.T.	86
6.2.1	Ορισμός Μοντέλου Ανάλυσης S.W.O.T.	86
6.2.2	Ανάλυση S.W.O.T. εφαρμογής iBoot	87
6.3	Μελλοντικές Επεκτάσεις	88
6.4	Συμπεράσματα	89
6.5	Σύνοψη Κεφαλαίου 6	90
	Παραρτήματα	91
	A' Απαιτήσεις, Εγκατάσταση & Ρύθμιση Πλατφόρμας	93
A'.1	Απαιτήσεις Συστήματος	93
A'.1.1	RHP	93
A'.1.2	Βάση Δεδομένων	93
A'.1.3	Διακομιστής Ιστού	93
A'.2	Οδηγίες Εγκατάστασης	95
A'.3	Αλλαγή Ρυθμίσεων Πλατφόρμας	95

Κατάλογος Σχημάτων

2.1	Διαδίκτυο	6
2.2	Διάγραμμα Αρχιτεκτονικής MVC	8
2.3	REST API	10
3.1	Διάγραμμα Περιπτώσεων Χρήσης iBoot	22
3.2	Διάγραμμα Βάσης Δεδομένων iBoot	42
4.1	iBoot - Εγγραφή	50
4.2	iBoot - Σύνδεση	51
4.3	iBoot - Υπενθύμιση Στοιχείων Πρόσβασης	51
4.4	iBoot - Αρχική Σελίδα	52
4.5	iBoot - Μενού Πλοήγησης Διαχειριστή	53
4.6	iBoot - Μενού Πλοήγησης Διαχειριστή Εργαστηρίου	53
4.7	iBoot - Μενού Πλοήγησης Επισκέπτη	53
4.8	iBoot - Διαχειριζόμενοι Υπολογιστές	54
4.9	iBoot - Μη Εκχωρημένοι Υπολογιστές	54
4.10	iBoot - Διαχείριση Ομάδων Υπολογιστών	56
4.11	iBoot - Διαχείριση Εργαστηρίων	57
4.12	iBoot - Διαχείριση Μπλοκ Εντολών Τύπου iPXE	58
4.13	iBoot - Διαχείριση Μενού Εκκίνησης	59
4.14	iBoot - Επεξεργασία Μενού Εκκίνησης	59
4.15	iBoot - Διαχείριση Χρονοδιαγραμμάτων	61
4.16	iBoot - Διαχείριση Χρηστών	62
4.17	iBoot - Συμβάντα	63
4.18	iBoot - Ρύθμιση Εκκίνησης Υπολογιστών	64
4.19	iBoot - Προδιαγραφή API	65
4.20	Η βασική δομή του CodeIgniter 4 framework	66
4.21	Τα αρχεία του καταλόγου Controllers του iBoot	67
4.22	Τα αρχεία του καταλόγου Database του iBoot	68
4.23	Τα αρχεία του καταλόγου Entities του iBoot	68
4.24	Τα αρχεία του καταλόγου Filters του iBoot	69
4.25	Τα αρχεία του καταλόγου Language του iBoot	69

4.26	Τα αρχεία του καταλόγου Models του iBoot	70
4.27	Τα αρχεία του καταλόγου Validation του iBoot	70
4.28	Τα αρχεία του καταλόγου Views του iBoot	71
4.29	Λειτουργία JWT	73
4.30	Δομή JWT	74
6.1	Ανάλυση SWOT εφαρμογής iBoot	87

Κατάλογος Πινάκων

2.1	Βασικά στοιχεία HTML	11
2.2	Στοιχεία εντολής CSS	12
2.3	Βασικές συναρτήσεις της PHP	13
3.1	Περίπτωση Χρήσης: Εγγραφή χρήστη στο σύστημα	23
3.2	Περίπτωση Χρήσης: Σύνδεση χρήστη στο σύστημα	23
3.3	Περίπτωση Χρήσης: Επανέκδοση κωδικού χρήστη	24
3.4	Περίπτωση Χρήσης: Υπενθύμιση Ονόματος Χρήστη	25
3.5	Περίπτωση Χρήσης: Αποσύνδεση από το σύστημα	25
3.6	Περίπτωση Χρήσης: Αλλαγή στοιχείων χρήστη	26
3.7	Περίπτωση Χρήσης: Διαγραφή χρήστη από το σύστημα	26
3.8	Περίπτωση Χρήσης: Εισαγωγή εργαστηρίου στο σύστημα	27
3.9	Περίπτωση Χρήσης: Διαγραφή εργαστηρίου από το σύστημα	28
3.10	Περίπτωση Χρήσης: Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων	28
3.11	Περίπτωση Χρήσης: Δημιουργία ομάδας υπολογιστών	29
3.12	Περίπτωση Χρήσης: Διαγραφή ομάδας υπολογιστών	30
3.13	Περίπτωση Χρήσης: Εισαγωγή υπολογιστή στο σύστημα	31
3.14	Περίπτωση Χρήσης: Ανάθεση υπολογιστή σε εργαστήριο	31
3.15	Περίπτωση Χρήσης: Αφαίρεση υπολογιστή από εργαστήριο	32
3.16	Περίπτωση Χρήσης: Επεξεργασία στοιχείων υπολογιστή	33
3.17	Περίπτωση Χρήσης: Διαγραφή υπολογιστή από το σύστημα	34
3.18	Περίπτωση Χρήσης: Εισαγωγή ονοματισμένου block εντολών τύπου iPXΕ στο σύστημα	35
3.19	Περίπτωση Χρήσης: Επεξεργασία στοιχείων ονοματισμένου block εντολών τύπου iPXΕ	35
3.20	Περίπτωση Χρήσης: Διαγραφή ονοματισμένου block εντολών τύπου iPXΕ από το σύστημα	36
3.21	Περίπτωση Χρήσης: Δημιουργία μενού εκκίνησης	37
3.22	Περίπτωση Χρήσης: Επεξεργασία στοιχείων μενού εκκίνησης	37
3.23	Περίπτωση Χρήσης: Διαγραφή μενού εκκίνησης	38
3.24	Περίπτωση Χρήσης: Δημιουργία χρονοδιαγράμματος μενού εκκίνησης	39
3.25	Περίπτωση Χρήσης: Επεξεργασία στοιχείων χρονοδιαγράμματος μενού εκκίνησης	39
3.26	Περίπτωση Χρήσης: Διαγραφή χρονοδιαγράμματος μενού εκκίνησης	40
3.27	Περίπτωση Χρήσης: Προβολή αρχείων συμβάντων	41
3.28	Περίπτωση Χρήσης: Διαγραφή αρχείων συμβάντων	41

3.29	Δομή πίνακα boot_menu	43
3.30	Δομή πίνακα boot_menu_blocks	43
3.31	Δομή πίνακα computers	43
3.32	Δομή πίνακα computer_groups	44
3.33	Δομή πίνακα forgot_password_tokens	44
3.34	Δομή πίνακα groups	45
3.35	Δομή πίνακα ipxe_blocks	45
3.36	Δομή πίνακα labs	45
3.37	Δομή πίνακα schedules	46
3.38	Δομή πίνακα users	46
3.39	Δομή πίνακα user_labs	47
5.1	Έλεγχοι Βασικών Λειτουργιών Χρηστών	79
5.2	Έλεγχοι Διαχείρισης Εργαστηρίων	80
5.3	Έλεγχοι Διαχείρισης Ομάδων Υπολογιστών	80
5.4	Έλεγχοι Διαχείρισης Υπολογιστών	80
5.5	Έλεγχοι Διαχείρισης Ονοματισμένων Block Εντολών Τύπου iPXE	81
5.6	Έλεγχοι Διαχείρισης Μενού Εκκίνησης	81
5.7	Έλεγχοι Χρονοδιαγραμμάτων	82
5.8	Έλεγχοι Αρχείων Καταγραφής Συμβάντων	82

Κατάλογος Απεικονίσεων

A'.1 Παράδειγμα ρύθμισης Apache2 webserver	94
A'.2 Παράδειγμα ρύθμισης Nginx webserver	94
A'.3 Παράδειγμα αρχείου .env	96

Κεφάλαιο 1

Εισαγωγή

1.1 Ορισμός του Προβλήματος

Το πρόβλημα που προσπαθεί να επιλύσει η παρούσα διπλωματική εργασία, με τον σχεδιασμό και την υλοποίηση της διαδικτυακής εφαρμογής iBoot, είναι η δημιουργία παραμετροποιημένων μενού εκκίνησης υπολογιστών, βάση χρονοδιαγράμματος, για το εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας.

Στην υφιστάμενη κατάσταση, η δημιουργία των μενού εκκίνησης γίνεται με χρήση κάποιου php script, στο οποίο οι αλλαγές γίνονται χειροκίνητα. Μετά από χρόνια προσθήκης εγγραφών και παραμετροποιήσεων, το εν λόγω script έχει γίνει δυσνόητο και δύσχρηστο, κάνοντας έτσι χρονοβόρα τη διαδικασία επεξεργασίας του κάθε φορά που πρέπει να γίνουν αλλαγές.

Μετά από εξέταση διάφορων εμπορικών λύσεων, έγινε αντιληπτό πως δεν υπάρχει κάποια λύση που να καλύπτει πλήρως τις ανάγκες του εργαστηρίου. Οι σχετικές εφαρμογές στοχεύουν στην κάλυψη αναγκών ίντερνετ καφέ και περιβαλλόντων εργασίας, στα οποία δεν υπάρχει τόσο η ανάγκη για δυνατότητες όπως χρονοπρογραμματισμό εκκίνησης ή δυναμική δημιουργία παραμετροποιημένων μενού εκκίνησης.

1.2 Κίνητρα και Στόχοι Υλοποίησης

Ενώ υπάρχουν αρκετές διαδικτυακές εφαρμογές διαχείρισης δικτυακής εκκίνησης υπολογιστών, καμιά δε φαίνεται να μπορεί να ικανοποιήσει πλήρως τις ανάγκες του εργαστηρίου. Κάποιες εφαρμογές υποστηρίζουν εκκίνηση με ένα μόνο μενού, ενώ άλλες έχουν δυνατότητα διαχωρισμού των συσκευών σε ομάδες και εκκίνηση της κάθε συσκευής με διαφορετικό μενού. Ορισμένες έχουν αρκετές περισσότερες δυνατότητες και εργαλεία, τα οποία όμως δε χρειάζεται το εργαστήριο, ενώ καμιά δεν προσφέρει ένα βασικό χαρακτηριστικό για το οποίο υπάρχει ανάγκη: τη δυνατότητα καθορισμού διαφορετικού μενού εκκίνησης, βάση ομάδας συσκευών αλλά και χρονοδιαγράμματος.

Έτσι γεννήθηκε η ιδέα δημιουργίας της διαδικτυακής εφαρμογής ανοικτού κώδικα 'iBoot', η οποία θα καλύψει τα κενά των διαθέσιμων εμπορικών λύσεων και θα δώσει λύση στο πρόβλημα της διαχείρισης της δικτυακής εκκίνησης των υπολογιστών του εργαστηρίου. Για να πετύχει τους στόχους της, η εφαρμογή iBoot θα πρέπει να έχει τα παρακάτω χαρακτηριστικά:

- **Ανοικτού Κώδικα:** Μια εφαρμογή ανοικτού κώδικα είναι πιο εύκολο να επεκταθεί, να ελεγχθεί για κενά ασφαλείας και η λειτουργία της να γίνει κατανοητή από τους χρήστες, τους διαχειριστές και όσους τη συντηρούν.

-
- **Ανάπτυξη με PHP, MySQL & Bootstrap:** Η ανάπτυξη θα πρέπει να γίνει με σύγχρονα και ασφαλή εργαλεία και τεχνολογίες, τα οποία να είναι σε θέση να συντηρήσουν και να επεκτείνουν οι φοιτητές του τμήματος.
 - **Απλή διεπαφή χρήστη:** Η διεπαφή χρήστη θα πρέπει να είναι απλή και η εφαρμογή θα πρέπει να μην έχει περισσότερα χαρακτηριστικά από όσα χρειάζονται στο εργαστήριο. Όσο περισσότερα γίνονται τα χαρακτηριστικά και οι δυνατότητες μιας εφαρμογής, τόσο πιο δύσκολη γίνεται η χρήση της και η περιήγηση σε αυτή, ενώ ταυτόχρονα μεγαλώνει η πιθανότητα ύπαρξης ευπαθειών προς εκμετάλλευση από κακόβουλους χρήστες.
 - **Προσαρμοστική Εμφάνιση:** Όπως συμβαίνει σε όλες τις σύγχρονες σελίδες και εφαρμογές, θα πρέπει η διεπαφή χρήστη να προσαρμόζεται στο μέγεθος της οθόνης του χρήστη, ώστε να προσφέρει την καλύτερη δυνατή εμπειρία χρήσης, τόσο σε υπολογιστές, όσο και κινητές συσκευές όπως τηλέφωνα και τάμπλετ.
 - **Ομαδοποίηση Υπολογιστών:** Η εφαρμογή θα πρέπει να υποστηρίζει την ομαδοποίηση υπολογιστών, για πιο ευέλικτη διαχείριση της εκκίνησής τους.
 - **Χρονοπρογραμματισμός Εκκίνησης:** Θα πρέπει να υποστηρίζεται η εκκίνηση με διαφορετικό μενού εκκίνησης τις ώρες των εργαστηρίων μαθημάτων του τμήματος.
 - **Παρακολούθηση Εκκίνησης:** Ο διαχειριστής θα πρέπει να μπορεί να παρακολουθεί τους υπολογιστές σχετικά με το ποιοι και πότε εκκινήθηκαν.
 - **Δυναμικά Μενού Εκκίνησης:** Τα μενού εκκίνησης πρέπει να είναι δυναμικά, δηλαδή να παράγονται τη στιγμή που ζητούνται από κάποιον υπολογιστή, με γνώμονα την τρέχουσα ημερομηνία/ώρα και την ομάδα στην οποία βρίσκεται ο υπολογιστής. Επίσης, θα πρέπει να μπορούν να υποστούν επεξεργασία ανά πάσα στιγμή από τους διαχειριστές της εφαρμογής, πραγματοποιώντας αλλαγές στο ίδιο το μενού ή στα μπλοκ iPXE που το αποτελούν.
 - **Ασφάλεια:** Η εφαρμογή θα πρέπει να αναπτυχθεί με βάση σύγχρονα πρότυπα και πρακτικές ασφαλείας, ώστε να διασφαλίζει τόσο την ακεραιότητα των περιεχόμενων δεδομένων της, όσο και την ιδιωτικότητα των χρηστών της.

1.3 Περιπτώσεις Παρόμοιων Εργαλείων

1.3.1 FOG Project

Το FOG (Free and Open-source Ghost) είναι ένα δωρεάν και ανοικτού κώδικα πρόγραμμα απεικόνισης δίσκων υπολογιστών. Υποστηρίζει Linux, Mac OS και διάφορες εκδόσεις των Windows (XP, Vista, 7, 8/8.1 και 10). Ενσωματώνει τεχνολογίες ανοικτού κώδικα με μια διεπαφή ιστού βασισμένη στην PHP. Το FOG χρησιμοποιεί μόνο TFTP και PXE για την εκκίνηση των υπολογιστών, οπότε εκείνοι δε χρειάζονται καθόλου CD ή δίσκους εκκίνησης [1].

Επιπλέον, με το FOG, πολλοί οδηγοί δικτύου περιλαμβάνονται ήδη στον πυρήνα του πελάτη (vanilla linux). Το FOG επιτρέπει επίσης την αντιγραφή μιας εικόνας από ένα μηχάνημα με μεγαλύτερο διαμέρισμα δίσκου σε ένα μηχάνημα με μικρότερο σκληρό δίσκο, εφόσον ο χώρος του τελικού δίσκου επαρκεί για την αποθήκευση των

δεδομένων του πηγαιού διαμερίσματος. Τέλος, επειδή το FOG επιτρέπει την πολλαπλή διανομή, γίνεται δυνατή η μετάδοση εικόνας σε πολλούς υπολογιστές ταυτόχρονα, χρησιμοποιώντας μία μόνο ροή.

Το FOG διαθέτει δυνατότητες όμοιες με ένα περιβάλλον εκκίνησης PXE (DHCP, iPXE, TFTP, γρήγορη λήψη HTTP μεγάλων αρχείων εκκίνησης όπως ο πυρήνας και το initrd). Χρησιμοποιεί snarins για τη διεξαγωγή εργασιών και σεναρίων στους πελάτες, την εγκατάσταση λογισμικού και τη διαχείριση εκτυπωτών. Μερικές ακόμη από τις λειτουργίες του είναι η παρακολούθηση της δραστηριότητας του χρήστη του υπολογιστή, η αυτόματη απενεργοποίηση του μηχανήματος έπειτα από προκαθορισμένα χρονικά όρια αδράνειας και η μετονομασία του κεντρικού υπολογιστή και ένταξή του σε τομέα. Επίσης, έχει εξελιχθεί σε λύση διαχείρισης δικτύου και απεικόνισης/κλωνοποίησης, ξεπερνώντας την απλή λύση απεικόνισης, διαθέτει Anti-Virus και έχει την δυνατότητα διαγραφής δίσκων, επαναφοράς διαγραμμένων αρχείων και σάρωσης δίσκων για bad blocks [2].

1.3.2 CCboot

Το CCBoot είναι μια κλειστού κώδικα, εμπορική λύση εκκίνησης υπολογιστών χωρίς δίσκο. Απευθύνεται σε internet cafe, σχολεία και εταιρείες που θέλουν να εκμεταλλευτούν τα πλεονεκτήματα της εικονοποίησης δίσκων και της δικτυακής εκκίνησης υπολογιστών. Με τη βοήθεια της τεχνολογίας εκκίνησης PXE, το Thin Client Software Solution του CCboot επιτρέπει την απομακρυσμένη εκκίνηση iSCSI και την εκκίνηση χωρίς δίσκο των Windows 7 και XP από ένα διακομιστή thin client στο τοπικό δίκτυο LAN.

Το CCboot έχει ορισμένα ιδιαίτερα χρήσιμα χαρακτηριστικά, όπως η σύνδεση τομέων, η γρήγορη ανάκτηση και η ενοποιημένη διαχείριση συσκευών. Εφόσον δε χρειάζεται σκληρούς δίσκους, διευκολύνει τις αναβαθμίσεις και την ομοιογένεια περιβάλλοντος σε όλες τις συσκευές του δικτύου. Τέλος, διαλειτουργεί τέλεια με τον τομέα των Windows, ελαχιστοποιεί την κατανάλωση ενέργειας και είναι εξοπλισμένο με καλές επιδόσεις κρυφής μνήμης, φυσική μνήμη και υποστήριξη κρυφής μνήμης SSD [3].

1.3.3 EigenBoot

Το Eigenboot είναι ένα σύστημα διαχείρισης εικόνων εκκίνησης που συγκεντρώνει και βελτιώνει τη διαχείριση των υπολογιστών γραφείου. Επιτρέπει στους υπολογιστές να εκκινούνται από ένα δίκτυο, αντί για έναν τοπικό δίσκο ή κάποια αφαιρούμενη συσκευή.

Το Eigenboot παρέχει επεκτασιμότητα και απόδοση, αξιοποιώντας τις υπολογιστικές δυνατότητες των τελικών συσκευών. Υποστηρίζει τη δυναμική κατανομή πόρων επιφάνειας εργασίας και την εξισορρόπηση φορτίου, η οποία αποτελεί ένα από τα πιο βασικά χαρακτηριστικά του. Επιτρέπει την ομαδοποίηση επιφάνειας εργασίας και μπορεί να προσφέρει διαφορετικά λειτουργικά συστήματα εκκίνησης ανά ομάδα. Μπορεί να σχεδιαστεί έτσι ώστε να επιτρέπει στον χρήστη να επιλέγει ένα λειτουργικό σύστημα κατά την εκκίνηση. Είναι συμβατό τόσο με λειτουργικά συστήματα Windows όσο και με Linux [4].

1.4 Σύνοψη Διπλωματικής Εργασίας

Η διαδικτυακή εφαρμογή διαχείρισης μενού δικτυακής εκκίνησης υπολογιστών 'iBoot', δημιουργήθηκε για να καλύψει τις ανάγκες του εργαστηρίου Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας, εφόσον δεν υπήρχαν εμπορικές λύσεις με τα αναζητούμενα χαρακτηριστικά

και τις απαραίτητες δυνατότητες.

Η εφαρμογή δημιουργήθηκε με PHP στο back-end, HTML, CSS, Javascript και Bootstrap στο front-end και αποθηκεύει τα δεδομένα της σε MySQL σχεσιακή βάση δεδομένων. Η χρήση της εφαρμογής γίνεται μέσω της διεπαφής χρήστη ή του REST API που έχουν αναπτυχθεί. Ο κώδικας της εφαρμογής είναι ανοικτός και δημοσιευμένος, και διαμοιράζεται με άδεια χρήσης MIT.

Η εφαρμογή υποστηρίζει τρία (3) είδη χρηστών.

- **Ανώνυμος Χρήστης:** Ο Ανώνυμος Χρήστης έχει πρόσβαση μόνο στις σελίδες σύνδεσης, εγγραφής και υπενθύμισης στοιχείων πρόσβασης.
- **Διαχειριστής Εργαστηρίου:** Ο Διαχειριστής Εργαστηρίου μπορεί να διαχειρίζεται υπολογιστές που βρίσκονται σε εργαστήρια που του έχουν ανατεθεί. Μπορεί επίσης να τοποθετεί σε εργαστήρια τα οποία διαχειρίζεται, υπολογιστές που δεν έχουν ήδη τοποθετηθεί σε άλλα εργαστήρια.
- **Διαχειριστής:** Ο χρήστης Διαχειριστής μπορεί να χρησιμοποιήσει όλες τις δυνατότητες της πλατφόρμας χωρίς περιορισμούς. Μπορεί να προσθέτει, να αφαιρεί και να επεξεργάζεται υπολογιστές, ομάδες, εργαστήρια, μπλοκ iPXE, μενού εκκίνησης, χρονοδιαγράμματα και χρήστες. Μπορεί επίσης να προβάλλει τα αρχεία καταγραφής συμβάντων της εφαρμογής.

Επίσης, οι υπολογιστές αλληλεπιδρούν με συγκεκριμένες σελίδες της εφαρμογής, για να εγγραφούν σε αυτή και για να λάβουν μενού εκκίνησης.

Έχουν ακολουθηθεί οι βέλτιστες πρακτικές και τα πιο σύγχρονα πρότυπα ασφαλείας στον σχεδιασμό και την υλοποίηση της εφαρμογής, με σκοπό την διασφάλιση της περιεχόμενης πληροφορίας. Οι κωδικοί πρόσβασης των χρηστών αποθηκεύονται στη βάση δεδομένων της εφαρμογής μόνο αφού περαστούν από κρυπτογραφικά ασφαλείς συναρτήσεις κατακερματισμού, και δεν εμφανίζονται ποτέ σε κανένα σημείο της εφαρμογής. Πριν τη φόρτωση κάθε σελίδας γίνεται έλεγχος αυθεντικοποίησης και δικαιωμάτων του χρήστη που θέλει να την προβάλλει, ενώ και για το API ισχύει ακριβώς το ίδιο, με την επισύναψη των JWT tokens σε κάθε αίτημα. Τέλος, η εφαρμογή δύναται να λειτουργήσει τόσο με πρωτόκολλο HTTP, όσο και με το ασφαλέστερο HTTPS, δίνοντας μάλιστα δυνατότητα κρυπτογράφησης και των cookies της συνεδρίας.

Η εγκατάσταση της διαδικτυακής εφαρμογής 'iBoot' στο εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας, θα εκσυγχρονίσει τη διαδικασία διαχείρισης των μενού δικτυακής εκκίνησης των υπολογιστών του, και θα διευκολύνει σημαντικά την καταγραφή και την παρακολούθησή τους.

1.5 Σύνοψη Κεφαλαίου 1

Στο ανωτέρω κεφάλαιο αναλύεται ο σχεδιασμός, η ανάπτυξη και η χρήση μιας πλατφόρμας ανοικτού κώδικα για τον έλεγχο της εκκίνησης υπολογιστών μέσω δικτύου, καθώς και οι λόγοι κατασκευής και οι λειτουργίες της. Επίσης, γίνεται αναφορά στα κίνητρα και τους στόχους υλοποίησης της εφαρμογής καθώς και σε περιπτώσεις παρόμοιων εργαλείων.

Στα επόμενα κεφάλαια, γίνεται αναφορά στη συνολική δομή της διαδικτυακής εφαρμογής, καθώς και στα εργαλεία και τις μεθόδους που χρησιμοποιήθηκαν καθ' όλη τη διάρκεια της ανάπτυξης.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Στο κεφάλαιο αυτό, αναλύονται λεπτομερώς όλα τα είδη τεχνολογιών που χρησιμοποιήθηκαν για την υλοποίηση της διαδικτυακής εφαρμογής που κατασκευάστηκε για την εκπόνηση της διπλωματικής εργασίας.

2.1 Διαδίκτυο

Το Διαδίκτυο αποτελείται από μια σειρά δικτύων που συνδέουν συσκευές σε όλο τον κόσμο. Αυτά τα δίκτυα χρησιμοποιούν τηλεφωνικές γραμμές για τη σύνδεση των διαφόρων συσκευών. Οι χρήστες χρησιμοποιούν το Διαδίκτυο από παρόχους υπηρεσιών Διαδικτύου. Τα ευρυζωνικά δίκτυα κινητής τηλεφωνίας και το Wi-Fi είναι ευρέως διαδεδομένα στον 21ο αιώνα, επιτρέποντας την ασύρματη σύνδεση. Η σουίτα πρωτοκόλλων TCP/IP είναι το κυρίως χρησιμοποιούμενο σύνολο πρωτοκόλλων δικτύωσης, που επιτρέπουν την επικοινωνία μεταξύ διαφορετικών δικτύων σε όλο τον κόσμο. Το Διαδίκτυο έχει φέρει επανάσταση σε όλους τους τομείς της σύγχρονης κοινωνίας, όπως η επικοινωνία, η εργασία, η εκπαίδευση και το εμπόριο, αφού επιτρέπει σε ανθρώπους από διαφορετικά μέρη του κόσμου να συνδεθούν μεταξύ τους [5].

Το Διαδίκτυο μπορεί να χρησιμοποιηθεί για σχεδόν κάθε σκοπό που εξαρτάται από την πληροφορία, και όποιος έχει την δυνατότητα να συνδεθεί σε ένα από τα δίκτυα που το συνθέτουν, έχει πρόσβαση σε αυτό. Υποστηρίζει την ανθρώπινη επικοινωνία μέσω των κοινωνικών μέσων, του ηλεκτρονικού ταχυδρομείου (e-mail), των “chat rooms”, και επιτρέπει στους ανθρώπους να συνεργάζονται είτε σύγχρονα, είτε ασύγχρονα σε πολλά διαφορετικά μέρη του κόσμου. Υποστηρίζει την πρόσβαση σε ψηφιακές πληροφορίες μέσω πολλών εφαρμογών, συμπεριλαμβανομένου του Παγκόσμιου Ιστού. Το Διαδίκτυο έχει διαδραματίσει σημαντικό ρόλο στην ανάπτυξη ενός μεγάλου και αυξανόμενου αριθμού “ηλεκτρονικών επιχειρήσεων” (συμπεριλαμβανομένων και των θυγατρικών των παραδοσιακών εταιρειών) που χρησιμοποιούν το Διαδίκτυο για το μεγαλύτερο μέρος των πωλήσεων και των υπηρεσιών τους.

2.1.1 Παγκόσμιος Ιστός

Αν και μερικές φορές χρησιμοποιούνται ως όμοιοι, οι όροι Διαδίκτυο και Παγκόσμιος Ιστός δεν έχουν την ίδια σημασία. Ο Παγκόσμιος Ιστός είναι μία από τις κύριες υπηρεσίες που παρέχονται μέσω του Διαδικτύου, ενώ ο όρος “Διαδίκτυο” αναφέρεται σε ολόκληρο το παγκόσμιο σύστημα επικοινωνίας, που περιλαμβάνει το υλικό και την υποδομή. Πιο επίσημα, ο Παγκόσμιος Ιστός μπορεί να οριστεί ως ένα σύστημα τεχνικοκοινωνικής αλληλεπίδρασης. Ένα σύστημα που βελτιώνει την ανθρώπινη διάνοια, την επικοινωνία και τη συνεργασία αναφέρεται ως τεχνοκοινωνικό σύστημα. Οι πληροφορίες που δημιουργούνται στον Παγκόσμιο Ιστό από τους χρήστες



Σχήμα 2.1: Διαδίκτυο Πηγή: [Eli18]

του μπορούν να αποθηκεύονται στο Διαδίκτυο. Υπάρχουν διάφοροι ιστότοποι στον Παγκόσμιο Ιστό, καθένας από τους οποίους έχει τη δική του διεύθυνση URL (Uniform Resource Locator). Η διεύθυνση URL έχει τη μορφή `http://www.example.com`. Το `http` συνιστά τη βάση δεδομένων επικοινωνίας για τον Παγκόσμιο Ιστό, το `www` πρόκειται για το World Wide Web, το οποίο είναι ένα πληροφοριακό σύστημα όπου τα έγγραφα και άλλοι δικτυακοί τόποι αναγνωρίζονται από τις διευθύνσεις URL τους, το `example` δηλώνει το όνομα του δικτυακού τύπου (domain name) και το `com` υποδεικνύει την περιοχή στην οποία ανήκει ο δικτυακός τύπος ή τον τύπο του δικτυακού τύπου [6].

2.1.2 Εφαρμογή Ιστού

Μια Εφαρμογή Web (Ιστού) συνιστά μια εφαρμογή που διανέμεται μέσω του Διαδικτύου με τη χρήση ενός προγράμματος περιήγησης και διατηρείται σε έναν απομακρυσμένο διακομιστή. Εξ ορισμού, οι υπηρεσίες διαδικτύου είναι Εφαρμογές Ιστού και πολλοί ιστότοποι -αν και όχι όλοι- περιέχουν τέτοιες εφαρμογές. Οι Εφαρμογές Ιστού μπορούν να δημιουργηθούν για ένα ευρύ φάσμα σκοπών και να χρησιμοποιηθούν από ανθρώπους ή οργανισμούς για πολλά διαφορετικά πράγματα. Το `webmail`, οι ηλεκτρονικές αριθμομηχανές και οι ιστότοποι ηλεκτρονικών αγορών είναι μερικά παραδείγματα συχνά χρησιμοποιούμενων Εφαρμογών Ιστού. Αν και η πλειονότητα των εφαρμογών ιστού είναι προσβάσιμες από οποιοδήποτε πρόγραμμα περιήγησης, ορισμένες απαιτούν συγκεκριμένο λογισμικό για την ορθή λειτουργία τους και τη βέλτιστη εμπειρία του χρήστη τους [7].

2.2 Τεχνικές Προγραμματισμού

2.2.1 Αντικειμενοστρεφής Προγραμματισμός

Ο αντικειμενοστρεφής προγραμματισμός (Object Oriented Programming / OOP), αποτελεί πρότυπο στην ανάπτυξη εφαρμογών, σε οποιαδήποτε γλώσσα. Βασίζεται στις ιδέες των κλάσεων και των αντικειμένων. Χρησιμοποιείται για την οργάνωση του λογισμικού σε απλές, επαναχρησιμοποιήσιμες κλάσεις σχεδίων κώδικα, οι οποίες στη συνέχεια χρησιμοποιούνται για τη δημιουργία ξεχωριστών περιπτώσεων αντικειμένων. Οι γλώσσες αντικειμενοστρεφούς προγραμματισμού JavaScript, C++, Java, Python και PHP είναι μερικά μόνο παραδείγματα [8].

Μια κλάση είναι ένα γενικευμένο πρότυπο που μπορεί να χρησιμοποιηθεί για τη δημιουργία πιο εξειδικευμένων, συγκεκριμένων πραγμάτων. Ευρείες κατηγορίες με κοινές ιδιότητες αναπαρίστανται συχνά με τη χρήση κλάσεων. Αυτές οι κλάσεις καθορίζουν τα χαρακτηριστικά μιας περίπτωσης τύπου, όπως το χρώμα, αλλά όχι την τιμή αυτών των χαρακτηριστικών για ένα συγκεκριμένο αντικείμενο.

Οι κλάσεις μπορούν επίσης να περιλαμβάνουν μεθόδους, οι οποίες είναι ειδικές λειτουργίες προσβάσιμες μόνο από αντικείμενα αυτού του είδους. Αυτές οι συναρτήσεις, οι οποίες ορίζονται εντός της κλάσης, εκτελούν ορισμένες χρήσιμες ενέργειες για το συγκεκριμένο είδος αντικείμενου.

Η δημιουργία μοναδικών αντικειμένων ακολουθεί το σχέδιο που παρέχουν τα πρότυπα κλάσεων. Κάθε αντικείμενο μπορεί να έχει διαφορετική τιμή για κάθε μία από τις δηλωμένες ιδιότητες της κλάσης [9].

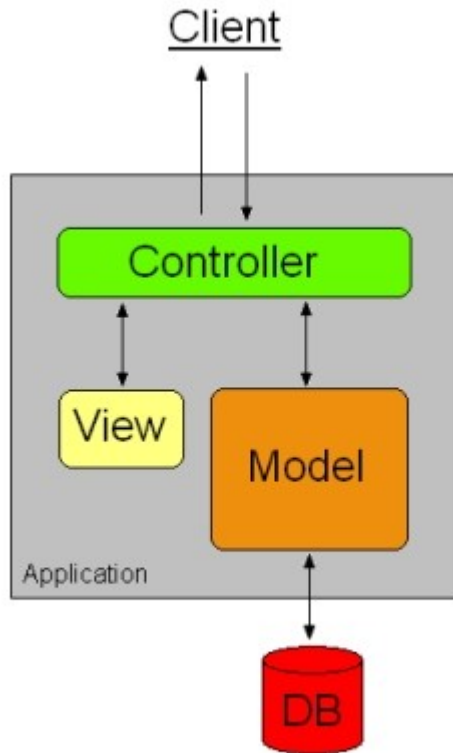
2.2.2 Αρχιτεκτονική MVC

Το MVC είναι ένα αρχιτεκτονικό πρότυπο, το οποίο υποδηλώνει ότι ελέγχει ολόκληρο το σχεδιασμό της εφαρμογής. Παρόλο που αναφέρεται συχνά ως πρότυπο σχεδίασης, αυτό μπορεί να είναι ανακριβές, διότι τα πρότυπα σχεδίασης χρησιμοποιούνται για την αντιμετώπιση ορισμένων τεχνικών ζητημάτων, αλλά τα πρότυπα αρχιτεκτονικής αντιμετωπίζουν αρχιτεκτονικά ζητήματα, έχοντας αντίκτυπο σε ολόκληρο τον σχεδιασμό του προγράμματος [10]. Κατά την ανάπτυξη μιας εφαρμογής, είναι απαραίτητη η οργάνωση του κώδικα, έτσι ώστε η πρόσβαση στα κατάλληλα αρχεία και η συντήρησή τους να είναι εύκολη και διαυφήνη. Η οργάνωση των αρχείων της εφαρμογής με τη χρήση του προτύπου Model, View, Controller (MVC) επιτυγχάνει τη διατήρηση των δεδομένων, της εμφάνισης και της ροής της εφαρμογής διακριτά [11].

Όπως φαίνεται και στο σχήμα 2.2, υπάρχουν τρία βασικά μέρη σε αυτό: τα Models για την επικοινωνία της εφαρμογής με τη βάση δεδομένων, τα Views για την απεικόνιση των πληροφοριών και οι Controllers για την υλοποίηση της επιχειρηματικής λογικής της εφαρμογής. Επίσης, ένα συμπληρωματικό βοηθητικό στοιχείο είναι τα Entities, τα οποία προσφέρουν αντικειμενοστρέφεια στις γραμμές αποτελεσμάτων που επιστρέφονται από τη βάση δεδομένων μέσω των Models.

Models

Τα μοντέλα διαχειρίζονται τα δεδομένα της εφαρμογής και βοηθούν στην επιβολή τυχόν ειδικών επιχειρησιακών κανόνων που μπορεί να χρειάζεται η εφαρμογή. Η δουλειά ενός μοντέλου είναι να διαχειρίζεται έναν συγκεκριμένο τύπο δεδομένων για την εφαρμογή. Αυτό μπορεί να είναι χρήστες, αναρτήσεις σε ιστολόγια, συναλλαγές κ.α. Σε αυτή την περίπτωση, η δουλειά του μοντέλου έχει δύο μέρη: να επιβάλλει επιχειρηματικούς κανόνες στα



Σχήμα 2.2: Διάγραμμα Αρχιτεκτονικής MVC Πηγή: [Blu11]

δεδομένα καθώς αυτά αντλούνται από τη βάση δεδομένων ή τοποθετούνται σε αυτήν και να χειρίζεται την πραγματική αποθήκευση και ανάκτηση των δεδομένων από τη βάση δεδομένων. Η επιβολή επιχειρηματικών κανόνων μπορεί να περιλαμβάνει την κανονικοποίηση των ακατέργαστων δεδομένων πριν από την αποθήκευση, ώστε να πληρούν τις απαραίτητες προδιαγραφές ασφαλείας, ή τη μορφοποίηση μιας στήλης με συγκεκριμένο τρόπο πριν από την παράδοσή της στον ελεγκτή. Διατηρώντας αυτές τις επιχειρηματικές απαιτήσεις στο μοντέλο, δεν επαναλαμβάνεται κώδικας σε πολλούς ελεγκτές, κατά την ανάκτηση δεδομένων από τη βάση ή πριν την αποθήκευσή τους σε αυτή.

Views

Το στοιχείο προβολής χειρίζεται την αναπαράσταση δεδομένων. Στην πραγματικότητα, δημιουργεί τη διεπαφή χρήστη (UI) για τον χρήστη. Τα δεδομένα που συγκεντρώνει το συστατικό μοντέλο για τις προβολές λαμβάνονται μέσω του ελεγκτή και όχι απευθείας, επομένως η προβολή επικοινωνεί μόνο με τον ελεγκτή [12].

Οι προβολές είναι τα πιο απλογραμμένα αρχεία και συχνά είναι μόνο HTML με πολύ λίγη PHP. Η PHP θα πρέπει να διατηρείται όσο το δυνατόν πιο βασική, συνήθως εμφανίζει τα περιεχόμενα μιας μεταβλητής ή εκτελεί επαναλήψεις σε διάφορα στοιχεία για την εμφάνιση των πληροφοριών τους.

Τα δεδομένα για τις προβολές προέρχονται από τους ελεγκτές, οι οποίοι τα στέλνουν στις προβολές ως μεταβλητές που μπορούν να εμφανιστούν με απλές κλήσεις echo. Άλλες προβολές μπορούν να εμφωλευθούν μέσα σε μια προβολή, καθιστώντας απλή την εμφάνιση μιας συνεπούς κεφαλίδας ή υποσέλιδου σε κάθε σελίδα.

Controllers

Οι ελεγκτές μπορούν να αναλάβουν δύο διαφορετικά καθήκοντα. Το πρώτο και πιο προφανές είναι ότι δέχονται την είσοδο του χρήστη και στη συνέχεια αποφασίζουν τι να κάνουν με αυτήν. Αυτό συχνά συνεπάγεται την παράδοση των δεδομένων σε ένα μοντέλο για τη διατήρησή τους ή την αίτηση δεδομένων από το μοντέλο για να παρουσιαστούν από την προβολή. Αυτό περιλαμβάνει επίσης τη φόρτωση επιπλέον βοηθητικών κλάσεων, αν είναι απαραίτητο, για την εκπλήρωση εξειδικευμένων δραστηριοτήτων που δεν καλύπτονται από το μοντέλο.

Ο ελεγκτής είναι επίσης υπεύθυνος για το χειρισμό όλων όσων σχετίζονται με τα αιτήματα HTTP, όπως η ανακατεύθυνση, ο έλεγχος ταυτότητας, η ασφάλεια ιστού, η κωδικοποίηση κ.ο.κ. Συνοπτικά, ο ελεγκτής διασφαλίζει ότι επιτρέπεται η πρόσβαση και ότι τα δεδομένα που χρειάζονται παρέχονται σε μορφή που μπορούν να χρησιμοποιήσουν.

Ως το συστατικό που παρέχει τη σύνδεση μεταξύ των προβολών και του μοντέλου, ο ελεγκτής αναφέρεται ως ο “κύριος ρόλος”, δεδομένου ότι λειτουργεί ως ενδιάμεσος. Ο ελεγκτής χρειάζεται μόνο να δίνει οδηγίες στο μοντέλο - δεν χρειάζεται να ανησυχεί για το χειρισμό της λογικής των δεδομένων. Αφού επεξεργαστεί τα δεδομένα που έχει λάβει από το μοντέλο, τα στέλνει στην προβολή και δίνει οδηγίες στον χρήστη για τον τρόπο απεικόνισής τους. Οι προβολές και τα μοντέλα δεν μπορούν να συνομιλούν άμεσα, παρά μόνο με την διαμεσολάβηση κάποιου ελεγκτή.

Entities

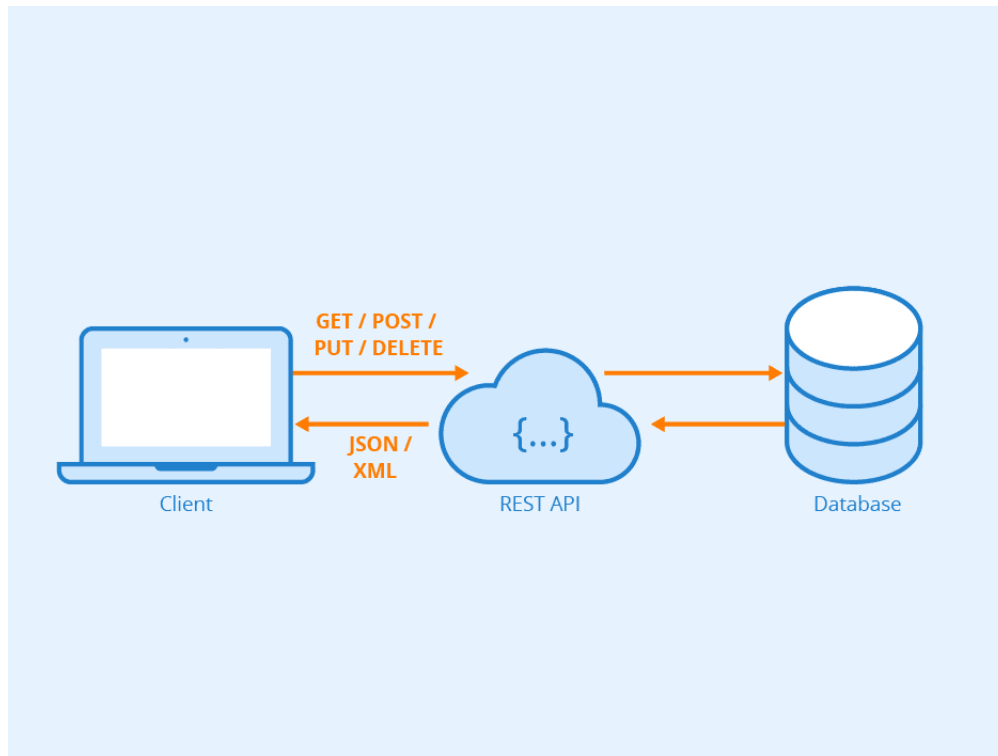
Οι κλάσεις οντοτήτων υποστηρίζονται ως πολίτες πρώτης κατηγορίας στο επίπεδο της βάσης δεδομένων, διατηρώντας παράλληλα την χρήση τους εντελώς προαιρετική.

Στον πυρήνα της, μια κλάση οντότητας είναι απλά μια κλάση που αναπαριστά μια μεμονωμένη γραμμή βάσης δεδομένων. Έχει ιδιότητες κλάσης για να αναπαριστά τις στήλες της βάσης δεδομένων και παρέχει οποιεσδήποτε πρόσθετες μεθόδους για την υλοποίηση της επιχειρηματικής λογικής για αυτή τη γραμμή. Το βασικό χαρακτηριστικό, όμως, είναι ότι δεν γνωρίζει τίποτα για το πώς να διατηρείται η ίδια. Αυτό είναι ευθύνη του μοντέλου ή της κλάσης αποθετηρίου. Με αυτόν τον τρόπο, αν αλλάξει κάτι σχετικά με τον τρόπο που πρέπει να αποθηκεύσετε το αντικείμενο, δεν χρειάζεται να αλλάξετε τον τρόπο με τον οποίο χρησιμοποιείται αυτό το αντικείμενο σε όλη την εφαρμογή.

2.2.3 RESTful API

Η μεταφορά κατάστασης παρουσίας (REpresentational State Transfer), κοινώς γνωστή ως REST, είναι ένας αρχιτεκτονικός σχεδιασμός για κατανεμημένα συστήματα υπερμέσων. Παρουσιάστηκε για πρώτη φορά στη διάσημη διατριβή του Roy Fielding από το 2000. Οι κατευθυντήριες έννοιες και οι περιορισμοί του REST είναι παρόμοιοι με εκείνους άλλων αρχιτεκτονικών στυλ. Εάν μια διεπαφή υπηρεσίας θέλει να αναφέρεται ως RESTful, πρέπει να τηρεί αυτά τα πρότυπα [13].

Ένα σύνολο ορισμών και πρωτοκόλλων, γνωστό ως API, χρησιμοποιείται για τη δημιουργία και την ενσωμάτωση λογισμικού εφαρμογών. Μερικές φορές αναφέρεται ως σύμβαση μεταξύ ενός παρόχου πληροφοριών και ενός χρήστη αυτών των πληροφοριών, η οποία περιγράφει το περιεχόμενο που ο καταναλωτής (η κλήση) και ο παραγωγός υποχρεούνται να παραδώσουν (η απάντηση). Το API περιγράφει τον τρόπο με τον οποίο ένας προ-



Σχήμα 2.3: REST API Πηγή: [Seo]

γραμματιστής θα πρέπει να σχεδιάσει ένα πρόγραμμα που ζητά υπηρεσίες από ένα λειτουργικό σύστημα ή μια άλλη εφαρμογή.

Ένα API μπορεί να θεωρηθεί ως μεσάζων μεταξύ των χρηστών ή των πελατών και των πόρων ή των επιχειρηματικών υπηρεσιών στις οποίες προσπαθούν να έχουν πρόσβαση. Επιπλέον, παρέχει σε μια εταιρεία έναν μηχανισμό για την κοινή χρήση περιουσιακών στοιχείων και δεδομένων, διατηρώντας παράλληλα την αυθεντικοποίηση, τον έλεγχο και την ασφάλεια - αποφασίζοντας ποιος έχει πρόσβαση σε τι [14].

Ένα RESTful API βασίζεται στην αναπαραστατική μεταφορά κατάστασης (REST), ένα αρχιτεκτονικό στυλ και μια προσέγγιση επικοινωνίας που χρησιμοποιείται συχνά στην κατασκευή υπηρεσιών ιστού. Είναι επίσης γνωστό ως RESTful υπηρεσία ιστού ή REST API.

Γενικά, η τεχνολογία REST προτιμάται έναντι άλλων, συγκρίσιμων τεχνολογιών. Αυτό συμβαίνει συχνά, καθώς η REST απαιτεί μικρότερο εύρος ζώνης και, ως εκ τούτου, είναι πιο κατάλληλη για την αποτελεσματική χρήση του διαδικτύου. Επιπλέον, γλώσσες υπολογιστών όπως η PHP, η JavaScript ή η Python μπορούν να χρησιμοποιηθούν για τη δημιουργία RESTful API [15].

2.3 Γλώσσες Προγραμματισμού Ιστού

2.3.1 HTML

Η HTML (Hyper Text Markup Language - Γλώσσα Σήμανσης Υπερκειμένου) αποτελεί την βασική γλώσσα για την κατασκευή ενός ιστοχώρου. Δεν θεωρείται γλώσσα προγραμματισμού, αλλά αποτελεί μία περιγραφική γλώσσα (markup language) η οποία περιέχει οδηγίες προς τους web browsers. Οι web browsers αφού διαβάσουν τις οδηγίες αυτές, τις μεταφράζουν στις κατάλληλες εντολές για να δημιουργηθεί το οπτικό περιεχόμενο που θα

παρουσιαστεί στον χρήστη. Η παραπάνω διαδικασία επιτυγχάνεται με την βοήθεια των HTML elements, τα οποία οριοθετούνται από ετικέτες (tags). Οι ετικέτες είναι γράμματα ή λέξεις, τα οποία περικλείουν γωνιώδεις αγκύλες και υπάρχουν ανά ζεύγη. Χωρίζονται σε tags έναρξης, τα οποία σηματοδοτούν την έναρξη μιας εντολής, και σε tags λήξης, τα οποία σηματοδοτούν την λήξη της εντολής, με μερικές εξαιρέσεις. Οι ετικέτες λήξης διαχωρίζονται από τις ετικέτες έναρξης με μία πλάγια γραμμή '/'. Ένα βασικό παράδειγμα ετικέτας είναι το: <html>...</html>, ανάμεσα στις ετικέτες περικλείεται κείμενο ή ακόμη και άλλες εσωτερικές ετικέτες (εμφωλευμένα tags). Ορισμένες βασικές ετικέτες παρατίθενται στον πίνακα 2.1.

Πίνακας 2.1: Βασικά στοιχεία HTML

<html>...</html>	Η αρχή και το τέλος του HTML αρχείου
<!DOCTYPE>	Η οδηγία που καθορίζει την έκδοση της HTML που χρησιμοποιείται
<head>...</head>	Οι σχετικές πληροφορίες με το έγγραφο (όπως η γλώσσα, η κωδικοποίηση και τα μεταδεδομένα)
<title>...</title>	Ο τίτλος του αρχείου
<body>...</body>	Τα οπτικά στοιχεία του αρχείου
<div>...</div>	Ομαδοποίηση στοιχείων εντός ετικέτας
<input>...</input>	Ορισμός πεδίου εισαγωγής
<!--...-->	Ορισμός σχολίων
<form>...</form>	Ορισμός φόρμας
<button>...</button>	Ορισμός κουμπιού
<a>...	Ορισμός υπερσυνδέσμου

2.3.2 CSS

Η CSS αποτελεί μία γλώσσα επικαλυπτόμενων στυλ μορφοποίησης (Cascading Style Sheets) και όπως η HTML δεν θεωρείται καθαρή γλώσσα προγραμματισμού. Συντελεί στον διαχωρισμό των εντολών εμφάνισης από τις εντολές του περιεχομένου της ιστοσελίδας. Χρησιμοποιείται για τη μορφοποίηση οποιασδήποτε ετικέτας HTML και τη δημιουργία ενός αποτελέσματος πιο όμορφου οπτικά για τον χρήστη. Η σύνταξη μιας εντολής στην CSS αποτελείται από τρία κύρια στοιχεία: το στοιχείο που θα τροποποιηθεί, τις ιδιότητες του στοιχείου που θα επηρεαστούν και η νέα τιμή των ιδιοτήτων. Στον πίνακα 2.2 αναφέρεται ένα παράδειγμα του τρόπου σύνταξης μιας εντολής στην γλώσσα CSS.

Όπως φαίνεται στον πίνακα 2.2, το στοιχείο p αποτελεί την ετικέτα της παραγράφου στην HTML και ονομάζεται επιλογέας (css selector). Τα font-family, color και text-align είναι οι ιδιότητες της εντολής CSS και δείχνουν τα

Πίνακας 2.2: Στοιχεία εντολής CSS

```
p {  
    font-family: Calibri;  
    color: #1B1811;  
    text-align: left;  
}
```

στοιχεία του επιλογέα τα οποία θα τροποποιηθούν. Ο διαχωρισμός των ιδιοτήτων από τις νέες τιμές που θα λάβουν σηματοδοτείται με την άνω και κάτω τελεία (:), στις οποίες το αριστερό μέρος βρίσκονται οι ιδιότητες και στο δεξί οι νέες τιμές τους.

2.3.3 JavaScript & Ajax

Η JavaScript είναι μια γλώσσα προγραμματισμού σεναρίου (scripting language), η οποία εκτελείται από τους περιηγητές ιστών (web browsers) χρησιμοποιώντας έναν σχετικό διερμηνευτή (Interpreter). Στόχο της αποτελεί η βελτίωση της εμπειρίας χρήσης. Με την JavaScript υπάρχει η δυνατότητα προσθήκης ή αφαίρεσης HTML στοιχείων και CSS κανόνων καθώς και η τροποποίηση ιδιοτήτων HTML στοιχείων με την μεταβολή στις τιμές τους. Αν και η JavaScript αποτελεί μια γλώσσα προγραμματισμού με το Client-Side χαρακτηριστικό, τον τελευταίο καιρό γίνεται χρήση της και από την πλευρά των υπολογιστών εξυπηρετητών.

Η Ajax (Asynchronous JavaScript and XML) αποτελείται από τον συνδυασμό των τεχνολογιών JavaScript και XML. Είναι μία τεχνολογία, η οποία προσδίδει διαδραστικές δυνατότητες σε μία ιστοσελίδα. Μέσω της Ajax γίνεται εφικτή η ανανέωση μέρους της ιστοσελίδας (στο παρασκήνιο θα γίνει επικοινωνία της τεχνολογίας με τον server, ο οποίος θα λάβει τα δεδομένα που ζητήθηκαν και με τη σειρά του θα τα εμφανίσει στον χρήστη), χωρίς να χρειαστεί να γίνει ανανέωση (refresh) ολόκληρης της ιστοσελίδας.

2.3.4 PHP

Η PHP ορίζεται ως μία αντικειμενοστρεφής γλώσσα προγραμματισμού γενικής χρήσης (παλαιότερα αποτελούσε γλώσσα προγραμματισμού σεναρίου), η οποία χρησιμοποιείται κυρίως για την ανάπτυξη διαδικτυακών εφαρμογών (web apps), δηλαδή αποτελεί την κατάλληλη γλώσσα για την δημιουργία ιστοχώρων με δυναμικό περιεχόμενο. Διατίθεται σε δύο μορφές, σε μορφή πηγαίου κώδικα και σε δυαδική μορφή και οι δύο αυτές μορφές έχουν ελεύθερη πρόσβαση. Η PHP χρησιμοποιείται για τον χειρισμό λειτουργιών και εργασιών τις οποίες θα υλοποιήσει και όχι για την οπτική διαμόρφωση μίας ιστοσελίδας.

Επομένως, ο χρήστης λαμβάνει τα αποτελέσματα του σεναρίου (στον browser ως απλές σελίδες HTML) και όχι τον κώδικα, ο οποίος εκτελείται στον server με τη χρήση του αντίστοιχου διερμηνευτή (interpreter) της κάθε γλώσσας. Ο διερμηνευτής, αφού διαβάσει τον κώδικα, εκτελεί τις δηλώσεις της γλώσσας ανά βήμα και τις μετατρέπει σε εκτελέσιμο κώδικα για το υπολογιστικό σύστημα.

Αναλυτικότερα, η PHP έχει την ικανότητα να δημιουργήσει, γράψει, διαβάσει, ανοίξει, κλείσει και διαγράψει αρχεία στη Βάση Δεδομένων. Υποστηρίζει ένα ευρύ φάσμα Βάσεων Δεδομένων, καθώς επίσης είναι δωρεάν και αρκετά εύκολη στην εκμάθηση, ενώ ταυτόχρονα είναι συμβατή και μπορεί να τρέξει σε οποιαδήποτε πλατφόρμα,

όπως για παράδειγμα Windows, Mac OS, Linux, Unix κ.α. Ένα αρχείο PHP (έγγραφο κειμένου αποθηκευμένο με την κατάληξη .php) είναι δυνατό να περιέχει εκτός από τον κώδικα PHP και κώδικα HTML, CSS και JavaScript.

Ορισμένες βασικές συναρτήσεις της PHP παρατίθενται στον πίνακα 2.3.

Πίνακας 2.3: Βασικές συναρτήσεις της PHP

htmlspecialchars()	Μετατροπή ειδικών χαρακτήρων σε οντότητες HTML
explode()	Μετατροπή μιας συμβολοσειράς σε πίνακα με χρήση ενός διαχωριστικού χαρακτήρα
rand()	Επιστροφή ενός τυχαίου αριθμού
str_replace()	Εύρεση και αντικατάσταση ενός μοτίβου σε μια συμβολοσειρά
date()	Επιστροφή μιας μορφοποιημένης ημερομηνίας
strlen()	Επιστροφή του μήκους μιας συμβολοσειράς
count()	Επιστροφή του αριθμού των στοιχείων ενός πίνακα
array_unique()	Απαλοιφή διπλότυπων στοιχείων ενός πίνακα
print_r()	Εκτύπωση μεταβλητής
echo	Εκτύπωση μίας ή περισσότερων συμβολοσειρών

2.3.5 SQL

Η τυποποιημένη γλώσσα προγραμματισμού Structured Query Language (SQL) [16], χρησιμοποιείται για τη διαχείριση σχεσιακών βάσεων δεδομένων και την εκτέλεση διαφόρων πράξεων στα δεδομένα που περιέχουν. Από την ίδρυσή της τη δεκαετία του 1970, η SQL χρησιμοποιείται ευρέως από διαχειριστές βάσεων δεδομένων καθώς και από προγραμματιστές που δημιουργούν σενάρια για την ολοκλήρωση δεδομένων και από αναλυτές δεδομένων που κατασκευάζουν και εκτελούν αναλυτικά ερωτήματα.

Με τη βοήθεια της ευέλικτης γλώσσας SQL, οι χρήστες μπορούν να λαμβάνουν, να αποθηκεύουν, να επεξεργάζονται και να διαγράφουν δεδομένα, καθώς και να δημιουργούν, να τροποποιούν και να αφαιρούν αντικείμενα βάσεων δεδομένων (όπως πίνακες, στήλες, διαδικασίες και χρήστες), καθώς και να δίνουν και να ανακαλούν προνόμια χρηστών και να ομαδοποιούν δηλώσεις σε συναλλαγές. Μια δήλωση που ζητά δεδομένα από τη βάση δεδομένων αναφέρεται ως ερώτημα και μια εντολή SQL είναι γνωστή ως δήλωση. Στις εντολές SQL περιλαμβάνονται προγνωστικά (όπως LIKE, BETWEEN, EXISTS), τελεστές (όπως AND, OR, NOT), ποσοδείκτες (όπως ANY, ALL, UNION), συναρτήσεις (όπως COUNT, SUM, AVG) και προτάσεις (όπως SELECT, FROM, WHERE). Όταν δεν είναι σημαντικό να γίνει διάκριση μεταξύ, για παράδειγμα, των ρητρών και των κατηγορημάτων, αναφερόμαστε σε αυτά ως έννοιες συλλογικά.

Η γλώσσα χειρισμού δεδομένων (DML, π.χ. SELECT, INSERT, UPDATE, DELETE) και η γλώσσα ορισμού δεδομένων (DDL, π.χ. CREATE, ALTER, DROP) είναι οι δύο υπογλώσσες της SQL που χρησιμοποιούνται συχνότερα [17].

Τα δεδομένα αποθηκεύονται, ανακτώνται και αναλύονται με τη χρήση λογισμικού που ονομάζεται σύστημα διαχείρισης βάσεων δεδομένων (DBMS). Οι χρήστες μπορούν να δημιουργούν, να διαβάζουν, να ενημερώνουν και να διαγράφουν δεδομένα σε βάσεις δεδομένων χρησιμοποιώντας ένα DBMS, το οποίο λειτουργεί ως διεπαφή μεταξύ αυτών και των βάσεων δεδομένων. Η ασφάλεια των δεδομένων, η ακεραιότητα των δεδομένων, η ταυτόχρονη χρήση και οι τυποποιημένες πρακτικές διαχείρισης δεδομένων βοηθούνται από αυτό.

Τα μοντέλα δεδομένων, οι κατανομές βάσεων δεδομένων, ο αριθμός των χρηστών και άλλοι παράγοντες μπορούν να χρησιμοποιηθούν για την κατηγοριοποίηση των συστημάτων διαχείρισης βάσεων δεδομένων. Οι σχεσιακές, καταμεμημένες, ιεραρχικές, αντικειμενοστραφείς και δικτυακές μορφές λογισμικού DBMS είναι οι πιο δημοφιλείς [18].

Στα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS), η SQL χρησιμοποιείται για τις ακόλουθες εργασίες: αλλαγή των δομών των πινάκων και των δεικτών της βάσης δεδομένων, προσθήκη, ενημέρωση και διαγραφή γραμμών δεδομένων και ανάκτηση υποσυνόλων πληροφοριών. Οι πληροφορίες αυτές μπορούν να χρησιμοποιηθούν για την επεξεργασία συναλλαγών, εφαρμογές ανάλυσης και άλλες εφαρμογές που απαιτούν αλληλεπίδραση με μια σχεσιακή βάση δεδομένων.

2.4 Πλαίσια και Βιβλιοθήκες Προγραμματισμού Ιστού

2.4.1 Bootstrap

Μια βιβλιοθήκη HTML, CSS και JS με έμφαση στη βελτιστοποίηση της διαδικασίας ανάπτυξης ιστοσελίδων ονομάζεται Bootstrap (σε αντίθεση με τις εφαρμογές ιστού). Το Bootstrap είναι ένα δωρεάν και ανοικτού κώδικα πλαίσιο CSS που δημιουργήθηκε αρχικά από το Twitter και σήμερα αναπτύσσεται από την ομάδα του Bootstrap στο Github. Σκοπός του είναι να αποτελέσει τα θεμέλια για την ανάπτυξη front-end ιστοσελίδων που δίνουν προτεραιότητα στη καλή εμπειρία χρήσης από κινητές συσκευές. Προσφέρει πρότυπα σχεδίασης HTML, CSS και (προαιρετικά) JavaScript για τυπογραφία, φόρμες, κουμπιά, πλοήγηση και άλλα στοιχεία διεπαφής [19].

Ο κύριος στόχος της προσθήκης του σε ένα έργο ιστού είναι η εφαρμογή των επιλογών χρώματος, μεγέθους, γραμματοσειράς και διάταξης του Bootstrap σε αυτό το έργο. Ως εκ τούτου, ο κύριος καθοριστικός παράγοντας είναι το κατά πόσον αυτές οι επιλογές αρέσουν στους υπεύθυνους προγραμματιστές. Όλα τα στοιχεία HTML έχουν βασικές δηλώσεις στυλ μόλις το Bootstrap εισαχθεί σε ένα έργο.

Ως αποτέλεσμα, τα κείμενα, οι πίνακες και τα στοιχεία φόρμας εμφανίζονται ομοιόμορφα σε όλα τα προγράμματα περιήγησης ιστού. Προκειμένου να εξατομικεύσουν περαιτέρω την εμφάνιση του περιεχομένου τους, οι προγραμματιστές μπορούν να κάνουν χρήση των κλάσεων CSS που ορίζονται στο Bootstrap. Για παράδειγμα, το Bootstrap προσφέρει ενσωματωμένη υποστήριξη για ανοιχτόχρωμους και σκούρους πίνακες, επικεφαλίδες σελίδων, μεγαλύτερα pull quotes και κείμενο με υπογράμμιση [20].

Επιπλέον, το Bootstrap περιλαμβάνει μια σειρά από στοιχεία JavaScript που μπορούν να χρησιμοποιηθούν ανεξάρτητα από άλλα πλαίσια, όπως το jQuery. Προσφέρουν επιπλέον στοιχεία UI, όπως πλαίσια διαλόγου, tooltips, μπάρες προόδου, drop-down μενού και καρousel [21]. Κάθε στοιχείο Bootstrap αποτελείται από ένα πλαίσιο

HTML, δηλώσεις CSS και περιστασιακά από συμπληρωματικό κώδικα JavaScript. Ταυτόχρονα, αυξάνουν τη λειτουργικότητα μερικών ήδη υπάρχοντων στοιχείων διεπαφής, όπως η λειτουργία αυτόματης συμπλήρωσης για τα πεδία εισαγωγής.

2.4.2 CodeIgniter

Το CodeIgniter είναι ένα πλαίσιο ανάπτυξης εφαρμογών MVC, διαθέσιμο για προγραμματιστές ιστοσελίδων PHP. Προσφέροντας μια ολοκληρωμένη συλλογή βιβλιοθηκών για συχνά εκτελούμενες λειτουργίες, καθώς και μια φιλική προς το χρήστη διεπαφή και μια λογική δομή πρόσβασης, στοχεύει να κάνει τη δημιουργία έργων σημαντικά ταχύτερη από τη συγγραφή κώδικα από το μηδέν. Το CodeIgniter προσπαθεί να γράφει όσο το δυνατόν λιγότερο κώδικα για την ολοκλήρωση μιας εργασίας [22].

Πρόκειται για ένα πλαίσιο με αμελητέες απαιτήσεις διαμόρφωσης. Προσφέρει εξαιρετικές επιδόσεις, διαθέτει λεπτομερή, σαφή τεκμηρίωση και δεν απαιτεί τη χρήση της γραμμής εντολών ή την τήρηση αυστηρών οδηγιών κωδικοποίησης. Αποφεύγει την πολυπλοκότητα και ευνοεί τις απλές διορθώσεις [23].

2.4.3 Swagger - OpenAPI

Το Swagger είναι μια συλλογή οδηγιών, απαιτήσεων και πόρων ανοικτού κώδικα για τη δημιουργία και τον ορισμό RESTful APIs, τα οποία υλοποιούν τις προδιαγραφές του OpenAPI. Το έργο ξεκίνησε το 2011 από τον Tony Tam, ο οποίος εργαζόταν στη δημιουργία εργαλείων για τον ιστότοπο λεξικού Wordnik και πλέον συντηρείται από την SmartBear Software. Οι προγραμματιστές μπορούν να γράφουν διαδραστική, αναγνώσιμη από μηχανήματα και ανθρώπους τεκμηρίωση API χρησιμοποιώντας το πλαίσιο Swagger [24].

Πληροφορίες όπως υποστηριζόμενες λειτουργίες, παράμετροι και έξοδοι, ανάγκες εξουσιοδότησης, προσβάσιμα τελικά σημεία και απαιτούμενη αδειοδότηση περιλαμβάνονται συχνά στις προδιαγραφές API. Ζητώντας από το API να παραδώσει ένα αρχείο τεκμηρίωσης από τις επισημάνσεις του, το Swagger μπορεί να παράγει αυτόματα αυτές τις πληροφορίες από τον πηγαίο κώδικα.

Οι χρήστες μπορούν να δημιουργούν, να τεκμηριώνουν, να δοκιμάζουν και να καταναλώνουν RESTful υπηρεσίες ιστού με τη βοήθεια του Swagger. Εφαρμόζεται τόσο σε μεθοδολογίες ανάπτυξης API από πάνω προς τα κάτω όσο και από κάτω προς τα πάνω. Το Swagger μπορεί να χρησιμοποιηθεί για τη δημιουργία ενός API στην προσέγγιση top-down ή “design-first”, πριν αναπτυχθεί οποιοσδήποτε κώδικας. Η τεκμηρίωση παράγεται από το Swagger χρησιμοποιώντας τον κώδικα που δημιουργείται για ένα API στην προσέγγιση “από κάτω προς τα πάνω” ή “code-first” [25].

2.5 Τεχνολογίες Ανάπτυξης Λογισμικού

2.5.1 JetBrains PhpStorm

Κατασκευασμένο από την τσεχική εταιρεία JetBrains, το PhpStorm είναι ένα ιδιόκτητο, διαπλατφορμικό IDE (ολοκληρωμένο περιβάλλον ανάπτυξης) για PHP [26]. Με on-the-fly ανάλυση κώδικα, πρόληψη σφαλμάτων και αυτοματοποιημένες αναδιαμορφώσεις για κώδικα PHP και JavaScript, το PhpStorm προσφέρει έναν επεξεργαστή για PHP, HTML και JavaScript. Η PHP υποστηρίζεται από τη συμπλήρωση κώδικα του PhpStorm, η οποία υποστηρίζει επίσης namespaces, closures, traits, generators, coroutines, τη λέξη-κλειδί finally, λίστες σε foreach και σύνταξη σύντομων πινάκων. Περιλαμβάνεται ένας πλήρης επεξεργαστής SQL με επεξεργάσιμα αποτελέσματα ερωτημάτων.

Το PhpStorm είναι γραμμένο σε Java. Εγκαθιστώντας plugins που έχουν κατασκευαστεί για το PhpStorm ή δημιουργώντας τα δικά τους, οι χρήστες μπορούν να επεκτείνουν τη λειτουργικότητα του IDE. Επιπλέον, το λογισμικό αλληλεπιδρά με εξωτερικούς πόρους όπως το XDebug. Το PhpStorm περιέχει όλες τις δυνατότητες που υπάρχουν στο WebStorm. Τα πρόσθετα JavaScript είναι ήδη προεγκατεστημένα στο WebStorm [27].

2.5.2 Git - GitHub

Git

Το Global Information Tracker (Git) είναι ένα σύστημα ελέγχου εκδόσεων (VCS) διαθέσιμο σε όλες τις κύριες πλατφόρμες ανάπτυξης μέσω μιας άδειας χρήσης ελεύθερου λογισμικού [28]. Κύριο χαρακτηριστικό του git είναι η διευκόλυνση της κατανεμημένης ανάπτυξης μέσω πολλαπλών προγραμματιστών σε πολλαπλές τοποθεσίες, γεγονός που οδηγεί στην κλιμάκωση του χειρισμού ακόμη και χιλιάδων προγραμματιστών που εργάζονται στα ίδια ή διαφορετικά μέρη ενός έργου. Άλλα χαρακτηριστικά του αποτελούν η γρήγορη και αποδοτική συγχώνευση με τη χρήση τεχνικών “delta” και συμπίεσης, η διατήρηση της ακεραιότητας των δεδομένων με την κρυπτογραφική συνάρτηση κατακερματισμού (SHA1) και η επιβολή της λογοδοσίας. Επίσης, υποστηρίζει και ενθαρρύνει τη διακλαδισμένη και σε άλλες περιπτώσεις τη συγχωνευμένη ανάπτυξη. Μία ακόμη ιδιότητα του git είναι πως διαθέτει πλήρη αποθετήρια. Ένα αποθετήριο αποτελείται από πολλούς φακέλους και αρχεία τα οποία είναι προσβάσιμα από οποιονδήποτε έχει πρόσβαση στο ιδιωτικό αποθετήριο [29].

Το Git διαθέτει έναν αριθμό από Graphical User Interfaces (GUI) πελάτες και ένας από αυτούς είναι το GitKraken [30], το οποίο αποτελεί ένα γραφικό περιβάλλον διαχείρισης git αποθετηρίων. Χρησιμοποιείται για την εύκολη και αποτελεσματική διαχείριση, την δημιουργία, αλλά και την συγχώνευση των διακλαδώσεων. Διαθέτει την δυνατότητα να ενσωματώνεται με τον λογαριασμό GitHub ή Bitbucket ενός χρήστη, καθώς επίσης μπορεί να προσαρμοστεί εύκολα στον χώρο εργασίας του χρήστη.

GitHub

Το GitHub είναι μία διαδικτυακή πλατφόρμα συνεργατικής φιλοξενίας κώδικα ανάπτυξης λογισμικού, που βασίζεται στο σύστημα ελέγχου εκδόσεων git. Μέσω του GitHub, οι προγραμματιστές έχουν την δυνατότητα να αντιγράψουν ένα αποθετήριο στον λογαριασμό τους (fork) και αφού πραγματοποιήσουν τις αλλαγές τους, τους δίνεται η δυνατότητα να προτείνουν να ενσωματωθούν στο αρχικό αποθετήριο από τους συντηρητές του (pull

request και merge). Οι προγραμματιστές μπορούν επίσης να ακολουθήσουν άλλους χρήστες, σχηματίζοντας έτσι ένα κύκλωμα ανταλλαγής πληροφοριών, όπου κάθε προγραμματιστής ενημερώνεται για εξελίξεις σχετικές με τα ενδιαφέροντα του [31]. Εκτός από τις λειτουργίες του Git, το GitHub προσφέρει και μερικά επιπρόσθετα χαρακτηριστικά. Βοηθά στη διαχείριση της πρόσβασης και της συνεργασίας για έργα, παρέχοντας χαρακτηριστικά όπως είναι η παρακολούθηση σφαλμάτων, η διαχείριση των διαφόρων εργασιών και τα αιτήματα χαρακτηριστικών.

2.6 Σύνοψη Κεφαλαίου 2

Στο δεύτερο κεφάλαιο εξετάστηκαν οι μέθοδοι και οι γλώσσες προγραμματισμού που εφαρμόστηκαν για την επεξεργασία και την ανάπτυξη της διαδικτυακής εφαρμογής. Εκτός από τα εργαλεία προγραμματισμού, η διατριβή περιλαμβάνει επίσης μια λεπτομερή περιγραφή των εργαλείων λογισμικού που χρησιμοποιήθηκαν για την οργάνωση και την κατασκευή του πληροφοριακού συστήματος.

Η κατασκευή της βάσης δεδομένων και η δομή και οργάνωση του συστήματος καλύπτονται στο επόμενο κεφάλαιο.

Κεφάλαιο 3

Δομή & Οργάνωση Συστήματος

Σε αυτό το κεφάλαιο, περιγράφονται λεπτομερώς οι λειτουργικές και μη-λειτουργικές απαιτήσεις του συστήματος διαχείρισης εκκίνησης υπολογιστών μέσω δικτύου. Επίσης, αναλύονται τα δύο είδη χρηστών που έχουν πρόσβαση στο σύστημα και οι δυνατότητες που τους παρέχονται από τη διαδικτυακή εφαρμογή του συστήματος. Τέλος, γίνεται ανάλυση του σχήματος της βάσης δεδομένων της διαδικτυακής εφαρμογής.

3.1 Ανάλυση Απαιτήσεων

Σε αυτή την ενότητα παρουσιάζονται αναλυτικά οι λειτουργικές και μη-λειτουργικές απαιτήσεις του συστήματος. Οι λειτουργικές απαιτήσεις είναι δηλώσεις που καθορίζουν τις υπηρεσίες οι οποίες προσφέρονται από το σύστημα, τον τρόπο ανταπόκρισης του σε συγκεκριμένες εισόδους και τη συμπεριφορά του σε συγκεκριμένες περιστάσεις. Οι μη-λειτουργικές απαιτήσεις είναι αυτές που ορίζουν κριτήρια, και όχι συγκεκριμένες συμπεριφορές, που μπορούν να χρησιμοποιηθούν για την αξιολόγηση της απόδοσης του συστήματος, ως προς την ποιότητα των υπηρεσιών του και την ικανοποίηση των αναγκών των χρηστών του.

3.1.1 Λειτουργικές Απαιτήσεις

Για την επίτευξη της ορθής ανάλυσης και σχεδίασης του συστήματος, είναι απαραίτητη η κατανόηση των λειτουργιών του και των δυνατοτήτων που προσφέρει στους χρήστες του. Οι χρήστες είναι είτε γενικοί διαχειριστές του συστήματος, είτε διαχειριστές συγκεκριμένων εργαστηρίων.

Οι γενικοί διαχειριστές του συστήματος θα πρέπει να έχουν τη δυνατότητα να προσθέτουν, να αφαιρούν και να επεξεργάζονται στοιχεία υπολογιστών, οι οποίοι δύνανται να εκκινηθούν δικτυακά, με μενού επιλογής λειτουργικού συστήματος παρεχόμενο από το σύστημα. Είναι επίσης απαραίτητο, να μπορούν να προσθέτουν και να διαχειρίζονται εργαστήρια στα οποία βρίσκονται οι υπολογιστές, και να αναθέτουν υπολογιστές σε αυτά. Μια ακόμη ενέργεια στην οποία θα μπορούν να προβούν οι γενικοί διαχειριστές, είναι να τοποθετούν τους υπολογιστές σε ομάδες, οι οποίες καθορίζουν τα μενού εκκίνησης που θα παρέχονται από το σύστημα, ανάλογα με τα ενεργά χρονοδιαγράμματα των μενού εκκίνησης. Τέλος, είναι αναγκαίο να επιτρέπεται στους γενικούς διαχειριστές του συστήματος ο ορισμός των διαθέσιμων block εντολών τύπου iPXE, η δημιουργία μενού εκκίνησης με χρήση των ορισμένων λειτουργικών συστημάτων και η ρύθμιση των χρονοδιαγραμμάτων τα οποία καθορίζουν τα μενού εκκίνησης που θα παρέχει το σύστημα στις ομάδες υπολογιστών.

Οι διαχειριστές εργαστηρίων θα πρέπει να μπορούν να πραγματοποιούν τις ίδιες ενέργειες με τους γενικούς διαχειριστές, αλλά μόνο για τα εργαστήρια που τους έχουν ανατεθεί. Θα μπορούν δηλαδή, να διαχειρίζονται τους

υπολογιστές, τις ομάδες υπολογιστών και τα μενού εκκίνησης των ομάδων υπολογιστών που βρίσκονται στα εργαστήρια, για τα οποία έχουν λάβει δικαίωμα διαχείρισης από τους γενικούς διαχειριστές.

Όλες οι λειτουργίες της πλατφόρμας θα πρέπει να παρέχονται μέσω προγραμματιστικών διεπαφών, ώστε να είναι δυνατή η επέκτασή της και η επίτευξη της διαλειτουργικότητάς της με άλλα πληροφοριακά συστήματα.

Τέλος, κάθε ενέργεια που αλλάζει την κατάσταση του συστήματος ή/και των πληροφοριών που είναι αποθηκευμένες σε αυτό, θα πρέπει να καταγράφεται στο αρχείο συμβάντων, μαζί με τη χρονοσφραγίδα και τον χρήστη που την πραγματοποίησε.

3.1.2 Μη-Λειτουργικές Απαιτήσεις

Η διεπαφή χρήστη θα πρέπει να είναι φιλική προς τον χρήστη. Πιο συγκεκριμένα, θα πρέπει να προσαρμόζεται στο μέγεθος της οθόνης της συσκευής που χρησιμοποιείται (υπολογιστής, τάμπλετ, κινητό). Θα πρέπει, επίσης, να είναι ξεκάθαρη ώστε να μην προκαλεί σύγχυση στον χρήστη κατά την εκτέλεση κάποιας λειτουργίας. Τέλος, θα πρέπει να είναι αποδοτική, δηλαδή να επιτρέπει στον χρήστη να διεκπεραιώνει εύστοχα και σε εύλογο χρονικό διάστημα τις ενέργειες που επιθυμεί.

Οι προγραμματιστικές διεπαφές που παρέχονται από την πλατφόρμα θα πρέπει να είναι συμβατές με την προδιαγραφή του OpenAPI 3, να συμμορφώνονται στα πρότυπα της αρχιτεκτονικής REST και να είναι πλήρως τεκμηριωμένες.

Οι πληροφορίες που αποθηκεύονται στη βάση δεδομένων του συστήματος θα πρέπει να είναι ελάχιστες απαιτούμενες για τη λειτουργία της πλατφόρμας και να υπακούν στους κανονισμούς περί ασφάλειας προσωπικών δεδομένων (GDPR). Οι κωδικοί πρόσβασης των χρηστών είναι απαραίτητο να αποθηκεύονται κρυπτογραφημένοι με κάποια σύγχρονη και ασφαλή μέθοδο. Τέλος, η βάση δεδομένων θα πρέπει να είναι κανονικοποιημένη, ώστε να αποδίδει με ταχύτητα και να μην διατηρεί διπλότυπα αποθηκευμένων πληροφοριών.

Είναι σημαντικό να υπάρχει προστασία από γνωστές απειλές, όπως Cross-Site Request Forgery (CSRF), SQL Injections και bots. Επιπρόσθετα, η συνεδρία κάθε συνδεδεμένου χρήστη θα πρέπει να λήγει μετά από αδράνεια 30 λεπτών, ώστε να αποφευχθεί πιθανή μη εξουσιοδοτημένη πρόσβαση στο σύστημα.

3.2 Περιπτώσεις Χρήσης

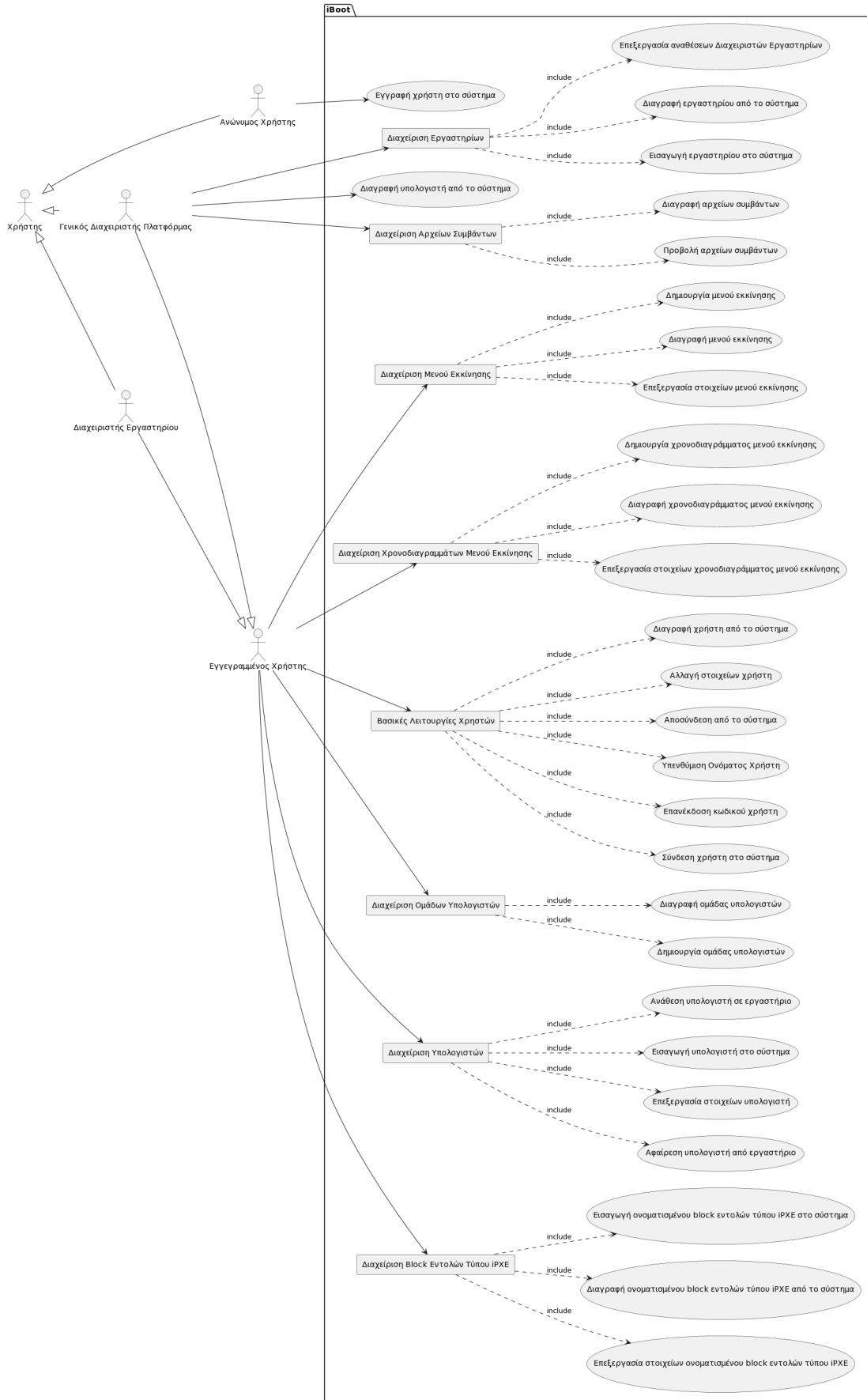
Στην παρούσα ενότητα παρουσιάζονται οι περιπτώσεις χρήσης της διαδικτυακής εφαρμογής, οι οποίες ονομαστικά είναι οι παρακάτω:

- Περίπτωση Χρήσης: Εγγραφή χρήστη στο σύστημα
- Περίπτωση Χρήσης: Σύνδεση χρήστη στο σύστημα
- Περίπτωση Χρήσης: Επανεκδοση κωδικού χρήστη
- Περίπτωση Χρήσης: Υπενθύμιση Ονόματος Χρήστη
- Περίπτωση Χρήσης: Αποσύνδεση από το σύστημα
- Περίπτωση Χρήσης: Αλλαγή στοιχείων χρήστη
- Περίπτωση Χρήσης: Διαγραφή χρήστη από το σύστημα

-
- Περίπτωση Χρήσης: Εισαγωγή εργαστηρίου στο σύστημα
 - Περίπτωση Χρήσης: Διαγραφή εργαστηρίου από το σύστημα
 - Περίπτωση Χρήσης: Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων
 - Περίπτωση Χρήσης: Δημιουργία ομάδας υπολογιστών
 - Περίπτωση Χρήσης: Διαγραφή ομάδας υπολογιστών
 - Περίπτωση Χρήσης: Εισαγωγή υπολογιστή στο σύστημα
 - Περίπτωση Χρήσης: Ανάθεση υπολογιστή σε εργαστήριο
 - Περίπτωση Χρήσης: Αφαίρεση υπολογιστή από εργαστήριο
 - Περίπτωση Χρήσης: Επεξεργασία στοιχείων υπολογιστή
 - Περίπτωση Χρήσης: Διαγραφή υπολογιστή από το σύστημα
 - Περίπτωση Χρήσης: Εισαγωγή ονοματισμένου block εντολών τύπου iPXE στο σύστημα
 - Περίπτωση Χρήσης: Επεξεργασία στοιχείων ονοματισμένου block εντολών τύπου iPXE
 - Περίπτωση Χρήσης: Διαγραφή ονοματισμένου block εντολών τύπου iPXE από το σύστημα
 - Περίπτωση Χρήσης: Δημιουργία μενού εκκίνησης
 - Περίπτωση Χρήσης: Επεξεργασία στοιχείων μενού εκκίνησης
 - Περίπτωση Χρήσης: Διαγραφή μενού εκκίνησης
 - Περίπτωση Χρήσης: Δημιουργία χρονοδιαγράμματος μενού εκκίνησης
 - Περίπτωση Χρήσης: Επεξεργασία στοιχείων χρονοδιαγράμματος μενού εκκίνησης
 - Περίπτωση Χρήσης: Διαγραφή χρονοδιαγράμματος μενού εκκίνησης
 - Περίπτωση Χρήσης: Προβολή αρχείων συμβάντων
 - Περίπτωση Χρήσης: Διαγραφή αρχείων συμβάντων

Το διάγραμμα περιπτώσεων χρήσης του συστήματος (UML Use Case Diagram), απεικονίζεται στο σχήμα 3.1.

Σημείωση: Η παραγωγή του διαγράμματος περιπτώσεων χρήσης έγινε με χρήση του λογισμικού PlantUML.



Σχήμα 3.1: Διάγραμμα Περιπτώσεων Χρήσης iBoot

Στους πίνακες που ακολουθούν παρουσιάζονται αναλυτικά οι περιπτώσεις χρήσης της πλατφόρμας.

3.2.1 Βασικές Λειτουργίες Χρηστών

Πίνακας 3.1: Περίπτωση Χρήσης: Εγγραφή χρήστη στο σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο εγγράφονται οι χρήστες στην πλατφόρμα
Χειριστές:	Ανώνυμος Χρήστης
Προϋποθέσεις:	Οι εγγραφές χρηστών στην πλατφόρμα πρέπει να είναι ενεργοποιημένες
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης συμπληρώνει τα στοιχεία του στη σελίδα εγγραφής2. Επιβεβαιώνεται η επιτυχής εισαγωγή των στοιχείων και γίνεται επαλήθευσή τους από client-side και server-side validators για την εξακρίβωση τύπου και πλήθους δεδομένων3. Η εγγραφή του χρήστη στο σύστημα ολοκληρώνεται, ο χρήστης μεταφέρεται στη σελίδα σύνδεσης και παράλληλα λαμβάνει email με σύνδεσμο επιβεβαίωσης της διεύθυνσης email του4. Η χρονοσφραγίδα επιτυχούς εγγραφής χρήστη στο σύστημα καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Ο χρήστης εισάγει εσφαλμένα στοιχεία, η πλατφόρμα απορρίπτει τη σύνδεση, εμφανίζει μήνυμα σφάλματος και ζητά από τον χρήστη να δοκιμάσει ξανά να εισαγάγει τα σωστά στοιχεία

Σημείωση: Στον πίνακα 3.1 περιγράφεται η περίπτωση χρήσης εγγραφής νέου χρήστη στην πλατφόρμα. Κατά την πρώτη περιήγηση στην αρχική σελίδα της εφαρμογής, μετά την εγκατάσταση της και όταν δεν υπάρχει εγγεγραμμένος Γενικός Διαχειριστής Πλατφόρμας, θα ξεκινήσει η διαδικασία εγγραφής Γενικού Διαχειριστή Πλατφόρμας, ανεξάρτητα από το αν οι εγγραφές χρηστών είναι ενεργοποιημένες. Οι επόμενοι χρήστες που θα εγγραφούν δε παίρνουν αυτόματα αυτό το ρόλο.

Πίνακας 3.2: Περίπτωση Χρήσης: Σύνδεση χρήστη στο σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο συνδέεται ο χρήστης στην πλατφόρμα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Ο χρήστης πρέπει να είναι εγγεγραμμένος στην πλατφόρμα

Πίνακας 3.2: Περίπτωση Χρήσης: Σύνδεση χρήστη στο σύστημα (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης δεν έχει cookie αυθεντικοποίησης χρήστη της εφαρμογής στον περιηγητή ιστού του 2. Ο χρήστης εισάγει το όνομα χρήστη και τον κωδικό πρόσβασης του λογαριασμού του 3. Επιβεβαιώνεται η επιτυχής εισαγωγή των στοιχείων σύνδεσης 4. Ο χρήστης συνδέεται στο σύστημα και αποθηκεύεται cookie αυθεντικοποίησης χρήστη της εφαρμογής στον περιηγητή ιστού του 5. Η χρονοσφραγίδα επιτυχούς σύνδεσης του χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Ο χρήστης εισάγει εσφαλμένα στοιχεία, η πλατφόρμα απορρίπτει τη σύνδεση, εμφανίζει μήνυμα σφάλματος, καταγράφει την αποτυχία σύνδεσης και ζητά από τον χρήστη να δοκιμάσει ξανά να εισαγάγει τα σωστά στοιχεία

Πίνακας 3.3: Περίπτωση Χρήσης: Επανέκδοση κωδικού χρήστη

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης δημιουργεί νέο κωδικό για τον λογαριασμό του σε περίπτωση που ξεχάσει τον παλιό
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης έχει ξεχάσει τον κωδικό του λογαριασμού του 2. Ο χρήστης περιηγείται στο σημείο επανέκδοσης κωδικού πρόσβασης 3. Εισάγει το όνομα χρήστη ή το email του στη φόρμα 4. Λαμβάνει στο email του έναν ειδικό σύνδεσμο επανέκδοσης του κωδικού του 5. Ακολουθεί τον σύνδεσμο και εισάγει τον νέο κωδικό του λογαριασμού του 6. Συνδέεται στην πλατφόρμα με χρήση του νέου κωδικού του 7. Η χρονοσφραγίδα αλλαγής του κωδικού πρόσβασης του χρήστη καταγράφεται στο αρχείο συμβάντων

Πίνακας 3.3: Περίπτωση Χρήσης: Επανεκδοση κωδικού χρήστη (συνέχεια)

Εναλλακτικό σενάριο:	Αν ο χρήστης εισάγει λάθος στοιχεία στο βήμα 3, δε θα λάβει το email επαναφοράς του κωδικού του
----------------------	---

Πίνακας 3.4: Περίπτωση Χρήσης: Υπενθύμιση Ονόματος Χρήστη

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης ζητά email υπενθύμισης για το όνομα χρήστη του λογαριασμού του
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης έχει ξεχάσει το όνομα χρήστη του λογαριασμού του 2. Ο χρήστης περιηγείται στο σημείο υπενθύμισης ονόματος χρήστη 3. Εισάγει το email του λογαριασμού του 4. Λαμβάνει στο email του υπενθύμιση του ονόματος χρήστη του 5. Η χρονοσφραγίδα αποστολής του μηνύματος υπενθύμισης του ονόματος χρήστη του χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης εισάγει λάθος email στο βήμα 3, δε θα λάβει το email υπενθύμισης του ονόματος χρήστη του

Πίνακας 3.5: Περίπτωση Χρήσης: Αποσύνδεση από το σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης αποσυνδέεται από την πλατφόρμα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης επιλέγει το κουμπί της Αποσύνδεσης από το κεντρικό μενού 2. Ο χρήστης αποσυνδέεται από το σύστημα και διαγράφεται το cookie αυθεντικοποίησης χρήστη της εφαρμογής από τον περιηγητή ιστού του
Εναλλακτικό σενάριο:	

Πίνακας 3.6: Περίπτωση Χρήσης: Αλλαγή στοιχείων χρήστη

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης αλλάζει τα στοιχεία του προφίλ του
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στο προφίλ του από το κεντρικό μενού2. Επιλέγει την επεξεργασία προφίλ3. Αλλάζει τα στοιχεία που επιθυμεί (όνομα χρήστη, κωδικό πρόσβασης, email, τηλέφωνο)4. Επιλέγει την αποθήκευση αλλαγών5. Η χρονοσφραγίδα επιτυχούς αλλαγής των στοιχείων του χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	<ul style="list-style-type: none">• Αν τα νέα στοιχεία είναι εσφαλμένα, η πλατφόρμα απορρίπτει την αποθήκευσή τους, εμφανίζει μήνυμα σφάλματος και ζητά από τον χρήστη να δοκιμάσει ξανά να εισαγάγει έγκυρα στοιχεία• Αν ο χρήστης είναι Γενικός Διαχειριστής Πλατφόρμας, μπορεί να περιηγηθεί στη σελίδα με τη λίστα χρηστών και από εκεί να επεξεργαστεί τα στοιχεία άλλων χρηστών και επιπρόσθετα να αλλάξει το ρόλο τους

Πίνακας 3.7: Περίπτωση Χρήσης: Διαγραφή χρήστη από το σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης αποσυνδέεται από την πλατφόρμα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία

Πίνακας 3.7: Περίπτωση Χρήσης: Διαγραφή χρήστη από το σύστημα (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στο προφίλ του από το κεντρικό μενού 2. Επιλέγει τη διαγραφή του χρήστη του από την πλατφόρμα 3. Εμφανίζεται ένα αναδυόμενο παράθυρο στο οποίο ο χρήστης επιβεβαιώνει την οριστική διαγραφή του από την πλατφόρμα 4. Ο χρήστης αποσυνδέεται και τα στοιχεία του διαγράφονται από την πλατφόρμα 5. Η χρονοσφραγίδα διαγραφής του χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	<ul style="list-style-type: none"> • Αν ο χρήστης δεν επιβεβαιώσει την οριστική διαγραφή του στο βήμα 3, τα στοιχεία του δε διαγράφονται και παραμένει συνδεδεμένος • Αν ο χρήστης είναι Γενικός Διαχειριστής Πλατφόρμας, μπορεί να περιηγηθεί στη σελίδα με τη λίστα χρηστών και από εκεί να διαγράψει άλλους χρήστες

3.2.2 Διαχείριση Εργαστηρίων

Πίνακας 3.8: Περίπτωση Χρήσης: Εισαγωγή εργαστηρίου στο σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο Γενικός Διαχειριστής προσθέτει ένα εργαστήριο το σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των εργαστηρίων 2. Επιλέγει την προσθήκη νέου εργαστηρίου 3. Εισαγάγει τα στοιχεία του νέου εργαστηρίου (όνομα και προαιρετικά διεύθυνση ή/και τηλέφωνο) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς εισαγωγής εργαστηρίου από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, το νέο εργαστήριο δε θα προστεθεί

Πίνακας 3.9: Περίπτωση Χρήσης: Διαγραφή εργαστηρίου από το σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο Γενικός Διαχειριστής αφαιρεί ένα εργαστήριο από το σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Το εργαστήριο πρέπει να έχει προστεθεί στο σύστημα
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των εργαστηρίων 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του εργαστηρίου που θέλει να διαγράψει 3. Επιβεβαιώνει την οριστική διαγραφή του εργαστηρίου στο αναδυόμενο παράθυρο που εμφανίζεται 4. Τα στοιχεία του εργαστηρίου διαγράφονται από την πλατφόρμα και οι υπολογιστές που του είχαν ανατεθεί μεταφέρονται στη λίστα αναμονής ανάθεσης σε εργαστήριο 5. Η χρονοσφραγίδα επιτυχούς διαγραφής εργαστηρίου από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του εργαστηρίου στο βήμα 3, τα στοιχεία του παραμένουν στην πλατφόρμα

Πίνακας 3.10: Περίπτωση Χρήσης: Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων

Περιγραφή:	Ο τρόπος με τον οποίο ο Γενικός Διαχειριστής Πλατφόρμας επεξεργάζεται τις αναθέσεις Διαχειριστών Εργαστηρίων
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Ο χρήστης προς ανάθεση πρέπει να έχει εγγραφεί στο σύστημα

Πίνακας 3.10: Περίπτωση Χρήσης: Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των εργαστηρίων 2. Επιλέγει την επεξεργασία στοιχείων 3. Επιλέγει τους Διαχειριστές Εργαστηρίου για κάθε εργαστήριο που θέλει να επεξεργαστεί 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας του κάθε εργαστηρίου από τον Γενικό Διαχειριστή Πλατφόρμας καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα χρηστών 2. Επιλέγει την επεξεργασία στοιχείων 3. Επιλέγει τα εργαστήρια στα οποία θέλει να είναι Διαχειριστές Εργαστηρίου σε όσους χρήστες θέλει να επεξεργαστεί 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας του κάθε χρήστη από τον Γενικό Διαχειριστή Πλατφόρμας καταγράφεται στο αρχείο συμβάντων

3.2.3 Διαχείριση Ομάδων Υπολογιστών

Πίνακας 3.11: Περίπτωση Χρήσης: Δημιουργία ομάδας υπολογιστών

Περιγραφή:	Ο τρόπος με τον οποίο δημιουργείται μια ομάδα υπολογιστών
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία

Πίνακας 3.11: Περίπτωση Χρήσης: Δημιουργία ομάδας υπολογιστών (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των ομάδων υπολογιστών 2. Επιλέγει την προσθήκη νέας εγγραφής 3. Εισαγάγει τα στοιχεία της νέας ομάδας υπολογιστών (όνομα, διεύθυνση και prefix εικόνων λειτουργικών συστημάτων) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς δημιουργίας ομάδας υπολογιστών από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, η νέα ομάδας υπολογιστών δε θα προστεθεί

Πίνακας 3.12: Περίπτωση Χρήσης: Διαγραφή ομάδας υπολογιστών

Περιγραφή:	Ο τρόπος με τον οποίο διαγράφεται μια ομάδα υπολογιστών
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Η ομάδα υπολογιστών πρέπει να έχει προστεθεί στο σύστημα
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των ομάδων υπολογιστών 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία της ομάδας υπολογιστών που θέλει να διαγράψει 3. Επιβεβαιώνει τη διαγραφή στο αναδυόμενο παράθυρο που θα εμφανιστεί 4. Τα στοιχεία της ομάδας υπολογιστών διαγράφονται από την πλατφόρμα 5. Η χρονοσφραγίδα επιτυχούς διαγραφής ομάδας υπολογιστών από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή της ομάδας υπολογιστών στο βήμα 3, τα στοιχεία της παραμένουν στην πλατφόρμα

3.2.4 Διαχείριση Υπολογιστών

Πίνακας 3.13: Περίπτωση Χρήσης: Εισαγωγή υπολογιστή στο σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο διαχειριστής προσθέτει έναν υπολογιστή στο σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Πρέπει να έχει ρυθμιστεί ο DHCP server του εργαστηρίου στο οποίο βρίσκεται ο υπολογιστής, ώστε να παρέχει τη διεύθυνση του συστήματος για δικτυακή εκκίνηση των υπολογιστών
Βασική ροή:	<ol style="list-style-type: none">1. Ο υπολογιστής εκκινείται και τα στοιχεία του υπολογιστή (το UUID της μητρικής πλακέτας και η φυσική διεύθυνση MAC της διεπαφής δικτύου) καταγράφονται στο σύστημα και εμφανίζονται στην οθόνη του υπολογιστή2. Ο χρήστης βλέπει την νέα εγγραφή στο σύστημα, στην οθόνη με τους υπολογιστές που δεν έχουν ανατεθεί σε εργαστήριο. Μπορεί να επαληθεύσει την ταυτότητα του υπολογιστή συγκρίνοντας τα στοιχεία που βλέπει στο σύστημα με αυτά που εμφανίζονται στην οθόνη του υπολογιστή3. Ο χρήστης συμπληρώνει τα απαραίτητα στοιχεία για τον υπολογιστή (όνομα και προαιρετικά σημειώσεις για τον υπολογιστή, το εργαστήριο στο οποίο βρίσκεται, τις ομάδες υπολογιστών στις οποίες ανήκει)4. Ο υπολογιστής είναι έτοιμος να εκκινηθεί μέσω του συστήματος και μπορούν να διαχειριστούν τα στοιχεία του, τόσο οι γενικοί διαχειριστές του συστήματος, όσο και οι διαχειριστές του εργαστηρίου στο οποίο ανήκει
Εναλλακτικό σενάριο:	Ο χρήστης γνωρίζει εκ των προτέρων τα στοιχεία του υπολογιστή και τα προσθέτει στο σύστημα

Πίνακας 3.14: Περίπτωση Χρήσης: Ανάθεση υπολογιστή σε εργαστήριο

Περιγραφή:	Ο τρόπος με τον οποίο ένας υπολογιστής ανατίθεται σε ένα εργαστήριο
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου

Πίνακας 3.14: Περίπτωση Χρήσης: Ανάθεση υπολογιστή σε εργαστήριο (συνέχεια)

Προϋποθέσεις:	<ul style="list-style-type: none"> • Ο υπολογιστής πρέπει να έχει προστεθεί στο σύστημα και να μην έχει ανατεθεί ήδη σε κάποιο άλλο εργαστήριο • Αν ο χρήστης είναι Διαχειριστής Εργαστηρίου, στο βήμα 2 μπορεί να αναθέσει υπολογιστή μόνο σε εργαστήριο το οποίο διαχειρίζεται ο ίδιος. Αν είναι Γενικός Διαχειριστής της πλατφόρμας μπορεί να αναθέσει τον υπολογιστή σε οποιοδήποτε εργαστήριο.
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει τον υπολογιστή στη λίστα αναμονής ανάθεσης σε εργαστήριο 2. Επιλέγει το εργαστήριο στο οποίο θα ανατεθεί ο υπολογιστής 3. Επιλέγει την αποθήκευση αλλαγών 4. Η χρονοσφραγίδα επιτυχούς ανάθεσης του υπολογιστή στο εργαστήριο από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 3, ο υπολογιστής δε θα ανατεθεί στο εργαστήριο

Πίνακας 3.15: Περίπτωση Χρήσης: Αφαίρεση υπολογιστή από εργαστήριο

Περιγραφή:	Ο τρόπος με τον οποίο ένας υπολογιστής διαγράφεται από ένα εργαστήριο
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	<ul style="list-style-type: none"> • Ο υπολογιστής πρέπει να έχει προστεθεί στο εργαστήριο • Αν ο χρήστης είναι Διαχειριστής Εργαστηρίου, στο βήμα 1 μπορεί να αφαιρέσει υπολογιστές μόνο από εργαστήριο το οποίο διαχειρίζεται ο ίδιος. Αν είναι Γενικός Διαχειριστής της πλατφόρμας δεν υπάρχει τέτοιος περιορισμός.

Πίνακας 3.15: Περίπτωση Χρήσης: Αφαίρεση υπολογιστή από εργαστήριο (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει τον υπολογιστή στη λίστα υπολογιστών του εργαστηρίου στο οποίο έχει ανατεθεί 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του υπολογιστή που θέλει να αφαιρέσει 3. Επιβεβαιώνει την αφαίρεση του υπολογιστή από το εργαστήριο στο αναδυόμενο παράθυρο που εμφανίζεται 4. Ο υπολογιστής αφαιρείται από το εργαστήριο και μεταφέρεται στη λίστα αναμονής ανάθεσης σε εργαστήριο 5. Η χρονοσφραγίδα επιτυχούς αφαίρεσης του υπολογιστή από το εργαστήριο από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει την αφαίρεση του υπολογιστή στο βήμα 3, ο υπολογιστής παραμένει στο εργαστήριο

Πίνακας 3.16: Περίπτωση Χρήσης: Επεξεργασία στοιχείων υπολογιστή

Περιγραφή:	Ο τρόπος με τον οποίο γίνεται επεξεργασία των στοιχείων ενός υπολογιστή
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	<ul style="list-style-type: none"> • Ο υπολογιστής πρέπει να έχει προστεθεί στο σύστημα • Αν ο χρήστης είναι Διαχειριστής Εργαστηρίου, στο βήμα 1 μπορεί να επεξεργαστεί τα στοιχεία μόνο υπολογιστών που έχουν ανατεθεί σε εργαστήριο το οποίο διαχειρίζεται ο ίδιος. Αν είναι Γενικός Διαχειριστής της πλατφόρμας δεν υπάρχει τέτοιος περιορισμός.

Πίνακας 3.16: Περίπτωση Χρήσης: Επεξεργασία στοιχείων υπολογιστή (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει τον υπολογιστή στη λίστα υπολογιστών του εργαστηρίου στο οποίο έχει ανατεθεί, ή στη λίστα αναμονής ανάθεσης σε εργαστήριο 2. Επιλέγει την επεξεργασία στοιχείων 3. Πραγματοποιεί αλλαγές στα στοιχεία του υπολογιστή (όνομα, MAC, UUID, σημειώσεις, εργαστήριο, ομάδες υπολογιστών στις οποίες ανήκει) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας των στοιχείων του υπολογιστή από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, τα στοιχεία του υπολογιστή παραμένουν ως έχουν

Πίνακας 3.17: Περίπτωση Χρήσης: Διαγραφή υπολογιστή από το σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ένας υπολογιστής διαγράφεται από το σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Ο υπολογιστής πρέπει να έχει προστεθεί στο σύστημα και να μην έχει ανατεθεί ήδη σε κάποιο άλλο εργαστήριο
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει τον υπολογιστή στη λίστα αναμονής ανάθεσης σε εργαστήριο 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του υπολογιστή που θέλει να αφαιρέσει 3. Επιβεβαιώνει τη διαγραφή του υπολογιστή στο αναδυόμενο παράθυρο που εμφανίζεται 4. Ο υπολογιστής διαγράφεται από την πλατφόρμα 5. Η χρονοσφραγίδα επιτυχούς διαγραφής του υπολογιστή από το σύστημα από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του υπολογιστή στο βήμα 3, τα στοιχεία του υπολογιστή παραμένουν στην πλατφόρμα

3.2.5 Διαχείριση Ονοματισμένων Block Εντολών Τύπου iPXE

Πίνακας 3.18: Περίπτωση Χρήσης: Εισαγωγή ονοματισμένου block εντολών τύπου iPXE στο σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο ο χρήστης προσθέτει ένα ονοματισμένο block εντολών τύπου iPXE στο σύστημα στο σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα τα ονοματισμένα block εντολών τύπου iPXE στο σύστημα2. Επιλέγει την προσθήκη νέας εγγραφής3. Εισαγάγει τα στοιχεία του νέου ονοματισμένου block εντολών τύπου iPXE (όνομα, περιγραφή και block εντολών τύπου iPXE)4. Επιλέγει την αποθήκευση αλλαγών5. Η χρονοσφραγίδα επιτυχούς εισαγωγής ονοματισμένου block εντολών τύπου iPXE στο σύστημα από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, το νέο ονοματισμένο block εντολών τύπου iPXE δε θα προστεθεί

Πίνακας 3.19: Περίπτωση Χρήσης: Επεξεργασία στοιχείων ονοματισμένου block εντολών τύπου iPXE

Περιγραφή:	Ο τρόπος με τον οποίο γίνεται επεξεργασία των στοιχείων ενός ονοματισμένου block εντολών τύπου iPXE
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το ονοματισμένο block εντολών τύπου iPXE πρέπει να έχει προστεθεί στο σύστημα

Πίνακας 3.19: Περίπτωση Χρήσης: Επεξεργασία στοιχείων ονοματισμένου block εντολών τύπου iPXE (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει την εγγραφή που θέλει να επεξεργαστεί στη λίστα των ονοματισμένων block εντολών τύπου iPXE 2. Επιλέγει την επεξεργασία στοιχείων 3. Πραγματοποιεί αλλαγές στα στοιχεία του ονοματισμένου block εντολών τύπου iPXE (όνομα, περιγραφή και block εντολών τύπου iPXE) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας των στοιχείων του ονοματισμένου block εντολών τύπου iPXE από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, τα στοιχεία του ονοματισμένου block εντολών τύπου iPXE παραμένουν ως έχουν

Πίνακας 3.20: Περίπτωση Χρήσης: Διαγραφή ονοματισμένου block εντολών τύπου iPXE από το σύστημα

Περιγραφή:	Ο τρόπος με τον οποίο μια εικόνα λειτουργικού συστήματος διαγράφεται από το σύστημα
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το ονοματισμένο block εντολών τύπου iPXE πρέπει να έχει προστεθεί στο σύστημα
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των εικόνων λειτουργικών συστημάτων 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του ονοματισμένου block εντολών τύπου iPXE που θέλει να διαγράψει 3. Επιβεβαιώνει τη διαγραφή στο αναδυόμενο παράθυρο που εμφανίζεται 4. Τα στοιχεία του ονοματισμένου block εντολών τύπου iPXE διαγράφονται από την πλατφόρμα 5. Η χρονοσφραγίδα επιτυχούς διαγραφής ονοματισμένου block εντολών τύπου iPXE από το σύστημα από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του ονοματισμένου block εντολών τύπου iPXE στο βήμα 3, τα στοιχεία του παραμένουν στην πλατφόρμα

3.2.6 Διαχείριση Μενού Εκκίνησης

Πίνακας 3.21: Περίπτωση Χρήσης: Δημιουργία μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο δημιουργείται ένα μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των μενού εκκίνησης2. Επιλέγει την προσθήκη νέας εγγραφής3. Εισαγάγει τα στοιχεία του νέου μενού εκκίνησης (όνομα και προαιρετικό block εντολών τύπου iPXE) και επιλέγει τα επιθυμητά από τα υπάρχοντα ονοματισμένα block εντολών τύπου iPXE, προαιρετικά αναθέτοντας στο κάθε ένα από αυτά κάποιο κουμπί συντόμευσης4. Επιλέγει την αποθήκευση αλλαγών5. Η χρονοσφραγίδα επιτυχούς δημιουργίας μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, το νέο μενού εκκίνησης δε θα προστεθεί

Πίνακας 3.22: Περίπτωση Χρήσης: Επεξεργασία στοιχείων μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο γίνεται επεξεργασία των στοιχείων ενός μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το μενού εκκίνησης πρέπει να έχει προστεθεί στο σύστημα

Πίνακας 3.22: Περίπτωση Χρήσης: Επεξεργασία στοιχείων μενού εκκίνησης (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει την εγγραφή που θέλει να επεξεργαστεί στη λίστα των μενού εκκίνησης 2. Επιλέγει την επεξεργασία στοιχείων 3. Πραγματοποιεί αλλαγές στα στοιχεία του μενού εκκίνησης (όνομα και προαιρετικό block εντολών τύπου iPXE) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας των στοιχείων του μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, τα στοιχεία του μενού εκκίνησης παραμένουν ως έχουν

Πίνακας 3.23: Περίπτωση Χρήσης: Διαγραφή μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο διαγράφεται ένα μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το μενού εκκίνησης πρέπει να έχει προστεθεί στο σύστημα
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των μενού εκκίνησης 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του μενού εκκίνησης που θέλει να διαγράψει 3. Επιβεβαιώνει τη διαγραφή στο αναδυόμενο παράθυρο που εμφανίζεται 4. Τα στοιχεία του μενού εκκίνησης διαγράφονται από την πλατφόρμα 5. Η χρονοσφραγίδα επιτυχούς διαγραφής μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του μενού εκκίνησης στο βήμα 3, τα στοιχεία του παραμένουν στην πλατφόρμα

3.2.7 Διαχείριση Χρονοδιαγραμμάτων Μενού Εκκίνησης

Πίνακας 3.24: Περίπτωση Χρήσης: Δημιουργία χρονοδιαγράμματος μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο δημιουργείται ένα χρονοδιάγραμμα μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των χρονοδιαγραμμάτων μενού εκκίνησης2. Επιλέγει την προσθήκη νέας εγγραφής3. Εισαγάγει τα στοιχεία του νέας χρονοδιαγράμματος μενού εκκίνησης (μενού εκκίνησης, ομάδα υπολογιστών, ημερομηνία/ώρες ενεργοποίησης, ενεργό/ανενεργό)4. Επιλέγει την αποθήκευση αλλαγών5. Η χρονοσφραγίδα επιτυχούς δημιουργίας χρονοδιαγράμματος μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, το νέο χρονοδιάγραμμα μενού εκκίνησης δε θα προστεθεί

Πίνακας 3.25: Περίπτωση Χρήσης: Επεξεργασία στοιχείων χρονοδιαγράμματος μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο γίνεται επεξεργασία των στοιχείων ενός χρονοδιαγράμματος μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το χρονοδιάγραμμα μενού εκκίνησης πρέπει να έχει προστεθεί στο σύστημα

Πίνακας 3.25: Περίπτωση Χρήσης: Επεξεργασία στοιχείων χρονοδιαγράμματος μενού εκκίνησης (συνέχεια)

Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης εντοπίζει την εγγραφή που θέλει να επεξεργαστεί στη λίστα των χρονοδιαγραμμάτων μενού εκκίνησης 2. Επιλέγει την επεξεργασία στοιχείων 3. Πραγματοποιεί αλλαγές στα στοιχεία του χρονοδιαγράμματος μενού εκκίνησης (μενού εκκίνησης, ομάδα υπολογιστών, ημερομηνία/ώρες ενεργοποίησης, ενεργό/ανενεργό) 4. Επιλέγει την αποθήκευση αλλαγών 5. Η χρονοσφραγίδα επιτυχούς επεξεργασίας των στοιχείων του χρονοδιαγράμματος μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν αποθηκεύσει τις αλλαγές στο βήμα 4, τα στοιχεία του χρονοδιαγράμματος μενού εκκίνησης παραμένουν ως έχουν

Πίνακας 3.26: Περίπτωση Χρήσης: Διαγραφή χρονοδιαγράμματος μενού εκκίνησης

Περιγραφή:	Ο τρόπος με τον οποίο διαγράφεται ένα χρονοδιάγραμμα μενού εκκίνησης
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου
Προϋποθέσεις:	Το χρονοδιάγραμμα μενού εκκίνησης πρέπει να έχει προστεθεί στο σύστημα
Βασική ροή:	<ol style="list-style-type: none"> 1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των χρονοδιαγραμμάτων μενού εκκίνησης 2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του χρονοδιαγράμματος μενού εκκίνησης που θέλει να διαγράψει 3. Επιβεβαιώνει τη διαγραφή στο αναδυόμενο παράθυρο που εμφανίζεται 4. Τα στοιχεία του χρονοδιαγράμματος μενού εκκίνησης διαγράφονται από την πλατφόρμα 5. Η χρονοσφραγίδα επιτυχούς διαγραφής χρονοδιαγράμματος μενού εκκίνησης από τον χρήστη καταγράφεται στο αρχείο συμβάντων
Εναλλακτικό σενάριο:	Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του χρονοδιαγράμματος μενού εκκίνησης στο βήμα 3, τα στοιχεία του παραμένουν στην πλατφόρμα

3.2.8 Διαχείριση Αρχείων Καταγραφής Συμβάντων

Πίνακας 3.27: Περίπτωση Χρήσης: Προβολή αρχείων συμβάντων

Περιγραφή:	Ο τρόπος με τον οποίο γίνεται η προβολή των αρχείων συμβάντων
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Καμία
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των αρχείων συμβάντων2. Επιλέγει το αρχείο συμβάντων που θέλει να προβάλει3. Αν το επιθυμεί, εφαρμόζει φίλτρα (Επίπεδο συμβάντος, ημερομηνία/ώρα, λέξη/φράση στο κείμενο του συμβάντος) για να βρει ευκολότερα την πληροφορία που αναζητά
Εναλλακτικό σενάριο:	

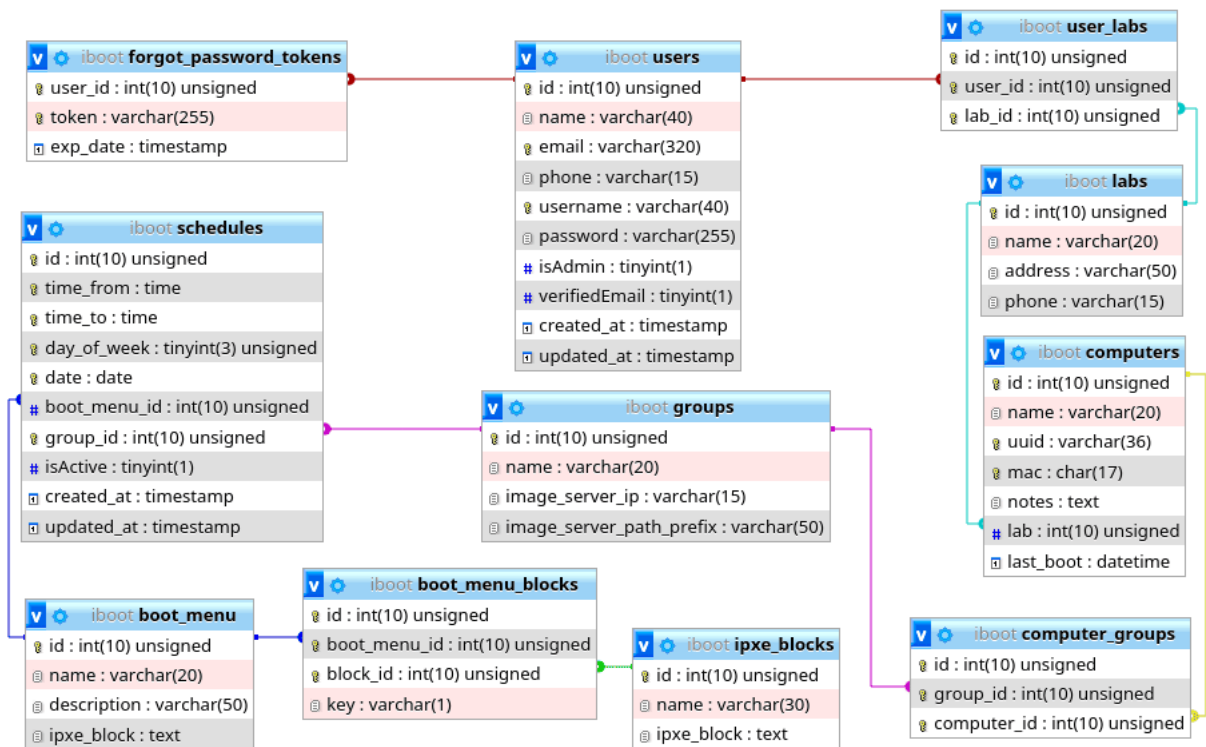
Πίνακας 3.28: Περίπτωση Χρήσης: Διαγραφή αρχείων συμβάντων

Περιγραφή:	Ο τρόπος με τον οποίο διαγράφονται τα αρχεία συμβάντων
Χειριστές:	Γενικός Διαχειριστής Πλατφόρμας
Προϋποθέσεις:	Το αρχείο συμβάντων πρέπει να υπάρχει
Βασική ροή:	<ol style="list-style-type: none">1. Ο χρήστης περιηγείται στη σελίδα με τη λίστα των αρχείων συμβάντων2. Επιλέγει το σύμβολο διαγραφής, δίπλα στα στοιχεία του χρονοδιαγράμματος μενού εκκίνησης που θέλει να διαγράψει3. Επιβεβαιώνει τη διαγραφή στο αναδυόμενο παράθυρο που εμφανίζεται4. Το αρχείο συμβάντων διαγράφεται από την πλατφόρμα
Εναλλακτικό σενάριο:	<ul style="list-style-type: none">• Στο βήμα 2, ο χρήστης μπορεί να επιλέξει το σύμβολο διαγραφής όλων των αρχείων συμβάντων• Αν ο χρήστης δεν επιβεβαιώσει τη διαγραφή του χρονοδιαγράμματος μενού εκκίνησης στο βήμα 3, τα στοιχεία του παραμένουν στην πλατφόρμα

3.3 Σχεδιασμός Βάσης Δεδομένων

Για τη δόμηση του σχεσιακού μοντέλου, έχουν οριστεί οι σχέσεις (relations) και οι πίνακες (tables) της βάσης δεδομένων. Το σύνολο των πληροφοριών που αποθηκεύονται στη βάση δεδομένων αποτελείται από μία συλλογή πινάκων. Οι γραμμές των πινάκων καλούνται πλειάδες (tuples) ή εγγραφές (records) των πινάκων (σχέσεις) και αντιστοιχούν σε οντότητες. Η εικόνα αναπαριστά τη δομή της βάσης δεδομένων με τους πίνακες, τα πεδία, τους τύπους δεδομένων και τις συσχετίσεις που υπάρχουν μεταξύ τους.

Οι σχέσεις και οι πίνακες της βάσης δεδομένων έχουν οριστεί για την κατασκευή του σχεσιακού μοντέλου. Μια ομάδα πινάκων αποτελεί το σύνολο των δεδομένων που διατηρείται στη βάση δεδομένων. Οι σειρές πινάκων, που αντιστοιχούν σε οντότητες, αναφέρονται ως πλειάδες (tuples) ή εγγραφές (records) των πινάκων. Το σχεσιακό διάγραμμα της βάσης δεδομένων του συστήματος (Entity-relationship model), δηλαδή η δομή της βάσης δεδομένων, συμπεριλαμβανομένων των πινάκων, των πεδίων, των τύπων δεδομένων και των σχέσεων, απεικονίζεται στο σχήμα 3.2.



Σχήμα 3.2: Διάγραμμα Βάσης Δεδομένων iBoot

Στους πίνακες που ακολουθούν, παρουσιάζονται αναλυτικά οι πίνακες της βάσης δεδομένων του συστήματος, καθώς και τα πεδία τους. Επίσης, ως σχόλιο σε κάθε πεδίο, περιγράφεται η χρησιμότητά του.

Πίνακας 3.29: Δομή πίνακα boot_menu

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(20)	Όχι			Όνομα μενού εκκίνησης
description	varchar(50)	Όχι			Περιγραφή μενού εκκίνησης
ipxe_block	text	Ναι	NULL		Προαιρετικό επιπρόσθετο block εντολών τύπου ipxe

Πίνακας 3.30: Δομή πίνακα boot_menu_blocks

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
boot_menu_id	int(10)	Όχι		boot_menu (id)	Αναγνωριστικό μενού εκκίνησης
block_id	int(10)	Όχι		ipxe_blocks (id)	Αναγνωριστικό εικόνας λειτουργικού συστήματος
key	varchar(1)	Ναι	NULL		Προαιρετικό κουμπί συντόμευσης για το block στο μενού εκκίνησης

Πίνακας 3.31: Δομή πίνακα computers

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(20)	Ναι	KENO		Όνομα υπολογιστή
uuid	varchar(36)	Όχι			Μοναδικό αναγνωριστικό του υπολογιστή, παρεχόμενο από το bios

Πίνακας 3.31: Δομή πίνακα computers (συνέχεια)

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
mac	char(17)	Όχι			Η φυσική διεύθυνση διεπαφής δικτύου του υπολογιστή
notes	text	Ναι	NULL		Σημειώσεις για τον υπολογιστή
lab	int(10)	Ναι	KENO	labs (id)	Αναγνωριστικό εργαστηρίου στο οποίο βρίσκεται ο υπολογιστής
last_boot	datetime	Ναι	KENO		Ημερομηνία και ώρα τελευταίας εκκίνησης του υπολογιστή μέσω του συστήματος

Πίνακας 3.32: Δομή πίνακα computer_groups

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
id	int(10)	Όχι			Αναγνωριστικό οντότητας
group_id	int(10)	Όχι		groups (id)	Αναγνωριστικό ομάδας υπολογιστών
computer_id	int(10)	Όχι		computers (id)	Αναγνωριστικό υπολογιστή

Πίνακας 3.33: Δομή πίνακα forgot_password_tokens

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
user_id	int(10)	Όχι		users (id)	Αναγνωριστικό χρήστη
token	varchar(255)	Ναι	KENO		Αλφαριθμητικό αλλαγής κωδικού πρόσβασης
exp_date	timestamp	Ναι	KENO		Ημερομηνία λήξης αλφαριθμητικού

Πίνακας 3.34: Δομή πίνακα groups

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(20)	Όχι			Όνομα ομάδας
image_server_ip	varchar(15)	Όχι			Διεύθυνση διακομιστή εικόνων
image_server_path_prefix	varchar(50)	Όχι			Πρόθεμα διαδρομής εικόνων στον διακομιστή

Πίνακας 3.35: Δομή πίνακα ipxe_blocks

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(30)	Όχι			Το όνομα της εικόνας λειτουργικού συστήματος
ipxe_block	text	Όχι			Το block εντολών τύπου ipxe

Πίνακας 3.36: Δομή πίνακα labs

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(20)	Όχι			Όνομα εργαστηρίου
address	varchar(50)	Ναι	KENO		Διεύθυνση εργαστηρίου
phone	varchar(15)	Ναι	KENO		Τηλέφωνο εργαστηρίου

Πίνακας 3.37: Δομή πίνακα schedules

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
id	int(10)	Όχι			Αναγνωριστικό οντότητας
time_from	time	Ναι	KENO		Ώρα έναρξης
time_to	time	Ναι	KENO		Ώρα λήξης
day_of_week	tinyint(3)	Ναι	KENO		Μέρα της εβδομάδας
date	date	Ναι	KENO		Ημερομηνία
boot_menu_id	int(10)	Όχι		boot_menu (id)	Αναγνωριστικό μενού εκκίνησης
group_id	int(10)	Όχι		groups (id)	Αναγνωριστικό ομάδας υπολογιστών
isActive	tinyint(1)	Όχι	1		Παίρνει τιμή '1' αν το χρονοδιάγραμμα είναι ενεργό, αλλιώς '0'
created_at	timestamp	Ναι	KENO		Ημερομηνία δημιουργίας του χρονοδιαγράμματος
updated_at	timestamp	Ναι	KENO		Ημερομηνία ενημέρωσης του χρονοδιαγράμματος

Πίνακας 3.38: Δομή πίνακα users

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
id	int(10)	Όχι			Αναγνωριστικό οντότητας
name	varchar(40)	Όχι			Όνοματεπώνυμο
email	varchar(320)	Όχι			Διεύθυνση email
phone	varchar(15)	Ναι	KENO		Αριθμός τηλεφώνου
username	varchar(40)	Όχι			Όνομα χρήστη

Πίνακας 3.38: Δομή πίνακα users (συνέχεια)

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
password	varchar(255)	Όχι			Κωδικός πρόσβασης
isAdmin	tinyint(1)	Όχι	0		Παίρνει τιμή '1' αν ο χρήστης είναι γενικός διαχειριστής, αλλιώς '0'
verifiedEmail	tinyint(1)	Όχι	0		Παίρνει τιμή '1' αν ο χρήστης έχει επιβεβαιώσει τη διεύθυνση email του, αλλιώς '0'
created_at	timestamp	Ναι	KENO		Ημερομηνία δημιουργίας του χρήστη
updated_at	timestamp	Ναι	KENO		Ημερομηνία ενημέρωσης του χρήστη

Πίνακας 3.39: Δομή πίνακα user_labs

Στήλη	Τύπος	Κενό	Προεπιλογή	Σύνδεσμος	Σχόλιο
<i>id</i>	int(10)	Όχι			Αναγνωριστικό οντότητας
user_id	int(10)	Όχι		users (id)	Αναγνωριστικό χρήστη
lab_id	int(10)	Όχι		labs (id)	Αναγνωριστικό εργαστηρίου

3.4 Σύνοψη Κεφαλαίου 3

Η δομή και η οργάνωση του εφαρμοζόμενου πληροφοριακού συστήματος είναι τα θέματα που καλύπτονται στην ανάλυση του κεφαλαίου 3. Στην αρχή, περιγράφηκαν οι στόχοι, οι λειτουργικές απαιτήσεις και οι περιπτώσεις χρήσης. Επιπλέον, εξετάστηκαν το σχεσιακό διάγραμμα και τα μοντέλα των πινάκων της βάσης δεδομένων και δόθηκε μια λεπτομερής εξήγηση του σχεδιασμού της βάσης δεδομένων.

Στο κεφάλαιο 4, αναλύονται τα βήματα που ο χρήστης πρέπει να ακολουθήσει για να χρησιμοποιήσει την εφαρμογή.

Κεφάλαιο 4

Ανάπτυξη Διαδικτυακής Εφαρμογής iBoot

Σε αυτό το κεφάλαιο, αναλύονται οι λειτουργίες που παρέχει η διαδικτυακή εφαρμογή στους χρήστες της μέσω Διεπαφής Χρήστη (User Interface) και παρέχεται επεξήγηση των αρχείων κώδικα, ο οποίος συντέλεσε στη δημιουργία του πληροφοριακού συστήματος. Ανάλογα με τον ρόλο του χρήστη που συνδέεται στην εφαρμογή, παρέχονται και οι αντίστοιχες λειτουργίες. Η εμφάνιση της εφαρμογής έχει ίδια αισθητικά χαρακτηριστικά για κάθε είδος χρήστη. Το User Interface θεωρείται από τα πιο σημαντικά μέρη μιας διαδικτυακής εφαρμογής καθώς, αν είναι σχεδιασμένο σωστά, ώστε να είναι εύχρηστο και διαισθητικό, κάνει εύκολη τη χρήση των λειτουργιών του συστήματος, ακόμη και σε χρήστες που πιθανώς δεν είναι εξειδικευμένοι στον τομέα της πληροφορικής.

Για την κατασκευή του User Interface, χρησιμοποιήθηκε η αρχή του Reactive Programming, δηλαδή της μεθόδου προγραμματισμού ιστού ώστε το περιεχόμενο της εφαρμογής να ανανεώνεται δυναμικά και ασύγχρονα με τις ενέργειες του χρήστη, σε ένα εν γένει στατικό περιβάλλον. Όσον αφορά το σχεδιασμό του γραφικού περιβάλλοντος, χρησιμοποιήθηκε Responsive Design, ώστε η εφαρμογή να έχει ομοιόμορφη, καλαίσθητη και λειτουργική εμφάνιση, τόσο σε συσκευές με μεγάλη διάμετρο οθόνης, όπως ηλεκτρονικούς υπολογιστές, όσο και σε συσκευές με μικρότερη διάμετρο οθόνης, όπως κινητά τηλέφωνα. Ο συνδυασμός των παραπάνω μεθόδων προγραμματισμού ιστού, προσφέρει ευχάριστη εμπειρία χρήσης της εφαρμογής, σε όλους τους χρήστες, ανεξάρτητα από τον τύπο συσκευής που χρησιμοποιούν.

Η διαδικτυακή εφαρμογή έχει υλοποιηθεί ως ένα γραφικό περιβάλλον ιστού που επικοινωνεί με τη βάση δεδομένων του μέσω ενός RESTful API. Για την ανάπτυξη της εφαρμογής έγινε χρήση του CodeIgniter PHP Framework. Η αναλυτική προδιαγραφή του API που παρέχει η πλατφόρμα, μετά την εγκατάσταση της, δημιουργήθηκε με χρήση του Swagger UI και βρίσκεται στην τοποθεσία `/api` και περιγράφει όλες τις λειτουργίες που είναι δυνατό να πραγματοποιηθούν με τη χρήση του.

4.1 Διεπαφή Χρήστη

4.1.1 Εγγραφή & Σύνδεση στο Σύστημα

Η σελίδα εγγραφής στο σύστημα απεικονίζεται στο σχήμα 4.1 και είναι προσβάσιμη στη διεύθυνση `/register`.

Ο χρήστης θα πρέπει να συμπληρώσει εκεί τα στοιχεία που ζητούνται και στη συνέχεια να πατήσει το κουμπί της εγγραφής για να δημιουργηθεί ο λογαριασμός του. Αν δεν υπάρχει κανένας εγγεγραμμένος διαχειριστής στο σύστημα, η ίδια σελίδα αξιοποιείται για την εγγραφή του. Στη συνέχεια, οι επόμενοι χρήστες που εγγράφονται δεν έχουν ρόλο διαχειριστή. Ο υπάρχων διαχειριστής μπορεί να δώσει ρόλο διαχειριστή σε άλλους χρήστες από τη

Εγγραφή

*Όνομα Χρήστη

*Κωδικός

*Επαλήθευση Κωδικού

*Όνομα

*Email

Τηλέφωνο

Εγγραφή

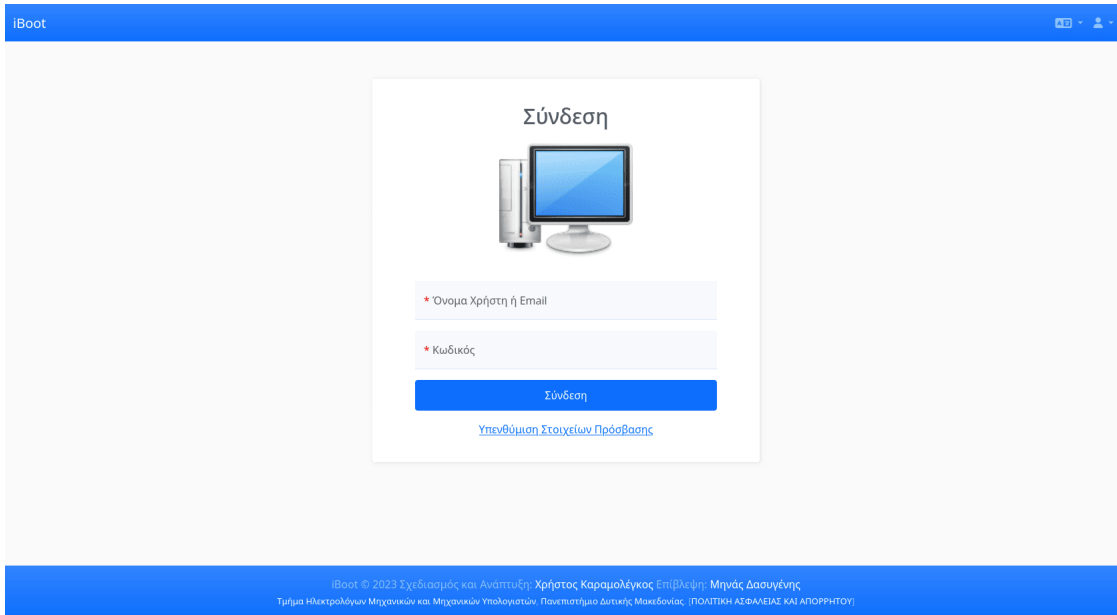
iBoot © 2023 Σχεδιασμός και Ανάπτυξη: Χρήστος Καραμολέγκος Επίβλεψη: Μηνάς Δασυγένης
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Δυτικής Μακεδονίας, ΠΡΟΛΙΤΗΚΗ ΑΣΦΑΛΕΙΑΣ ΚΑΙ ΑΠΟΡΡΗΤΟΥ

Σχήμα 4.1: iBoot - Εγγραφή

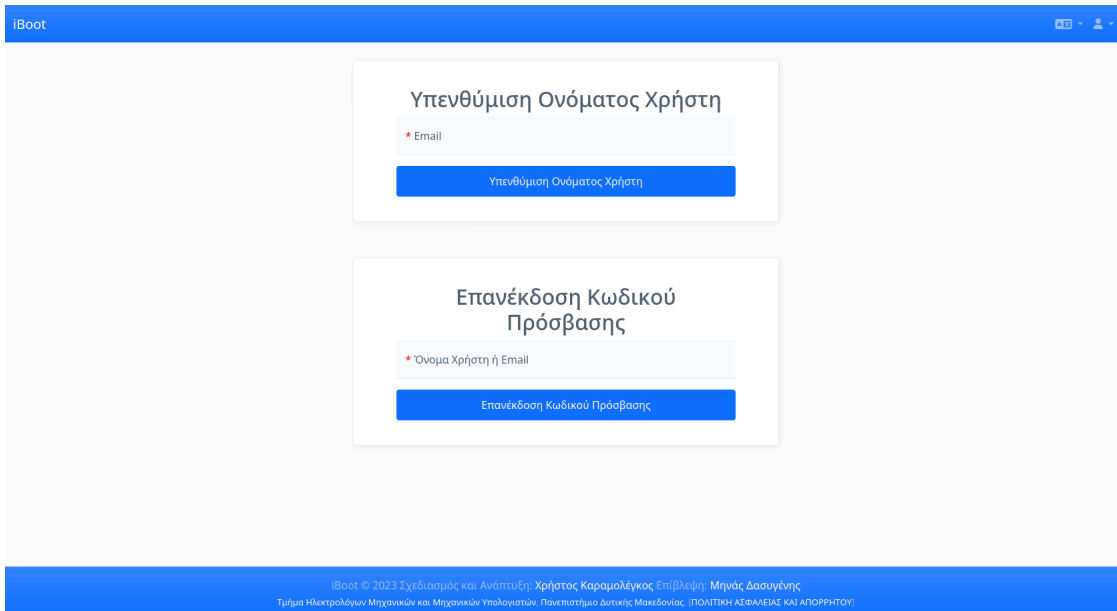
σελίδα διαχείρισης χρηστών 4.16.

Η σελίδα σύνδεσης στο σύστημα απεικονίζεται στο σχήμα 4.2 και είναι προσβάσιμη στη διεύθυνση `/login`.

Αν κάποιος χρήστης ξεχάσει τα στοιχεία πρόσβασής του και επιθυμεί να τα αλλάξει, θα πρέπει να επισκευτεί τη διεύθυνση `/forgotCredentials`, η οποία απεικονίζεται στο σχήμα 4.3. Εκεί, ο χρήστης έχει τη δυνατότητα να αιτηθεί την επανέκδοση του κωδικού πρόσβασής του, με χρήση του ονόματος χρήστη ή του email του. Μετά την υποβολή της φόρμας επανέκδοσης κωδικού πρόσβασης, ο χρήστης θα λάβει ένα email με έναν σύνδεσμο, τον οποίο θα πρέπει να ακολουθήσει για να δηλώσει τον νέο κωδικό πρόσβασής του. Σε περίπτωση που δε θυμάται το όνομα χρήστη του, μπορεί να βάλει το email του στη φόρμα υπενθύμισης ονόματος χρήστη και να λάβει ένα email που να αναφέρει το όνομα χρήστη του.



Σχήμα 4.2: iBoot - Σύνδεση

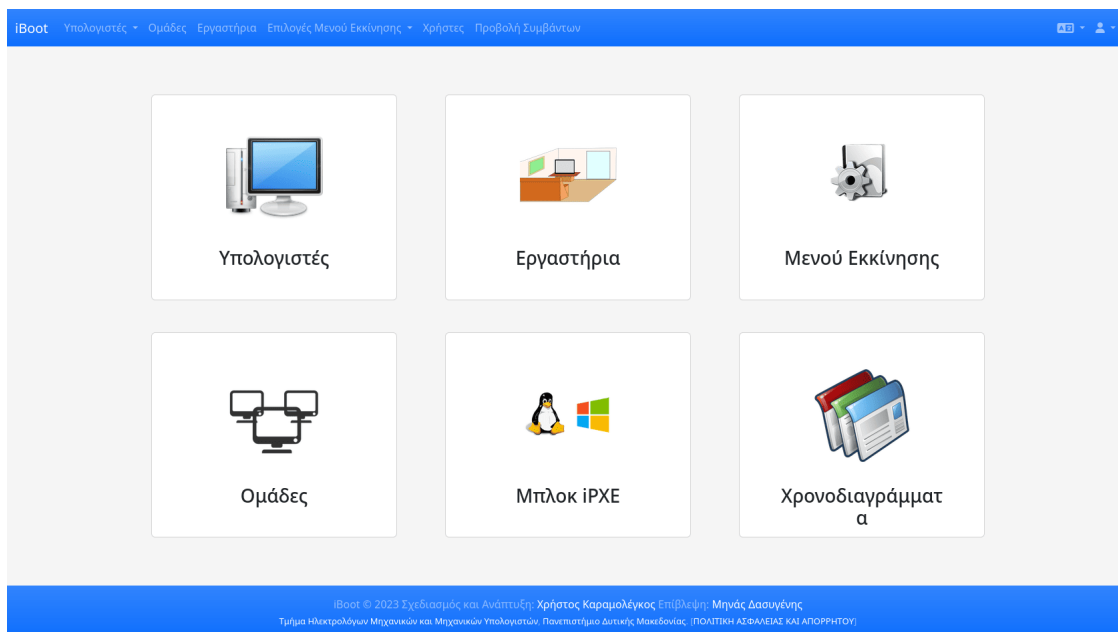


Σχήμα 4.3: iBoot - Υπενθύμιση Στοιχείων Πρόσβασης

4.1.2 Αρχική Σελίδα

Η αρχική σελίδα, προσβάσιμη στη διεύθυνση /**dashboard**, στην οποία μεταφέρεται ο χρήστης μετά τη σύνδεσή του (αν δεν είχε ζητήσει κάποια διαφορετική σελίδα).

Όπως φαίνεται στο σχήμα 4.4, η αρχική σελίδα περιέχει συνδέσμους προς άλλες σημαντικές σελίδες της εφαρμογής.



Σχήμα 4.4: iBoot - Αρχική Σελίδα

4.1.3 Μενού Πλοήγησης

Στο σχήμα 4.5, απεικονίζεται η μπάρα του μενού πλοήγησης για τους χρήστες με ρόλο διαχειριστή.

Στο αριστερό μέρος της μπάρας υπάρχει το όνομα της εφαρμογής, το οποίο αν πατηθεί επιστρέφει τον χρήστη την αρχική της σελίδα. Αμέσως μετά εμφανίζεται μια σειρά από επιλογές για τις διαθέσιμες στον χρήστη σελίδες.

Η πρώτη επιλογή στο αριστερό μέρος της μπάρας του μενού πλοήγησης, αφορά την απεικόνιση υπολογιστών. Όταν ο χρήστης περιηγηθεί με το ποντίκι του πάνω στην επιλογή (σε υπολογιστή) ή επιλέξει να την επεκτείνει (σε κινητές συσκευές), θα του παρουσιαστούν οι επιλογές απεικόνισης είτε Διαχειριζόμενων (Σχήμα 4.8) είτε Μη Εκχωρημένων Υπολογιστών (Σχήμα 4.9).

Η δεύτερη επιλογή κατευθύνει τον χρήστη στη σελίδα απεικόνισης των Ομάδων υπολογιστών (Σχήμα 4.10), ενώ η τρίτη σε αυτή της απεικόνισης των Εργαστηρίων (Σχήμα 4.11).

Η επόμενη ομάδα επιλογών, μπορεί να επεκταθεί με πανομοιότυπο τρόπο με την πρώτη, και αφορά την απεικόνιση των Μπλοκ Εντολών τύπου iPXE (Σχήμα 4.12), των Μενού Εκκίνησης (Σχήμα 4.13) και των Χρονοδιαγραμμάτων (Σχήμα 4.15).

Στη συνέχεια, βρίσκεται η επιλογή των Χρηστών της εφαρμογής (Σχήμα 4.16) και στο τέλος του αριστερού τμήματος της μπάρας πλοήγησης βρίσκεται η επιλογή απεικόνισης των αρχείων καταγραφής συμβάντων (Σχήμα 4.17).

Στο δεξί μέρος της μπάρας πλοήγησης υπάρχουν πρώτα ο επιλογές γλώσσας της εφαρμογής και έπειτα η ομάδα επιλογών της προβολής των στοιχείων του προφίλ του χρήστη και της αποσύνδεσής του.



Σχήμα 4.5: iBoot - Μενού Πλοήγησης Διαχειριστή

Όπως είναι εμφανές στο σχήμα 4.6, η μπάρα πλοήγησης των χρηστών με ρόλο διαχειριστή εργαστηρίου είναι σχεδόν πανομοιότυπη με εκείνη των χρηστών με ρόλο διαχειριστή. Η διαφορά του βρίσκεται στο ότι από την πρώτη απουσιάζουν οι επιλογές απεικόνισης χρηστών και προβολής των αρχείων καταγραφής συμβάντων, καθώς οι χρήστες που δεν έχουν ρόλο διαχειριστή δεν έχουν και πρόσβαση σε αυτές τις σελίδες.



Σχήμα 4.6: iBoot - Μενού Πλοήγησης Διαχειριστή Εργαστηρίου

Οι επιλογές του αριστερού τμήματος της μπάρας πλοήγησης απουσιάζουν εντελώς όταν ο χρήστης είναι μη αυθεντικοποιημένος επισκέπτης, όπως φαίνεται στο σχήμα 4.7. Εκείνος δηλαδή βλέπει μόνο το όνομα της εφαρμογής στο αριστερό μέρος της μπάρας πλοήγησης και στο δεξί μέρος της τον επιλογέα γλώσσας της εφαρμογής και την ομάδα επιλογών για πραγματοποίηση σύνδεσης και εγγραφής (αν αυτή είναι ενεργοποιημένη ως δυνατότητα του συστήματος).



Σχήμα 4.7: iBoot - Μενού Πλοήγησης Επισκέπτη

4.1.4 Υπολογιστές

Στη σελίδα Διαχειριζόμενων Υπολογιστών, προσβάσιμη στη διεύθυνση `/computers_managed`, ο χρήστης μπορεί να δει τους υπολογιστές των οποίων τα στοιχεία έχει δικαίωμα να διαχειριστεί. Αν ο χρήστης είναι διαχειριστής, βλέπει όλους τους υπολογιστές που έχουν εγγραφεί στην πλατφόρμα. Αν είναι διαχειριστής εργαστηρίου, τότε βλέπει μόνο τους υπολογιστές που βρίσκονται σε εργαστήρια που αυτός διαχειρίζεται.

Όπως φαίνεται στο σχήμα 4.8, ο πίνακας με τα στοιχεία των υπολογιστών αποτελείται από τις παρακάτω στήλες:

- Υπολογιστής: Το όνομα που έχει δοθεί στον υπολογιστή
- UUID: Το μοναδικό αναγνωριστικό UUID του υπολογιστή
- MAC: Η φυσική διεύθυνση διεπαφής δικτύου (MAC address) του υπολογιστή
- Σημείωση: Πεδίο ελεύθερου κειμένου για διατήρηση σημειώσεων σχετικά με τον υπολογιστή
- Ομάδες: Πεδίο πολλαπλής επιλογής, περιέχει τις ομάδες των οποίων είναι μέλος ο υπολογιστής
- Εργαστήριο: Πεδίο επιλογής, περιέχει το εργαστήριο στο οποίο βρίσκεται ο υπολογιστής
- Τελευταία Εκκίνηση: Χρονοσφραγίδα τελευταίας εκκίνησης του υπολογιστή μέσω του συστήματος.

Υπολογιστής	UUID	MAC	Σημειώσεις	Ομάδες	Εργαστήριο	Εκκινήθηκε	Διαγραφή
pc1	538c60cb-5c7c-4...	01:ca:6e:acf8:8d	Μια σημείωση	Group1	Lab1		✗
pc2	16051fe3-211e-4...	84:2e:29:21:b8:37			Lab3		✗
pc3	538c60cb-5c7c-4...	03:ca:6e:acf8:8d		Group1, Group2	Lab4	πριν από 12 ώρες	✗
pc5	538c60cb-5c7c-4...	05:ca:6e:acf8:8d		Group1	Lab4	πριν από 5 ώρες	✗
laptop1	bdc4584-a7da-...	54:ee:75:b7:e6:0d	Added from 2.87...	Group1	Lab1	πριν από 2 ημέρες	✗
test-pc	38cab980-9a59-...	08:00:27:94:dd:a8	Added from 188...	Group2	Lab2		✗

Σχήμα 4.8: iBoot - Διαχειριζόμενοι Υπολογιστές

- Διαγραφή: Κουμπί για τη διαγραφή του υπολογιστή από το σύστημα

Διαχειριζόμενοι Υπολογιστές θεωρούνται όσοι έχουν τοποθετηθεί σε κάποιο εργαστήριο, από έναν διαχειριστή ή διαχειριστή εργαστηρίου.

Όσοι υπολογιστές είναι εγγεγραμμένοι στο σύστημα αλλά δεν έχουν τοποθετηθεί σε κάποιο εργαστήριο, θεωρούνται Μη Εκχωρημένοι και εμφανίζονται σε ξεχωριστό πίνακα, ο οποίος απεικονίζεται στο σχήμα 4.9 και είναι προσβάσιμος στη διεύθυνση /computers_unassigned. Στον πίνακα με τους Μη Εκχωρημένους Υπολογιστές έχουν πρόσβαση όλοι οι χρήστες, ώστε να μπορούν να τοποθετούν τους υπολογιστές σε εργαστήρια, χωρίς ανάγκη παρέμβασης κάποιου διαχειριστή.

Υπολογιστής	UUID	MAC	Σημειώσεις	Ομάδες	Εργαστήριο	Εκκινήθηκε	Διαγραφή
pc4	538c60cb-5c7c-4...	04:ca:6e:acf8:8d			-		✗
	538c60cb-5c7c-4...	02:ca:6e:acf8:8d			-		✗
	938c60cb-5c7c-4...	91:ca:6e:acf8:8d	Added from 2.87...		-		✗

Σχήμα 4.9: iBoot - Μη Εκχωρημένοι Υπολογιστές

Φυσικά, οι διαχειριστές εργαστηρίων μπορούν να τοποθετούν τους υπολογιστές μόνο σε εργαστήρια που οι

ίδιοι διαχειρίζονται, ενώ οι διαχειριστές μπορούν να τοποθετούν οποιονδήποτε υπολογιστή σε οποιοδήποτε εργαστήριο, ακόμη και αν κάποιος άλλος χρήστης είχε τοποθετήσει τον υπολογιστή σε άλλο εργαστήριο νωρίτερα. Αντίστοιχα, αν κάποιος αφαιρέσει έναν υπολογιστή από εργαστήριο, τότε αυτός θα εμφανίζεται στον πίνακα των Μη Εκχωρημένων και οποιοσδήποτε θα έχει τη δυνατότητα να τον προσθέσει σε κάποιο εργαστήριο στο οποίο έχει πρόσβαση.

Στις σελίδες που αφορούν τους υπολογιστές, έχει προστεθεί ένα πλαίσιο επιλογής (checkbox) για την ενεργοποίηση και απενεργοποίηση της λειτουργίας αυτόματης επαναφόρτωσης των δεδομένων των υπολογιστών. Αν το πλαίσιο είναι επιλεγμένο, η δυνατότητα επεξεργασίας στοιχείων απενεργοποιείται και κάθε πέντε δευτερόλεπτα γίνεται κλήση AJAX στο endpoint του REST API της εφαρμογής, ώστε ο πίνακας των υπολογιστών να ξαναγεμίζεται με ενημερωμένα δεδομένα. Αυτό, σε συνδυασμό με φίλτρο λεπτών στο πεδίο της τελευταίας εκκίνησης, επιτρέπει τη ζωντανή παρακολούθηση της εκκίνησης των υπολογιστών.

4.1.5 Ομάδες Υπολογιστών

Στη σελίδα Ομάδων Υπολογιστών, προσβάσιμη στη διεύθυνση `/groups`, ο χρήστης μπορεί να δει τις ομάδες υπολογιστών που έχουν δημιουργηθεί και τους υπολογιστές που έχουν προστεθεί σε αυτές. Αν ο χρήστης είναι διαχειριστής, βλέπει όλους τους υπολογιστές που έχει μια ομάδα, διαφορετικά μόνο εκείνους που βρίσκονται σε εργαστήρια τα οποία διαχειρίζεται.

Όπως φαίνεται στο σχήμα 4.10, ο πίνακας με τα στοιχεία των ομάδων αποτελείται από τις παρακάτω στήλες:

- Ομάδα: Το όνομα που έχει δοθεί στην ομάδα
- IP διακομιστή: Η διεύθυνση του διακομιστή στον οποίο μπορούν οι υπολογιστές - μέλη της ομάδας να αναζητήσουν εικόνες λειτουργικών συστημάτων
- Πρόθεμα διαδρομής διακομιστή: Το πρόθεμα της διαδρομής στον διακομιστή, όπου βρίσκονται οι εικόνες των λειτουργικών συστημάτων
- Υπολογιστές: Πεδίο πολλαπλής επιλογής, περιέχει τους υπολογιστές οι οποίοι είναι μέλη της ομάδας
- Διαγραφή: Κουμπί για τη διαγραφή της ομάδας από το σύστημα

Τα πεδία *IP διακομιστή* και *Πρόθεμα διαδρομής διακομιστή* μπορούν να χρησιμοποιηθούν μέσω των προκαθορισμένων μεταβλητών $\{group.ip\}$ και $\{group.prefix\}$ αντίστοιχα, σε καταχωρήσεις μενού iPXE.

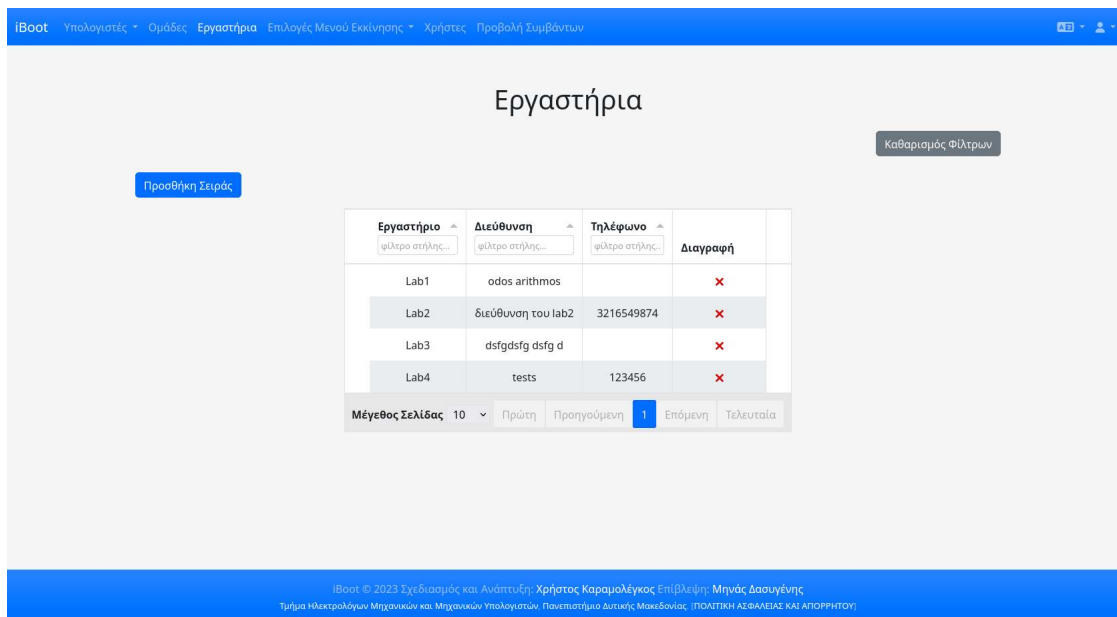
Σχήμα 4.10: iBoot - Διαχείριση Ομάδων Υπολογιστών

4.1.6 Εργαστήρια

Στη σελίδα των Εργαστηρίων, προσβάσιμη στη διεύθυνση `/labs`, ο χρήστης μπορεί να δει τα στοιχεία των εργαστηρίων που έχουν δημιουργηθεί.

Όπως φαίνεται στο σχήμα 4.11, ο πίνακας με τα στοιχεία των εργαστηρίων αποτελείται από τις παρακάτω στήλες:

- Εργαστήριο: Το όνομα που έχει δοθεί στο εργαστήριο
- Διεύθυνση: Η διεύθυνση του εργαστηρίου
- Τηλέφωνο: Το τηλέφωνο του εργαστηρίου, αν έχει προστεθεί
- Διαγραφή: Κουμπί για τη διαγραφή του εργαστηρίου από το σύστημα



Σχήμα 4.11: iBoot - Διαχείριση Εργαστηρίων

4.1.7 Block εντολών τύπου iPXE

Στη σελίδα των Μπλοκ iPXE, προσβάσιμη στη διεύθυνση `/ipxeblocks`, ο χρήστης μπορεί να δει τα στοιχεία των Μπλοκ iPXE που έχουν δημιουργηθεί.

Όπως φαίνεται στο σχήμα 4.12, ο πίνακας με τα στοιχεία των Μπλοκ iPXE αποτελείται από τις παρακάτω στήλες:

- Μπλοκ iPXE: Το όνομα που έχει δοθεί στο Μπλοκ iPXE
- Καταχώρηση Μενού iPXE: Το περιεχόμενο της καταχώρησης μπλοκ iPXE
- Διαγραφή: Κουμπί για τη διαγραφή του Μπλοκ iPXE από το σύστημα

Η *Καταχώρηση Μενού iPXE* είναι πεδίο ελεύθερου κειμένου, όπου ο χρήστης πρέπει να καταχωρήσει ένα έγκυρο μπλοκ iPXE. Μπορεί να χρησιμοποιήσει τις μεταβλητές $\{group.ip\}$ και $\{group.prefix\}$, έτσι ώστε να γίνεται αυτόματη αντικατάστασή τους στο μενού που θα δημιουργείται, ανάλογα με το ποιο group περιείχε τον υπολογιστή που διαβάζει το μπλοκ.

Αν υπάρχει μπλοκ με όνομα *default*, τότε σε περίπτωση που ζητήσει μενού κάποιος υπολογιστής που δεν έχει ενεργό πρόγραμμα εκκίνησης εκείνη τη στιγμή, θα λάβει ένα μενού με αυτό το μπλοκ.

[iBoot](#)
[Υπολογιστές](#)
[Ομάδες](#)
[Εργαστήρια](#)
[Επιλογές Μενού Εκκίνησης](#)
[Χρήστες](#)
[Προβολή Συμβάντων](#)

[Καθαρισμός Φίλτρων](#)

Μπλοκ iPXE

[Προσθήκη Σειράς](#)

id	Μπλοκ iPXE	Καταχώρηση Μενού iPXE	Διαγραφή
1	ubuntu jammy	kernel casper/vmlinuz initrd casper/initrd append root=/dev/ram0 ramdisk_size=1500000 ip=dhcp cloud-config- url=/dev/null url=http://\$(group.ip)\$(group... live-server-amd64.iso autoinstall ds=nocloud-net; s=http://\$(group.ip)\$(group...	✘
2	ubuntu 2	kernel casper/vmlinuz initrd casper/initrd append root=/dev/ram0 ramdisk_size=1500000 ip=dhcp cloud-config- url=/dev/null url=http://192.168.42.151 /jammy-live-server-amd64.iso autoinstall ds=nocloud-net; s=http://192.168.42.151 /jammy/ # Don't forget the slash at the end.	✘
3	ubuntu 3	kernel casper/vmlinuz initrd casper/initrd append root=/dev/ram0 ramdisk_size=1500000 ip=dhcp cloud-config- url=/dev/null url=http://192.168.42.151 /jammy-live-server-amd64.iso autoinstall ds=nocloud-net; s=http://192.168.42.151 /jammy/ # Don't forget the slash at the end.	✘
4	default	kernel casper/vmlinuz initrd casper/initrd append root=/dev/ram0 ramdisk_size=1500000 ip=dhcp cloud-config- url=/dev/null url=http://192.168.42.151 /jammy-live-server-amd64.iso autoinstall ds=nocloud-net; s=http://192.168.42.151 /jammy/ # Don't forget the slash at the end.	✘

Μέγεθος Σελίδας 10

[Πρώτη](#)
[Προηγούμενη](#)
1
[Επόμενη](#)
[Τελευταία](#)

iBoot © 2023 Σχεδιασμός και Ανάπτυξη: Χρήστος Καραμολύγκος/Επιβλέπων: Μηνάς Δασυγένης
 Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Δυτικής Μακεδονίας, (ΠΟΛΙΤΕΚΝΗ ΑΣΦΑΛΕΙΑΣ ΚΑΙ ΑΠΟΡΡΙΠΤΟΥ)

Σχήμα 4.12: iBoot - Διαχείριση Μπλοκ Εντολών Τύπου iPXE

4.1.8 Μενού Εκκίνησης

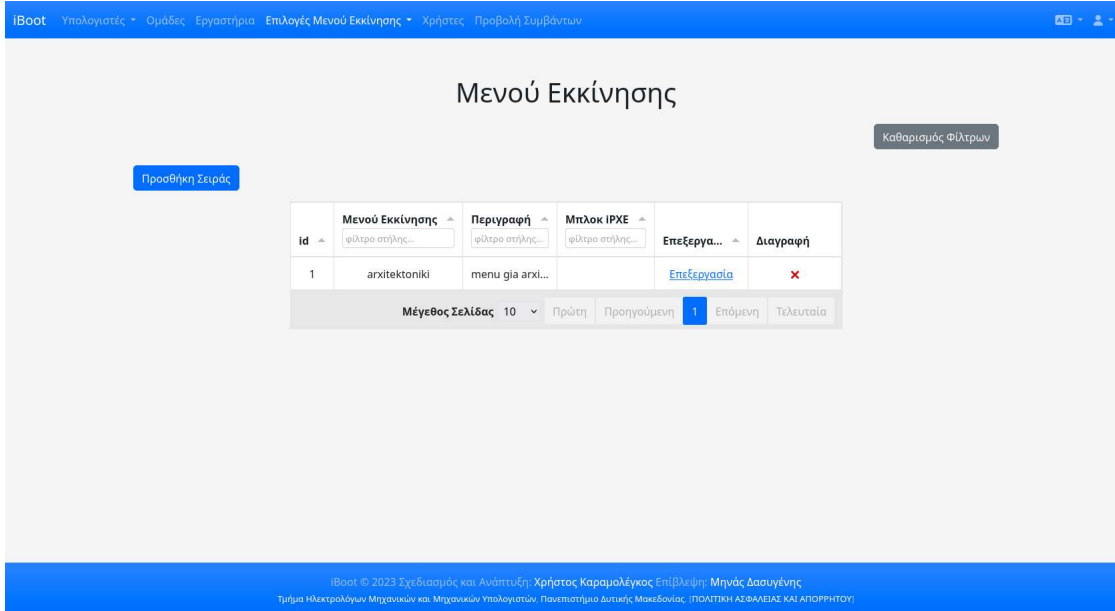
Στη σελίδα Μενού Εκκίνησης, προσβάσιμη στη διεύθυνση `/boot_menu`, ο χρήστης μπορεί να προβάλει και να επεξεργαστεί τα μενού εκκίνησης που έχουν δημιουργηθεί, καθώς και να δημιουργήσει νέα.

Όπως φαίνεται στο σχήμα 4.13, ο πίνακας με τα στοιχεία των ομάδων αποτελείται από τις παρακάτω στήλες:

- **id:** Το id του μενού εκκίνησης. Προβάλλεται για διευκόλυνση του διαχειριστή στη χειροκίνητη δοκιμή των μενού εκκίνησης
- **Μενού Εκκίνησης:** Το όνομα που έχει δοθεί στο μενού εκκίνησης
- **Περιγραφή:** Η περιγραφή του μενού εκκίνησης, για να μπορεί πιο εύκολα ο χρήστης να ξεχωρίσει τα μενού μεταξύ τους
- **Μπλοκ iPXE:** Ένα iPXE μπλοκ το οποίο αντιστοιχεί μόνο στο συγκεκριμένο μενού και εμφανίζεται κατά

την εκκίνηση πριν από όλες τις άλλες εγγραφές που περιλαμβάνονται σε αυτό

- Επεξεργασία: Κουμπί για την επεξεργασία του μενού εκκίνησης, ως προς τα μπλοκ iPXΕ που περιέχει
- Διαγραφή: Κουμπί για τη διαγραφή του μενού εκκίνησης από το σύστημα



Σχήμα 4.13: iBoot - Διαχείριση Μενού Εκκίνησης

Πατώντας στην επεξεργασία ενός μενού, ο χρήστης μεταφέρεται στη σελίδα `/boot_menu/{id}`, όπου μπορεί να προσθέσει ή να αφαιρέσει μπλοκ iPXΕ από τα υπάρχοντα στο συγκεκριμένο μενού εκκίνησης. Η σελίδα επεξεργασίας μενού απεικονίζεται στο σχήμα 4.14.



Σχήμα 4.14: iBoot - Επεξεργασία Μενού Εκκίνησης

4.1.9 Χρονοδιαγράμματα

Στη σελίδα Χρονοδιαγράμματα, προσβάσιμη στη διεύθυνση `/schedules`, ο χρήστης μπορεί να προβάλει και να επεξεργαστεί τα χρονοδιαγράμματα που έχουν δημιουργηθεί, καθώς και να δημιουργήσει νέα. Τα υπάρχοντα ενεργά χρονοδιαγράμματα απεικονίζονται σε ένα ημερολόγιο, ώστε να είναι ευκολότερος ο προσδιορισμός τους από τον χρήστη, και η κατανόηση του μενού που θα δημιουργηθεί για μια ομάδα σε κάποια συγκεκριμένη χρονική στιγμή.

Όπως φαίνεται στο σχήμα 4.15, ο πίνακας με τα στοιχεία των χρονοδιαγραμμάτων αποτελείται από τις παρακάτω στήλες:

- Ώρα Από: Η ώρα από την οποία θα είναι ενεργό το χρονοδιάγραμμα
- Ώρα Έως: Η ώρα έως την οποία θα είναι ενεργό το χρονοδιάγραμμα
- Ημέρα Εβδομάδας: Η ημέρα της εβδομάδας για την οποία θα είναι ενεργό το χρονοδιάγραμμα
- Ημερομηνία: Η συγκεκριμένη ημερομηνία για την οποία θα είναι ενεργό το χρονοδιάγραμμα
- Μενού Εκκίνησης: Το μενού εκκίνησης που θα δοθεί στα μηχανήματα που θα ζητήσουν να εκκινηθούν εντός χρονικών πλαισίων του χρονοδιαγράμματος
- Ομάδα: Η ομάδα της οποίας οι υπολογιστές-μέλη μπορούν να αιτηθούν μενού εκκίνησης από το συγκεκριμένο χρονοδιάγραμμα
- Ενεργό: Πεδίο με πλαίσιο επιλογής που καθορίζει αν είναι ενεργό ή όχι το χρονοδιάγραμμα
- Δημιουργήθηκε στις: Πεδίο μόνο για ανάγνωση, περιέχει την χρονοσφραγίδα δημιουργίας του χρονοδιαγράμματος
- Ενημερώθηκε στις: Πεδίο μόνο για ανάγνωση, περιέχει την χρονοσφραγίδα τελευταίας ενημέρωσης του χρονοδιαγράμματος
- Διαγραφή: Κουμπί για τη διαγραφή του χρονοδιαγράμματος από το σύστημα

Για τη δημιουργία χρονοδιαγράμματος, είναι απαραίτητο μόνο ένα από τα πεδία *Ημέρα Εβδομάδας* και *Ημερομηνία*. Αν έχει οριστεί το πεδίο *Ημέρα Εβδομάδας*, τότε το χρονοδιαγραμμα θα είναι επαναλαμβανόμενο και θα ενεργοποιείται κάθε εβδομάδα, την ημέρα που είναι δηλωμένη στο πεδίο. Αν έχει οριστεί το πεδίο *Ημερομηνία*, τότε το χρονοδιαγραμμα δεν θα είναι επαναλαμβανόμενο και θα ενεργοποιηθεί μόνο τη συγκεκριμένη ημερομηνία που είναι δηλωμένη στο πεδίο. Αν τύχει για μια χρονική στιγμή να είναι ταυτόχρονα ενεργά για την ίδια ομάδα υπολογιστών δύο χρονοδιαγράμματα, εκ των οποίων το ένα έχει ορισμένο το πεδίο *Ημέρα Εβδομάδας* και το άλλο το πεδίο *Ημερομηνία*, τότε το χρονοδιάγραμμα με τη συγκεκριμένη ημερομηνία έχει προτεραιότητα και υπερκαλύπτει το άλλο.

iBoot Υπολογιστές Ομάδες Εργαστήρια Επιλογές Μενού Εκκίνησης Χρήστες Προβολή Συμβάντων

Χρονοδιαγράμματα

6 - 12 Αυγ 2023 day week

	Κυριακή	Δευτέρα	Τρίτη	Τετάρτη	Πέμπτη	Παρασκευή	Σάββατο
6 π.μ.							
7 π.μ.							
8 π.μ.		8:50 μ.μ. - 11:59 μ.μ. BM: 1 for G: 1					
9 π.μ.							
10 π.μ.							
11 π.μ.							
12 μ.μ.							
1 μ.μ.							
2 μ.μ.							
3 μ.μ.							
4 μ.μ.							
5 μ.μ.							
6 μ.μ.		6:00 μ.μ. - 8:00 BM: 1 for G: 2					
7 μ.μ.							
8 μ.μ.							
9 μ.μ.							
10 μ.μ.							
11 μ.μ.							

Καθαρισμός Φίλτρων

Προσθήκη Σειράς

Ώρα Από	Ώρα Έως	Ημέρα Εβδομάδας	Ημερομηνία	Μενού Εκκίνησης	Ομάδα	Ενεργό	Δημιουργήθηκε στις	Ενημερώθηκε
--:--	--:--	φίλτρο στήλης...	mm / mm /	φίλτρο στήλης...	φίλτρο στή...	<input type="checkbox"/>		
16:00	18:00		21/07/2023	αρχιτεκτονικι	Group1	<input checked="" type="checkbox"/>	20/07/2023 15:04:07	05/08/2023
18:00	20:00	Δευτέρα		αρχιτεκτονικι	Group2	<input checked="" type="checkbox"/>	20/07/2023 17:15:05	28/07/2023
08:00	23:59	Δευτέρα		αρχιτεκτονικι	Group1	<input checked="" type="checkbox"/>	23/07/2023 21:20:56	24/07/2023
18:00	20:00		28/07/2023	αρχιτεκτονικι	Group2	<input checked="" type="checkbox"/>	28/07/2023 18:48:59	28/07/2023
16:30	17:30		21/07/2023	αρχιτεκτονικι	Group1	<input checked="" type="checkbox"/>	28/07/2023 19:00:29	28/07/2023
17:00	17:30		29/07/2023	αρχιτεκτονικι	Group2	<input checked="" type="checkbox"/>	29/07/2023 17:01:52	29/07/2023
13:00	14:00		01/08/2023	αρχιτεκτονικι	Group1	<input checked="" type="checkbox"/>	01/08/2023 13:43:23	01/08/2023

Μέγεθος Σελίδας 10 Πρώτη Προηγούμενη 1 Επόμενη Τελευταία

iBoot © 2023 Σχεδιασμός και Ανάπτυξη: Χρήστος Καραμολέγκος Επιβλέψη: Μηνάς Δασυγένης
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Δυτικής Μακεδονίας, (ΠΟΛΙΤΙΚΗ ΑΣΦΑΛΕΙΑΣ ΚΑΙ ΑΠΟΡΡΗΤΟΥ)

Σχήμα 4.15: iBoot - Διαχείριση Χρονοδιαγραμμάτων

4.1.10 Χρήστες

Στη σελίδα Χρήστες, προσβάσιμη μόνο από διαχειριστές στη διεύθυνση `/users`, ο χρήστης μπορεί να προβάλει και να επεξεργαστεί τα στοιχεία των χρηστών της εφαρμογής.

Όπως φαίνεται στο σχήμα 4.16, ο πίνακας με τα στοιχεία των χρηστών αποτελείται από τις παρακάτω στήλες:

- Ονοματεπώνυμο: Το ονοματεπώνυμο του χρήστη
- Email: Η διεύθυνση email του χρήστη
- Τηλέφωνο: Ο αριθμός τηλεφώνου του χρήστη
- Όνομα Χρήστη: Το όνομα χρήστη (username) που χρησιμοποιείται για ταυτοποίηση στο σύστημα
- Κωδικός: Ο κωδικός πρόσβασης του χρήστη
- Διαχειριστής: Πεδίο με πλαίσιο επιλογής που καθορίζει αν είναι διαχειριστής ή όχι ο χρήστης
- Επιβεβαιωμένο Email: Πεδίο με πλαίσιο επιλογής που καθορίζει αν είναι επιβεβαιωμένο ή όχι το email του χρήστη
- Εργαστήρια: Λίστα των εργαστηρίων που έχουν ανατεθεί στον χρήστη, αν αυτός δεν είναι διαχειριστής
- Δημιουργήθηκε στις: Πεδίο μόνο για ανάγνωση, περιέχει την χρονοσφραγίδα δημιουργίας του χρονοδιαγράμματος
- Ενημερώθηκε στις: Πεδίο μόνο για ανάγνωση, περιέχει την χρονοσφραγίδα τελευταίας ενημέρωσης του χρονοδιαγράμματος
- Διαγραφή: Κουμπί για τη διαγραφή του χρήστη από το σύστημα

Όνοματεπώνυμο	Email	Τηλέφωνο	Όνομα Χρήστη	Κωδικός	Διαχειριστής	Επιβεβαιωμένο Email	Εργαστήρια
iBoot Admin	ibootAdmin@ckaramolegkos.gr	1234567890	ibootAdmin		✓	✓	
iBoot User 1	ibootUser1@ckaramolegkos.gr		ibootUser1		✗	✓	Lab
iBoot User 2	ibootUser2@ckaramolegkos.gr		ibootUser2		✗	✗	

Σχήμα 4.16: iBoot - Διαχείριση Χρηστών

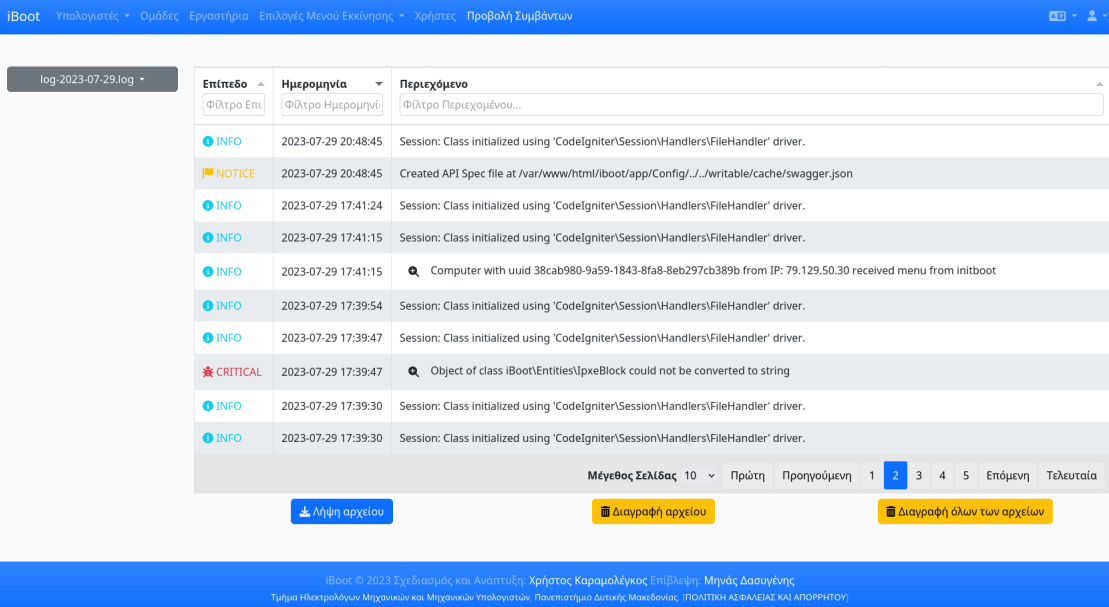
Το πεδίο Κωδικός φαίνεται κενό, καθώς οι κωδικοί αποθηκεύονται κρυπτογραφημένοι στη βάση δεδομένων της εφαρμογής και οι κρυπτογραφημένες συμβολοσειρές δεν εμφανίζονται σε κανένα σημείο της εφαρμογής. Το πεδίο είναι εγγράψιμο και υπάρχει για την αλλαγή του κωδικού του κάθε χρήστη.

4.1.11 Συμβάντα

Στη σελίδα των Συμβάντων, προσβάσιμη μόνο από διαχειριστές στη διεύθυνση /logs, ο χρήστης μπορεί να δει τις καταγραφές συμβάντων που έχουν δημιουργηθεί από την εφαρμογή.

Όπως φαίνεται στο σχήμα 4.17, ο πίνακας με τα στοιχεία των Μπλοκ iPXE αποτελείται από τις παρακάτω στήλες:

- Επίπεδο: Το όνομα που έχει δοθεί στο Μπλοκ iPXE
- Ημερομηνία: Το περιεχόμενο της καταχώρησης μπλοκ iPXE
- Περιεχόμενο: Κουμπί για τη διαγραφή του Μπλοκ iPXE από το σύστημα



The screenshot shows the iBoot logs interface. At the top, there is a navigation bar with the iBoot logo and several menu items. Below the navigation bar, there is a dropdown menu for the log file, currently set to 'log-2023-07-29.log'. The main content area is a table with three columns: 'Επίπεδο' (Level), 'Ημερομηνία' (Date), and 'Περιεχόμενο' (Content). The table contains several rows of log entries, including INFO, NOTICE, and CRITICAL messages. At the bottom of the table, there are buttons for 'Λήψη αρχείου' (Download file), 'Διαγραφή αρχείου' (Delete file), and 'Διαγραφή όλων των αρχείων' (Delete all files). The footer of the page contains copyright information and the name of the development team.

Επίπεδο	Ημερομηνία	Περιεχόμενο
INFO	2023-07-29 20:48:45	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
NOTICE	2023-07-29 20:48:45	Created API Spec file at /var/www/html/iboot/app/Config/...writable/cache/swagger.json
INFO	2023-07-29 17:41:24	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
INFO	2023-07-29 17:41:15	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
INFO	2023-07-29 17:41:15	Computer with uuid 38cab980-9a59-1843-8fa8-8eb297cb389b from IP: 79.129.50.30 received menu from initboot
INFO	2023-07-29 17:39:54	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
INFO	2023-07-29 17:39:47	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
CRITICAL	2023-07-29 17:39:47	Object of class iBoot\Entities\IpxeBlock could not be converted to string
INFO	2023-07-29 17:39:30	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.
INFO	2023-07-29 17:39:30	Session: Class initialized using 'CodeIgniter\Session\Handlers\FileHandler' driver.

Σχήμα 4.17: iBoot - Συμβάντα

Στο αριστερό μέρος της σελίδας, υπάρχει ένας επιλογέας του αρχείου καταγραφής συμβάντων που θέλει να προβάλει ο χρήστης, με προεπιλεγμένο αυτό της τρέχουσας ημερομηνίας. Επιλέγοντας ένα αρχείο, αν είναι πολύ μεγάλο για προβολή ο χρήστης προτρέπεται να το μεταφορτώσει για να το προβάλει τοπικά στη συσκευή του, διαφορετικά οι εγγραφές του αρχείου γεμίζουν τον πίνακα.

Ο χρήστης έχει τη δυνατότητα να επιλέξει αν θα εμφανίζονται στον πίνακα 10, 25, 50 ή 100 εγγραφές, ενώ αν υπάρχουν περισσότερες σελιδοποιούνται αυτόματα. Επίσης, μπορεί να φιλτράρει τις εγγραφές ανάλογα με το επίπεδό τους, την ημερομηνία ή το περιεχόμενό τους, κάνοντας έτσι εύκολη την αναζήτηση συγκεκριμένων εγγραφών.

Τέλος, δίνεται η δυνατότητα να γίνει λήψη ή διαγραφή του αρχείου που προβάλει ο χρήστης, καθώς και διαγραφή όλων των αρχείων καταγραφής συμβάντων.

4.1.12 Ρύθμιση Εκκίνησης Υπολογιστών

Στη σελίδα των Ρύθμισης Εκκίνησης Υπολογιστών, προσβάσιμη στη διεύθυνση `/boot`, δίνονται οδηγίες παραμετροποίησης του DHCP Server από τον οποίο θα ζητήσουν διαδικτυακή εκκίνηση οι υπολογιστές όταν εκκινήθουν. Συγκεκριμένα, υπάρχουν οδηγίες ρύθμισης για ICS DHCPD και Dnsmasq, ενώ ο χρήστης παροτρύνεται να αναζητήσει οδηγίες ρύθμισης στις οδηγίες χρήστη του DHCP Server του, αν χρησιμοποιεί κάποιον διαφορετικό. Η σελίδα Ρύθμισης Εκκίνησης Υπολογιστών απεικονίζεται στο σχήμα 4.18.



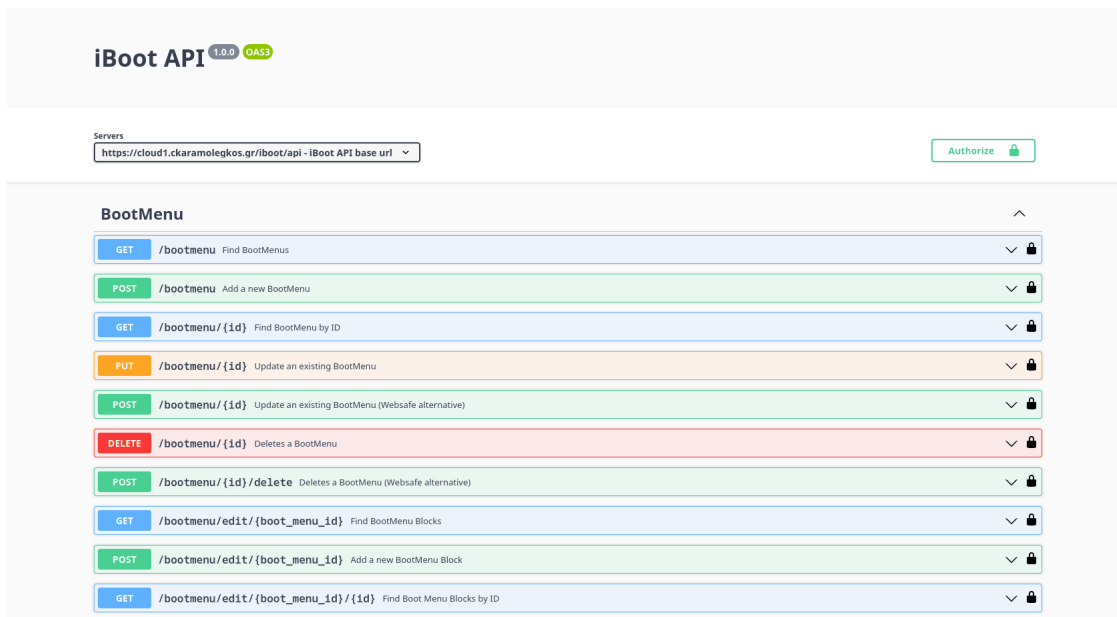
Σχήμα 4.18: iBoot - Ρύθμιση Εκκίνησης Υπολογιστών

4.1.13 Προδιαγραφή API

Η σελίδα της Προδιαγραφής API, προσβάσιμη στη διεύθυνση `/api`, περιέχει μια λεπτομερή καταγραφή των REST API endpoints της εφαρμογής. Οι εγγραφές είναι λειτουργικές, αφού ο χρήστης μπορεί να πραγματοποιήσει αιτήματα στο API της εφαρμογής μέσω της σελίδας, πατώντας το κουμπί “Try it out” της εγγραφής που θέλει να δοκιμάσει. Εκτελώντας ένα αίτημα σε κάποιο endpoint του API, εμφανίζεται και η αντίστοιχη curl εντολή που θα μπορούσε να εκτελέσει ο χρήστης για να πραγματοποιήσει το ίδιο αίτημα.

Φυσικά, όλα τα endpoints είναι κλειδωμένα και απαιτούν τη χρήση JWT token για να αυθεντικοποιηθεί ο χρήστης που εκτελεί το κάθε αίτημα και να διαπιστωθεί ότι έχει τα απαραίτητα δικαιώματα. Το endpoint της σύνδεσης χρήστη `/api/user/login` είναι ανοιχτό για POST αιτήματα, ώστε ο χρήστης να μπορεί με χρήση του ονόματος χρήστη και του κωδικού πρόσβασής του, να λάβει το JWT token το οποίο πρέπει να επισυνάπτει σε όλα τα επόμενα αιτήματα προς τα κλειδωμένα endpoints, για την εξυπηρέτησή τους. Πατώντας το κουμπί “Authorize” στο πάνω μέρος της σελίδας, ο χρήστης μπορεί να προσθέσει στο πλαίσιο που εμφανίζεται το JWT token του, ώστε να επισυνάπτεται αυτόματα σε όλα τα επόμενα αιτήματα που θα πραγματοποιήσει. Αν ο χρήστης έχει συνδεθεί στην εφαρμογή ήδη, όταν επισκεφτεί τη σελίδα Προδιαγραφής API, το token του επισυνάπτεται αυτόματα.

Η σελίδα της Προδιαγραφής API απεικονίζεται στο σχήμα 4.19.



Σχήμα 4.19: iBoot - Προδιαγραφή API

4.2 Επεξήγηση Αρχείων

Καθώς η διαδικτυακή εφαρμογή αναπτύχθηκε με χρήση του CodeIgniter 4 Framework, χρησιμοποιεί και τη δομή αρχείων του [32].

4.2.1 Βασική Δομή CodeIgniter 4

Η βασική δομή της εφαρμογής αποτελείται από πέντε καταλόγους: `app/`, `public/`, `writable/`, `tests/` και `vendor/` ή `system/`.

app

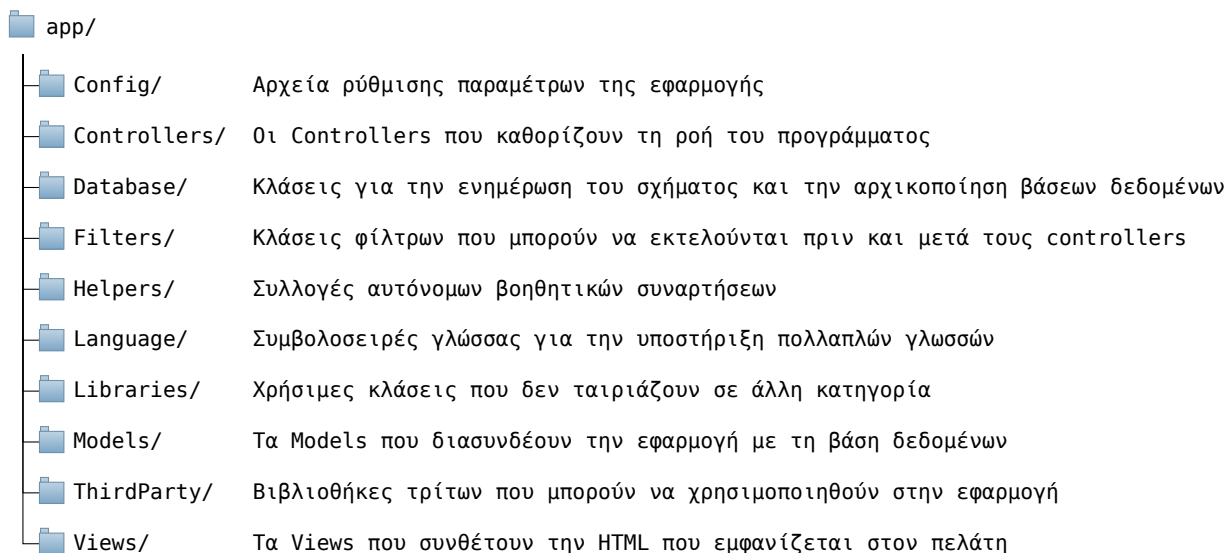
Στον κατάλογο `app` βρίσκεται όλος ο κώδικας της εφαρμογής. Οι ακόλουθοι φάκελοι αποτελούν τα βασικά περιεχόμενα:

Στον κατάλογο `app` έχει δοθεί το `php namespace` 'iBoot', έτσι ώστε τα ονόματα των αρχείων που βρίσκονται στους υποκαταλόγους του να μην συγκρούονται με ομοίως ονομαζόμενα αρχεία του Framework ή άλλων χρησιμοποιούμενων βιβλιοθηκών.

public

Ο κατάλογος `public` περιέχει το προσβάσιμο από το πρόγραμμα περιήγησης τμήμα της διαδικτυακής εφαρμογής, αποτρέποντας την άμεση πρόσβαση στον πηγαίο κώδικα. Περιέχει το κύριο αρχείο `.htaccess`, το `index.php` και όλα τα επιπρόσθετα στοιχεία της εφαρμογής, όπως CSS, javascript και εικόνες.

Αυτός ο φάκελος προορίζεται να είναι η "διαδικτυακή ρίζα" της εφαρμογής και ο διακομιστής ιστού θα πρέπει να ρυθμιστεί ώστε να δείχνει σε αυτόν.



Σχήμα 4.20: Η βασική δομή του CodeIgniter 4 framework

writable

Αυτός ο κατάλογος περιέχει όλους τους καταλόγους στους οποίους μπορεί να χρειαστεί να γίνει εγγραφή αρχείων κατά τη διάρκεια της ζωής της εφαρμογής. Αυτό περιλαμβάνει καταλόγους για την αποθήκευση προσωρινών αρχείων cache, αρχείων καταγραφής συμβάντων και τυχόν μεταφορτώσεις που μπορεί να στείλει ένας χρήστης. Θα πρέπει να προστεθούν εκεί όλοι οι κατάλογοι στους οποίους θα χρειαστεί να γράψει η εφαρμογή. Αυτό επιτρέπει τον ορισμό των υπόλοιπων κύριων καταλόγων ως μη εγγράψιμους, ως πρόσθετο μέτρο ασφαλείας κατά της ανεπιθύμητης τροποποίησης του κώδικα της εφαρμογής.

tests

Αυτός ο κατάλογος έχει οριστεί για να περιέχει τα αρχεία δοκιμών. Ο κατάλογος `_support` περιέχει διάφορες κλάσεις προσομοίωσης και άλλα βοηθητικά προγράμματα που μπορούν να χρησιμοποιηθούν για τη συγγραφή και εκτέλεση δοκιμών. Αυτός ο κατάλογος δεν χρειάζεται να υπάρχει στην παραγωγική εγκατάσταση της εφαρμογής.

system

Αυτός ο κατάλογος περιέχει τα αρχεία που συνθέτουν το ίδιο το CodeIgniter 4 Framework. Τα αρχεία στον κατάλογο συστήματος δεν πρέπει ποτέ να τροποποιούνται. Αντ' αυτού, μπορούν να επεκταθούν οι υπάρχουσες κλάσεις ή να δημιουργηθούν νέες, ώστε να παρέχουν την επιθυμητή λειτουργικότητα.

Όλα τα αρχεία σε αυτόν τον κατάλογο βρίσκονται κάτω από το `php` namespace `'CodeIgniter'`.

Η εγκατάσταση του framework έχει γίνει με τη βοήθεια του Composer, οπότε στην περίπτωση μας, ο κατάλογος συστήματος βρίσκεται στη διαδρομή `'vendor/codeigniter4/framework/system'`.

4.2.2 Δομή Αρχείων iBoot

Τα αρχεία που έχουν δημιουργηθεί για τη λειτουργία της εφαρμογής και του REST API της, τον καθορισμό της εμφάνισής της, αλλά και της επικοινωνίας της με τη βάση δεδομένων της, περιέχονται στον φάκελο `app` και

Controllers/	Αρχεία ελεγκτών υπεύθυνων για την επιχειρησιακή λογική της εφαρμογής
Api/	Ελεγκτές του REST API της εφαρμογής
ApiLogViewer.php	Η συνάρτηση προβολής και διαγραφής των αρχείων καταγραφής συμβάντων
BootMenu.php	Οι CRUD συναρτήσεις των Μενού Εκκίνησης
BootMenuBlocks.php	Οι CRUD συναρτήσεις των Μπλοκ iPXΕ των Μενού Εκκίνησης
Computer.php	Οι CRUD συναρτήσεις των Υπολογιστών
Group.php	Οι CRUD συναρτήσεις των Ομάδων
IpxeBlock.php	Οι CRUD συναρτήσεις των Μπλοκ iPXΕ
Lab.php	Οι CRUD συναρτήσεις των Εργαστηρίων
Schedule.php	Οι CRUD συναρτήσεις των Χρονοδιαγραμμάτων
Swagger.php	Ο ελεγκτής απεικόνισης της προδιαγραφής του REST API
User.php	Οι CRUD συναρτήσεις των Χρηστών
BaseController.php	Ο βασικός ελεγκτής τον οποίο επεκτείνουν οι υπόλοιποι
BootMenu.php	Ο ελεγκτής προβολής των Μενού Εκκίνησης
Computers.php	Ο ελεγκτής προβολής των Υπολογιστών
Dashboard.php	Ο ελεγκτής προβολής της αρχικής σελίδας
Groups.php	Ο ελεγκτής προβολής των Ομάδων
Home.php	Ο ελεγκτής αρχικής ανακατεύθυνσης και προβολής των σελίδων εκκίνησης
IpxeBlocks.php	Ο ελεγκτής προβολής των Μπλοκ iPXΕ
Labs.php	Ο ελεγκτής προβολής των Εργαστηρίων
Locale.php	Ο ελεγκτής αλλαγής γλώσσας
LogViewer.php	Ο ελεγκτής προβολής των αρχείων καταγραφής συμβάντων
Schedules.php	Ο ελεγκτής προβολής των Χρονοδιαγραμμάτων
User.php	Ο ελεγκτής προβολής των Χρηστών

Σχήμα 4.21: Τα αρχεία του καταλόγου Controllers του iBoot

περιγράφονται αναλυτικά παρακάτω.

Controllers

Στον κατάλογο Controllers βρίσκονται οι ελεγκτές που είναι υπεύθυνοι για την επιχειρησιακή λογική της εφαρμογής. Τα περιεχόμενα του καταλόγου Controllers απεικονίζονται στο σχήμα 4.21.

Database

Στον κατάλογο Database βρίσκονται τα αρχεία υπεύθυνα για τον καθορισμό του σχήματος της βάσης δεδομένων και της αρχικοποίησης δεδομένων, αν αυτό χρειάζεται. Τα περιεχόμενα του καταλόγου Database απεικονίζονται στο σχήμα 4.22.

Database/	Κλάσεις σχεδιασμού και αρχικοποίησης της βάσης δεδομένων
Migrations/	Κλάσεις ενημέρωσης του σχήματος της βάσης δεδομένων
2022-06-27-172100_InitDB.php	Κατασκευάζει το σχήμα της βάσης δεδομένων
Seeds/	Κλάσεις αρχικοποίησης τη βάση δεδομένων με εγγραφές

Σχήμα 4.22: Τα αρχεία του καταλόγου Database του iBoot

Entities/	Κλάσεις οντοτήτων της εφαρμογής
BootMenu.php	Η οντότητα των Μενού Εκκίνησης
BootMenuBlocks.php	Η οντότητα των Μπλοκ iPXΕ των Μενού Εκκίνησης
Computer.php	Η οντότητα των Υπολογιστών
Group.php	Η οντότητα των Ομάδων
IpxeBlock.php	Η οντότητα των Μπλοκ iPXΕ
Lab.php	Η οντότητα των Εργαστηρίων
Schedule.php	Η οντότητα των Χρονοδιαγραμμάτων
User.php	Η οντότητα των Χρηστών

Σχήμα 4.23: Τα αρχεία του καταλόγου Entities του iBoot

Entities

Στον κατάλογο Entities βρίσκονται οι κλάσεις οντοτήτων της εφαρμογής. Οι οντότητες λειτουργούν σαν στιγμιότυπα γραμμών της βάσης δεδομένων, οπότε λειτουργικά παρεμβάλλονται μεταξύ των μοντέλων και της απεικόνισης πληροφορίας στην εφαρμογή. Εκτός από τα δεδομένα που επιστρέφονται από την αντίστοιχη γραμμή της βάσης δεδομένων, μια οντότητα περιέχει και όλες τις απαραίτητες συναρτήσεις για τα περιεχόμενα του καταλόγου Entities απεικονίζονται στο σχήμα 4.23.

Filters

Στον κατάλογο Filters βρίσκονται συναρτήσεις φίλτρων που εκτελούνται ακριβώς πριν ή μετά από εκτέλεση συναρτήσεων ελεγκτών. Αυτές οι συναρτήσεις μπορούν να αλλάξουν τη συμπεριφορά του ελεγκτή, να προσθέσουν ή να επαβεβαιώσουν την ύπαρξη πληροφοριών και να αλλάξουν τη ροή εκτέλεσης εφόσον αυτό είναι απαραίτητο. Τα περιεχόμενα του καταλόγου Filters απεικονίζονται στο σχήμα 4.24.

Language

Στον κατάλογο Language βρίσκεται ένας υποκατάλογος για κάθε υποστηριζόμενη γλώσσα της εφαρμογής. Στον κάθε υποκατάλογο βρίσκονται τα αρχεία που περιέχουν τις μεταφράσεις των λεκτικών για την εκάστοτε γλώσσα. Τα λεκτικά έχουν χωριστεί σε αρχεία βάση της κατηγοριοποίησης τους, δηλαδή βρίσκονται στο αρχείο *Error.php* αν είναι λεκτικά που εμφανίζονται σε μηνύματα σφαλμάτων, στο αρχείο *Validation.php* αν είναι λεκτικά που αφορούν μηνύματα προβλημάτων επαλήθευσης εισόδων από τον χρήστη και στο αρχείο *Text.php* αν είναι λεκτικά γενικού περιεχομένου που εμφανίζονται κατά την περιήγηση του χρήστη στην εφαρμογή. Τα περιεχόμενα

Filters/	Αρχεία φίλτρων υπεύθυνων για αλλαγές στη ροή εκτέλεσης
ApiAuth.php	Φίλτρο για έλεγχο αυθεντικοποίησης αιτήματος προς το REST API
Auth.php	Φίλτρο για έλεγχο αυθεντικοποίησης χρήστη στην εφαρμογή
Locale.php	Φίλτρο για αλλαγή γλώσσας της εφαρμογής
NoAuth.php	Φίλτρο για επιβολή μη αυθεντικοποίησης χρήστη στην εφαρμογή
RefreshUserToken.php	Φίλτρο για αναγέννηση του token αυθεντικοποίησης χρήστη

Σχήμα 4.24: Τα αρχεία του καταλόγου Filters του iBoot

Language/	Αρχεία ελεγκτών υπεύθυνων για την επιχειρησιακή λογική της εφαρμογής
e1/	Οι ελληνικές μεταφράσεις λεκτικών της εφαρμογής
Error.php	Οι ελληνικές μεταφράσεις λεκτικών σφαλμάτων
Text.php	Οι ελληνικές μεταφράσεις λεκτικών γενικού κειμένου
Validation.php	Οι ελληνικές μεταφράσεις λεκτικών επαλήθευσης
en/	Οι αγγλικές μεταφράσεις λεκτικών της εφαρμογής
Error.php	Οι αγγλικές μεταφράσεις λεκτικών σφαλμάτων
Text.php	Οι αγγλικές μεταφράσεις λεκτικών γενικού κειμένου
Validation.php	Οι αγγλικές μεταφράσεις λεκτικών επαλήθευσης

Σχήμα 4.25: Τα αρχεία του καταλόγου Language του iBoot

του καταλόγου Controllers απεικονίζονται στο σχήμα 4.25.

Models

Στον κατάλογο Models βρίσκονται τα μοντέλα που είναι υπεύθυνα για την αλληλεπίδραση της εφαρμογής με πίνακες της βάσης δεδομένων της. Κάθε μοντέλο εξυπηρετεί την αλληλεπίδραση με έναν μόνο πίνακα της βάσης, είτε πρόκειται για ανάγνωση υπαρχόντων είτε για καταχώρηση νέων δεδομένων. Στην περίπτωση της ανάγνωσης δεδομένων, αν το αποτέλεσμα μιας επιλογής είναι μια γραμμή του πίνακα της βάσης, τότε η γραμμή επιστρέφεται ως στιγμιότυπο της αντίστοιχης οντότητας, ενώ αν το αποτέλεσμα είναι περισσότερες γραμμές, τότε επιστρέφονται ως πίνακας οντοτήτων. Τα περιεχόμενα του καταλόγου Controllers απεικονίζονται στο σχήμα 4.26.

Validation

Στον κατάλογο Validation βρίσκονται αρχεία κλάσεων που περιέχουν συναρτήσεις επαλήθευσης δεδομένων. Τα περιεχόμενα του καταλόγου Validation απεικονίζονται στο σχήμα 4.27.

Views

Στον κατάλογο Views βρίσκονται τα αρχεία προβολών που είναι υπεύθυνα για την απεικόνιση σελίδων της εφαρμογής. Περιέχει έναν υποκατάλογο *partials* στον οποίο βρίσκονται τμήματα προβολών, τα οποία μπορούν να ενσωματωθούν σε άλλες προβολές. Η βασική προβολή της εφαρμογής ορίζεται στο αρχείο *template.php* και

Models/	Μοντέλα για την αλληλεπίδραση με πίνακες της βάσης δεδομένων
BootMenuBlocksModel.php	Μοντέλο συνδετικού πίνακα μεταξύ Μενού Εκκίνησης και Μπλοκ iPXE
BootMenuModel.php	Μοντέλο πίνακα των Μενού Εκκίνησης
ComputerGroupsModel.php	Μοντέλο συνδετικού πίνακα μεταξύ Υπολογιστών και Ομάδων
ComputerModel.php	Μοντέλο πίνακα των Υπολογιστών
ForgotPasswordTokenModel.php	Μοντέλο πίνακα των tokens ανάκτησης κωδικού πρόσβασης
GroupModel.php	Μοντέλο πίνακα των Ομάδων
IpxeBlockModel.php	Μοντέλο πίνακα των Μπλοκ iPXE
LabModel.php	Μοντέλο πίνακα των Εργαστηρίων
ScheduleModel.php	Μοντέλο πίνακα των Χρονοδιαγραμμάτων
UserLabsModel.php	Μοντέλο συνδετικού πίνακα μεταξύ Χρηστών και Εργαστηρίων
UserModel.php	Μοντέλο πίνακα των Χρηστών

Σχήμα 4.26: Τα αρχεία του καταλόγου Models του iBoot

Validation/	Αρχεία κανόνων υπεύθυνων για την επαλήθευση εισόδων
ComputerRules.php	Κανόνες μορφής των UUID και MAC Address των υπολογιστών
UserRules.php	Κανόνας για την αυθεντικοποίηση χρηστών

Σχήμα 4.27: Τα αρχεία του καταλόγου Validation του iBoot

εκεί ενσωματώνονται τα τμήματα προβολών από τον κατάλογο *partials*. Η βασική προβολή δε χρησιμοποιείται αυτούσια, αλλά επεκτείνεται από τις υπόλοιπες που προσθέτουν το περιεχόμενο προς εμφάνιση. Τα περιεχόμενα του καταλόγου Views απεικονίζονται στο σχήμα 4.28.

Views/	Αρχεία προβολών υπεύθυνων για την απεικόνιση
partials/	Αρχεία που περιέχουν τμήματα προς ενσωμάτωση σε προβολές
footer.php	Τμήμα προβολής απεικόνισης υποσέλιδου
head.php	Τμήμα προβολής απεικόνισης κεφαλίδας
navbar.php	Τμήμα προβολής απεικόνισης μπάρας μενού πλοήγησης
boot.php	Προβολή ρύθμισης εκκίνησης υπολογιστών
dashboard.php	Προβολή αρχικής σελίδας εφαρμογής
forgotCredentials.php	Προβολή σελίδας υπενθύμισης στοιχείων χρήστη
initboot.php	Προβολή παραμετροποιημένων μενού εκκίνησης υπολογιστών
login.php	Προβολή φόρμας σύνδεσης χρήστη
logs.php	Προβολή απεικόνισης αρχείων καταγραφής συμβάντων
profile.php	Προβολή απεικόνισης στοιχείων προφίλ χρήστη
reissuePassword.php	Προβολή επανέκδοσης κωδικού πρόσβασης χρήστη
signup.php	Προβολή φόρμας εγγραφής χρήστη
swagger.php	Προβολή απεικόνισης προδιαγραφής API
table.php	Προβολή απεικόνισης πινάκων
template.php	Βασική προβολή την οποία επεκτείνουν οι υπόλοιπες

Σχήμα 4.28: Τα αρχεία του καταλόγου Views του iBoot

4.3 Ασφάλεια Συστήματος

Η ασφάλεια των εφαρμογών ιστού είναι ζωτικής σημασίας επειδή οι εφαρμογές ιστού αποτελούν συχνά τον πρωταρχικό στόχο των επιτιθέμενων που επιδιώκουν να αποκτήσουν μη εξουσιοδοτημένη πρόσβαση σε ευαίσθητες πληροφορίες ή να θέσουν σε κίνδυνο τα συστήματα ενός οργανισμού. Για την διασφάλιση της ακεραιότητας της εφαρμογής και των δεδομένων της, την προστασία της ιδιωτικότητας του κάθε χρήστη της και την εξασφάλιση της εύρυθμης και απρόσκοπτης λειτουργίας της, έχουν αξιοποιηθεί οι πρακτικές ασφαλείας που περιγράφονται παρακάτω.

4.3.1 Honey pots

Η κλάση Honeyrot καθιστά δυνατό τον προσδιορισμό του πότε ένα Bot κάνει ένα αίτημα στην διαδικτυακή εφαρμογή. Για να λειτουργεί, θα πρέπει να είναι ενεργοποιημένη στο αρχείο `app/Config/Filters.php`. Η επίτευξη της λειτουργικότητας γίνεται με την επισύναψη πεδίων εισαγωγής δεδομένων σε οποιαδήποτε φόρμα. Αυτά τα πεδία φόρμας είναι κρυμμένα από έναν άνθρωπο αλλά προσβάσιμα σε ένα Bot. Όταν εισάγονται δεδομένα στο πεδίο, θεωρείται ότι το αίτημα προέρχεται από ένα Bot και επιστρέφεται ένα `HoneyrotException`, διακόπτοντας την εκτέλεση του αιτήματος [33].

4.3.2 CSRF

Η κλάση Security περιέχει μεθόδους που βοηθούν στην προστασία διαδικτυακής εφαρμογής από επιθέσεις Cross-Site Request Forgery.

Η πλαστογράφηση αιτήσεων Cross-Site Request Forgery (CSRF) είναι μια επίθεση που αναγκάζει έναν τελικό χρήστη να εκτελέσει ανεπιθύμητες ενέργειες σε μια εφαρμογή ιστού στην οποία έχει πιστοποιηθεί. Με λίγη βοήθεια κοινωνικής μηχανικής (όπως η αποστολή ενός συνδέσμου μέσω ηλεκτρονικού ταχυδρομείου ή συνομιλίας), ένας επιτιθέμενος μπορεί να εξαπατήσει τους χρήστες μιας εφαρμογής ιστού ώστε να εκτελέσουν ενέργειες της επιλογής του επιτιθέμενου. Εάν το θύμα είναι ένας κανονικός χρήστης, μια επιτυχημένη επίθεση CSRF μπορεί να αναγκάσει τον χρήστη να εκτελέσει αιτήματα αλλαγής κατάστασης, όπως μεταφορά χρημάτων, αλλαγή της διεύθυνσης ηλεκτρονικού ταχυδρομείου του κ.ο.κ. Εάν το θύμα είναι ένας διαχειριστικός λογαριασμός, το CSRF μπορεί να θέσει σε κίνδυνο ολόκληρη την εφαρμογή ιστού.

Για την πρόληψη και αντιμετώπιση των επιθέσεων CSRF υλοποιούνται οι παρακάτω μέθοδοι προστασίας και εφαρμόζονται σε όλα τα αιτήματα προς την εφαρμογή, πλην αυτών προς διευθύνσεις του REST API της.

Double Submit Cookie

Το Double Submit Cookie είναι μια τεχνική όπου στέλνουμε μια τυχαία τιμή τόσο σε ένα cookie όσο και ως παράμετρος αιτήματος, με τον διακομιστή να επαληθεύει αν η τιμή του cookie και η τιμή του αιτήματος ταιριάζουν. Όταν ένας χρήστης επισκέπτεται (ακόμη και πριν από την αυθεντικοποίηση για την αποτροπή του CSRF σύνδεσης), ο ιστότοπος παράγει μια κρυπτογραφικά ισχυρή τυχαία τιμή και τη θέτει ως cookie στον υπολογιστή του χρήστη ξεχωριστά από το αναγνωριστικό συνεδρίας. Στη συνέχεια, ο ιστότοπος απαιτεί κάθε αίτημα συναλλαγής να περιλαμβάνει αυτή την τυχαία τιμή ως κρυφή τιμή φόρμας ή στην επικεφαλίδα του αιτήματος. Εάν και οι δύο ταιριάζουν στην πλευρά του διακομιστή, ο διακομιστής το δέχεται ως νόμιμο αίτημα, ενώ εάν δεν ταιριάζουν, θα απορρίψει το αίτημα.

Με λίγα λόγια, ένας εισβολέας δεν μπορεί να έχει πρόσβαση στην τιμή του cookie κατά τη διάρκεια ενός cross-site αιτήματος. Αυτό τους εμποδίζει να συμπεριλάβουν μια τιμή που ταιριάζει στην κρυφή τιμή της φόρμας ή ως παράμετρος/κεφαλίδα αιτήματος.

Τυχοποίηση token

Για να μετριαστούν οι επιθέσεις πλευρικού καναλιού συμπίεσης όπως το BREACH και να αποτραπεί η προσπάθεια ενός εισβολέα από το να μαντέψει τα διακριτικά CSRF, είναι ενεργοποιημένη η τυχοποίηση των διακριτικών. Έτσι, μια τυχαία μάσκα προστίθεται στο κάθε token και χρησιμοποιείται για την κρυπτογράφησή του.

Αναγέννηση Token

Τα tokens αναδημιουργούνται σε κάθε υποβολή του CSRF cookie. Η αναγέννηση των tokens παρέχει αυστηρότερη ασφάλεια, αλλά μπορεί να οδηγήσει σε μειωμένη ευχρηστία, καθώς άλλα tokens που καθίστανται άκυρα μπορεί να χρειαστεί να χρησιμοποιηθούν εκ νέου σε περιπτώσεις όπως πλοήγηση προς τα πίσω/εμπρός, πολλαπλές καρτέλες/παράθυρα, ασύγχρονες ενέργειες κ.λπ. Το παραπάνω αρνητικό δε θεωρείται μεγάλης σημαντικότητας σε σχέση με την αυξημένη ασφάλεια που προσφέρει το μέτρο της αναγέννησης των tokens.

4.3.3 JWT

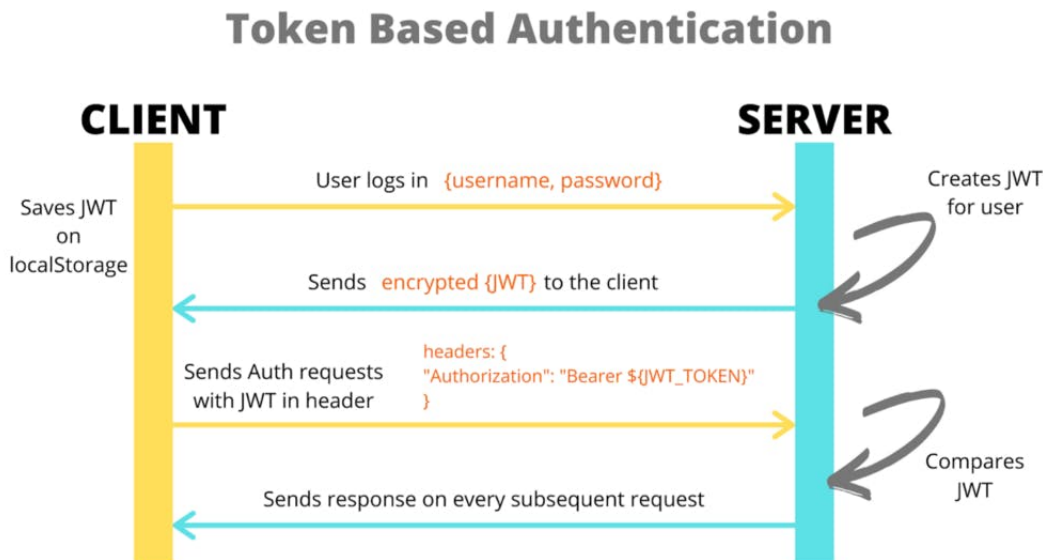
Η αυθεντικοποίηση στο REST API της εφαρμογής, έχει υλοποιηθεί με χρήση JSON Web Tokens (JWT).

Λειτουργία JWT

Αρχικά, ο χρήστης ή η εφαρμογή-πελάτης στέλνει ένα αίτημα σύνδεσης. Σε αυτό το βήμα, ουσιαστικά, ένα όνομα χρήστη, ένας κωδικός πρόσβασης ή οποιοσδήποτε άλλος τύπος διαπιστευτηρίων σύνδεσης που παρέχει ο χρήστης θα ταξιδέψει στο API. Μόλις επαληθευτεί, το API θα δημιουργήσει ένα JSON Web Token και θα το υπογράψει χρησιμοποιώντας ένα μυστικό κλειδί. Στη συνέχεια, το API θα επιστρέψει αυτό το token πίσω στην εφαρμογή-πελάτη.

Τέλος, η εφαρμογή-πελάτης θα λάβει το token, θα το επαληθεύσει στη δική της πλευρά για να διασφαλίσει ότι είναι αυθεντικό και στη συνέχεια θα το χρησιμοποιήσει σε κάθε επόμενο αίτημα. Ως εκ τούτου, μπορεί να πιστοποιήσει τον χρήστη χωρίς να χρειάζεται να στείλει πλέον τα διαπιστευτήριά του.

Η διαδικασία έκδοσης και χρήσης JWT token απεικονίζεται στο σχήμα 4.29.



Σχήμα 4.29: Λειτουργία JWT Πηγή: [Gha20a]

Δομή JWT

Το ίδιο το token, το οποίο επιστρέφεται από το API, είναι απλώς μια κωδικοποιημένη συμβολοσειρά. Αποτελείται από τρία διαφορετικά τμήματα, τα οποία διαχωρίζονται μεταξύ τους με έναν χαρακτήρα τελείας. Αυτά είναι το header, το payload και το signature.

Το header περιέχει δεδομένα σχετικά με τον τύπο του κουπονιού και τον αλγόριθμο που χρησιμοποιείται για τη δημιουργία του.

Το payload περιέχει δεδομένα σχετικά με το αίτημα και τον χρήστη που την υπέβαλε. Υπάρχει ένα σύνολο τυποποιημένων ζευγών κλειδιών/τιμών που ορίζονται ως μέρος του JWT, τα οποία μπορούν να χρησιμοποιηθούν:

- Sub (Subject): Προσδιορίζει τον χρήστη που κάνει το αίτημα και πιστοποιείται.

- Iss (Issuer): Ο διακομιστής που εξέδωσε το token. Στην περίπτωση μας, θα είχε νόημα να συμπεριλάβουμε το URI που χρησιμοποιείται
- Aud (Audience): Παρέχει κάποια μορφή ταυτοποίησης του παραλήπτη αυτού του κουπονιού.
- Exp (Ημερομηνία λήξης): Τα κουπόνια συνήθως δεν διαρκούν για πάντα. Η Exp εξασφαλίζει ότι όποιος χρησιμοποιεί το token παρέχει ένα πρόσφατα δημιουργημένο token.

Επίσης, είναι δυνατόν να καθοριστούν επιπρόσθετα ζεύγη κλειδιών/τιμών για την κάλυψη των αναγκών της υλοποίησης.

Τέλος, το signature είναι απλώς μια κωδικοποιημένη συμβολοσειρά που χρησιμοποιείται τόσο από τον διακομιστή όσο και από τον πελάτη για την επαλήθευση της αυθεντικότητας του payload.

Η αποκωδικοποιημένη δομή ενός παραδείγματος JWT token απεικονίζεται στο σχήμα 4.30.

The image shows a screenshot of the JWT.io website. At the top, there's a navigation bar with the JWT logo, a 'Debugger' button, and links for 'Libraries', 'Ask', and 'Get a T-shirt!'. Below the navigation bar, there's a dropdown menu for 'ALGORITHM' set to 'HS256'. The main content area is split into two columns: 'Encoded' and 'Decoded'. The 'Encoded' column shows the JWT token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoiYXZgeFONFh7HgQ`. The 'Decoded' column shows the token's structure: 'HEADER:' with `{ "alg": "HS256", "typ": "JWT" }`; 'PAYLOAD:' with `{ "sub": "1234567890", "name": "John Doe", "admin": true }`; and 'VERIFY SIGNATURE' with the formula `HMACSHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload), secret)` and a checked box for 'secret base64 encoded'. At the bottom, a blue banner with a checkmark icon says 'Signature Verified'.

Σχήμα 4.30: Δομή JWT Πηγή: [Gha20b]

4.3.4 Αντιμετώπιση SQL Injections

Τι είναι SQL Injection;

Οι επιθέσεις τύπου SQL Injection προσπαθούν να εκμεταλλευτούν μια ευπάθεια στην ασφάλεια μιας διαδικτυακής εφαρμογής, η οποία επιτρέπει σε έναν εισβολέα να παρεμβαίνει στα ερωτήματα που κάνει η εφαρμογή στη βάση δεδομένων της. Αν υπάρχει ευπάθεια και ο εισβολέας καταφέρει να την εκμεταλλευτεί, μπορεί να αποκτήσει πρόσβαση σε δεδομένα που κανονικά δεν θα έπρεπε να είναι σε θέση να ανακτήσει ή ακόμη και να προκαλέσει καταστροφή της βάσης και απώλεια δεδομένων. Η αντιμετώπιση τέτοιων ευπαθειών είναι μείζονος σημασίας στις διαδικτυακές εφαρμογές, αφού οι επιθέσεις SQL Injection αποτελούν έναν από τους πιο διαδεδομένους τρόπους επίθεσης, είτε από μόνες τους, είτε σε συνδυασμό με άλλου τύπου επιθέσεις όπως DDoS attacks, DNS hijacking και Cross-Site Scripting (XSS).

Prepared SQL Statements

Οι προμεταγλωττισμένες εντολές SQL βοηθούν στην αντιμετώπιση των επιθέσεων τύπου SQL Injection. Με τη χρήση προμεταγλωττισμένων εντολών, οι τιμές των μεταβλητών που λαμβάνονται από τον χρήστη, αφού περάσουν από φίλτρο αντικατάστασης χαρακτήρων με ειδική σημασία στην SQL, ενσωματώνονται στην προμεταγλωττισμένη εντολή για να συνθέσουν το τελικό query που θα εκτελεστεί στη βάση δεδομένων. Έτσι, ενδεχόμενες κακόβουλες υποβολές τιμών από χρήστες, δε φτάνουν στο να εκτελεστούν στη βάση δεδομένων και να προκαλέσουν απώλεια ή υποκλοπή πληροφοριών.

Ληφθέντα μέτρα αντιμετώπισης SQL Injections

Η κλάση Query Builder του CodeIgniter4 [34] έχει χρησιμοποιηθεί για την επικοινωνία της εφαρμογής με τη βάση δεδομένων της. Οι συναρτήσεις, που παρέχονται από τη συγκεκριμένη κλάση, καλούνται διαδοχικά και κάθε μια, με την εκτέλεσή της, προσθέτει από μια παράμετρο ώστε να συντεθεί το τελικό SQL Query προς εκτέλεση. Δέχονται ως ορίσματα τις απαραίτητες μεταβλητές, στις οποίες πραγματοποιούν αυτόματα την αντικατάσταση των ειδικών χαρακτήρων. Όταν κληθεί μια συνάρτησης ανάκτησης πληροφορίας από τη βάση, οι μεταβλητές ενσωματώνονται στο prepared query που έχει συνθέσει η κλήση των προηγούμενων συναρτήσεων, το ερώτημα εκτελείται και επιστρέφεται το αποτέλεσμα.

Για επιπρόσθετη ασφάλεια, καθώς η αντικατάσταση ειδικών χαρακτήρων δεν επαρκεί από μόνη της, αλλά και για κανονικοποίηση και επαλήθευση των τιμών των μεταβλητών του κάθε Μοντέλου, πριν ακόμη αυτές εισαχθούν στη βάση δεδομένων, έχει χρησιμοποιηθεί επίσης η βιβλιοθήκη Validation [35]. Στο κάθε Μοντέλο διατηρείται ένας πίνακας, ο οποίος περιέχει τα απαραίτητα φίλτρα επαλήθευσης για την κάθε τιμή μεταβλητής που δόθηκε από τον χρήστη ως είσοδος. Ο έλεγχος πραγματοποιείται πριν την εισαγωγή των δεδομένων στη βάση, η οποία πραγματοποιείται μόνο αν επικυρωθεί ότι οι τιμές που εξετάζονται είναι στη σωστή μορφή και τύπο.

Για παράδειγμα, έχει οριστεί ότι το αναγνωριστικό (ID) κάθε γραμμής ενός πίνακα, είναι ένας θετικός ακέραιος αριθμός. Αν ο χρήστης δοκιμάσει να εισάγει ως αναγνωριστικό κάποια είσοδο που δεν ικανοποιεί τον παραπάνω κανόνα, όπως κάποιο χαρακτήρα, η είσοδος θα απορριφθεί κατά τον έλεγχο εγκυρότητάς της. Ομοίως, δε θα γίνει αποδεκτή η είσοδος, αν στο πεδίο του email δεν είναι έγκυρο email ή αν το email που δόθηκε υπάρχει ήδη στον πίνακα, αφού έχει οριστεί και ως μοναδικό.

Ο συνδυασμός των παραπάνω πρακτικών, δηλαδή του ελέγχου εγκυρότητας των εισόδων και της ενσωμάτωσής τους σε prepared SQL statements, αποτελεί μια πολύ ισχυρή άμυνα ενάντια σε επιθέσεις τύπου SQL Injection.

4.4 Σύνοψη Κεφαλαίου 4

Στο κεφάλαιο 4, εξετάστηκαν όλες οι δυνατότητες της διαδικτυακής εφαρμογής και τα βήματα που πρέπει να ακολουθήσει ένας νέος χρήστης για να τις χρησιμοποιήσει. Επιπλέον, παρασχέθηκε οπτικό υλικό για κάθε λειτουργία στην οποία εμφανιζόταν στον χρήστη η διεπαφή χρήστη της διαδικτυακής εφαρμογής κατά την πλοήγησή του. Στη συνέχεια, εξηγήθηκαν διεξοδικά τόσο το front-end όσο και το back-end της δομής του πληροφοριακού συστήματος. Προκειμένου να διασφαλιστεί η ασφάλεια κατά τη χρήση της διαδικτυακής εφαρμογής, παρέχεται επεξήγηση των πρωτοκόλλων και των προσεγγίσεων ασφαλείας.

Στο επόμενο κεφάλαιο θα γίνει αξιολόγηση του συστήματος, ως προς την ορθή λειτουργία του και τις δυνατότητες κλιμάκωσής του.

Κεφάλαιο 5

Αξιολόγηση Συστήματος

Σε αυτό το κεφάλαιο, αναλύονται λεπτομερώς οι δοκιμές που έγιναν για την εξακρίβωση της ορθής λειτουργίας της δικτυακής εφαρμογής. Επίσης, καταγράφεται το πλάνο ελέγχου ορθής λειτουργίας της πλατφόρμας και γίνεται μελέτη κλιμάκωσης του συστήματος.

5.1 Δοκιμή Πλατφόρμας

5.1.1 Οι δοκιμές στην ανάπτυξη λογισμικού

Η δοκιμή λογισμικού είναι η διαδικασία αντικειμενικής αξιολόγησης και επαλήθευσης της λειτουργικότητας μιας εφαρμογής λογισμικού για τον εντοπισμό πιθανών σφαλμάτων ή κακής συμπεριφοράς. Προσδιορίζει αν το λογισμικό που αναπτύχθηκε ευθυγραμμίζεται με τις καθορισμένες απαιτήσεις και εντοπίζει τυχόν ελαττώματα στο λογισμικό, ώστε να διασφαλίζεται η παραγωγή ενός προϊόντος υψηλής ποιότητας. Εντοπίζει τυχόν σφάλματα, κενά, παραλείψεις ή ελλείψεις απαιτήσεων που δεν ευθυγραμμίζονται με τις απαραίτητες.

Η δοκιμή λογισμικού περιλαμβάνει την εκτέλεση ενός συστατικού λογισμικού ή συστήματος για την αξιολόγηση μιας ή περισσότερων ιδιοτήτων ενδιαφέροντος. Μπορεί να εκτελεστεί είτε χειροκίνητα είτε με τη χρήση αυτοματοποιημένων εργαλείων.

Δεδομένου ότι υπάρχει σχεδόν άπειρος αριθμός πιθανών δοκιμών ακόμη και για απλά στοιχεία λογισμικού, όλες οι δοκιμές λογισμικού χρησιμοποιούν κάποια στρατηγική για την επιλογή δοκιμών που είναι κατάλληλες για τον διαθέσιμο χρόνο και τους διαθέσιμους πόρους.

Η διαδικασία των δοκιμών είναι επαναληπτική, πράγμα που σημαίνει ότι η διόρθωση ενός σφάλματος μπορεί να αποκαλύψει άλλα που είναι βαθύτερα ή να δημιουργήσει νέα.

5.1.2 Οφέλη των δοκιμών λογισμικού

Ακολουθούν ορισμένα από τα οφέλη των δοκιμών λογισμικού:

- **Εξοικονόμηση πόρων:** Ο έγκαιρος προσδιορισμός των αστοχιών και ελλείψεων του λογισμικού οδηγεί στη γρήγορη διόρθωσή του, κάτι που εξοικονομεί χρόνο ανάπτυξης.
- **Ασφάλεια:** Οι δοκιμές ασφαλείας χρησιμοποιούνται για τον προσδιορισμό του επιπέδου ασφαλείας μιας εφαρμογής και οι δοκιμαστές προσπαθούν να εντοπίσουν τυχόν ευπάθειες που ενδέχεται να θέσουν σε κίνδυνο την ασφάλειά της. Ο στόχος είναι να εντοπιστούν και να εξαλειφθούν τυχόν ελαττώματα ή κενά ασφαλείας της εφαρμογής.

-
- **Διασφάλιση ποιότητας:** Η ποιότητα ενός προϊόντος μπορεί να διατηρηθεί μόνο όταν είναι απαλλαγμένο από σφάλματα και ανταποκρίνεται σε όλες τις απαιτήσεις των χρηστών.
 - **Επιτάχυνση ανάπτυξης:** Οι δοκιμές λογισμικού επιταχύνουν τη διαδικασία ανάπτυξης με τον εντοπισμό ελαττωμάτων. Η ανίχνευση σε πρώιμο στάδιο διευκολύνει τη διόρθωση χωρίς να επηρεάζονται αρνητικά άλλες λειτουργίες του συστήματος.

5.1.3 Τύποι δοκιμών λογισμικού

Υπάρχουν διάφοροι τύποι δοκιμών λογισμικού, καθένας από τους οποίους έχει μοναδικούς στόχους και στρατηγικές.

- **Αποδοχής:** Επαληθεύουν αν το σύστημα λειτουργεί όπως προβλέπεται.
- **Ολοκλήρωσης:** Εξετάζουν τη δυνατότητα των διάφορων μονάδων του συστήματος να λειτουργούν μαζί.
- **Μονάδας:** Επικυρώνουν ότι κάθε μονάδα λογισμικού λειτουργεί όπως αναμένεται. Μια μονάδα είναι το μικρότερο ελέγξιμο συστατικό μιας εφαρμογής.
- **Λειτουργική:** Ελέγχουν τις επιχειρηματικές λειτουργίες, με βάση τις λειτουργικές απαιτήσεις.
- **Επιδόσεων:** Διασφαλίζουν ότι το λογισμικό πληροί τις απαιτήσεις επιδόσεων, όπως ο χρόνος απόκρισης και η χρήση μνήμης.
- **Καταπόνησης:** Δοκιμάζουν τη μέγιστη καταπόνηση που μπορεί να αντέξει το σύστημα πριν αποτύχει.
- **Ευχρηστίας:** Επικυρώνουν την ικανότητα του χρήστη να χρησιμοποιεί αποτελεσματικά ένα σύστημα ή μια εφαρμογή.

Η επικύρωση των βασικών απαιτήσεων αποτελεί κρίσιμο κομμάτι της αξιολόγησης ενός προϊόντος λογισμικού. Εξίσου σημαντική, η διερευνητική δοκιμή βοηθά στην αποκάλυψη απρόβλεπτων σεναρίων που μπορεί να οδηγήσουν σε σφάλματα λογισμικού.

5.2 Πλάνο Ελέγχου Ορθής Λειτουργίας

Για το πλάνο ελέγχου ορθής λειτουργίας της πλατφόρμας, θα πρέπει να εξακριβωθεί πως ικανοποιούνται οι περιπτώσεις χρήσεις που έχουν οριστεί στην ενότητα 3.2.

Ονομαστικά, οι έλεγχοι που πραγματοποιήθηκαν είναι οι παρακάτω:

- Έλεγχοι Βασικών Λειτουργιών Χρηστών
- Έλεγχοι Διαχείρισης Εργασιών
- Έλεγχοι Διαχείρισης Ομάδων Υπολογιστών
- Έλεγχοι Διαχείρισης Υπολογιστών
- Έλεγχοι Διαχείρισης Ονοματισμένων Block Εντολών Τύπου iPXE

- Έλεγχοι Διαχείρισης Μενού Εκκίνησης
- Έλεγχοι Χρονοδιαγραμμάτων
- Έλεγχοι Αρχείων Καταγραφής Συμβάντων

Για την πραγματοποίηση των ελέγχων ορθής χρήσης κάθε λειτουργίας που περιγράφεται στις περιπτώσεις χρήσης της πλατφόρμας, έγινε μια εγκατάσταση της εφαρμογής σε δοκιμαστικό περιβάλλον, το οποίο όμως σχεδιάστηκε ώστε να πληρεί όλες τις προδιαγραφές παραγωγικής λειτουργίας, όπως για παράδειγμα ρύθμιση HTTPS, αποστολής email κλπ. Στη συνέχεια δημιουργήθηκαν οι κατάλληλοι χειριστές και περάστηκαν δοκιμαστικά στοιχεία στη βάση δεδομένων.

Στους πίνακες που ακολουθούν παρουσιάζονται οι έλεγχοι ορθής χρήσης που πραγματοποιήθηκαν για κάθε λειτουργία της πλατφόρμας, καθώς και τα αποτελέσματά τους.

5.2.1 Βασικές Λειτουργίες Χρηστών

Πίνακας 5.1: Έλεγχοι Βασικών Λειτουργιών Χρηστών

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Εγγραφή χρήστη στο σύστημα	Ανώνυμος Χρήστης	✓
Σύνδεση χρήστη στην πλατφόρμα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Επανεκδοση κωδικού χρήστη	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Υπενθύμιση Ονόματος Χρήστη	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Αποσύνδεση από το σύστημα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Αλλαγή στοιχείων χρήστη	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή χρήστη από το σύστημα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

5.2.2 Διαχείριση Εργαστηρίων

Πίνακας 5.2: Έλεγχοι Διαχείρισης Εργαστηρίων

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Εισαγωγή εργαστηρίου στο σύστημα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή εργαστηρίου από το σύστημα	Γενικός Διαχειριστής Πλατφόρμας	✓
Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων	Γενικός Διαχειριστής Πλατφόρμας	✓

5.2.3 Διαχείριση Ομάδων Υπολογιστών

Πίνακας 5.3: Έλεγχοι Διαχείρισης Ομάδων Υπολογιστών

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Δημιουργία ομάδας υπολογιστών	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή ομάδας υπολογιστών	Γενικός Διαχειριστής Πλατφόρμας	✓
Επεξεργασία αναθέσεων Διαχειριστών Εργαστηρίων	Γενικός Διαχειριστής Πλατφόρμας	✓

5.2.4 Διαχείριση Υπολογιστών

Πίνακας 5.4: Έλεγχοι Διαχείρισης Υπολογιστών

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Εισαγωγή υπολογιστή στο σύστημα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Ανάθεση υπολογιστή σε εργαστήριο	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

Πίνακας 5.4: Έλεγχοι Διαχείρισης Υπολογιστών (συνέχεια)

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Αφαίρεση υπολογιστή από εργαστήριο	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Επεξεργασία στοιχείων υπολογιστή	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή υπολογιστή από το σύστημα	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

5.2.5 Διαχείριση Ονοματισμένων Block Εντολών Τύπου iPXE

Πίνακας 5.5: Έλεγχοι Διαχείρισης Ονοματισμένων Block Εντολών Τύπου iPXE

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Εισαγωγή block εντολών τύπου iPXE	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Επεξεργασία στοιχείων block εντολών τύπου iPXE	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή block εντολών τύπου iPXE	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

5.2.6 Διαχείριση Μενού Εκκίνησης

Πίνακας 5.6: Έλεγχοι Διαχείρισης Μενού Εκκίνησης

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Δημιουργία μενού εκκίνησης	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Επεξεργασία στοιχείων μενού εκκίνησης	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

Πίνακας 5.6: Έλεγχοι Διαχείρισης Μενού Εκκίνησης (συνέχεια)

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Διαγραφή μενού εκκίνησης	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

5.2.7 Διαχείριση Χρονοδιαγραμμάτων

Πίνακας 5.7: Έλεγχοι Χρονοδιαγραμμάτων

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Δημιουργία χρονοδιαγράμματος	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Επεξεργασία στοιχείων χρονοδιαγράμματος	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓
Διαγραφή χρονοδιαγράμματος	Γενικός Διαχειριστής Πλατφόρμας, Διαχειριστής Εργαστηρίου	✓

5.2.8 Διαχείριση Αρχείων Καταγραφής Συμβάντων

Πίνακας 5.8: Έλεγχοι Αρχείων Καταγραφής Συμβάντων

Περιγραφή Ελέγχου	Χειριστές	Αποτέλεσμα
Δημιουργία αρχείων καταγραφής συμβάντων	Γενικός Διαχειριστής Πλατφόρμας	✓
Διαγραφή αρχείων καταγραφής συμβάντων	Γενικός Διαχειριστής Πλατφόρμας	✓

5.3 Μελέτη Κλιμάκωσης

Τελικό στάδιο αξιολόγησης της πλατφόρμας που αναπτύχθηκε, ήταν η δοκιμή της σε πραγματικές συνθήκες και με μεγάλο όγκο δεδομένων.

Αποφασίστηκε η εφαρμογή να χρησιμοποιηθεί για την κάλυψη των αναγκών ενός workshop, στο οποίο οι 50 συμμετέχοντες θα έπρεπε να έχουν πανομοιότυπο περιβάλλον εργασίας με προεγκατεστημένα και προρυθμισμένα εργαλεία. Για το σκοπό αυτό, δημιουργήθηκε μια παραμετροποιημένη εικόνα του λειτουργικού συστήματος Linux Mint, στην οποία είχαν γίνει όλες οι απαραίτητες εγκαταστάσεις και ρυθμίσεις. Η εικόνα τοποθετήθηκε σε έναν

TFTP server, στο τοπικό δίκτυο του εργαστηρίου όπου έλαβε χώρα το workshop. Ο DHCP server του εργαστηρίου ρυθμίστηκε επίσης ώστε να παραπέμπει τους υπολογιστές που εκκινούνταν, στην παραγωγική εγκατάσταση της εφαρμογής.

Νωρίτερα από το workshop είχε γίνει μια εκκίνηση του κάθε υπολογιστή, ώστε τα στοιχεία του (UUID και διεύθυνση MAC) να αποθηκευτούν στην πλατφόρμα και να είναι έτοιμοι για ανάθεση σε ομάδες. Δημιουργήθηκε μια ομάδα με την κατάλληλη διαδρομή και πρόθεμα για τον TFTP server του εργαστηρίου, στην οποία και έγινε ανάθεση όλων των υπολογιστών. Επίσης, δημιουργήθηκε μια εγγραφή μπλοκ εντολών τύπου iPXE, κατάλληλου για την τροποποιημένη εικόνα λειτουργικού συστήματος με την οποία έπρεπε να εκκινηθούν οι υπολογιστές, καθώς και ένα μενού εκκίνησης, το οποίο περιείχε το συγκεκριμένο μπλοκ εντολών. Τέλος, δημιουργήθηκε ένα χρονοδιάγραμμα για την ημερομηνία και ώρα του εργαστηρίου, στο οποίο καθοριζόταν πως στους υπολογιστές της ομάδας θα έπρεπε να σταλεί το συγκεκριμένο μενού εκκίνησης.

Τα αποτελέσματα ήταν πολύ θετικά, με τους υπολογιστές να εκκινούνται σωστά και χωρίς καθυστερήσεις. Αυτό απέδειξε έμπρακτα ότι η εφαρμογή είναι όχι μόνο λειτουργική, αλλά και αποδοτική σε μεγάλο όγκο δεδομένων.

5.4 Σύνοψη Κεφαλαίου 5

Στο κεφάλαιο 5, έγινε αναφορά στην αξιολόγηση του συστήματος μετά τις απαραίτητες δοκιμές. Σκοπός της αξιολόγησης του συστήματος είναι η εξαγωγή συμπερασμάτων σχετικά με την ορθή λειτουργία της πλατφόρμας και τις δυνατότητες επέκτασης και κλιμάκωσής της.

Τα συμπεράσματα, τα οφέλη και οι συστάσεις για μελλοντικές βελτιώσεις, όπως και η συνολική επισκόπηση της διπλωματικής εργασίας, παρατίθενται στο επόμενο κεφάλαιο.

Κεφάλαιο 6

Επίλογος

Στο τελευταίο κεφάλαιο, πραγματοποιείται συνοπτική παρουσίαση των θεμάτων που αναλύθηκαν έως τώρα. Στη συνέχεια παρουσιάζονται τα συμπεράσματα που προέκυψαν κατά τη διαδικασία σχεδίασης και κατασκευής της διαδικτυακής εφαρμογής. Επιπλέον, γίνεται λεπτομερής ανάλυση SWOT (Strengths, Weaknesses, Opportunities, Threats) του συστήματος, καθώς και προτάσεις για μελλοντικές βελτιστοποιήσεις και επεκτάσεις των λειτουργιών που παρέχει.

6.1 Ανακεφαλαίωση Διπλωματικής Εργασίας

Οι ανάγκες δικτυακής εκκίνησης υπολογιστών του Εργαστηρίου Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας, απαιτούσαν μια διαδικτυακή εφαρμογή με συγκεκριμένα χαρακτηριστικά και δυνατότητες που δεν ήταν διαθέσιμες σε εμπορικές λύσεις. Για την κάλυψη αυτών των αναγκών, αναπτύχθηκε η διαδικτυακή εφαρμογή 'iBoot'.

Η εφαρμογή 'iBoot' αναπτύχθηκε χρησιμοποιώντας PHP στο back-end και HTML, CSS, JavaScript και Bootstrap στο front-end. Τα δεδομένα της εφαρμογής αποθηκεύονται σε μια σχεσιακή βάση δεδομένων MySQL. Η πρόσβαση στην εφαρμογή μπορεί να γίνει είτε μέσω μιας διεπαφής χρήστη είτε μέσω ενός REST API που αναπτύχθηκε για αυτόν το σκοπό. Ο πηγαίος κώδικας της εφαρμογής είναι δημόσια διαθέσιμος και διαμοιράζεται υπό τους όρους της ελεύθερης άδειας χρήσης λογισμικού MIT.

Η εφαρμογή διευκολύνει την πρόσβαση σε τρεις τύπους χρηστών. Ένας ανώνυμος χρήστης έχει πρόσβαση μόνο στις σελίδες σύνδεσης, εγγραφής και υπενθύμισης των διαπιστευτηρίων σύνδεσης. Ο διαχειριστής εργαστηρίου μπορεί να διαχειρίζεται υπολογιστές σε συγκεκριμένα εργαστήρια. Μπορούν επίσης να τοποθετηθούν υπολογιστές σε εργαστήρια που διαχειρίζεται ο διαχειριστής και οι οποίοι δεν έχουν ήδη τοποθετηθεί σε άλλα εργαστήρια. Ο διαχειριστής μπορεί να χρησιμοποιεί όλες τις δυνατότητες της πλατφόρμας χωρίς περιορισμούς. Ο διαχειριστής έχει την εξουσία να προσθέτει, να αφαιρεί και να τροποποιεί υπολογιστές, ομάδες, εργαστήρια, μπλοκ iPXE, μενού εκκίνησης, χρονοδιαγράμματα και χρήστες. Ο Διαχειριστής επιτρέπεται επίσης να βλέπει τα αρχεία καταγραφής συμβάντων της εφαρμογής. Οι υπολογιστές επικοινωνούν με συγκεκριμένες σελίδες της εφαρμογής για να εγγραφούν και να λάβουν μενού εκκίνησης.

Η εφαρμογή έχει σχεδιαστεί και υλοποιηθεί σύμφωνα με τις βέλτιστες πρακτικές και τα πιο πρόσφατα πρότυπα ασφαλείας για την προστασία των πληροφοριών που περιέχονται σε αυτήν. Οι κωδικοί πρόσβασης των χρηστών αποθηκεύονται στη βάση δεδομένων της εφαρμογής αφού υποστούν επεξεργασία μέσω κρυπτογραφικά ασφαλών συναρτήσεων κατακερματισμού και δεν εμφανίζονται ποτέ στην εφαρμογή. Η πιστοποίηση ταυτότητας και τα

δικαιώματα του χρήστη που επιθυμεί να προβάλει τη σελίδα ελέγχονται πριν από τη φόρτωση, ενώ το API υλοποιεί την ίδια διαδικασία προσθέτοντας σε κάθε αίτηση JWT tokens. Η εφαρμογή είναι σε θέση να λειτουργεί τόσο με το πρωτόκολλο HTTP όσο και με το πιο ασφαλές πρωτόκολλο HTTPS και μπορεί ακόμη και να κρυπτογραφήσει τα cookies συνόδου.

Το Εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του Πανεπιστημίου Δυτικής Μακεδονίας θα εκσυγχρονίσει τη διαδικασία διαχείρισης του μενού εκκίνησης του δικτύου των υπολογιστών του με την εγκατάσταση της διαδικτυακής εφαρμογής “iBoot”, η οποία θα διευκολύνει επίσης σημαντικά την καταγραφή και την παρακολούθηση των υπολογιστών αυτών.

6.2 Μοντέλο Ανάλυσης S.W.O.T.

6.2.1 Ορισμός Μοντέλου Ανάλυσης S.W.O.T.

Η ανάλυση SWOT, ή πίνακας SWOT, είναι ένα εργαλείο στρατηγικού σχεδιασμού και διαχείρισης που χρησιμοποιείται για να επιτρέψει σε άτομα ή οργανισμούς να εντοπίσουν τα δυνατά σημεία, τις αδυναμίες, τις ευκαιρίες και τις απειλές που σχετίζονται με τον επιχειρηματικό ανταγωνισμό ή τον προγραμματισμό έργων.

Η τεχνική αυτή προορίζεται για χρήση κατά τα προκαταρκτικά στάδια των διαδικασιών λήψης αποφάσεων για την αξιολόγηση της στρατηγικής θέσης. Σκοπός της είναι να εντοπίσει τόσο εσωτερικούς όσο και εξωτερικούς παράγοντες που είναι είτε ευνοϊκοί είτε δυσμενείς για την επίτευξη των στόχων του έργου ή της πρωτοβουλίας.

Ο όρος αποτελεί ακρωνύμιο των τεσσάρων στοιχείων τα οποία αξιολογεί η μέθοδος:

- **Δυνάμεις:** χαρακτηριστικά της επιχείρησης ή του έργου που παρέχουν πλεονέκτημα έναντι του ανταγωνισμού
- **Αδυναμίες:** χαρακτηριστικά που θέτουν την επιχείρηση ή το έργο σε μειονεκτική θέση σε σχέση με τον ανταγωνισμό
- **Ευκαιρίες:** παράγοντες του περιβάλλοντος που μπορεί να εκμεταλλευτεί η επιχείρηση ή το έργο
- **Απειλές:** παράγοντες του περιβάλλοντος που θα μπορούσαν να αποτελέσουν προκλήσεις για την επιχείρηση ή το έργο

Τα αποτελέσματα της αξιολόγησης εμφανίζονται συνήθως είτε σε μορφή πίνακα είτε σε παραγράφοις.

Εσωτερικοί - Εξωτερικοί Παράγοντες

Οι εσωτερικές πτυχές μιας επιχείρησης θεωρούνται συνήθως ως πλεονεκτήματα ή αδυναμίες, ενώ οι ευκαιρίες ή οι απειλές θεωρούνται ως εξωτερικές.

Οι εσωτερικοί παράγοντες ενός οργανισμού αξιολογούνται ως πλεονεκτήματα ή αδυναμίες, με βάση τον αντίκτυπό τους στους στόχους της επιχείρησης. Κάτι που θα μπορούσε να θεωρηθεί ως πλεονέκτημα για έναν στόχο μπορεί να θεωρηθεί ως αδυναμία για έναν άλλο, για παράδειγμα, ο ανταγωνισμός ή οι περισπασμοί. Οι παράγοντες μπορεί να περιλαμβάνουν το προσωπικό, τη χρηματοδότηση, τις δυνατότητες παραγωγής και όλα τα 4P του μείγματος μάρκετινγκ (προϊόν, τιμή, θέση και προώθηση).

Οι εξωτερικοί παράγοντες περιέχουν κοινωνικοπολιτισμικές αλλαγές, μακροοικονομικές εξελίξεις, νομοθετικές εξελίξεις, τεχνολογικές εξελίξεις, καθώς και μετατοπίσεις στην αγορά.

6.2.2 Ανάλυση S.W.O.T. εφαρμογής iBoot

Στο σχήμα 6.1 απεικονίζεται η ανάλυση S.W.O.T. της διαδικτυακής εφαρμογής iBoot. Τα ευρήματα της ανάλυσης αναλύονται στη συνέχεια, το καθένα στην αντίστοιχη υποενότητα.



Σχήμα 6.1: Ανάλυση SWOT εφαρμογής iBoot

Δυνάμεις

- **Προσαρμοσμένη εφαρμογή:** Η εφαρμογή iBoot αναπτύχθηκε ειδικά για να ανταποκρίνεται στις απαιτήσεις του Εργαστηρίου Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων, διασφαλίζοντας ότι διαθέτει τα απαραίτητα χαρακτηριστικά και δυνατότητες.
- **Ανοιχτός κώδικας:** Ο πηγαίος κώδικας της εφαρμογής είναι διαθέσιμος στο κοινό και δημοσιεύεται με την άδεια MIT, επιτρέποντας τη διαφάνεια και τις πιθανές συνεισφορές της κοινότητας.
- **Βέλτιστες πρακτικές και πρότυπα ασφαλείας:** Η εφαρμογή έχει σχεδιαστεί και υλοποιηθεί σύμφωνα με τις βέλτιστες πρακτικές και τα πιο πρόσφατα πρότυπα ασφαλείας, διασφαλίζοντας την προστασία των πληροφοριών και την κρυπτογράφηση των cookies συνόδου.

Αδυναμίες

- **Εξάρτηση από την PHP:** Το back-end της εφαρμογής iBoot είναι κατασκευασμένο με τη χρήση PHP, η οποία μπορεί να δημιουργήσει προκλήσεις όσον αφορά την επεκτασιμότητα και τη συμβατότητα με μελλοντικές τεχνολογίες.

-
- **Εξάρτηση από σχεσιακή βάση δεδομένων:** Η εφαρμογή βασίζεται σε μια σχεσιακή βάση δεδομένων MySQL για την αποθήκευση δεδομένων, γεγονός που ενδέχεται να περιορίσει την επεκτασιμότητα και την απόδοση κατά τον χειρισμό μεγάλων όγκων δεδομένων.

Ευκαιρίες

- **Πιθανές συνεργασίες:** Η φύση της εφαρμογής ως ανοικτού κώδικα δημιουργεί ευκαιρίες συνεργασίας με άλλα ιδρύματα ή προγραμματιστές που μπορούν να συμβάλουν στην περαιτέρω βελτίωσή της.
- **Επέκταση σε άλλα εργαστήρια:** Η επιτυχία και ο θετικός αντίκτυπος της εφαρμογής iBoot στο Εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων μπορεί να οδηγήσει στην υιοθέτησή της από άλλα εργαστήρια εντός του Πανεπιστημίου Δυτικής Μακεδονίας ή και πέραν αυτού.
- **Ενσωμάτωση με άλλα συστήματα:** Η εφαρμογή μπορεί να διερευνήσει ευκαιρίες για ενσωμάτωση με άλλα συστήματα ή πλατφόρμες, επιτρέποντας βελτιωμένη λειτουργικότητα και διαλειτουργικότητα.

Απειλές

- **Ανταγωνισμός από εμπορικές λύσεις:** Παρόλο που η εφαρμογή iBoot αναπτύχθηκε για να ικανοποιήσει συγκεκριμένες απαιτήσεις, υπάρχει πιθανότητα ανταγωνισμού από εμπορικές λύσεις που μπορεί να προσφέρουν περισσότερα χαρακτηριστικά ή ευρύτερο φάσμα δυνατοτήτων.
- **Τεχνολογική απαξίωση:** Οι ταχείες εξελίξεις στην τεχνολογία ενδέχεται να καταστήσουν ορισμένες πτυχές της εφαρμογής ξεπερασμένες, απαιτώντας συνεχείς ενημερώσεις και προσαρμογές για να παραμείνει επίκαιρη.
- **Τρωτά σημεία ασφαλείας:** Παρά την τήρηση των βέλτιστων πρακτικών και των προτύπων ασφαλείας της εφαρμογής, υπάρχει ο κίνδυνος τρωτών σημείων ασφαλείας και πιθανών παραβιάσεων, οι οποίες θα μπορούσαν να θέσουν σε κίνδυνο τις πληροφορίες και τη λειτουργικότητα της εφαρμογής.

6.3 Μελλοντικές Επεκτάσεις

Όλες οι διαδικτυακές εφαρμογές οφείλουν να αναβαθμίζονται συνεχώς ώστε, πέρα από την προσθήκη νέων δυνατοτήτων και τη βελτίωση των υπαρχόντων, να παραμένουν λειτουργικές και ασφαλείς. Συχνά χρειάζεται επίσης και η ενημέρωση βιβλιοθηκών και άλλων εξαρτήσεων της εφαρμογής, ώστε να καλύπτονται πιθανά κενά ασφαλείας και να επιλύονται προβλήματα που ίσως υπάρχουν. Η διαδικτυακή εφαρμογή που αναπτύχθηκε δεν αποτελεί εξαίρεση. Η βιωσιμότητα της είναι άμεσα εξαρτώμενη από τις εργασίες συντήρησης και ενημέρωσης της, οι οποίες θα πρέπει να πραγματοποιούνται τακτικά.

Μερικές ιδέες για περαιτέρω ανάπτυξη της εφαρμογής παρουσιάζονται στη συνέχεια.

- **Διεύρυνση λειτουργιών του REST API:** Το REST API θα μπορούσε να διευρυνθεί, προσθέτοντας endpoints που θα προσφέρουν διαλειτουργικότητα με άλλες εφαρμογές και υπηρεσίες. Αν αξιολογηθεί ως σημαντικό, θα μπορούσαν να προστεθούν συναρτήσεις ανάκτησης τμήματος δεδομένων, με περισσότερους ελέγχους πρόσβασης. Τέλος, θα ήταν ίσως χρήσιμη η προσθήκη μιας προαιρετικής παραμέτρου σελιδοποίησης, ώστε να αντλείται συγκεκριμένος αριθμός εγγραφών από την κάθε κλήση στα endpoints του API. Κάτι τέτοιο θα βελτίωνε την απόδοση της εφαρμογής σε μεγάλους αριθμούς εγγραφών.

-
- **Ανάπτυξη εναλλακτικών front-end:** Η ύπαρξη του REST API ευνοεί τη δημιουργία εναλλακτικών front-end για την εφαρμογή, αφού ουσιαστικά το front-end με το back-end είναι αποσυνδεδεμένα και επικοινωνούν μέσω του API της εφαρμογής. Ένα εναλλακτικό front-end θα μπορούσε να προσφέρει μεγαλύτερη συμβατότητα με συγκεκριμένους τύπους συσκευών, να εμφανίζει τις πληροφορίες της εφαρμογής με έναν πιο μοντέρνο και παραμετροποιήσιμο τρόπο, ή ακόμη και να πακεταριστεί σε εγγενή εφαρμογή για κάποιο λειτουργικό σύστημα υπολογιστών.
 - **Υποστήριξη διαφορετικών μεθόδων αυθεντικοποίησης:** Η εγγραφή και σύνδεση στο σύστημα θα μπορούσε να αξιοποιεί υπάρχοντα πρωτόκολλα διαπιστευτηρίων, ώστε να υποστηρίζεται και η διαλειτουργικότητα με την υπόλοιπη υποδομή κάποιου εργαστηρίου. Τέτοια πρωτόκολλα είναι τα LDAP, OAuth2, SAML και άλλα. Επίσης, θα μπορούσε να αναπτυχθεί υποστήριξη για σύνδεση με χρήση πολλαπλών παραγόντων ή και σύνδεσης χωρίς κωδικό πρόσβασης με magic link στο email του χρήστη.
 - **Προσθήκη λειτουργίας Wake-on-Lan:** Θα μπορούσε να διερευνηθεί η δυνατότητα δημιουργίας και αποστολής Wake-on-Lan πακέτων στο δίκτυο των υπολογιστών, ώστε η διαδικασία εκκίνησής τους να μπορεί να ξεκινήσει απευθείας από την εφαρμογή και χωρίς φυσική πρόσβαση στο κάθε μηχάνημα.
 - **Τροποποίηση ημερολογίου σε επεξεργάσιμο:** Το ημερολόγιο των χρονοδιαγραμμάτων θα μπορούσε να γίνει επεξεργάσιμο και έτσι δημιουργία νέων ή η τροποποίηση υπαρχόντων χρονοδιαγραμμάτων να γίνεται απευθείας σε αυτό.
 - **Προσθήκη δυνατότητας ανακατάταξης των Μπλοκ iPXE:** Θα μπορούσε να προστεθεί η δυνατότητα ανακατάταξης των Μπλοκ iPXE μέσα σε κάθε μενού, ώστε ο χρήστης να μπορεί να αλλάξει τη σειρά με την οποία εμφανίζονται κατά την εκκίνηση υπολογιστών, αντί για την μη-παραμετροποιήσιμη σειρά απεικόνισης που υπάρχει τώρα, καταταγμένη βάση της χρονολογικής σειράς με την οποία προστέθηκαν τα Μπλοκ στο κάθε μενού.
 - **Ενσωμάτωση relay DHCP server:** Η εφαρμογή θα μπορούσε να ενσωματώσει έναν relay DHCP server, ώστε να μπορεί να εγκατασταθεί στο τοπικό δίκτυο και να λειτουργήσει χωρίς επεξεργασία των ρυθμίσεων του προϋπάρχοντος DHCP server του δικτύου.

6.4 Συμπεράσματα

Από την παρούσα διπλωματική εργασία μπορεί να προκύψει μια πληθώρα συμπερασμάτων, σχετικά είτε με τις διαδικτυακές εφαρμογές γενικότερα, είτε ειδικότερα για τη συγκεκριμένη εφαρμογή διαχείρισης δικτυακής εκκίνησης υπολογιστών που αναπτύχθηκε.

Αρχικά, έγινε ανάλυση του διαδικτύου και τον ρόλο των εφαρμογών του στη σύγχρονη εποχή. Επίσης, έγινε αναφορά σε διάφορες γλώσσες διαδικτυακού προγραμματισμού, αλλά και τεχνικές οι οποίες αποσκοπούν να διευκολύνουν και να επιταχύνουν την ανάπτυξη εφαρμογών και να τις καταστήσουν πιο συντηρήσιμες και ασφάλειες. Μπορούμε από τα παραπάνω να συμπεράνουμε, ότι παρά την ταχεία τεχνολογική εξέλιξη, και την πληθώρα τεχνικών και εργαλείων που συνεχώς βελτιώνονται και αλλάζουν, η σημασία των διαδικτυακών εφαρμογών ως μέσων επίλυσης πρακτικών προβλημάτων παραμένει αναλλοίωτη σε βάθος χρόνου. Αυτό κάνει και σημαντικό προσόν την τεχνογνωσία σχεδίασης, ανάπτυξης και συντήρησής τους.

Στη συνέχεια, αναλύθηκαν η σχεδίαση και υλοποίηση της διαδικτυακής εφαρμογής δικτυακής εκκίνησης υπολογιστών iBoot. Γίνεται εύκολα εμφανές, ότι για την επιτυχημένη ανάπτυξη μιας εφαρμογής, πρέπει να έχει γίνει αναλυτική σχεδίαση των δεδομένων που θα μεταχειρίζεται, των λειτουργιών της και του τρόπου απεικόνισής της. Φυσικά, για τη σχεδίαση της λύσης, πρέπει να έχει προηγηθεί μια καλή ανάλυση του προβλήματος προς επίλυση. Συμπεράνουμε δηλαδή, πώς η σωστή ακολουθία φάσεων ανάπτυξης μιας εφαρμογής, της οποίας βήματα δε πρέπει να παραλείπονται ή να παραβλέπονται, είναι:

1. Διεξοδική ανάλυση του προβλήματος.
2. Καθορισμός προδιαγραφών και σχεδιασμός λύσης.
3. Ανάπτυξη λύσης, βάση των ορισμένων προδιαγραφών.
4. Έλεγχος ποιότητας λύσης και ανατροφοδότηση.

Επιπρόσθετα, έγινε αξιολόγηση του συστήματος που παράχθηκε, με γνώμονα τόσο την ποιότητα και τη χρησιμότητα της λύσης που προσφέρει, όσο και τη βιωσιμότητα του, καθώς και την ικανότητα του να κλιμακωθεί, ώστε να σε αυξημένες ανάγκες. Από αυτό, μπορούμε να αντλήσουμε το συμπέρασμα πως, όπως αναφέρθηκε και παραπάνω, αν και συμβαίνουν στην τελευταία φάση του κύκλου ζωής ανάπτυξης μιας εφαρμογής, ο έλεγχος της και η ανατροφοδότηση είναι ιδιαίτερα σημαντικά. Σε αυτή τη φάση αξιολογείται, ουσιαστικά, αν η εφαρμογή πέτυχε το σκοπό για τον οποίο δημιουργήθηκε, αν είναι ασφαλής και σταθερή, και πώς μπορεί να βελτιωθεί περαιτέρω.

Τέλος, όσον αφορά τη συγκεκριμένη εφαρμογή, μπορούμε να συμπεράνουμε πως η διαδικασία διαχείρισης δικτυακής εκκίνησης υπολογιστών για ένα πανεπιστημιακό εργαστήριο, μπορεί να γίνει αρκετά περίπλοκη σε βάθος χρόνου και καθώς αυξάνονται οι απαιτήσεις. Η χρήση της εφαρμογής θα έχει σημαντικό όφελος για το εργαστήριο, αφού θα απλοποιήσει τη διαδικασία και θα αποδεσμεύσει πολύτιμο χρόνο των διαχειριστών, οι οποίοι θα μπορούν πλέον να προβάλλουν και να τροποποιούν πολύ εύκολα τα χρονοδιαγράμματα εκκίνησης των υπολογιστών του εργαστηρίου.

6.5 Σύνοψη Κεφαλαίου 6

Το κεφάλαιο 6 σηματοδοτεί το πέρας της παρούσας διπλωματικής εργασίας. Στο παρόν κεφάλαιο παρέχεται μια ανάλυση SWOT του περιεχομένου της διαδικτυακής πλατφόρμας, η οποία περιγράφει συνοπτικά τα πλεονεκτήματα και τα μειονεκτήματα που σχετίζονται με την υλοποίηση του παρόντος έργου, καθώς και τις δυνατότητες και τα οφέλη που παρουσιάζει και τυχόν πιθανούς κινδύνους που μπορεί να αντιμετωπίσει. Παρέχονται επίσης ορισμένες ιδέες για πιθανές μελλοντικές βελτιώσεις και προσαρμογές της πλατφόρμας. Το κεφάλαιο αυτό ολοκληρώνεται με τα τελικά συμπεράσματα επί της διπλωματικής εργασίας.

Παράρτημα

Παράρτημα Α΄

Απαιτήσεις, Εγκατάσταση & Ρύθμιση Πλατφόρμας

Παρακάτω θα βρείτε τις απαιτήσεις συστήματος και τις οδηγίες εγκατάστασης και ρύθμισης της πλατφόρμας.

Α΄.1 Απαιτήσεις Συστήματος

Α΄.1.1 PHP

Απαιτείται PHP έκδοσης 8.1 ή νεότερη. Είναι, επίσης, απαραίτητες οι παρακάτω επεκτάσεις της PHP.

- php-intl
- php-mbstring
- php-json
- php-mysqlnd
- php-xml
- php-curl

Τέλος, για το php-curl, είναι απαραίτητη και η εγκατάσταση της βιβλιοθήκης libcurl στο μηχάνημα.

Η πλατφόρμα έχει δοκιμαστεί επιτυχώς με PHP έκδοσης 8.2 και χρήση php-fpm.

Α΄.1.2 Βάση Δεδομένων

Για τη βάση δεδομένων της πλατφόρμας απαιτείται MySQL/MariaDB έκδοσης 5.1 ή νεότερη.

Η πλατφόρμα έχει δοκιμαστεί επιτυχώς με MariaDB έκδοσης 8.1, με χρήση php-mysql και MySQLi DBDriver.

Α΄.1.3 Διακομιστής Ιστού

Ως webserver, μπορεί να χρησιμοποιηθεί οποιοσδήποτε έχει τη δυνατότητα εκτέλεσης PHP κώδικα. Στις απεικονίσεις Α΄.1 και Α΄.2 παρέχονται ενδεικτικά παραδείγματα ρύθμισης για Apache2 και Nginx webserver.

Στα παρακάτω παραδείγματα έχουν γίνει οι εξής παραδοχές:

- Τα αρχεία της πλατφόρμας βρίσκονται στη διαδρομή `/srv/iboot` του συστήματος.
- Η δημόσια διεύθυνση της πλατφόρμας είναι η `iboot.example.com`.

Apache2

Απεικόνιση Α΄.1: Παράδειγμα ρύθμισης Apache2 webserver

```
<VirtualHost *:80>
    ServerName iboot.example.com
    ServerAdmin webmaster@example.com
    DocumentRoot /srv/iboot/public
    ErrorLog /var/log/apache2/iboot-error.log
    CustomLog /var/log/apache2/iboot-access.log combined
    <Directory /srv/iboot/public>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Για λειτουργία της πλατφόρμας με Apache2 webserver, είναι απαραίτητη και η ενεργοποίηση του rewrite module του, το οποίο γίνεται με χρήση της εντολής `sudo a2enmod rewrite`.

Nginx

Απεικόνιση Α΄.2: Παράδειγμα ρύθμισης Nginx webserver

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    root /srv/iboot/public;

    index index.html index.php;

    server_name iboot.example.com;

    # set expiration of assets to MAX for caching
    location ~* \.(ico|css|js|gif|jpe?g|png)(\?[0-9]+)?$ {
        expires max;
        log_not_found off;
    }

    location / {
        try_files $uri $uri/ /index.php$is_args$args;
```

```

}

location ~ /\.php$ {
    include snippets/fastcgi-php.conf;
    # With php-fpm (or other unix sockets):
    fastcgi_pass unix:/run/php/php8.1-fpm.sock;
    # With php-cgi (or other tcp sockets):
    # fastcgi_pass 127.0.0.1:9000;
}

error_page 404 /index.php;

location ~ /\. {
    deny all;
}
}

```

Σημείωση: Τα παραπάνω παραδείγματα ρύθμισης webserver **αντενδείκνυνται** για παραγωγική λειτουργία του συστήματος, καθώς είναι ελλιπή ως προς τη διευθέτηση σοβαρών ζητημάτων ασφαλείας, όπως κρυπτογράφηση TLS.

A'.2 Οδηγίες Εγκατάστασης

1. Αντιγραφή των αρχείων της πλατφόρμας στο root path του webserver ή σε υποφάκελό του.
2. Εκτέλεση της εντολής `composer install --no-dev` για την εγκατάσταση των απαραίτητων βιβλιοθηκών.
3. Παραχώρηση κατάλληλων δικαιωμάτων στα αρχεία, ώστε να είναι προσβάσιμα από τον webserver.
4. Δημιουργία βάσης δεδομένων για την πλατφόρμα και του χρήστη με δικαίωμα πρόσβασης σε αυτή.
5. Αλλαγή ρυθμίσεων και επίσκεψη της πλατφόρμας για ολοκλήρωση εγκατάστασης και εγγραφή Γενικού Διαχειριστή.

A'.3 Αλλαγή Ρυθμίσεων Πλατφόρμας

Τα αρχεία που βρίσκονται στη διαδρομή `app/Config/` περιέχουν τις ρυθμίσεις της εφαρμογής με επεξηγηματικά σχόλια για κάθε μια ξεχωριστά. Αλλαγές στις ρυθμίσεις μπορούν να γίνουν με απευθείας επεξεργασία αυτών των αρχείων. Ωστόσο, είναι δυνατή η δημιουργία ενός `.env` αρχείου, το οποίο παρέχει μεταβλητές περιβάλλοντος, των οποίων οι τιμές υπερκαλύπτουν αυτές των αρχείων της διαδρομής `app/Config/`.

Μαζί με τον κώδικα της εφαρμογής, παρέχεται ένα `env` αρχείο το οποίο μπορεί κανείς να αντιγράψει ή μετονομάσει σε `.env` και να κάνει εκεί τις αλλαγές που χρειάζονται για τη ρύθμιση της εγκατάστασης. Στην απεικόνιση A'.3 παρουσιάζεται ένα ενδεικτικό αρχείο `.env`.

Απεικόνιση Α'.3: Παράδειγμα αρχείου .env

```
#-----  
# APP  
#-----  
app.baseUrl = 'http://iboot.example.com/'  
app.forceGlobalSecureRequests = false  
  
#-----  
# DATABASE  
#-----  
database.default.hostname = localhost  
database.default.database = iboot  
database.default.username = iboot  
database.default.password = iboot  
database.default.DBDriver = MySQLi  
database.default.DBPrefix = ibt_  
  
#-----  
# EMAIL  
#-----  
email.fromEmail = 'no-reply@iboot.example.com'  
email.fromName = 'iBoot'  
# email.protocol = 'mail'  
# email.mailPath = '/usr/sbin/sendmail'  
# email.SMTPHost = ''  
# email.SMTPUser = ''  
# email.SMTPPass = ''  
# email.SMTPPort = ''  
  
#-----  
# JWT  
#-----  
jwt.secret = '36wfpzOaVveefvs5mcHVAhsqEpKTzpfLz+LkAct'  
  
#-----  
# REGISTRATION  
#-----  
registration.enabled = true
```

Βιβλιογραφία

- [1] FOG Project Team. *FOG Project*. 2020. URL: <https://fogproject.org/>.
- [2] FOG Project Team. *What is FOG - FOG Project 1.5.9 documentation*. 2020. URL: <https://docs.fogproject.org/en/latest/introduction/introduction.html>.
- [3] CCboot Team. *Kazakstan - Force*. 2022. URL: <https://www.ccboot.com/>.
- [4] Colama. Αύγ. 2023. URL: <http://colama.net/eigenboot>.
- [5] Robert Khan. *Internet*. Απρ. 2022. URL: <https://www.britannica.com/technology/Internet#ref255529>.
- [6] Sareh Aghaei, Mohammad Ali Nematbakhsh και Hadi Khosravi Farsani. «Evolution of the world wide web: From WEB 1.0 TO WEB 4.0». Στο: *International Journal of Web & Semantic Technology* 3.1 (2012), σσ. 1–10.
- [7] Tech Target Contributor. *What is Web Application (Web Apps) and its Benefits*. Αύγ. 2019. URL: <https://www.techtarget.com/searchsoftwarequality/definition/Web-application-Web-app>.
- [8] Ben Smith. «Object-oriented programming». Στο: *AdvancED ActionScript 3.0: Design Patterns*. Springer, 2011, σσ. 1–25.
- [9] Erin Doherty. *What is object-oriented programming? OOP explained in depth*. Απρ. 2020. URL: <https://www.educative.io/blog/object-oriented-programming>.
- [10] Zanfina Svirca. *Everything you need to know about MVC architecture | by Zanfina Svirca | Towards Data Science*. Μάι. 2020. URL: <https://towardsdatascience.com/everything-you-need-to-know-about-mvc-architecture-3c827930b4c1>.
- [11] CodeIgniter Foundation. *Models, Views, and Controllers ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/concepts/mvc.html.
- [12] tutorialspoint. *MVC Framework - Introduction*. 2022. URL: https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm.
- [13] Lokesh Gupta. *What is REST - REST API Tutorial*. Μάι. 2020. URL: <https://restfulapi.net/>.
- [14] RedHat. *What is a REST API?* Μάι. 2020. URL: <https://www.redhat.com/en/topics/api/what-is-a-rest-api>.
- [15] Alexander S. Gillis. *What is REST API (RESTful API)?* Σεπτ. 2020. URL: <https://www.techtarget.com/searchapparchitecture/definition/RESTful-API>.

-
- [16] Peter Loshin. *What is Structured Query Language (SQL)?* Φεβ. 2022. URL: <https://www.techtarget.com/searchdatamanagement/definition/SQL>.
- [17] Toni Taipalus και Ville Seppänen. «SQL education: A systematic mapping study and future research agenda». Στο: *ACM Transactions on Computing Education (TOCE)* 20.3 (2020), σσ. 1–33.
- [18] *What is a Database Management System | DBMS | Monitoring*. URL: <https://www.appdynamics.com/topics/database-management-systems>.
- [19] Jacob Thornton και Mark Otto. *Get started with Bootstrap · Bootstrap v5.2*. URL: <https://getbootstrap.com/docs/5.2/getting-started/introduction/>.
- [20] Hackr Team. *What is Bootstrap? Pros and Cons Of This Framework*. Μάι. 2022. URL: <https://hackr.io/blog/what-is-bootstrap-framework>.
- [21] Suraj Shahu Gaikwad και PRATIBHA Adkar. «A review paper on bootstrap framework». Στο: *IRE Journals* 2.10 (2019), σσ. 349–351.
- [22] Paul Jackson. *What is CodeIgniter? How Does PHP CI Framework Works?* Ιαν. 2020. URL: <https://www.guru99.com/what-is-codeigniter.html>.
- [23] CodeIgniter Foundation. *Welcome to CodeIgniter4 ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/intro/index.html.
- [24] Swagger Team. *API Documentation & Design Tools for Teams | Swagger*. 2022. URL: <https://swagger.io/>.
- [25] TechTarget Contributor. *What is Swagger? Definition from WhatIs.com*. Αύγ. 2019. URL: <https://www.techtarget.com/searchapparchitecture/definition/Swagger>.
- [26] Semen Grigorev κ.ά. «String-embedded language support in integrated development environment». Στο: *Proceedings of the 10th Central and Eastern European Software Engineering Conference in Russia*. 2014, σσ. 1–11.
- [27] *PhpStorm: The Lightning-Smart IDE for PHP Programming by JetBrains*. URL: <https://www.jetbrains.com/phpstorm/>.
- [28] Diomidis Spinellis. «Git». Στο: *IEEE software* 29.3 (2012), σσ. 100–101.
- [29] Jon Loeliger και Matthew McCullough. *Version Control with Git: Powerful tools and techniques for collaborative software development*. ” O’Reilly Media, Inc.”, 2012.
- [30] *GitKraken, what is it? and common actions?* 24 Μάι. 2019. URL: <https://dev.to/iphytech/gitkraken-what-is-it-and-common-actions-5531>.
- [31] Eirini Kalliamvakou κ.ά. «The promises and perils of mining github». Στο: *Proceedings of the 11th working conference on mining software repositories*. 2014, σσ. 92–101.
- [32] CodeIgniter Foundation. *Application Structure ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/concepts/structure.html.
- [33] CodeIgniter Foundation. *Honeypot Class ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/concepts/mvc.html.

-
- [34] CodeIgniter Foundation. *Query Builder Class ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/database/query_builder.html.
- [35] CodeIgniter Foundation. *Validation ;CodeIgniter 4.3.6 documentation*. Ιούλ. 2023. URL: https://www.codeigniter.com/user_guide/libraries/validation.html.

Πηγές Εικόνων

- [Eli18] Elisa.garcia.1994. *Plataforma Internet*. This file is licensed under the Creative Commons Attribution-Share Alike 4.0 International license. 30 Ιούν. 2018. URL: https://commons.wikimedia.org/wiki/File:Plataforma_internet.png.
- [Blu11] BlueSky23. *MVC Diagram*. This file is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license. 2011. URL: https://commons.wikimedia.org/wiki/File:MVC_Diagram_3.jpg.
- [Seo] Seobility. *REST API*. License: CC BY-SA 4.0. URL: <https://www.seobility.net/en/wiki/images/thumb/f/f1/Rest-API.png/675px-Rest-API.png>.
- [Gha20a] Shreya Ghatge. *Using session cookies vs. JWT for authentication*. Ιούν. 2020. URL: <https://hackernoon.com/images/pazJZnCJTqSZxQS4tltZo4Gatbo1-fo8h3y11.jpg>.
- [Gha20b] Shreya Ghatge. *Using session cookies vs. JWT for authentication*. Ιούν. 2020. URL: <https://hackernoon.com/images/pazJZnCJTqSZxQS4tltZo4Gatbo1-h7an3ylo.jpg>.