

Πανεπιστήμιο Δυτικής Μακεδονίας

Πολυτεχνική Σχολή

Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

**Διπλωματική Εργασία**

**Σχεδιασμός και υλοποίηση έξυπνου βομβητή για παραγγελίες σε κέντρα  
εστίασης και ανάπτυξη υποστηρικτικής εφαρμογής ιστοχώρου**

---

**Design and implementation of a smart buzzer for orders in restaurants and  
development of a supportive website application**

**Ιωάννης Θαλαλαίος**

Επιβλέπων Καθηγητής: **Δρ Μηνάς Δασυγένης**

Εργαστήριο Ρομποτικής, Ενσωματωμένων και Ολοκληρωμένων Συστημάτων

Κοζάνη, Οκτώβριος 2021



## Περίληψη

Τα κέντρα εστίασης στην σύγχρονη εποχή έχουν υψηλές απαιτήσεις στο τομέα της εξυπηρέτησης των πελατών. Ο αυξανόμενος ρυθμός ανάπτυξης της τεχνολογίας δίνει την δυνατότητα να εκπληρωθούν οι απαιτήσεις και οι στόχοι που θέτει κάθε κέντρο εστίασης. Η μείωση του χρόνου αναμονής παραγγελιών, η ταχύτερη εξυπηρέτηση πελατών, η ευκολία και η εγκυρότητα της διαδικασίας παραγγελίας είναι τα κύρια ζητούμενά τους. Αξιοποιώντας ψηφιακές τεχνολογίες αιχμής, είτε σε υλικό είτε σε λογισμικό, μπορεί να επιτευχθούν οι ανωτέρω στόχοι και να γίνει αποτελεσματικότερη η λειτουργία τους.

Στην παρούσα διπλωματική εργασία παρουσιάζεται η υλοποίηση διαδικτυακού Πληροφοριακού Συστήματος, που υποστηρίζει ένα κέντρο εστίασης και πιο συγκεκριμένα ενός ταχυφαγείου. Όσον αφορά το υλικό στην υλοποίηση αυτή αξιοποιείται η πλακέτα esp8266. Η πλακέτα αυτή αποτελείται από μια οθόνη εμφάνισης για τον αριθμό της παραγγελίας και έχει τοποθετηθεί σε ένα ειδικά κατασκευασμένο, από 3D εκτυπωτή, κουτί. Επίσης χρησιμοποιείται led και buzzer-βομβητής για ηχητική και φωτεινή σήμανση κατά την ολοκλήρωση της παραγγελίας. Παράλληλα έχει υλοποιηθεί εφαρμογή ιστού, η οποία διασυνδέεται μέσω ασύρματης επικοινωνίας WiFi με την ανωτέρω πλακέτα για να γίνεται άμεσα ενημέρωση των πελατών του ταχυφαγείου για τις παραγγελίες τους, όταν είναι έτοιμες. Η εφαρμογή ιστού παρέχει τις ακόλουθες λειτουργίες χρήστη: προσθήκης, επεξεργασίας, ιστορικού, στατιστικών σχετικά με τις παραγγελίες του, παρέχει περιβάλλον πλήρους διαχείρισής της και έχει υλοποιηθεί με χρήση τεχνολογιών ιστού ανοιχτού κώδικα (HTML, CSS, JavaScript, jQuery, AJAX, Bootstrap, PHP, MySQL). Αξίζει να σημειωθεί ότι, σε αντίθεση με υπάρχουσες εμπορικές υλοποιήσεις, η υλοποίηση αυτή είναι ανοιχτού υλικού και λογισμικού δίνοντας την δυνατότητα και σε όποιους το επιθυμήσουν χωρίς κόστος να την αναπτύξουν περαιτέρω.

### Λέξεις Κλειδιά

Προγραμματισμός Διαδικτύου, PHP, HTML, CSS, Javascript, Bootstrap, ESP, Arduino IDE, Ενσωματωμένα Συστήματα.



## **Abstract**

Restaurants in the modern era have high demands in the field of customer service. The increasing rate of development of technology enables the fulfillment of the requirements and goals set by each restaurant. Reducing order waiting time, faster customer service, ease, and validity of the ordering process are their main requirements. Utilizing state-of-the-art digital technologies, whether in hardware or software, the above objectives can be achieved and their operation made more efficient.

This dissertation has presented the implementation of an online Information System, which supports a restaurant and more precise a fast-food restaurant. Regarding the hardware in this implementation, the esp8266 board is utilized. This board consists of a display to show the order number and has been placed in a special box made by a 3D printer. Also used a led and a buzzer, for sound and light signaling during the completion of the order. At the same time, a web application has been implemented which is interconnected via wireless communication Wi-Fi with the above board to immediately inform the customers, of the fast-food restaurant, when their orders are ready. The web application provides the user functions of adding, editing, history, statistics related to his orders, as well as provides a complete management environment and has been created with open-source web technologies (HTML, CSS, JavaScript, jQuery, AJAX, Bootstrap, PHP, MySQL). It is worth noting that, in contrast to existing commercial implementations, this implementation is open hardware and software allowing those who wish to develop it at no cost.

### **Keywords**

Web programming, PHP, HTML, CSS, JavaScript, bootstrap, ESP, Arduino IDE, embedded systems.



## Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν.1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο

“Σχεδιασμός και υλοποίηση έξυπνου βομβητή για παραγγελίες σε κέντρα εστίασης και ανάπτυξη υποστηρικτικής εφαρμογής ιστοχώρου” 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.

Copyright (C) Ιωάννης Θαλαλαίος, Δρ Μηνάς Δασυγένης, 2021 , Κοζάνη





## Περιεχόμενα

1ο Κεφάλαιο - Εισαγωγή.....	16
1.1 Πλεονεκτήματα ηλεκτρονικής παραγγελιοληψίας .....	17
1.2 Σκοπός του ΠΣ.....	17
1.3 Ερευνητική επισκόπηση .....	18
1.4 Σύνοψη της διπλωματικής εργασίας.....	19
1.5 Σύνοψη κεφαλαίου.....	20
2ο Κεφάλαιο - Θεωρητικό υπόβαθρο.....	21
2.1 Διαδίκτυο των Πραγμάτων .....	21
2.2 Πλακέτα esp8266.....	21
2.3 Arduino IDE .....	22
2.4 3D εκτυπωτής Prusa i3 .....	22
2.5 PrusaSlicer .....	22
2.6 Fritzing.....	23
2.7 HTML .....	23
2.8 CSS .....	23
2.9 JavaScript.....	24
2.10 jQuery .....	24
2.11 AJAX .....	25
2.12 Bootstrap.....	25
2.13 PHP .....	25
2.14 MySQL/MariaDB .....	26
2.15 Notepad++ .....	26
2.16 WinSCP .....	26
2.17 PuTTY .....	27
2.18 Σύνοψη κεφαλαίου.....	27

3ο Κεφάλαιο - Ανάλυση Πληροφοριακού Συστήματος eBuzzer.....	28
3.1 Σύλληψη ιδέας .....	28
3.2 Απαιτήσεις Πληροφοριακού Συστήματος eBuzzer .....	28
3.2.1 Ρόλος διαχειριστή.....	28
3.2.2 Ρόλος απλού χρήστη .....	29
3.2.3 Απαιτήσεις συσκευής με χρήση της πλακέτας esp8266 .....	29
3.2.4 Απαιτήσεις λειτουργιών και βάσης δεδομένων .....	30
3.2.5 Use Case Diagram.....	30
3.3 Σχεσιακό Διάγραμμα της Βάσης Δεδομένων .....	31
3.4 Ανάλυση της Βάσης Δεδομένων του ΠΣ.....	33
3.4.1 Πίνακας user.....	33
3.4.2 Πίνακας orders .....	34
3.4.3 Πίνακας history .....	35
3.4.4 Πίνακας categories .....	36
3.4.5 Πίνακας esp .....	36
3.4.6 Πίνακας product .....	37
3.4.7 Πίνακας purchase .....	38
3.5 Ανάλυση υποδομής, αρχείων και κώδικα του ΠΣ .....	39
3.5.1 Ανάλυση υποδομής του ΠΣ .....	39
3.5.2 Ανάλυση φακέλων και αρχείων του ΠΣ .....	40
3.5.3 Ανάλυση κώδικα του ΠΣ .....	42
3.6 Διασύνδεση πληροφοριακού συστήματος με την πλακέτα esp8266.....	47
3.7 Ασφάλεια πληροφοριακού συστήματος .....	50
3.7.1 Σύνδεση με τη βάση δεδομένων .....	50
3.7.2 Επικύρωση δεδομένων εισόδου με τη συνάρτηση filter_var() .....	51
3.7.3 Έλεγχος πρόσβασης με τη χρήση SESSION .....	52

3.8	Σύνοψη κεφαλαίου.....	52
4ο Κεφάλαιο -	Διεπαφή χρήστη και λειτουργίες.....	53
4.1	Εγγραφή, σύνδεση και αποσύνδεση στο ΠΣ .....	53
4.2	Λειτουργίες χρήστη διαχειριστή .....	55
4.3	Διενέργεια παραγγελίας μέσω του ΠΣ.....	59
4.4	Σύνοψη κεφαλαίου.....	61
5ο Κεφάλαιο -	Επίλογος.....	62
5.1	Σύνοψη του ΠΣ.....	62
5.2	Συμπεράσματα .....	63
5.3	Ανάλυση SWOT .....	63
5.3.1	Δυνατά σημεία (strengths) .....	63
5.3.2	Αδύνατα σημεία (weaknesses).....	63
5.3.3	Ευκαιρίες (opportunities) .....	64
5.3.4	Απειλές (threats).....	64
5.4	Εκτίμηση συνεχόμενης λειτουργίας (scaling) .....	64
5.5	Μελλοντικές επεκτάσεις.....	65
5.6	Σύνοψη κεφαλαίου.....	65
6ο Κεφάλαιο -	Βιβλιογραφία.....	66
7ο Κεφάλαιο -	Παράρτημα.....	71
7.1	Εγκατάσταση εφαρμογής.....	71
7.2	Πλάνο ελέγχου ορθής λειτουργίας ΠΣ .....	71
7.3	Συντήρηση βάσης δεδομένων .....	73

## Κατάλογος εικόνων

Εικόνα 1:	Διάγραμμα των ακροδεκτών του esp8266 NodeMCU.....	22
-----------	---	----

Εικόνα 2: Κύκλωμα βομβητή με esp8266 .....	30
Εικόνα 3: Διάγραμμα χρήσης ΠΣ .....	31
Εικόνα 4: Σχεσιακό Διάγραμμα Βάσης Δεδομένων eBuzzer .....	32
Εικόνα 5: Πίνακες της βάσης Δεδομένων.....	33
Εικόνα 6: Πίνακας user .....	33
Εικόνα 7: Πίνακας orders.....	34
Εικόνα 8: Πίνακας history .....	35
Εικόνα 9: Πίνακας categories .....	36
Εικόνα 10: Πίνακας esp8266 .....	37
Εικόνα 11: Πίνακας product .....	38
Εικόνα 12: Πίνακας purchase .....	38
Εικόνα 13: Κατάλογος root eBuzzer.....	40
Εικόνα 14: Αρχεία φακέλων admin, includes, user .....	41
Εικόνα 15: https.php .....	42
Εικόνα 16: config.php.....	42
Εικόνα 17: connectPDO.php.....	43
Εικόνα 18: register.php .....	43
Εικόνα 19: regValues.php.....	44
Εικόνα 20: login.php.....	44
Εικόνα 21: checkLogin.php .....	45
Εικόνα 22: searchUser.js.....	46

Εικόνα 23: logout.php .....	46
Εικόνα 24: Κανόνας εμφάνισης επιλεγμένος από το style.css .....	47
Εικόνα 25: Αρχικοποίηση μεταβλητών esp8266 .....	47
Εικόνα 26: Σύνδεση πλακέτας esp8266 με το ασύρματο δίκτυο.....	48
Εικόνα 27: Σύνδεση πλακέτας esp8266 με ΠΣ.....	48
Εικόνα 28: Ανταλλαγή δεδομένων πλακέτας esp8266 με ΠΣ.....	49
Εικόνα 29: Αρχείο api.php.....	49
Εικόνα 30: Εντολές PDO για σύνδεση με βάση δεδομένων.....	50
Εικόνα 31: Παράδειγμα χρήσης PDO εντολών .....	51
Εικόνα 32: Παράδειγμα χρήσης filter_var() .....	52
Εικόνα 33: Έλεγχος πρόσβασης με SESSION .....	52
Εικόνα 34: eBuzzer χωρίς εξουσιοδοτημένη πρόσβαση .....	53
Εικόνα 35: Φόρμα εγγραφής.....	54
Εικόνα 36: Μήνυμα λάθους στη φόρμα εγγραφής .....	54
Εικόνα 37: Σύνδεση χρήστη.....	55
Εικόνα 38: Διαχείριση χρηστών.....	55
Εικόνα 39: Διαχείριση κατηγοριών .....	56
Εικόνα 40: Επεξεργασία κατηγορίας .....	56
Εικόνα 41: Διαχείριση προϊόντων.....	57
Εικόνα 42: Διαχείριση πλακετών esp .....	57
Εικόνα 43: Παραγγελία σε εξέλιξη.....	58

Εικόνα 44: Αντιστοίχιση με πλακέτα esp .....	58
Εικόνα 45: Ολοκλήρωση παραγγελίας .....	58
Εικόνα 46: Ιστορικό παραγγελιών .....	59
Εικόνα 47: Διενέργεια παραγγελίας 1ο στάδιο .....	59
Εικόνα 48: Διενέργεια παραγγελίας 2ο στάδιο .....	59
Εικόνα 49: Προσθήκη προϊόντος στο καλάθι .....	60
Εικόνα 50: Ολοκλήρωση διενέργειας παραγγελίας .....	60
Εικόνα 51: Κατάσταση παραγγελίας .....	60

### **Κατάλογος πινάκων**

Πίνακας 1: Μετρικές ΠΣ.....	62
Πίνακας 2: Μέγιστες τιμές πενταετούς πλάνου συνεχόμενης λειτουργίας .....	64
Πίνακας 3: Πενταετές πλάνο φυσιολογικής συνεχόμενης λειτουργίας.....	65



## **1ο Κεφάλαιο - Εισαγωγή**

Τον τελευταίο αιώνα οι επιχειρήσεις σε παγκόσμιο επίπεδο ολοένα και αυξάνονται. Τις τελευταίες δεκαετίες οι ψηφιακές τεχνολογίες εξελίσσονται με ραγδαίο ρυθμό, με αποτέλεσμα να οδηγούν ολοένα και περισσότερες επιχειρήσεις να τις αξιοποιούν, ώστε να γίνονται αποτελεσματικότερες. Η παγκοσμιοποίηση με την σειρά της οδηγεί σε ένα πλαίσιο υψηλότερων απαιτήσεων των πελατών έναντι των υπηρεσιών και των προϊόντων που δέχονται από τις επιχειρήσεις, συντελώντας στην περαιτέρω αύξηση του ανταγωνισμού μεταξύ των επιχειρήσεων. Στο πλαίσιο αυτό, το βασικό μέλημα των επιχειρήσεων είναι η αύξηση του κέρδους ή η μείωση του κόστους.

Επίσης τα τελευταία χρόνια ζούμε σε μία εποχή της υπερπαραγωγής προϊόντων και υπηρεσιών παντός τύπου, που με την σειρά της οδηγεί τον πελάτη να αναζητά είτε την ποιότητα, είτε την οικονομικότερη λύση, είτε την βέλτιστη λύση, δηλαδή σχέση ποιότητας τιμής. Οι γρήγοροι ρυθμοί της σύγχρονης εποχής καθιστούν τον πελάτη να έχει όλο και λιγότερο χρόνο στην διάθεσή του για να τον αξιοποιήσει όπως αυτός επιθυμεί. Η εξυπηρέτηση πελατών με όσο το δυνατόν ταχύτερες και ποιοτικότερες υπηρεσίες είναι δύο από τους βασικούς στόχους των επιχειρήσεων. Η ηλεκτρονική παραγγελιοληψία ενσωματώνεται σε ολοένα και περισσότερα εστιατόρια διανομής φαγητού (delivery) για να μπορούν να ανταποκριθούν στις απαιτήσεις αυτές.

Στον Ελληνικό χώρο οι επιχειρήσεις εστίασης παντός τύπου είναι αυξημένες και αυξάνονται σύμφωνα με στατιστικά στοιχεία της Ελληνικής Στατιστικής Υπηρεσίας για την περίοδο 2011 έως το 2018 [1]. Πρόσφατα λόγω της πανδημίας COVID19 ένας μεγάλος αριθμός των επιχειρήσεων εστίασης έχει προσθέσει την ηλεκτρονική παραγγελιοληψία στον τρόπο λειτουργίας του, με σκοπό να ανταπεξέλθει στις νέες συνθήκες που έχουν διαμορφωθεί.

Η δημιουργία ενός Πληροφοριακού Συστήματος (ΠΣ) για την κάλυψη των παραπάνω αναγκών σε συνδυασμό με την αύξηση των επιχειρήσεων εστίασης στην Ελλάδα, ήταν ευκαιρία για να δοθούν οι κατάλληλες λύσεις στην παραγγελιοληψία. Η προγραμματιστική αυτή πρόκληση με οδήγησε να υλοποιήσω ένα ΠΣ για να αντιμετωπιστούν τα προβλήματα της ουράς αναμονής κατά το στάδιο της παραγγελίας, της μείωσης του κόστους της επιχείρησης και της καλύτερης εποπτείας των παραγγελιών των πελατών.



## 1.1 Πλεονεκτήματα ηλεκτρονικής παραγγελιοληψίας

Η παραδοσιακή παραγγελιοληψία γινόταν και γίνεται ως επί το πλείστον μέσω του τηλεφώνου. Σε ώρες αιχμής προέκυπταν προβλήματα, όπως χρόνος αναμονής, δυσκολία στην καταγραφή και παρακολούθηση της παραγγελίας με αποτέλεσμα την δυσφορία των πελατών. Τα παραπάνω προβλήματα μπορούν να αντιμετωπιστούν αποτελεσματικότερα με την χρήση της ηλεκτρονικής παραγγελιοληψίας, μέσω της οποίας υπάρχει καλύτερη διενέργεια, καταγραφή, παρακολούθηση και εκτέλεση των παραγγελιών. Επιπροσθέτως η ενσωμάτωσή της οδηγεί στην μείωση του κόστους λειτουργίας ενός εστιατορίου, επειδή απαιτείται λιγότερο προσωπικό και χρόνος για την διενέργεια των ηλεκτρονικών παραγγελιοληψιών. Οι πελάτες μπορούν να διενεργούν την ηλεκτρονική παραγγελιοληψία μέσω του Διαδικτύου και των υπηρεσιών του, από μόνοι τους, με χρήση κάποιας έξυπνης συσκευής (τηλέφωνο, ταμπλέτα, φορητός υπολογιστής). Το Διαδίκτυο και οι υπηρεσίες αποτελούν τα θεμέλια της ηλεκτρονικής παραγγελιοληψίας. Επίσης το Διαδίκτυο των Πραγμάτων - (Internet of Things - IoT) φέρνει επανάσταση με τις επιλογές που δίνει. Ένα σύνολο αισθητήρων, μικροεπεξεργαστών και μικροϋπολογιστών μπορούν να αξιοποιηθούν και να ικανοποιήσουν κάθε είδους ιδέα, που μπορεί να είχε κάποιος πριν την εποχή του IoT και που δεν ήταν εφικτό να την υλοποιήσει τότε. Συνήθως η εκάστοτε επιχείρηση εστίασης που θέλει να προσφέρει τέτοιες υπηρεσίες υιοθετεί ένα υπάρχον ΠΣ για να καλύψει το σύνολο ή μέρος των αναγκών αυτών. Ένα τέτοιο ΠΣ πρέπει να είναι κατάλληλα σχεδιασμένο για τον σκοπό αυτό.

## 1.2 Σκοπός του ΠΣ

Στην παρούσα διπλωματική εργασία σχεδιάστηκε και υλοποιήθηκε ένα ΠΣ με στόχο να μπορεί να ανταποκριθεί στις ανάγκες της σύγχρονης εποχής και να μπορεί να υποστηρίξει την λειτουργία ενός εστιατορίου και πιο συγκεκριμένα ενός ταχυφαγείου. Το ΠΣ δίνει την δυνατότητα στο εστιατόριο να δέχεται παραγγελίες μέσω του Διαδικτύου, είτε απομακρυσμένα, είτε δια ζώσης. Επίσης εμπλουτίζει τις δυνατότητες του εστιατορίου με τέτοιο τρόπο, που ο πελάτης πλέον μπορεί, είτε από το κατάστημα, είτε από το σπίτι του να ενημερωθεί από το ΠΣ και να διενεργήσει την παραγγελία του χωρίς την παρέμβαση τρίτου. Χρησιμοποιώντας το IoT και την πλακέτα esp8266, ο πελάτης ενημερώνεται για την παραγγελία όταν αυτή είναι έτοιμη. Για την κατασκευή του ΠΣ χρησιμοποιήθηκαν τεχνολογίες ανοιχτού κώδικα (HTML, CSS, JavaScript, jQuery, AJAX, Bootstrap, PHP, MySQL).

Το ΠΣ που σχεδιάστηκε έχει δύο είδη χρηστών: τον απλό χρήστη και τον διαχειριστή. Στον απλό χρήστη – πελάτη και μέσω του front-end του ΠΣ παρέχονται λειτουργίες για την

προβολή των απαραίτητων πληροφοριών και την διενέργεια των παραγγελιών που επιθυμεί, αφού συνδεθεί στο σύστημα. Έχει την δυνατότητα σε μία παραγγελία να επιλέξει πολλά προϊόντα και μάλιστα σε όσα είναι φαγώσιμα να κάνει επιλογή για ορισμένα από τα χαρακτηριστικά τους. Μπορεί επίσης, να επεξεργάζεται τις παραγγελίες του αλλά και να τις διαγράφει, ενώ τέλος υπάρχει δυνατότητα να βλέπει το ιστορικό των ολοκληρωμένων παραγγελιών του. Στον χειριστή δίνεται η δυνατότητα της πλήρους διαχείρισης των χρηστών, παραγγελιών, προϊόντων, κατηγοριών προϊόντων, ιστορικού και πλακετών (esp8266). Όπως είναι φανερό δεν έχουν τα δύο είδη χρηστών τα ίδια δικαιώματα. Επίσης το ΠΣ αξιοποιεί συσκευές IoT και πιο συγκεκριμένα πλακέτες esp8266, οι οποίες τοποθετούνται μέσα σε ειδικά σχεδιασμένο και εκτυπωμένο από 3D εκτυπωτή κουτί. Το κουτί αυτό λειτουργεί σαν βομβητής για να ενημερώνει μέσω ηχητικού σήματος και φωτεινής ένδειξης με τον κωδικό παραγγελίας τον πελάτη για την ετοιμότητα της δια ζώσης παραγγελίας του. Επίσης να αναφέρουμε ότι το ΠΣ αυτό είναι ανοιχτού κώδικα, ανοιχτού λογισμικού και ανοιχτού υλικού και είναι διαθέσιμο για να το χρησιμοποιήσει και να το εξελίξει όποιος επιθυμεί χωρίς κόστος.

### 1.3 Ερευνητική επισκόπηση

Διενεργώντας έρευνα για την παρούσα κατάσταση, όσον αφορά υλοποιήσεις ΠΣ και πιο συγκεκριμένα σχεδιασμού και υλοποίησης παραγγελιοληψίας θα δούμε ότι υπάρχουν αρκετές από αυτές.

Αρχικά είναι το σύστημα της Witec [2] το οποίο υποστηρίζει αυτόνομη παραγγελιοδοσία πελάτη, παραγγελιοληψία μέσω σελίδας Facebook και ιστοσελίδας με ενσωματωμένη παραγγελιοληψία, συστήματα αναμονής πελατών και συστήματα προτεραιότητας. Όταν κάνει παραγγελία ο πελάτης του χορηγείται βομβητής με αριθμό παραγγελίας που έχει εμβέλεια έως 100 μέτρα και ο βομβητής ενημερώνει τον πελάτη για την ετοιμότητα της παραγγελίας με χρήση δόνησης ή/και ήχου.

Στην συνέχεια έχουμε το σύστημα της enokey [3] με τους μεγαλύτερους πελάτες, όπως είναι MCDONALD'S και TIM HORTONS, στον τομέα της παραγγελιοληψίας φαγητού μετατρέποντας την διαδικασία παραγγελιοληψίας και αναμονής για την παραλαβή του φαγητού ενδιαφέρουσα. Το σύστημα αυτό έχει ευρείες δυνατότητες και περιγράφοντας τη διαδικασία παραγγελίας για την εφαρμογή του στα MCDONALD'S [4] [5] αξίζει να αναφέρουμε ότι ένας πελάτης εντελώς αυτόνομα κάνει την παραγγελία του από εξαιρετικά

σχεδιασμένα κιόσκια και κατά την αναμονή του μπορεί να ψυχαγωγηθεί σε διαδραστικά τραπέζια.

Επόμενο σύστημα είναι το σύστημα SIGNALGRYD [6] με το οποίο απλά και εύκολα κάνει ο πελάτης την παραγγελία του και του παραδίδεται ένας βομβητής ο οποίος στην συνέχεια τον ενημερώνει για την ετοιμότητα της παραγγελίας. Μπορεί να χρησιμοποιηθεί από επιχειρήσεις εστίασης, όπως εστιατόρια, καφέ και μπαρ, αλλά και άλλα είδη επιχειρήσεων όπως φιλοξενίας, τουρισμού, υγειονομικής περίθαλψης και πολλές άλλες.

Τέλος, είναι η πρόταση για υλοποίηση [7] ηλεκτρονικού συστήματος παραγγελίας φαγητού το οποίο αξιοποιεί την πλακέτα esp8266 και σκοπός της είναι η ασφαλή μεταφορά της παραγγελίας από το εστιατόριο στον πελάτη χρησιμοποιώντας ένα κουτί μεταφορά που κλειδώνει και ανοίγει από το πελάτη με κωδικό μίας χρήσης (One Time Password - OTP) που του αποστέλλεται στο κινητό του τηλέφωνο.

Οι υλοποιήσεις που αναφέρθηκαν παραπάνω χρησιμοποιούνται στον τομέα της εστίασης και παραγγελιοληψίας και αξιοποιούν είτε κάποιο βομβητή είτε μέσω εφαρμογής κινητού ενημερώνουν τον πελάτη για την ετοιμότητα της παραγγελίας. Οι περισσότερες είναι συστήματα που πρέπει κάποιος να πληρώσει για να τα αξιοποιήσει σε αντίθεση με το ΠΣ της διπλωματικής αυτής το οποίο είναι χωρίς κόστος. Τέλος μια από τις προτάσεις χρησιμοποιεί την πλακέτα esp8266 στο ΠΣ της για να εξασφαλίσει την ασφαλή και χωρίς αλλοίωση μεταφορά των προϊόντων μέσω κωδικού μίας χρήσης, το οποίο είναι λίγο διαφορετικό σε σχέση με τα άλλα.

#### **1.4 Σύνοψη της διπλωματικής εργασίας**

Στην εργασία αυτή γίνεται λεπτομερής περιγραφή της διαδικασίας σχεδιασμού ενός ΠΣ για τη διενέργεια παραγγελιών σε επιχειρήσεις εστίασης και πιο συγκεκριμένα εστιατορίων γρήγορου φαγητού. Το πρώτο κεφάλαιο αναφέρεται επιγραμματικά στο θέμα και την υπάρχουσα κατάσταση με ορισμένα στοιχεία που αφορούν τις συνθήκες στο χώρο της εστίασης εντός της ελληνικής επικράτειας. Το δεύτερο κεφάλαιο περιέχει τις τεχνολογίες και τα προγραμματιστικά εργαλεία που αξιοποιήθηκαν. Το τρίτο κεφάλαιο εμπεριέχει την ανάλυση του ΠΣ που δημιουργήθηκε. Στο τέταρτο κεφάλαιο γίνεται παρουσίαση του ΠΣ με τις λειτουργίες και την διεπαφή του χρήστη. Το πέμπτο κεφάλαιο περιλαμβάνει τη σύνοψη, τα συμπεράσματα, τις δυσκολίες και τις πιθανές επεκτάσεις της διπλωματικής εργασίας.

## **1.5 Σύνοψη κεφαλαίου**

Στο κεφάλαιο αυτό έγινε μια επιγραμματική αναφορά για το θέμα και το ΠΣ που σχεδιάστηκε και υλοποιήθηκε στην διπλωματική εργασία. Παράλληλα έγινε μία αναφορά στην παρούσα κατάσταση με αντίστοιχες υλοποιήσεις με τους σκοπούς και τα εργαλεία που αξιοποιήθηκαν. Όπως επίσης έγινε αναφορά των στοιχείων για τις συνθήκες στο χώρο της εστίασης εντός της ελληνικής επικράτειας. Στο επόμενο κεφάλαιο γίνεται ανάλυση του θεωρητικού υπόβαθρου που απαιτείται για τον σχεδιασμό και την υλοποίηση του ΠΣ που προτείνεται.

## 2ο Κεφάλαιο - Θεωρητικό υπόβαθρο

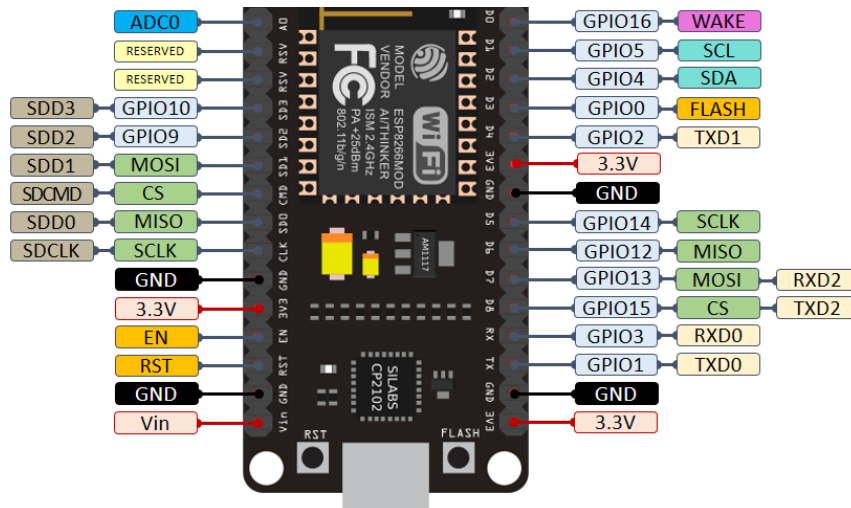
### 2.1 Διαδίκτυο των Πραγμάτων

Η εξέλιξη της τεχνολογίας γενικότερα και ειδικότερα του Διαδικτύου των Πραγμάτων (Internet of Things – IoT) έχει αλλάξει την καθημερινότητα πολιτών, επιχειρήσεων και όχι μόνο με στόχο την σύνδεση όλων το συσκευών στο Διαδίκτυο, ώστε να μπορούμε να αλληλοεπιδρούμε με αυτές ανταλλάσσοντας πληροφορίες. Το IoT έχει αξιοποιηθεί ευρέως με διάφορους τρόπους. Ενδεικτικά να αναφέρουμε ορισμένες εργασίες που έχουν δημοσιευθεί και αφορούν είτε προτάσεις είτε εφαρμογές του IoT, όπως για έξυπνες κατοικίες [8], τη διαχείριση καλλιεργειών [9], τον αθλητισμό [10], σε σουπερμάρκετ [11], στη βιομηχανία [12] και σε επιχειρήσεις εστίασης [13].

### 2.2 Πλακέτα esp8266

Το IoT στις εφαρμογές του αξιοποιεί διάφορες ηλεκτρονικές διατάξεις και μικροεπεξεργαστές. Μία από αυτές είναι η πλακέτα esp8266 η οποία έχει πολλές δυνατότητες και για την οποία γίνεται εκτενή αναφορά στο βιβλίο του Marco Schwartz με τίτλο “Internet of Things with esp8266” [14]. Επιγραμματικά αναφέρουμε ορισμένες από τις δυνατότητές τις, όπως είναι ο έλεγχος led, ανάγνωση από GPIO, συλλογή πληροφοριών από ιστοσελίδα, ανάγνωση πληροφοριών από αισθητήρα, καταγραφή δεδομένων από το Dweet.io, αποτύπωση δεδομένων χρησιμοποιώντας το Freeboard.io, έλεγχο led από οπουδήποτε, διαλειτουργικότητα με υπηρεσίες ιστού, άντληση δεδομένων καιρού από το Yahoo, αποστολή δεδομένων θερμοκρασίας και υγρασίας στο Twitter, δημιουργία δημοσιεύσεων στο Facebook, απλή επικοινωνία μηχανής σε μηχανή, αποστολή ειδοποιήσεων με email, αποστολή δεδομένων με μήνυμα κειμένου, λήψη ειδοποιήσεων push, διαχείριση και έλεγχο κλειδαριάς από οπουδήποτε, λήψη ειδοποιήσεων όταν η κλειδαριά είναι ανοιχτή, διασύνδεση υπηρεσιών Bitcoin, αυτοματισμοί σπιτιού και έλεγχος από οπουδήποτε, αυτοματισμοί κήπου, δημιουργία ειδοποιήσεων για πότισμα φυτών, παρακολούθηση της θερμοκρασίας και υγρασία, διαχείριση Robot από οπουδήποτε, δημιουργία της δικής σου πλατφόρμας εξ αποστάσεως ελέγχου.

Αναλυτικότερα η πλακέτα esp8266 που φαίνεται στην παρακάτω Εικόνα 1 δέχεται τροφοδοσία από USB και παρέχει τροφοδοσία 3.3 Volt και 5.0 Volt μέσω των 3V3 και Vin ακροδεκτών της. Επίσης διαθέτει 17 GPIO ακροδέκτες εισόδου/εξόδου και 4 GND ακροδέκτες γείωσης.



Εικόνα 1: Διάγραμμα των ακροδεκτών του esp8266 NodeMCU

## 2.3 Arduino IDE

Το Arduino IDE (Integrated Development Environment) [15] είναι μια εφαρμογή ανοιχτού λογισμικού ανοιχτού κώδικα για συγγραφή κώδικα για λειτουργικά συστήματα Windows, macOS και Linux. Μπορεί να χρησιμοποιηθεί για πλακέτες Arduino για να προγραμματιστούν οι είσοδοι και έξοδοι του, αλλά και για άλλες συμβατές πλακέτες όπως το esp8266. Η γλώσσα που υποστηρίζει είναι η C και η C++ με ειδικούς κανόνες και δομές κώδικα. Παρέχει βιβλιοθήκη για το Wiring έργο, το οποίο παρέχει πολλές κοινές διαδικασίες εισόδου και εξόδου. Απαιτεί μόνο δύο βασικές συναρτήσεις την void setup και την void loop για να δημιουργηθεί ένα πρόγραμμα.

## 2.4 3D εκτυπωτής Prusa i3

Ο 3D εκτυπωτής Prusa i3 [16] κατασκευάστηκε από την Τσέχικη εταιρία Prusa Research με το εμπορικό σήμα Original Prusa i3 και ο εφευρέτης τους είναι ο Josef Průša. Οι εκτυπωτές αυτοί σύμφωνα με το 3DHubs το 2016 [17] ήταν οι πιο χρησιμοποιημένοι εκτυπωτές στον κόσμο. Το συγκρίσιμο χαμηλό κόστος και η ευκολία κατασκευής και τροποποίησης, τους έκανε δημοφιλείς στην εκπαίδευση, σε χομπίστες και επαγγελματίες. Δεδομένου ότι η σειρά i3 είναι ανοιχτού κώδικα, έχουν γίνει πολλές παραλλαγές από επιχειρήσεις και ιδιώτες.

## 2.5 PrusaSlicer

Το PrusaSlicer παλαιότερα γνωστό ως Prusa Edition ή Slic3r PE [18] είναι βασισμένο στο έργο ανοιχτού κώδικα Slic3r με πλούσια χαρακτηριστικά και συχνή ενημέρωση και παρέχει όλα όσα χρειάζεστε κάποιος για την εξαγωγή αρχείων εκτύπωσης για 3D εκτυπωτές

Original Prusa και όχι μόνο. Το PrusaSlicer είναι διαθέσιμο για εγκατάσταση σε λειτουργικά συστήματα Microsoft Windows, MacOS και Linux. Το έργο Slic3r γεννήθηκε το 2011 [19] μέσα στην κοινότητα RepRap με την καθοδήγηση του Alessandro Ranellucci. Όταν ο Josef Průša έψαχνε για ένα κορυφαίο λογισμικό τεμαχισμού για τους 3D εκτυπωτές Original Prusa επέλεξε το Slic3r το οποίο ήταν και παραμένει λογισμικό ανοιχτού κώδικα.

## 2.6 Fritzing

Το fritzing είναι μία εφαρμογή ανοιχτού λογισμικού και ανοιχτού κώδικα για την δημιουργία και τον σχεδιασμό κυκλωμάτων. Η εφαρμογή αυτή έχει δημιουργηθεί από την ομάδα του Καθηγητή Reto Wettach, , André Knörrig, Jonathan Cohen και Stefan Hermann του πανεπιστήμιου Fachhochschule Potsdam University of Applied Sciences και υποστηρίζεται από την ομάδα Friends-of-Fritzing e.V. με έδρα το Βερολίνο της Γερμανίας [20]. Μπορεί να χρησιμοποιηθεί από κατασκευαστές, ερευνητές και χομπίστες για να σχεδιασμό κυκλωμάτων.

## 2.7 HTML

Η Hypertext Markup Language (HTML) ως γλώσσα προγραμματισμού αποτελεί το βασικό εργαλείο για την δημιουργία ιστοσελίδων και εφαρμογών ιστού. Χρησιμοποιώντας ετικέτες, όπως `<p> eBuzzer </p>` ή `` εμφανίζει τα περιεχόμενα που περιγράφονται από την ετικέτα σε μία ιστοσελίδα. Με την χρήση ετικετών μπορεί να εμφανίσει στοιχεία, όπως κείμενο, εικόνες, βίντεο, συνδέσμους, φόρμες.

Τα τελευταία χρόνια έχουν εξελιχθεί πολύ οι τεχνολογίες ιστού δίνοντας περισσότερες δυνατότητες σε προγραμματιστές να υλοποιήσουν εφαρμογές για μεγαλύτερη ποικιλία συσκευών και συστημάτων. Η συνεχής εξέλιξη του προτύπου HTML το οποίο διαχρονικά ξεκίνησε από την έκδοση 1.0 το 1991 έως την έκδοση 5.2 το 2017 [21] και πλέον μπορεί κάποιος να το παρακολουθεί ως ένα πρότυπο διαρκής εξέλιξης [22]. Συνεπώς η γλώσσα προγραμματισμού HTML αποτελεί την βάση στην οποία στηρίζονται όλες οι εφαρμογές ιστού.

## 2.8 CSS

Πολύ σημαντική τεχνολογία που χρησιμοποιείται σε συνδυασμό με την HTML και στην ουσία η συνύπαρξή τους είναι σχεδόν υποχρεωτική είναι η τεχνολογία Cascading Stylesheets (CSS) [23] με την οποία μπορούμε να δώσουμε στυλ σε ιστοσελίδες και εφαρμογές ιστού. Στην ουσία με τη χρήση CSS καθορίζεται η εμφάνιση σε ιστοσελίδες και εφαρμογές ιστού. Επίσης,

ανάλογα με τη συσκευή που χρησιμοποιείται, καθορίζεται η εμφάνιση στις ιστοσελίδες και στις εφαρμογές ιστού.

Η τεχνολογία CSS υποστηρίζεται και εξελίσσεται από ομάδα ανθρώπων εντός του Διεθνούς οργανισμού World Wide Web Consortium (W3C) [24] και είναι μέλη της CSS Working Group [25]. Η ομάδα αυτή ανά διαστήματα δημοσιεύει τα λεγόμενα CSS specification [26].

Η CSS ως γλώσσα προγραμματισμού<sup>1</sup> αποτελεί και αυτή με την σειρά της, σε συνδυασμό με την HTML, την βάση στην οποία στηρίζονται όλες οι εφαρμογές ιστού κυρίως ως προς τον τρόπο εμφάνισης.

## 2.9 JavaScript

Η γλώσσα προγραμματισμού JavaScript (JS) [27] αποτελεί και αυτή μία από τις βασικές τεχνολογίες του παγκόσμιου ιστού (World Wide Web – WWW). Η JS ξεκίνησε από τη Netscape [28] επίσημα το 1995 και κατέληξε σήμερα να είναι μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού στον παγκόσμιο ιστό. Τα τελευταία χρόνια έχουν δημιουργηθεί με βάση την JS ορισμένα framework τα οποία παρέχουν ορισμένες λειτουργίες ώστε να μην χρειαστεί να ξεκινήσεις από το μηδέν. Ορισμένα από τα JS Frameworks είναι το Node.js [29], Angular [30], React [31], το Vue.js [32], το Ember.js [33], το Meteor [34]. Επίσης υπάρχουν πολύ διαδεδομένες βιβλιοθήκες σε JavaScript, όπως η jQuery [35] που είναι σχεδιασμένη για λειτουργίες από την πλευρά του χρήστη. Η τεχνική Asynchronous JavaScript and XML (AJAX) που χρησιμοποιείται στον προγραμματισμό ιστού βασίζεται στο προγραμματιστικό της μέρος στη JS.

## 2.10 jQuery

Η βιβλιοθήκη jQuery [36] της JS, είναι σχεδιασμένη για να απλοποιεί λειτουργίες από την πλευρά του χρήστη (client side). Οι λειτουργίες αυτές μπορεί να είναι έλεγχος γεγονότων (event handling), κινούμενη εικόνα με CSS (CSS animation) και AJAX. Είναι βιβλιοθήκη ανοιχτού κώδικα και χρησιμοποιείται από το 73% των δέκα εκατομμυρίων δημοφιλέστερων ιστοσελίδων στον κόσμο [37]. Η βιβλιοθήκη jQuery αρχικά γράφτηκε από τον John Resig και

---

<sup>1</sup> Την CSS την αναφέρει εκτός από τεχνολογία και ως γλώσσα στο CSS Snapshot 2020 [26]



η πρώτη επίσημη έκδοσή της ήταν στις 26 Αυγούστου του 2006. Αυτή τη στιγμή υποστηρίζεται από την ομάδα The jQuery Team του ιδρύματος OpenJS Foundation [38].

## 2.11 AJAX

Η τεχνική AJAX είναι ένα σύνολο από τεχνικές και τεχνολογίες ιστού για την ασύγχρονη επικοινωνία των εφαρμογών ιστού. Στην ουσία αυτό που επιτυγχάνει η τεχνική αυτή είναι να δίνει την αίσθηση desktop εφαρμογή σε εφαρμογές ιστού. Αυτό επιτυγχάνεται με την ασύγχρονη μεταφορά των δεδομένων που ανταλλάσσουν ο εξερευνητής ιστού (web browser) με τον εξυπηρετητή ιστού (web server). Η διαδικασία αυτή γίνεται στο παρασκήνιο χωρίς να ξαναφορτώνεται η σελίδα ιστού, παρά μόνο τα δεδομένα που απαιτείται να αλλάξουν. Η τεχνική αυτή πρωτοεμφανίστηκε τον Μάρτιο του 1999 [39].

## 2.12 Bootstrap

Το Bootstrap [40] είναι ένα ανοιχτού κώδικα CSS framework με βασικό προσανατολισμό στις λειτουργίες του την δημιουργία εφαρμογών ιστού που ανταποκρίνονται σε μεγάλο εύρος συσκευών, ως προς την εμφάνισή τους, έτσι ώστε οι εφαρμογές αυτές να είναι responsive. Αρχικά το bootstrap είχε ονομαστεί Twitter Blueprint, το οποίο είχε υλοποιηθεί από τους Mark Otto και Jacob Thornton στην εταιρία Twitter ως framework για την συνοχή των εσωτερικών εργαλείων της εταιρίας. Η μετονομασία του από Twitter Blueprint σε Bootstrap έγινε 11 Αυγούστου του 2011 ως έργο ανοιχτού κώδικα. Το bootstrap υποστηρίζεται από την ομάδα Bootstrap Core Team [41] με την έκδοση 5.1.0 να είναι η τρέχουσα έκδοση.

## 2.13 PHP

Η γλώσσα προγραμματισμού PHP (Personal Home Page) [42] είναι scripting γλώσσα προγραμματισμού γενικού σκοπού που χρησιμοποιείται κυρίως για ανάπτυξη εφαρμογών ιστού. Η PHP είναι μια γλώσσα προγραμματισμού «server side». Δηλαδή η εκτέλεση των προγραμμάτων PHP λαμβάνει χώρα στον εξυπηρετητή ιστού από τον διερμηνευτή PHP, είτε ως πρόσθετο (module), είτε ως υπηρεσία δαίμονας (daemon), είτε ως κοινή διεπαφή πύλης (Common Gateway Interface - CGI). Το αποτέλεσμα της εκτέλεσης των προγραμμάτων PHP μπορεί να είναι οποιουδήποτε τύπου, όπως HTML ή δυαδικά δεδομένα εικόνας, και μπορεί να είναι ολόκληρο ή μέρος από μια απόκριση http (http response). Όπως και σε άλλες γλώσσες προγραμματισμού έτσι και στην PHP έχουν δημιουργηθεί αρκετά PHP framework, όπως το Laravel [43], CodeIgniter [44], Symfony [45], CakePHP [46], Yii [47], Laminas (Zend Framework) [48]. Η PHP ξεκίνησε από τον Rasmus Lerdorf το 1994 με την πρώτη έκδοση της

στις 8 Ιουνίου του 1995. Από το 2001 και μέχρι σήμερα η PHP υποστηρίζεται από την ομάδα The PHP Group [49] με τρέχουσες εκδόσεις την 7.4.22 και την 8.0.9, που αναπτύσσονται παράλληλα. Σύμφωνα με στατιστικά στοιχεία της W<sup>3</sup>Techs [50] η PHP χρησιμοποιείται από το 79.0% των ιστοτόπων με server-side προγραμματισμό.

## 2.14 MySQL/MariaDB

Το MySQL [51] είναι ένα από τα δημοφιλέστερα Συστήματα Διαχείρισης Σχεσιακής Βάσης Δεδομένων – ΣΔΣΒΔ (Relational Database Management System – RDBMS) ανοιχτού κώδικα που χρησιμοποιείται για να υποστηρίξει εφαρμογές ιστού και όχι μόνο. Το όνομά του το πήρε από το My (όνομα της κόρης του συνιδρυτή Michael Widenius) και το SQL (Structured Query Language). Στην Σχεσιακή Βάση Δεδομένων – ΣΒΔ (Relational Database – RDB) τα δεδομένα οργανώνονται σε πίνακες, που συσχετίζονται μεταξύ τους με τέτοιο τρόπο, ώστε να βοηθούν στην καλύτερη οργάνωση και δομή τους. Η Γλώσσα Δομημένων Ερωτημάτων - ΓΔΕ (Structured Query Language - SQL) είναι γλώσσα προγραμματισμού για ΣΔΣΒΔ. Στις 29 Οκτωβρίου του 2009, για να διατηρήσει τον χαρακτήρα του ανοιχτού κώδικα λογισμικού το MySQL, δημιουργήθηκε το MariaDB [52] σαν συνέχεια του με σκοπό να διατηρήσει υψηλή συμβατότητα με το MySQL. Το MySQL/MariaDB είναι επίσης βασικό συστατικό στο LAMP (Linux, Apache, MySQL/MariaDB, PHP/Perl/Python), το WAMP, (Windows, Apache, MySQL/MariaDB, and PHP), το XAMPP (X, Apache, MySQL/MariaDB, PHP, Perl).

## 2.15 Notepad++

Το Notepad++ [53] είναι ένας ανοιχτού λογισμικού ανοιχτού κώδικα επεξεργαστής κειμένου εξειδικευμένος για συγγραφή κώδικα για μεγάλη ποικιλία από γλώσσες προγραμματισμού. Διατίθεται δωρεάν υπό GPL License version 3 και είναι βασισμένο στο Scintilla [54] και είναι γραμμένο σε C++. Έχει χρωματική απεικόνιση του κώδικα ανάλογα τη γλώσσα προγραμματισμού συγγραφής ώστε να βοηθάει τον εκάστοτε προγραμματιστή στη συγγραφή των προγραμμάτων του.

## 2.16 WinSCP

Το WinSCP [55] είναι ένα ανοιχτού λογισμικού ανοιχτού κώδικα για ftp, sftp, WebDAV, S3, SCP πρόγραμμα πελάτη για το λειτουργικό σύστημα Windows της Microsoft. Ο βασικός σκοπός του είναι η μεταφορά αρχείων από ένα τοπικό σύστημα σε ένα απομακρυσμένο και αντίστροφα. Ξεκίνησε ως έργο [56] του Πανεπιστημίου Οικονομικών της Πράγας τον Μάρτιο

του 2000. Εκτός από μεταφορά αρχείων προσφέρει δέσμες ενεργειών και λειτουργίες διαχείρισης αρχείων.

## 2.17 PuTTY

Το PuTTY [57] είναι μία δωρεών εφαρμογή ανοιχτού λογισμικού ανοιχτού κώδικα για Telnet και SSH, μαζί με έναν εξομοιωτή τερματικού xTerm. Η εφαρμογή έχει υλοποιηθεί από τον Simon Tatham και η πρώτη έκδοση δημοσιεύτηκε 8 Ιανουαρίου του 1999 [58] για λειτουργικό σύστημα Microsoft Windows. Αργότερα δημοσιεύτηκαν εκδόσεις και για άλλα λειτουργικά συστήματα παρόμοια με UNIX.

## 2.18 Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο παρουσιάστηκαν τα προγραμματιστικά εργαλεία και τα εργαλεία υλικού που αξιοποιήθηκαν για τον σχεδιασμό και την υλοποίηση της διπλωματικής αυτής. Δόθηκαν γενικές και ιστορικές πληροφορίες σχετικά με αυτά, ώστε να δοθεί η δυνατότητα άμεσης εύρεσης πληροφοριών σε όποιον επιθυμεί. Στο κεφάλαιο που ακολουθεί θα γίνει ανάλυση του ΠΣ eBuzzer, όσον αφορά την σύλληψη της ιδέας, τις απαιτήσεις, τον σχεδιασμό της βάσης, την ασφάλεια, τον τρόπο διασύνδεσης με την πλακέτα esp8266, το πλάνο ελέγχου ορθής λειτουργίας, την υποδομή, τα αρχεία και τον κώδικα του.

## 3ο Κεφάλαιο - Ανάλυση Πληροφοριακού Συστήματος eBuzzer

Στο συγκεκριμένο κεφάλαιο θα αναλυθεί με λεπτομέρεια το ΠΣ eBuzzer που δημιουργήθηκε στην παρούσα διπλωματική εργασία. Πραγματοποιείται η ανάλυση του σχεδιασμού και της υλοποίησής του η οποία περιλαμβάνει: τη σύλληψη της ιδέας, τις απαιτήσεις του ΠΣ, την βάση δεδομένων, τη διασύνδεση με την πλακέτα esp8266, την ανάλυση της υποδομής, του κώδικα και την ασφάλεια.

### 3.1 Σύλληψη ιδέας

Η ιδέα για την υλοποίηση ΠΣ υποστήριξης ταχυφαγείου προήλθε μετά από την ανακοίνωση των θεμάτων από τον επιβλέποντα καθηγητή μου, σε συνδυασμό με την εμπειρία μου ως επισκέπτης, σε εστιατόριο που λειτουργούσε με ανάλογο τρόπο. Ο πελάτης με την επίσκεψή του στο εν λόγω εστιατόριο έκανε την παραγγελία του. στην συνέχεια το προσωπικό του εστιατορίου του παρέδιδε μια συσκευή. Μόλις η παραγγελία ήταν έτοιμη, ο πελάτης ενημερωνόταν με ηχητική ειδοποίηση από την συσκευή για την παραλαβή της. Αυτή η διαδικασία αποτέλεσε το έναυσμα για τη σχεδίαση και την υλοποίηση του ΠΣ eBuzzer. Επομένως, από τα θέματα που ανακοινώθηκαν από τον επιβλέπον καθηγητή, επιλέχθηκε η υλοποίηση ΠΣ που θα υποστηρίξει εστιατόρια και πιο συγκεκριμένα ταχυφαγεία με δυνατότητες παραγγελιών εξ' αποστάσεως μέσω ιστοσελίδας. Αντιθέτως, για τις δια ζώσης παραγγελίες θα χρησιμοποιεί συσκευή ενημέρωσης του πελάτη για την ετοιμότητα της παραγγελίας του.

### 3.2 Απαιτήσεις Πληροφοριακού Συστήματος eBuzzer

Αναλύοντας τις απαιτήσεις του ΠΣ εντοπίζουμε ότι ως μια εφαρμογή ιστού έχει λειτουργίες χρήστη (front-end) και λειτουργίες διαχείρισης (back-end). Επομένως θα πρέπει να υπάρχουν τουλάχιστον δύο ρόλοι στο ΠΣ, όσον αφορά τους χρήστες του. Ο ρόλος του διαχειριστή και ο ρόλος του απλού χρήστη (πελάτη). Επίσης, υπάρχουν και οι απαιτήσεις του ΠΣ σε σχέση με τις λειτουργίες που θα υποστηρίζει τη βάση δεδομένων του, καθώς και απαιτήσεις σχετικά με τη συσκευή που θα χρησιμοποιεί την πλακέτα esp8266 προς ενημέρωση του πελάτη για την ετοιμότητα της παραγγελίας.

#### 3.2.1 Ρόλος διαχειριστή

Ο διαχειριστής έχει πλήρη δικαιώματα χρήσης και εκτέλεσης όλων των λειτουργιών του ΠΣ και πιο συγκεκριμένα τις παρακάτω:

- Σύνδεση στο σύστημα με το όνομα χρήστη (username) και τον κωδικό πρόσβασης (password) του.
- Διαχείριση χρηστών (αναζήτηση, εισαγωγή, επεξεργασία, διαγραφή, χαρακτηρισμός κατηγορίας χρήστη).
- Διαχείριση κατηγοριών προϊόντων (εισαγωγή, επεξεργασία, διαγραφή).
- Διαχείριση προϊόντων (εισαγωγή, επεξεργασία, διαγραφή).
- Διαχείριση πλακετών esp8266 (εισαγωγή, επεξεργασία, διαγραφή).
- Διαχείριση παραγγελιών (αναζήτηση, εισαγωγή, επεξεργασία, χαρακτηρισμός παραγγελίας, αντιστοίχιση παραγγελίας με esp8266, διαγραφή).
- Προβολή ιστορικού παραγγελιών όλων των χρηστών.
- Αποσύνδεση από το σύστημα.

### 3.2.2 Ρόλος απλού χρήστη

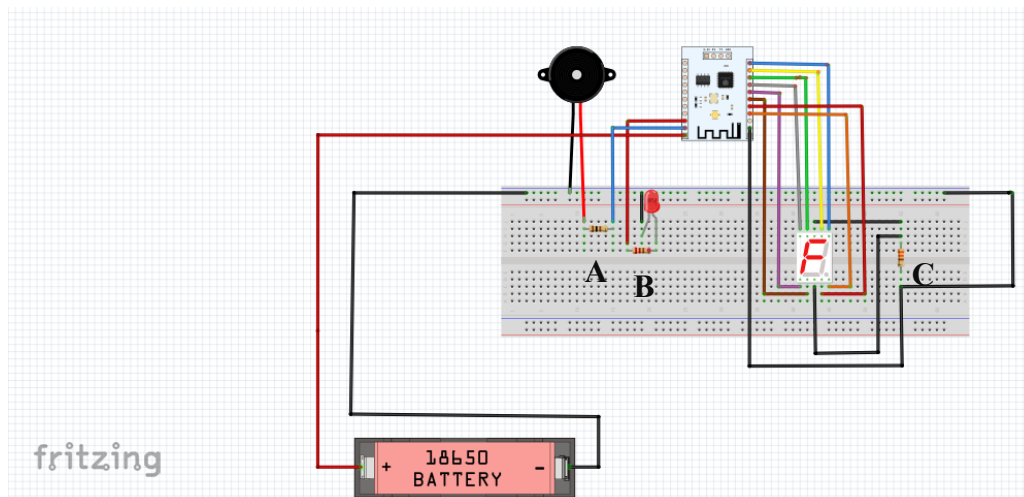
Ο απλός χρήσης έχει δικαιώματα χρήσης και εκτέλεσης συγκεκριμένων λειτουργιών του ΠΣ και πιο συγκεκριμένα τις παρακάτω:

- Εγγραφή χρήστη (Καταχώρηση όλων των στοιχείων του: Όνομα, Επώνυμο, Όνομα Χρήστη - Username, Κωδικό Πρόσβασης – Password, Ηλεκτρονικό Ταχυδρομείο, Διεύθυνση, Τηλέφωνο).
- Σύνδεση στο σύστημα με το όνομα χρήστη (username) και τον κωδικό πρόσβασης (password) του.
- Διαχείριση του προφίλ του (Επεξεργασία όλων των στοιχείων του: Όνομα, Επώνυμο, Όνομα Χρήστη - Username, Κωδικό Πρόσβασης – Password, Ηλεκτρονικό Ταχυδρομείο, Διεύθυνση, Τηλέφωνο).
- Διενέργεια παραγγελιών με χρήση καλαθιού (εισαγωγή, διαγραφή, ολοκλήρωση παραγγελίας).
- Προβολή ιστορικού παραγγελιών.
- Αποσύνδεση από το σύστημα.

### 3.2.3 Απαιτήσεις συσκευής με χρήση της πλακέτας esp8266

Για την ομαλή λειτουργία του συστήματος και την κατασκευή της συσκευής με χρήση της πλακέτας esp8266, έπρεπε να δημιουργηθεί το σχεδιάγραμμα του κυκλώματος με τις συνδέσεις στα επιμέρους στοιχεία που απαιτούνται. Στην παρακάτω Εικόνα 2 φαίνεται το κύκλωμα με όλες τις συνδέσεις και τα στοιχεία της συσκευής, το οποίο δημιουργήθηκε με το

πρόγραμμα fritzing. Όπως φαίνεται αποτελείται από μία μπαταρία, ένα μεγάφωνο, ένα led, μία οθόνη και ένα esp8266 με τις συνδέσεις τους. Επίσης, στη θέση A έχει τοποθετηθεί αντίσταση 110 Ω για την προστασία του μεγάφωνου, στην θέση B τοποθετήθηκε αντίσταση 220 Ω για την προστασία του led και στην θέση C έχει τοποθετηθεί αντίσταση 330 Ω για την προστασία της οθόνης.



Εικόνα 2: Κύκλωμα βομβητή με esp8266

### 3.2.4 Απαιτήσεις λειτουργιών και βάσης δεδομένων

Οι λειτουργίες του ΠΣ πρέπει να είναι σε αρμονία με την βάση δεδομένων που το υποστηρίζει. Οι απαιτήσεις του συστήματος όσον αφορά τις λειτουργίες είναι οι εξής:

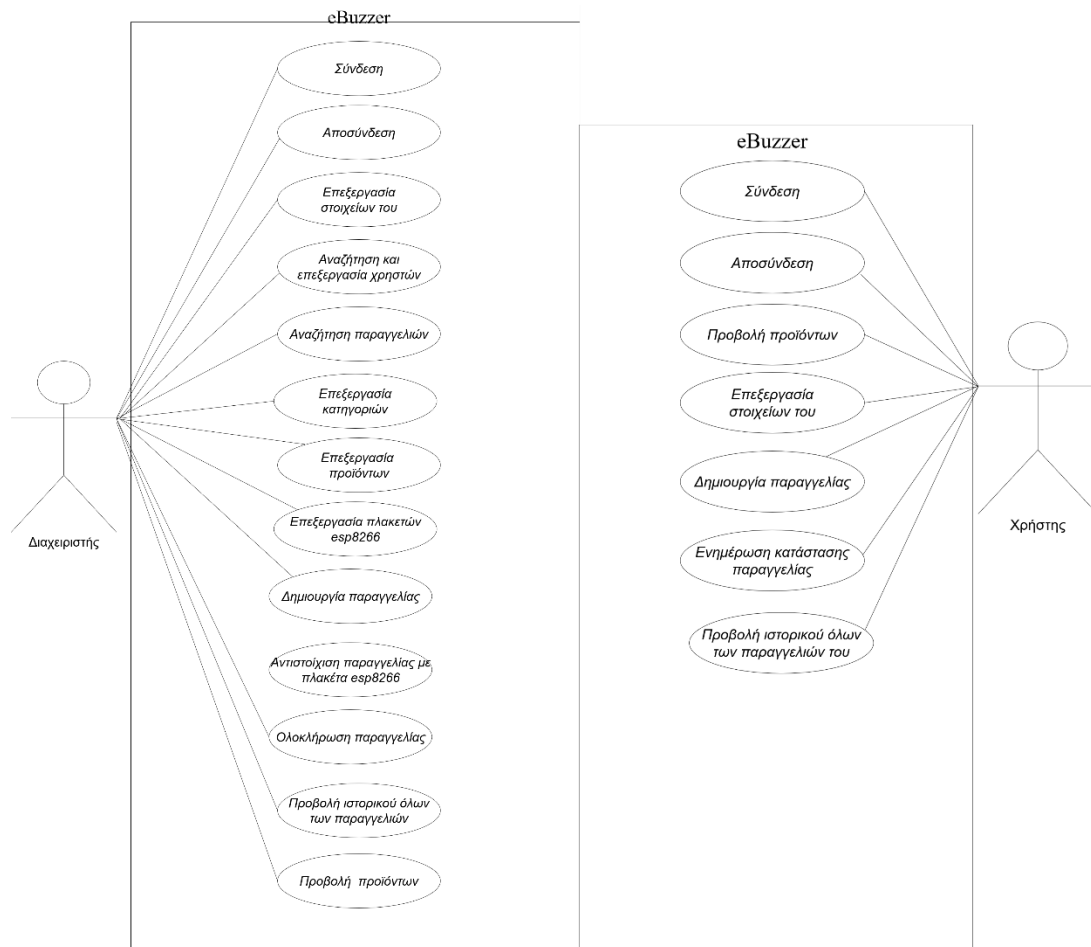
- Ένας χρήστης μπορεί να κάνει πολλές παραγγελίες, όμως μία παραγγελία μπορεί να γίνει μόνο από έναν χρήστη.
- Ένα esp8266 μπορεί να αντιστοιχηθεί σε πολλές παραγγελίες, όμως μια παραγγελία μπορεί να την αντιστοιχίσουμε με ένα μόνο esp8266.
- Μία παραγγελία μπορεί να έχει πολλά προϊόντα και ένα προϊόν μπορεί να ανήκει σε πολλές παραγγελίες.
- Σε μία κατηγορία μπορούν να ανήκουν πολλά προϊόντα, αλλά ένα προϊόν μπορεί να ανήκει σε μόνο μία κατηγορία.
- Να τηρείται ιστορικό για τις ολοκληρωμένες παραγγελίες.

### 3.2.5 Use Case Diagram

Ένα διάγραμμα περιπτώσεων χρήσης (Use Case Diagram) στην πιο απλή μορφή του αναπαριστά την αλληλεπίδραση ενός χρήστη με το σύστημα και δείχνει τη σχέση μεταξύ του χρήστη και τις διάφορες περιπτώσεις χρήσης με τις οποίες εμπλέκεται.

Ένα διάγραμμα περιπτώσεων χρήσης μπορεί να προσδιορίσει τους τύπους χρηστών ενός συστήματος και τις περιπτώσεις χρήσης τους, επίσης συνήθως συνοδεύεται και από άλλους τύπους διαγραμμάτων. Οι περιπτώσεις χρήσης συνήθως αντιπροσωπεύονται από κύκλο.

Στην Εικόνα 3 που ακολουθεί φαίνεται το διάγραμμα χρήσης του ΠΣ, στο οποίο φαίνονται όλες οι λειτουργίες που μπορεί να υλοποιήσει ο απλός χρήστης αλλά και ο διαχειριστής του.







Εικόνα 3: Διάγραμμα χρήσης ΠΣ

Αμέσως μετά ακολουθεί περιγραφή και ανάλυση των απαιτήσεων του ΠΣ σε σχέση με τη βάση δεδομένων και το σχεσιακό διάγραμμα της.

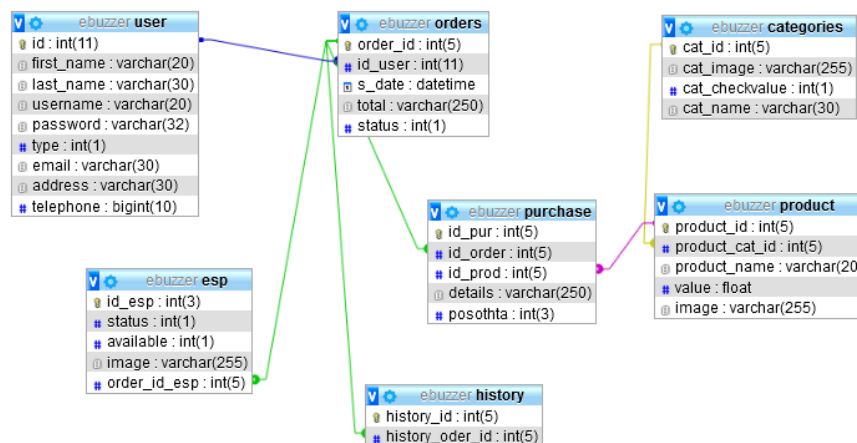
### 3.3 Σχεσιακό Διάγραμμα της Βάσης Δεδομένων

Μετά την ανάλυση των απαιτήσεων του ΠΣ ακολούθησε ο σχεδιασμός της βάσης δεδομένων ξεκινώντας από τη δημιουργία του Σχεσιακού Διαγράμματος της Βάσης Δεδομένων (ΣΔΒΔ). Κατά τον σχεδιασμό του ΣΔΒΔ ακολουθήθηκαν οι κανόνες κανονικοποίησης (Normalization). Εντοπίστηκε μία σχέση πολλά προς πολλά μεταξύ των παραγγελιών και των

προϊόντων και αντιμετωπίστηκε προσθέτοντας μία νέα οντότητα τις αγορές purchase. Στο σχεσιακό διάγραμμα της βάσης δεδομένων του ΠΣ, όπως φαίνεται στην Εικόνα 4 αναπαρίστανται τα δεδομένα και οι μεταξύ τους σχέσεις με την μορφή πινάκων. Ο κάθε πίνακας αναπαρίσταται με ένα τετράγωνο στον πάνω μέρος του οποίου υπάρχει το όνομα του πίνακα και κάτω από το όνομα του πίνακα υπάρχουν υπό την μορφή γραμμών τα πεδία του πίνακα. Το κάθε πεδίο περιέχει ένα μοναδικό όνομα, έναν τύπο δεδομένων και ένα εικονίδιο. Ανάλογα το εικονίδιο, αυτό συμβολίζει είτε το κύριο κλειδί , είτε τον τύπο δεδομένων αλφαριθμητικού , είτε τον τύπο δεδομένων αριθμού , είτε τον τύπο δεδομένων ημερομηνίας . Στον τύπο δεδομένων των περισσότερων πεδίων υπάρχει μια παρένθεση με έναν αριθμό, ο οποίος συμβολίζει το μέγεθός τους για τους αλφαριθμητικούς και το μέγεθος εμφάνισής τους για τους ακεραίους. Παράδειγμα στο password: varchar(32) ο αριθμός 32 συμβολίζει το μέγεθος του varchar, το οποίο αποτελείται έως και 32 χαρακτήρες.

Επίσης, στο σχεσιακό διάγραμμα της βάσης δεδομένων του ΠΣ υπάρχουν οι συνδέσεις μεταξύ των πινάκων, οι οποίες συμβολίζουν τις σχέσεις τους και συνδέουν τα κύρια κλειδιά (primary keys) με τα ξένα κλειδιά (foreign keys). Οι σχέσεις των πινάκων μεταξύ τους είναι ως εξής:

- Ο πίνακας user είναι ένα προς πολλά ή κανένα με τον πίνακα orders.
- Ο πίνακας orders είναι ένα προς πολλά με τον πίνακα esp, history, purchase.
- Ο πίνακας product είναι ένα προς πολλά με τον πίνακα purchase.
- Ο πίνακας categories είναι ένα προς πολλά με τον πίνακα product.



Εικόνα 4: Σχεσιακό Διάγραμμα Βάσης Δεδομένων eBuzzer



### 3.4 Ανάλυση της Βάσης Δεδομένων του ΠΣ

Το ΠΣ αποτελείται από επτά (7) πίνακες, που φαίνονται στην Εικόνα 5, για την αποθήκευση των δεδομένων, ώστε να λειτουργήσει όπως σχεδιάστηκε. Στην συνέχεια θα ακολουθήσει λεπτομερής ανάλυση του κάθε πίνακα.

Table	Action
<input type="checkbox"/> categories	★ Browse
<input type="checkbox"/> esp	★ Browse
<input type="checkbox"/> history	★ Browse
<input type="checkbox"/> orders	★ Browse
<input type="checkbox"/> product	★ Browse
<input type="checkbox"/> purchase	★ Browse
<input type="checkbox"/> user	★ Browse
<b>7 tables</b>	<b>Sum</b>

Εικόνα 5: Πίνακες της βάσης Δεδομένων

#### 3.4.1 Πίνακας user

Στον πίνακα user είναι αποθηκευμένες οι εγγραφές με τους χρήστες του συστήματος και είναι ένας από τους βασικούς πίνακες σε κάθε ΠΣ που έχει διαχείριση χρηστών στις λειτουργίες του. Στην Εικόνα 6 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι εννέα (9) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/> 1	id 🔑	int(11)
<input type="checkbox"/> 2	first_name	varchar(20)
<input type="checkbox"/> 3	last_name	varchar(30)
<input type="checkbox"/> 4	username	varchar(20)
<input type="checkbox"/> 5	password	varchar(32)
<input type="checkbox"/> 6	type	int(1)
<input type="checkbox"/> 7	email	varchar(30)
<input type="checkbox"/> 8	address	varchar(30)
<input type="checkbox"/> 9	telephone	bigint(10)

Εικόνα 6: Πίνακας user

- **id:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης έντεκα (11) ψηφίων και χαρακτηρίζει μοναδικά τον κάθε χρήστη. Αυξάνεται αυτόματα (auto\_increment) σε κάθε νέα εγγραφή στον πίνακα.

- **first\_name:** Είναι το πεδίο που αποθηκεύεται το όνομα για κάθε χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους είκοσι (20) χαρακτήρων.
- **last\_name:** Είναι το πεδίο που αποθηκεύεται το επώνυμο για κάθε χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους τριάντα (30) χαρακτήρων.
- **username:** Είναι το πεδίο που αποθηκεύεται το όνομα χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους είκοσι (20) χαρακτήρων.
- **password:** Είναι το πεδίο που αποθηκεύεται ο κωδικός για κάθε χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους τριάντα δύο (32) χαρακτήρων. Ο κωδικός αποθηκεύεται κρυπτογραφημένος με md5 αλγόριθμο.
- **type:** Είναι το πεδίο που δέχεται τιμή 0 ή 1 ανάλογα αν ο χρήστης είναι απλός ή διαχειριστής. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός με ένα (1) ψηφίο εμφάνισης.
- **email:** Είναι το πεδίο που αποθηκεύεται η ηλεκτρονική διεύθυνση του κάθε χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους τριάντα (30) χαρακτήρων.
- **address:** Είναι το πεδίο που αποθηκεύεται η διεύθυνση για κάθε χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους τριάντα (30) χαρακτήρων.
- **telephone:** Είναι το πεδίο που αποθηκεύεται το τηλέφωνο του κάθε χρήστη με τύπος δεδομένων μεγάλο ακέραιο εμφάνισης δέκα (10) ψηφίων.

### 3.4.2 Πίνακας orders

Στον πίνακα orders είναι αποθηκευμένες οι εγγραφές με τις παραγγελίες του χρήστη. Ένας χρήσης μπορεί να κάνει πολλές παραγγελίες, αλλά όχι ταυτόχρονα. Για την κάθε παραγγελία αντιστοιχίζεται και μια πλακέτα esp. Επίσης κάθε παραγγελία μπορεί να έχει πολλά προϊόντα και κατά την ολοκλήρωσή της αποθηκεύεται εγγραφή στον πίνακα history. Στην Εικόνα 7 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι πέντε (5) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/> 1	order_id	int(5)
<input type="checkbox"/> 2	id_user	int(11)
<input type="checkbox"/> 3	s_date	datetime
<input type="checkbox"/> 4	total	varchar(250)
<input type="checkbox"/> 5	status	int(1)

Εικόνα 7: Πίνακας orders

- **order\_id:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης πέντε (5) ψηφίων και χαρακτηρίζει μοναδικά την κάθε παραγγελία.
- **id\_user:** Είναι το ξένο κλειδί του πεδίου id του πίνακα user και συσχετίζει μοναδικά μία παραγγελία με ένα χρήστη. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης έντεκα (11) ψηφίων.
- **s\_date:** Είναι το πεδίο που αποθηκεύεται η ημερομηνία και ώρα ολοκλήρωσης για κάθε παραγγελία με τύπο datetime.
- **total:** Είναι το πεδίο που αποθηκεύεται το τελικό ποσό για κάθε παραγγελία με τύπο δεδομένων αλφαριθμητικό μεγέθους διακοσίων πενήντα (250) χαρακτήρων.
- **status:** Είναι το πεδίο του πίνακα που αποθηκεύεται η κατάσταση της παραγγελίας με τύπο δεδομένων ακέραιο αριθμό εμφάνισης ενός (1) ψηφίου. Όταν είναι η τιμή του είναι 0 τότε η παραγγελία δεν έχει αντιστοιχηθεί ακόμη με διαθέσιμη πλακέτα esp8266. Όταν είναι η τιμή του είναι 1 τότε η παραγγελία έχει αντιστοιχηθεί με μία πλακέτα esp8266. Όταν η τιμή του είναι 2 η παραγγελία έχει ολοκληρωθεί.

### 3.4.3 Πίνακας history

Στον πίνακα history είναι αποθηκευμένες οι εγγραφές με τις ολοκληρωμένες παραγγελίες των χρηστών. Στην Εικόνα 8 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι πέντε (5) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.


#	Name	Type
1	history_id	int(5)
2	history_order_id	int(5)

Εικόνα 8: Πίνακας history

- **history\_id:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης πέντε (5) ψηφίων και χαρακτηρίζει μοναδικά την κάθε εγγραφή στο ιστορικό ολοκληρωμένων παραγγελιών.
- **history\_order\_id:** Είναι το ξένο κλειδί του πεδίου order\_id του πίνακα orders και συσχετίζει μία ολοκληρωμένη παραγγελία με μία εγγραφή ιστορικού. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης πέντε (5) ψηφίων.

### 3.4.4 Πίνακας categories

Στον πίνακα categories είναι αποθηκευμένες οι εγγραφές με τις κατηγορίες των προϊόντων του συστήματος τα οποία μπορεί να επιλέξει ο χρήστης σε κάθε παραγγελία του. Σε μία κατηγορία μπορούν να ανήκουν πολλά προϊόντα. Στην Εικόνα 9 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι τέσσερα (4) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/> 1	cat_id 	int(5)
<input type="checkbox"/> 2	cat_image	varchar(255)
<input type="checkbox"/> 3	cat_checkvalue	int(1)
<input type="checkbox"/> 4	cat_name	varchar(30)

↑  Check all With selected:

Εικόνα 9: Πίνακας categories

- **cat\_id:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης πέντε (5) ψηφίων και χαρακτηρίζει μοναδικά την κάθε κατηγορία.
- **cat\_image:** Είναι το πεδίο που αποθηκεύεται η διαδρομή του αρχείου εικόνας που αντιπροσωπεύει την κατηγορία προϊόντος με τύπο δεδομένων αλφαριθμητικό μεγέθους διακοσίων πενήντα πέντε (255) χαρακτήρων.
- **cat\_checkvalue:** Είναι το πεδίο που αποθηκεύεται η πληροφορία για την κάθε κατηγορία αν είναι φαγώσιμο ή όχι με τύπο δεδομένων ακέραιο αριθμό εμφάνισης ένα (1) ψηφίο. Αν είναι φαγώσιμο η τιμή είναι 1, αντιθέτως αν δεν είναι φαγώσιμο η τιμή είναι 0.
- **cat\_name:** Είναι το πεδίο που αποθηκεύεται το όνομα για κάθε κατηγορία με τύπο δεδομένων αλφαριθμητικό μεγέθους τριάντα (30) χαρακτήρων.

### 3.4.5 Πίνακας esp

Ο πίνακας esp έχει αποθηκευμένες τις εγγραφές για τις πλακέτες esp8266 και τα χαρακτηριστικά τους, οι οποίες αντιστοιχίζονται με τις παραγγελίες του χρήστη και τον συνοδεύουν μέχρι την ολοκλήρωσή τους. Μία πλακέτα esp8266 μπορεί να αντιστοιχηθεί σε πολλές παραγγελίες, όταν είναι διαθέσιμη, αλλά όχι ταυτόχρονα. Στην Εικόνα 10 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι πέντε (5) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/>	1 id_esp 🔑	int(3)
<input type="checkbox"/>	2 status	int(1)
<input type="checkbox"/>	3 available	int(1)
<input type="checkbox"/>	4 image	varchar(255)
<input type="checkbox"/>	5 order_id_esp 🔑	int(5)

Εικόνα 10: Πίνακας esp8266

- **id\_esp:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης τριών (3) ψηφίων και χαρακτηρίζει μοναδικά την κάθε πλακέτα esp.
- **status:** Είναι το πεδίο του πίνακα που αποθηκεύεται η κατάσταση της πλακέτας esp8266 με τύπο δεδομένων ακέραιο αριθμό εμφάνισης ενός (1) ψηφίου. Όταν είναι η τιμή του είναι 0 τότε η πλακέτα esp8266 δεν έχει αντιστοιχηθεί ακόμη με την παραγγελία. Όταν είναι η τιμή του είναι 1 τότε η πλακέτα esp8266 έχει αντιστοιχηθεί με την παραγγελία. Όταν η τιμή του είναι 2 τότε η πλακέτα esp8266 ενεργοποιείται και ενημερώνει τον πελάτη για την ολοκλήρωση της παραγγελίας.
- **available:** Είναι το πεδίο του πίνακα αποθηκεύεται η διαθεσιμότητα της κάθε πλακέτας esp με τύπο δεδομένων ακέραιο αριθμό εμφάνισης ενός (1) ψηφίου. Όταν είναι 1 η πλακέτα esp είναι διαθέσιμη, ενώ όταν είναι 0 μη διαθέσιμη.
- **image:** Είναι το πεδίο που αποθηκεύεται η διαδρομή του αρχείου εικόνας που αντιπροσωπεύει την κάθε πλακέτα esp με τύπο δεδομένων αλφαριθμητικό μεγέθους διακοσίων πενήντα πέντε (255) χαρακτήρων.
- **order\_id\_esp:** Είναι το ξένο κλειδί του πεδίου id\_esp του πίνακα esp και συσχετίζει μοναδικά μία παραγγελία με μια πλακέτα esp. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης πέντε (5) ψηφίων.

### 3.4.6 Πίνακας product

Στον πίνακα product είναι αποθηκευμένες οι εγγραφές με τα προϊόντα του συστήματος τα οποία μπορεί να παραγγείλει ο χρήστης σε κάθε παραγγελία του. Στην Εικόνα 11 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι πέντε (5) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/> 1	product_id	int(5)
<input type="checkbox"/> 2	product_cat_id	int(5)
<input type="checkbox"/> 3	product_name	varchar(20)
<input type="checkbox"/> 4	value	float
<input type="checkbox"/> 5	image	varchar(255)

Εικόνα 11: Πίνακας product

- **product\_id:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης πέντε (5) ψηφίων και χαρακτηρίζει μοναδικά το κάθε προϊόν.
- **product\_cat\_id:** Είναι το ξένο κλειδί του πεδίου cat\_id του πίνακα categories και συσχετίζει μοναδικά ένα προϊόν με μία κατηγορία προϊόντων. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης πέντε (5) ψηφίων.
- **product\_name:** Είναι το πεδίο που αποθηκεύεται το όνομα για κάθε προϊόν με τύπο δεδομένων αλφαριθμητικό μεγέθους είκοσι (20) χαρακτήρων.
- **value:** Είναι το πεδίο που αποθηκεύεται η τιμή του προϊόντος με τύπο δεδομένων πραγματικό αριθμό κινητής υποδιαστολής.
- **image:** Είναι το πεδίο που αποθηκεύεται η διαδρομή του αρχείου εικόνας που αντιπροσωπεύει το προϊόν με τύπο δεδομένων αλφαριθμητικό μεγέθους διακοσίων πενήντα πέντε (255) χαρακτήρων.

### 3.4.7 Πίνακας purchase

Στον πίνακα purchase είναι αποθηκευμένες οι εγγραφές με τις παραγγελίες του χρήστη σε σχέση με τα προϊόντα που έχει παραγγείλει. Είναι η οντότητα που χρησιμοποιήθηκε για να αντιμετωπιστεί η σχέση πολλά προς πολλά μεταξύ παραγγελιών και προϊόντων. Στην Εικόνα 12 που ακολουθεί, φαίνονται αναλυτικά τα πεδία του πίνακα, που στο σύνολό τους είναι πέντε (5) και για τα οποία γίνεται ανάλυση αμέσως παρακάτω.

#	Name	Type
<input type="checkbox"/> 1	id_pur	int(5)
<input type="checkbox"/> 2	id_order	int(5)
<input type="checkbox"/> 3	id_prod	int(5)
<input type="checkbox"/> 4	details	varchar(250)
<input type="checkbox"/> 5	posothta	int(3)

Εικόνα 12: Πίνακας purchase

- **id\_pur:** Είναι το πρωτεύον κλειδί του πίνακα με τύπο δεδομένων ακέραιο αριθμό εμφάνισης πέντε (5) ψηφίων και χαρακτηρίζει μοναδικά την κάθε κίνηση παραγγελίας στο καλάθι.

- **id\_order:** Είναι το ξένο κλειδί του πεδίου order\_id του πίνακα orders και συσχετίζει μοναδικά μία κίνηση με μία παραγγελία. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης πέντε (5) ψηφίων.
- **id\_prod:** Είναι το ξένο κλειδί του πεδίου product\_id του πίνακα products και συσχετίζει μοναδικά μία κίνηση με ένα προϊόν. Ο τύπος δεδομένων του πεδίου είναι ακέραιος αριθμός εμφάνισης πέντε (5) ψηφίων.
- **details:** Είναι το πεδίο που αποθηκεύονται ορισμένες έξτρα επιλογές σε σχέση με το προϊόν κατά την φάση της παραγγελίας από το χρήστη με τύπο δεδομένων αλφαριθμητικό μεγέθους διακοσίων πενήντα (250) χαρακτήρων.
- **posothta:** Είναι το πεδίο του πίνακα που αποθηκεύεται η ποσότητα του προϊόντος κατά την φάση της παραγγελίας του χρήστη με τύπο δεδομένων ακέραιο αριθμό εμφάνισης τριών (3) ψηφίων.

### 3.5 Ανάλυση υποδομής, αρχείων και κώδικα του ΠΣ

Στο κεφάλαιο αυτό θα γίνει ανάλυση της υποδομής, των αρχείων και του κώδικα του ΠΣ που δημιουργήθηκαν, ώστε αυτό να λειτουργεί ομαλά και σύμφωνα με τις απαιτήσεις που προαναφέρθηκαν στην προηγούμενη ενότητα. Για την λειτουργία του ΠΣ πρέπει πρώτα να εγκατασταθεί η απαραίτητη υποδομή.

#### 3.5.1 Ανάλυση υποδομής του ΠΣ

Η υποδομή του ΠΣ που υλοποιήθηκε αποτελείται από τον εξυπηρετητή του Πανεπιστημίου Δυτικής Μακεδονίας <https://zafora.ece.uowm.gr> με λειτουργικό σύστημα FreeBSD 12.2-RELEASE-p7 στο οποίο είναι εγκατεστημένα:

- Εξυπηρετητής ιστού Apache έκδοση 2.4.48.
- Εξυπηρετητής βάσεων δεδομένων MySQL έκδοση 5.7.35.
- Υποστήριξη με PHP έκδοση 7.4.23 (cli).
- Εξυπηρετητής ασφαλούς σύνδεσης σε τερματικό OpenSSH έκδοση 7.9p1.
- Πρωτόκολλο ασφαλών συνδέσεων OpenSSL 1.1.1h-freebsd.

Ωστόσο έχει εγκατασταθεί σε τοπικό ηλεκτρονικό υπολογιστή και η διαδικασία εγκατάστασης θα αναλυθεί στο παράρτημα με ανάλογους εξυπηρετητές. Αφού εξασφαλίσουμε την εγκατάσταση της υποδομής στην συνέχεια θα γίνει ανάλυση των φακέλων και αρχείων του ΠΣ.

### 3.5.2 Ανάλυση φακέλων και αρχείων του ΠΣ

Οι φάκελοι και τα αρχεία που χρησιμοποιούνται για την υλοποίηση του ΠΣ έχουν δοθεί με CamelCase ονοματολογία [59]. Τα αρχεία αυτά περιέχουν κυρίως πηγαίο κώδικα των γλωσσών προγραμματισμού που αναλύθηκαν στο προηγούμενο κεφάλαιο και είναι κυρίως αρχεία τύπου PHP, HTML, JS και CSS. Στα αρχεία αυτά, τα ονόματα των μεταβλητών που αποδοθήκαν ακολουθούν Under\_score ονοματολογία [59]. Οι ονοματολογίες για τα αρχεία και τα ονόματα μεταβλητών επιλέχθηκαν να είναι από τα δύο πιο γνωστά στυλ ονοματολογίας κώδικα. Το εκάστοτε αρχείο μπορεί να συμμετέχει σε μία ή και περισσότερες λειτουργίες του ΠΣ, όπως και μία λειτουργία αντίστοιχα μπορεί να εκτελεί ένα ή περισσότερα αρχεία κώδικα. Επίσης, κάποια αρχεία είναι κοινά και για τον απλό χρήστη και για τον διαχειριστή, ενώ κάποια άλλα είναι για αποκλειστική πρόσβαση του διαχειριστή. Ο έλεγχος πρόσβασης γίνεται με την χρήση των μεταβλητών \$\_SESSION της συνάρτησης session\_start() της PHP και με αυτόν τον τρόπο διασφαλίζεται η ορθή πρόσβαση στα αρχεία.

Τα αρχεία του ΠΣ είναι οργανωμένα σε φακέλους για την αποτελεσματικότερη διαχείρισή τους και την καλύτερη οργάνωσή τους ανάλογα την γλώσσα συγγραφής ή τη λειτουργία που εξυπηρετούν. Οι κατάλογοι που φαίνονται στην παρακάτω Εικόνα 13 είναι υποφάκελοι του root (eBuzzer/) και είναι οι εξής:

```
[ictest00592@zafora ~/public_html/eBuzzer]$ ls -a --color
.          admin          includes        login.php      user
..         api.php        index.php       register.php   view.php
[ictest00592@zafora ~/public_html/eBuzzer]$
```

Εικόνα 13: Κατάλογος root eBuzzer

- **admin:** Στον φάκελο αυτό είναι αποθηκευμένα τα αρχεία που έχουν αποκλειστική πρόσβαση από τους διαχειριστές του ΠΣ.
- **includes:** Σε αυτό τον φάκελο βρίσκονται όλα τα αρχεία που είναι κοινόχρηστα και ορισμένα από αυτά έχουν διαβαθμισμένη πρόσβαση ανάλογα τον χρήστη που ελέγχεται από κώδικα PHP. Ο φάκελος αυτός αποτελεί την βιβλιοθήκη του συστήματος.
- **user:** Στον φάκελο αυτό υπάρχουν τα αρχεία που χρησιμοποιούνται από τους απλούς χρήστες του ΠΣ.

Στην παρακάτω Εικόνα 14 φαίνονται τα αρχεία των φακέλων admin, includes και user, όπως παρατηρούμε στο σύνολό τους είναι αρχεία τύπου PHP.



```
[ictest00592@zafora ~/public_html/eBuzzer]$ ls -a --color admin
.
..
action.php
addProduct.php
addProductDetails.php
adminSession.php
cartPopover.php
category.php
editSales.php
esp.php
fcart.php
getCategory.php
getProduct.php
header.php
history.php
historyInfo.php
index.php
insertCheckbox.php
listCategory.php
order.php
product.php
requireAdmin.php
requireOrder.php
sales.php
salesInfo.php
searchOrders.php
searchUsers.php
selectCategory.php
selectProduct.php
sendSalesData.php
sendSalesDataFromView.php

[ictest00592@zafora ~/public_html/eBuzzer]$ ls -a --color includes/
.
..
bootstrap
category
checkLogin.php
config.php
connectPDO.php
css
edits
esp
fetch
footer.php
footer2.php
functions.php
header.php
https.php
images
js
logout.php
modalLeftRight.php
product
regValues.php
require.php
require2.php

[ictest00592@zafora ~/public_html/eBuzzer]$ ls -a --color user
.
..
action.php
addProduct.php
addProductDetails.php
cartPopover.php
fcart.php
getCategory.php
getProduct.php
header.php
history.php
historyInfo.php
index.php
listCategory.php
order.php
requireOrder.php
requireUser.php
selectCategory.php
selectProduct.php
userSession.php

[ictest00592@zafora ~/public_html/eBuzzer]$ █
```

Εικόνα 14: Αρχεία φακέλων admin, includes, user

Επίσης παρατηρούμε ότι στον φάκελο includes υπάρχουν ορισμένοι υποφάκελοι που είναι οι εξής:

- **bootstrap:** Στον φάκελο αυτό υπάρχουν αποθηκευμένα τα αρχεία του bootstrap framework.
- **category:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία που έχουν να κάνουν με την προσθήκη κατηγορίας και είναι αρχεία τύπου PHP.
- **css:** Στον φάκελο αυτό έχει τοποθετηθεί το CSS αρχείο που έχει δημιουργηθεί για το ΠΣ και έχει όλους τους προσαρμοσμένους κανόνες εμφάνισης.
- **edits:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία PHP που χρησιμοποιούνται για την διαχείριση του χρήστη.
- **esp:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία PHP που χρησιμοποιούνται για την διαχείριση των πλακετών esp.
- **fetch:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία PHP που χρησιμοποιούνται για την μεταφορά των πληροφοριών από το ΠΣ προς τους χρήστες και αντίστροφα χρησιμοποιώντας τις τεχνικές ajax.
- **images:** Στον φάκελο αυτό έχουν τοποθετηθεί οι εικόνες του ΠΣ για την κυρίως εμφάνισή του.

- **js:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία JS που χρησιμοποιούνται από το ΠΣ για τις διάφορες λειτουργίες του.
- **product:** Στον φάκελο αυτό έχουν τοποθετηθεί τα αρχεία PHP που χρησιμοποιούνται για την διαχείριση των προϊόντων.

Στην συνέχεια θα ακολουθήσει η ανάλυση του κώδικα σε επιλεγμένα αρχεία του ΠΣ που διαδραματίζουν σημαντικό ρόλο σε ορισμένες λειτουργίες του.

### 3.5.3 Ανάλυση κώδικα του ΠΣ

Ξεκινώντας την ανάλυση του κώδικα πρώτα θα αναλύσουμε το αρχείο https.php το οποίο εξασφαλίζει την ασφαλή σύνδεση με τον εξυπηρετητή ιστού μέσω HTTPS πρωτοκόλλου ελέγχοντας την μεταβλητή της PHP `$_SERVER['HTTPS']`. Στην παρακάτω Εικόνα 15 φαίνεται ο κώδικας του.

```
<?php
if(!isset($_SERVER['HTTPS']) || $_SERVER['HTTPS'] == ""){
$redirect = "https://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
header("HTTP/1.1 301 Moved Permanently");
header("Location: $redirect");
die("Please visit <a href=$redirect>$redirect </a>");
}
?>
```

Εικόνα 15: https.php

Στην συνέχεια γίνεται ανάλυση το αρχείου config.php το οποίο περιλαμβάνει τις ρυθμίσεις σύνδεσης με τη βάση δεδομένων του ΠΣ. Στην Εικόνα 16 φαίνονται οι ρυθμίσεις αυτές και η απόδοση τιμών σε μεταβλητές που είναι απαραίτητες για την σύνδεση μέσω της κλάσης PDO, όπως φαίνεται στην Εικόνα 17 που δείχνει των κώδικα του αρχείου connectPDO.php. Επίσης αναφέρουμε ότι το αρχείο require.php στην ουσία χρησιμοποιεί τα δύο παραπάνω αρχεία config.php και connectPDO.php.

```
<?php
define("MYSQLHOST", "/zstorage/home/ece00592/mysql/run/mysql.sock");
define("MYSQLUSER", "root");
define("MYSQLPASS", "██████████");
define("MYSQLPREF", "prt");
define("MYSQLBASE", "ebuzzer");
define("MYSQLHTYP", "unix_socket");
define("HTMLHOME", "zafora.ict.e.uowm.gr/~ece00592/");
define("DISPERROR", "1");
define("HELPO", TRUE);
define("MYSQLPORT", "3306");
?>
```

Εικόνα 16: config.php

```

<?php
function db_open() {
    $host=MYSQLHOST;
    $dbname=MYSQLBASE;
    $user=MYSQLUSER;
    $passwd=MYSQLPASS;
    try{
        $conn = new PDO("mysql:".$MYSQLHTYP."=$host;dbname=$dbname", $user, $passwd,
            array(PDO::ATTR_ERRMODE => PDO::ERRMODE_WARNING, PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES utf8'));
    }
    catch(PDOException $pe){
        die('Connection error: ' . $pe->getMessage());
    }
    return $conn;
}
?>

```

Εικόνα 17: connectPDO.php

Η πρώτη ενέργεια που πρέπει να κάνει ο χρήστης για να μπορεί να έχει πρόσβαση στο σύστημα είναι η εγγραφή του σε αυτό μέσω του αρχείου register.php, το οποίο εκτελεί τις λειτουργίες εγγραφής του χρήστη στο σύστημα. Στην Εικόνα 18 φαίνονται τα στοιχεία της φόρμας που απαιτούνται από το χρήστη για την εγγραφή του στο σύστημα μέσω του αρχείου regValues.php.

```

<div class="main-container">
    <div class="header">
        <div class="logo">
            <h6>Εγγραφή Χρήστη:</h6>
        </div>
    </div>
    <div class="container2">
        <div class="row">
            <form id="reg" name="reg" action="includes/regValues.php" method="post" onsubmit="return validate()" >
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε το όνομα χρήστη" name="username" id="usname" /></div>
                    <span id="available"> </span>
                <div class="col-100">
                    <input required type="password" name="pass1" placeholder="Εισάγετε τον κωδικό πρόσβασης" id="pass" /></div>
                <div class="col-100">
                    <input required type="password" name="pass2" placeholder="Επιβεβαιώστε τον κωδικό πρόσβασης" id="pass" /></div>
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε το όνομά σας" name="uname" id="usname" /></div>
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε το επίθετό σας" name="fullname" id="usname" /></div>
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε την ηλεκτρονική σας διεύθυνση" name="mail" id="usname" /></div>
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε την τοποθεσία σας" name="address" id="usname" /></div>
                <div class="col-100">
                    <input required type="number_format" placeholder="Εισάγετε το τηλέφωνό σας" name="phone" id="usname" /></div>
                <div class="col-100-btn">
                    <input type="submit" value="ΕΓΓΡΑΦΗ" id="form-submit" />
                </div>
            </form>
        </div>
    </div>
</div>

```

Εικόνα 18: register.php

Στο αρχείο regValues.php που φαίνεται στην Εικόνα 19 γίνεται ο έλεγχος τιμών της φόρμας χρησιμοποιώντας τις συναρτήσεις της PHP filter\_var και md5. Μετά την εγγραφή στο σύστημα ο χρήστης μπορεί να συνδεθεί σε αυτό χρησιμοποιώντας το login.php το οποίο αναλαμβάνει να κάνει τον έλεγχο της σύνδεσης των χρηστών.

```

<?php
//access administrator, register values for new user
session_start();
require_once ('require.php');
require_once ('functions.php');
$conn=db_open();
if(!empty($_POST["username"] && $_POST["pass1"] && $_POST["uname"] && $_POST["fullname"] && $_POST["mail"] && $_POST["address"] && $_POST["phone"]))
{
    if($_SESSION['goRegister'] == 1)
    {
        $username=filter_var($_POST["username"],FILTER_SANITIZE_STRING);
        $password=md5(filter_var($_POST["pass1"]));
        $name=filter_var($_POST["uname"],FILTER_SANITIZE_STRING);
        $full_name=filter_var($_POST["fullname"],FILTER_SANITIZE_STRING);
        $email=filter_var($_POST["mail"],FILTER_SANITIZE_EMAIL);
        $address=filter_var($_POST["address"],FILTER_SANITIZE_STRING);
        $phone_number=filter_var($_POST["phone"],FILTER_SANITIZE_NUMBER_INT);
        $type=1;
        try
        {
            $sql=" INSERT INTO user(username ,password, first_name, last_name ,email ,address ,telephone ,type )
            VALUES (:username, :password, :first_name, :last_name, :email, :address, :telephone, :type) " ;
            $query = $conn->prepare($sql);
            $query->execute(array(':username'=>$username, ':password'=>$password, ':first_name'=>$name,
            ':last_name'=>$full_name, ':email'=>$email, ':address'=>$address, ':telephone'=>$phone_number, ':type'=>$type));
            $_SESSION['yadd'] = 1;
        }
        catch(PDOException $e)
        {
            echo $sql . "<br>" . $e->getMessage();
            unset($_SESSION['yadd']);
            $_SESSION['nadd'] = 0;
            die();
        }
    }
    else
    {
        $_SESSION['nadd'] = 0;
    }
}
$conn=null;
returntostart("../register.php");
-?>

```

Εικόνα 19: regValues.php

Όπως φαίνεται την παρακάτω Εικόνα 20 αφού ο χρήστης καταχωρίσει τα στοιχεία πρόσβασης του (username, password), με μέθοδο HTML POST καλείται το αρχείο checkLogin.php το οποίο αναλαμβάνει τον έλεγχο για την είσοδο στο ΠΣ.

```

<div class="main-container">
    <div class="header">
        <div class="logo">
            <h3>Είσοδος Χρήστη:</h3>
        </div>
    </div>
    <div class="container2">
        <div class="row">
            <form id="check_con" action="includes/checkLogin.php" method="post">
                <div class="col-100">
                    <input required type="text" placeholder="Εισάγετε το όνομα χρήστη" name="usname" id="usname" />
                </div>
                <div class="col-100">
                    <input required type="password" name="pass" placeholder="Εισάγετε Κωδικό Πρόσβασης" id="pass" />
                </div>
                <div class="col-100-btn">
                    <input type="submit" class="login-form" value="ΕΙΣΟΔΟΣ" id="form-submit" />
                </div>
            </form>
            <form id="register" action="register.php" method="post">
                <div class="col-100-btn">
                    <input type="submit" value="ΕΓΓΡΑΦΗ" id="form-submit" /></div>
            </form>
        </div>
    </div>
</div>

```

Εικόνα 20: login.php

Στην συνέχεια το checkLogin.php, όπως φαίνεται και στην παρακάτω Εικόνα 21, με τη χρήση των συναρτήσεων της PHP stripslashes, filter\_var, md5, των κλάσεων PDO, PDOException και των μεθόδων τους γίνεται σύνδεση με τη βάση δεδομένων. Εφόσον

πραγματοποιηθεί επιτυχώς η ταυτοποίηση του χρήστη, τον συνδέει στο σύστημα με τα ανάλογα δικαιώματα και λειτουργίες, αλλιώς με κατάλληλο μήνυμα απορρίπτεται η σύνδεση.

```

<?php
// access = administrator purpose = check the values of login
session_start();
require_once ('require.php');
require_once ('functions.php');
$con=db_open();
if(!empty($_POST['usname']) && (!empty($_POST['pass'])))
{
    $name = stripslashes($_POST["usname"]);
    $name=filter_var($_POST["usname"],FILTER_SANITIZE_STRING);
    $password1 = stripslashes($_POST["pass"]);
    $password1=filter_var($_POST["pass"]);
    $en_pass=md5($password1);
    try
    {
        $sql="SELECT username,password,type,id,first_name FROM user WHERE username=:username AND password=:password ";
        $result = $con->prepare($sql);
        $result->bindParam(':username',$username);
        $result->bindParam(':password',$password);
        $result->bindParam(':type',$type);
        $result->bindParam(':id',$id);
        $result->bindParam(':first_name',$first_name);
        $result->execute(array(':username'=>$name,':password'=>$en_pass));
        $result1 = $result->fetchAll();
    }
    catch(PDOException $e)
    {
        echo $sql . "<br>" . $e->getMessage();
        die();
    }
}

```

Εικόνα 21: checkLogin.php

Στη συνέχεια, αφού γίνει η σύνδεση στο σύστημα, τότε ο χρήστης μπορεί να διενεργήσει παραγγελίες μέσω των selectCategory.php, selectProduct.php, addProduct.php και order.php.

Ακολούθως αξίζει να αναφέρουμε και να αναλύσουμε την τεχνική AJAX που χρησιμοποιείται κατά την δυναμική ανταλλαγή δεδομένων μεταξύ πελάτη (client) και εξυπηρετητή (server) χωρίς να γίνεται επαναφόρτωση ολοκλήρωσης της σελίδας. Για την τεχνική αυτή χρησιμοποιούνται τα αρχεία searchUser.js, searchOrder.js και getData.js. Όπως φαίνεται για παράδειγμα στην παρακάτω Εικόνα 22 για τα δεδομένα του χρήστη καλείται η συνάρτηση load\_data η οποία μέσα περιέχει την τεχνική AJAX, όπου δέχεται μία παράμετρο (query) και κατά την κλήση της σε περίπτωση επιτυχίας επιστρέφει στο αναφερόμενο element με βάση το id την τιμή του με μέθοδο POST. Την ενεργοποίηση της συνάρτησης την υλοποιούμε με τη μέθοδο keyup, η οποία σε κάθε πληκτρολόγηση που κάνει ο χρήστης στο element με id search\_text αναζητά δυναμικά το username ή το id.

```

<script>
$(document).ready(function(){
    load_data();

    function load_data(query)
    {
        $.ajax({
            url:"../includes/fetch/showUsersDynamic.php",
            method:"POST",
            data:{query:query},
            dataType:"text",
            success:function(data)
            {
                $('#result').html(data);
            }
        });
    }

    $('#search_text').keyup(function(){
        var search = $(this).val();
        if(search != '')
        {
            load_data(search);
        }
        else
        {
            load_data();
        }
    });
});
</script>

```

Εικόνα 22: searchUser.js

Αφού ο χρήστης ολοκληρώσει την παραγγελία του και θελήσει να αποσυνδεθεί τότε η αποσύνδεση γίνεται χρησιμοποιώντας το logout.php και διαγράφοντας το SESSION του. Στην Εικόνα 23 φαίνεται ο κώδικας που εκτελείται κατά την αποσύνδεση του χρήστη.

```

<?php
    session_start();
    setcookie(ID_my_site, gone, $past);
    setcookie(Key_my_site, gone, $past);

    if(isset($_SESSION["root"])){
        setcookie("root", false);
        setcookie ("root", "", time()-3600 , '/' );
        unset($_SESSION["root"]);
    }

    if(isset($_SESSION["who"])){
        setcookie("who", false);
        setcookie ("who", "", time()-3600 , '/' );
        unset($_SESSION["who"]);
    }

    if(isset($_SESSION['name'])){
        setcookie("name", false);
        setcookie ("name", "", time()-3600 , '/' );
        unset($_SESSION["name"]);
    }

    if(isset($_SESSION["user"])){
        setcookie("user", false);
        setcookie ("user", "", time()-3600 , '/' );
        unset($_SESSION["user"]);
    }

    session_destroy();
    header("Location: ../index.php");
?>

```

Εικόνα 23: logout.php

Τέλος όσον αφορά την εμφάνιση του ΠΣ πέρα από τα CSS αρχεία του bootstrap έχει δημιουργηθεί ένα επιπλέον αρχείο style.css. Στην εικόνα που ακολουθεί φαίνεται πως ξεκινάει ένας κανόνας εμφάνισης που αφορά οθόνη έως 650 εικονοστοιχεία.

```
@media screen and (max-width:650px) {
    .main-content {
        text-align:center;
    }

    #quantity {
        width:25%;
        margin-top:25%;
    }

    .success {
        position:absolute;
        top:16.5%;
        right:0%;
        background-color: rgb(106,168,126);
        padding:5px 10px;
        border-radius:5px;
    }
}
```

Εικόνα 24: Κανόνας εμφάνισης επιλεγμένος από το style.css

### 3.6 Διασύνδεση πληροφοριακού συστήματος με την πλακέτα esp8266

Αρχικά, για την διασύνδεση της πλακέτας esp8266 με το ΠΣ υλοποιήθηκε ο κώδικας που θα εκτελείται σε αυτήν. Στην παρακάτω Εικόνα 25 φαίνεται η αρχικοποίηση μεταβλητών, όπως το ssid, password, host, request, esp\_id, και οι κατάλληλες βιβλιοθήκες που απαιτούνται για τη διασύνδεση αυτή.

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266HTTPClient.h>

const char *ssid = "Home_Gate";
const char *password = " ";
const char *host = "zafora.ece.uowm.gr";
String request="http://zafora.ece.uowm.gr/~ece00592/eBuzzer/api.php";
int esp_id = 1;
```

Εικόνα 25: Αρχικοποίηση μεταβλητών esp8266

Στην συνέχεια στην Εικόνα 26 γίνεται η σύνδεση της πλακέτας esp8266 με το ασύρματο δίκτυο Home\_Gate, παρέχοντάς της πρόσβαση στο Διαδίκτυο. Αυτό επιτυγχάνεται με την συνάρτηση WiFi.begin() της βιβλιοθήκης ESP8266WiFi.

```

void setup()
{
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  Serial.println("");
  Serial.println("Connecting");
  if (WiFi.status() == WL_CONNECTED)
  {
    Serial.println("");
    Serial.println("Connected! IP address:");
    Serial.println(WiFi.localIP());
  }
}

```

Εικόνα 26: Σύνδεση πλακέτας esp8266 με το ασύρματο δίκτυο

Ακολούθως, στην παρακάτω εικόνα φαίνεται ο τρόπος σύνδεσης της πλακέτας esp8266 με το ΠΣ. Αυτό επιτυγχάνεται με τη χρήση της βιβλιοθήκης HTTPClient, όπου ορίζεται αντικείμενό της με όνομα http, την κλήση της μεθόδου http.begin() και εκτελείται συνέχεια όσο η πλακέτα είναι συνδεδεμένη στο ασύρματο δίκτυο. Στο σημείο αυτό να αναφερθεί ότι, η πλακέτα esp8266 σε περίπτωση απρόσμενης αποσύνδεσης από το ασύρματο δίκτυο κάνει από μόνη της προσπάθειες επανασύνδεσης.

```

void loop() {
  if (WiFi.status() == WL_CONNECTED)
  {
    HTTPClient http;
    Serial.print("connecting to ");
    Serial.println(host);
    WiFiClient wificlient;
    http.begin(wificlient, request);
    const int httpPort = 80;
    if (!wificlient.connect(host, httpPort))
    {
      Serial.println("Connection failed to host");
      return;
    }
  }
}

```

Εικόνα 27: Σύνδεση πλακέτας esp8266 με ΠΣ

Έπειτα, στην Εικόνα 28 που ακολουθεί φαίνεται ο τρόπος ανταλλαγής των δεδομένων της πλακέτας esp8266 με ΠΣ. Αυτός ο τρόπος υλοποιείται με την βοήθεια του αντικειμένου http που ορίστηκε προηγουμένως και των μεθόδων http.POST(), http.GET() και http.getString().



```

while (wificlient.available())
{
  postData = "espID=" + esp_id ;
  int httpCode = http.POST(postData);
  int get_status = http.GET();
  Serial.print("Get status:");
  Serial.println(httpCode);
  if (get_status != 200)
  {
    Serial.println("Get failed! " + String(get_status));
  }
  else
  {
    String payload = http.getString();
    Serial.println("Get data:");
    Serial.println(payload);
  }
}

```

Εικόνα 28: Ανταλλαγή δεδομένων πλακέτας esp8266 με ΠΣ

Τέλος, στην Εικόνα 29 φαίνεται ο κώδικας του αρχείου api.php με τον οποίο γίνεται η ανταλλαγή των δεδομένων της πλακέτας με το ΠΣ. Σε πρώτη φάση ελέγχεται η τιμή της μεταβλητής espID, η οποία στέλνεται από την πλακέτα esp8266 με μέθοδο POST. Στην επόμενη φάση και αφού η τιμή της μεταβλητής espID έχει αρχικοποιηθεί γίνεται η σύνδεση με τη βάση δεδομένων του ΠΣ με χρήση του ερωτήματος που είναι αποθηκευμένο στην μεταβλητή \$sql. Στην τελική φάση και μετά την εκτέλεση του ερωτήματος επιστρέφονται στην πλακέτα esp8266 ο αριθμός και η κατάσταση της παραγγελίας, με τη βοήθεια των μεταβλητών \$order\_id και \$status.

```

<?php
//access everyone
require_once ('includes/require.php');
if(isset($_POST['espID']))
{
  $esp_id=filter_var($_POST['espID'],FILTER_SANITIZE_NUMBER_INT);
  $conn=db_open();
  try
  {
    $sql="select status,order_id_esp from esp where id_esp=:id_esp";
    $query=$conn->prepare ($sql);
    $query->execute(array(':id_esp' =>$esp_id));
  }
  catch (PDOException $e)
  {
    echo $sql . "<br>" . $e->getMessage();
    die();
  }

  foreach ($query->fetchAll() as $row)
  {
    $status=$row['status'];
    $order_id=$row['order_id_esp'];
    echo "order_id:",$order_id;
    echo "status:",$status;
  }
}
$conn=null;
?>

```

Εικόνα 29: Αρχείο api.php

### 3.7 Ασφάλεια πληροφοριακού συστήματος

Το θέμα της ασφάλειας είναι πολύ σημαντικά για ένα ΠΣ, επομένως και για τον προγραμματιστή που θα το υλοποιήσει. Η έννοια της ασφάλειας γενικότερα είναι δύσκολο να αντιμετωπιστεί, παρόλα αυτά χρησιμοποιούνται τεχνικές στην φάση της σχεδίασης και του προγραμματισμού του, ώστε να υλοποιηθεί ένα πιο ασφαλές ΠΣ.

Η πιο βασική τεχνική που επιβάλλεται να ενσωματωθεί σε ένα ΠΣ είναι η χρήση του πρωτοκόλλου ασφαλούς σύνδεσης https (Hypertext Transfer Protocol Secure) αντί του http (Hypertext Transfer Protocol). Το https προσφέρει ασφαλή σύνδεση μέσω κρυπτογράφησης των δεδομένων από τον εξυπηρετητή προς τον πελάτη και αντίστροφα, μέσω της TCP (Transmission Control Protocol) θύρας 443 αντί της 80. Για να εξασφαλιστεί η ασφαλής σύνδεση πρέπει σε όλα τα αρχεία του ΠΣ να γίνεται ανακατεύθυνση (redirect) της σύνδεσης σε https. Αυτό επιτυγχάνεται με την χρήση της PHP και της εντολής include με την οποία ενσωματώνουμε τις κρίσιμες ενέργειες σε κάθε αρχείο, αυτό φαίνεται στην Εικόνα 15 όπου με κατάλληλες εντολές γίνεται η εν λόγω ανακατεύθυνση.

#### 3.7.1 Σύνδεση με τη βάση δεδομένων

Επόμενη τεχνική ασφάλειας που χρησιμοποιήθηκε αφορά την σύνδεση με την βάση δεδομένων και είναι η χρήση PDO (PHP Data Objects), ώστε να επιτευχθεί υψηλότερο επίπεδο ασφάλειας. Η χρήση προκατασκευασμένων εντολών PDO αποτελεί έναν τρόπο αντιμετώπισης της απειλής με SQL injection. Στην Εικόνα 30 φαίνεται ο τρόπος χρήσης των εντολών PDO ώστε να επιτευχθεί η σύνδεση αυτή. Με τη χρήση της συνάρτησης db\_open() γίνονται όλες οι συνδέσεις με τη βάση δεδομένων.

```
<?php
function db_open()
{
    $host = MYSQLHOST;
    $dbname = MYSQLBASE;
    $user = MYSQLUSER;
    $passwd = MYSQLPASS;
    try {
        $conn = new PDO("mysql:" . MYSQLHTYP . "=$host;dbname=$dbname", $user, $passwd, array(
            PDO::ATTR_ERRMODE => PDO::ERRMODE_WARNING,
            PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES utf8'
        ));
    }
    catch (PDOException $pe) {
        die('Connection error:' . $pe->getMessage());
    }
    return $conn;
}
?>
```

Εικόνα 30: Εντολές PDO για σύνδεση με βάση δεδομένων

Όπως φαίνεται και στην παρακάτω Εικόνα 31 στην αρχή γίνεται η σύνδεση με την βάση δεδομένων μέσω της μεταβλητής `$con=db_open()`. Στην συνέχεια δημιουργείται το sql ερώτημα με τη χρήση της μεταβλητής `$sql1`, το οποίο προετοιμάζεται με την εντολή `$query1=$con->prepare($sql1)` και εκτελείται με την εντολή `$query1->execute(array())`.

```

<?php
session_start();
require_once('../require.php');
require_once('../functions.php');
$con=db_open();
if(isset($_POST['id']))
{
    if(!empty($_POST["cat_checkbox"]))
    {
        $checkbox_value=1;
    }
    else
    {
        $checkbox_value=0;
    }
    try
    {
        $id2=filter_var($_POST['id'],FILTER_SANITIZE_NUMBER_INT);
        $sql1="select cat_image from categories WHERE cat_id=:cat_id ";
        $query1=$con->prepare($sql1);
        $query1->execute(array(':cat_id' => $id2));
        $count=$query1->fetchAll();
    }
    catch(PDOException $e)
    {
        echo $sql1 . "<br>" . $e->getMessage();
        die();
    }
}

```

Εικόνα 31: Παράδειγμα χρήσης PDO εντολών

### 3.7.2 Επικύρωση δεδομένων εισόδου με τη συνάρτηση `filter_var()`

Επιπλέον χρησιμοποιήθηκε η τεχνική ασφάλειας με την χρήση της συνάρτησης της PHP `filter_var()`, ώστε να ελέγχεται και να φιλτράρεται η είσοδος των δεδομένων στο ΠΣ από ανεπιθύμητες ενέργειες. Η συνάρτηση `filter_var()` μας βοηθάει να εξασφαλίσουμε τα δεδομένα εισαγωγής, είτε με την μέθοδο POST, είτε με την μέθοδο GET, να είναι σύμφωνα με το φίλτρο που ορίζουμε σαν παράμετρο. Όπως φαίνεται στην παρακάτω Εικόνα 32, για την μεταβλητή `$username` χρησιμοποιείται η συνάρτηση `filter_var()` με την παράμετρο `FILTER_SANITIZE_STRING` για να εξασφαλίσει ότι το περιεχόμενο της μεταβλητής θα είναι αλφαριθμητικό. Ομοίως η μεταβλητή `$email` χρησιμοποιεί την συνάρτηση `filter_var()` με παράμετρο `FILTER_SANITIZE_EMAIL` για να εξασφαλίσει ότι το περιεχόμενο της μεταβλητής θα είναι αλφαριθμητικό τύπου email (αφαιρεί όλους τους χαρακτήρες που δεν είναι συμβατοί με διεύθυνση ηλεκτρονικού ταχυδρομείου). Τέλος, η μεταβλητή `$phone_number` χρησιμοποιεί την συνάρτηση `filter_var()` με παράμετρο `FILTER_SANITIZE_NUMBER_INT` για να εξασφαλίσει ότι το περιεχόμενο της μεταβλητής θα είναι ακέραιος αριθμός.

```

<?php
$username = filter_var($_POST["username"], FILTER_SANITIZE_STRING);
$password = md5(filter_var($_POST["pass1"]));
$name = filter_var($_POST["uname"], FILTER_SANITIZE_STRING);
$full_name = filter_var($_POST["fullname"], FILTER_SANITIZE_STRING);
$email = filter_var($_POST["mail"], FILTER_SANITIZE_EMAIL);
$address = filter_var($_POST["address"], FILTER_SANITIZE_STRING);
$phone_number = filter_var($_POST["phone"], FILTER_SANITIZE_NUMBER_INT);
?>

```

Εικόνα 32: Παράδειγμα χρήσης filter\_var()

### 3.7.3 Έλεγχος πρόσβασης με τη χρήση SESSION

Τέλος, για τον έλεγχο πρόσβασης στο ΠΣ χρησιμοποιούνται οι μεταβλητές \$\_SESSION της PHP σε συνδυασμό με κατάλληλους ελέγχους για την εκτέλεση διαβαθμισμένων λειτουργιών ανάλογα με τον χρήστη. Όπως φαίνεται στην παρακάτω Εικόνα 33 πρώτα καλείται η συνάρτηση της PHP session\_start() και στην συνέχεια ελέγχονται οι μεταβλητές \$\_SESSION['root'] και \$\_SESSION['user']. Ο σκοπός των ελέγχων αυτών είναι να εξασφαλιστεί η διαβαθμισμένη πρόσβαση στις λειτουργίες του ΠΣ ανάλογα με τον χρήστη. Σε κάθε περίπτωση που η πρόσβαση γίνεται από μη εξουσιοδοτημένο χρήστη αυτός θα ανακατευθύνεται στην αρχική σελίδα του ΠΣ με την βοήθεια της συνάρτησης returntostart().

```

<?php
session_start();
if (!$SESSION['root'] == 1) {
    returntostart("../index.php");
}

if (!$SESSION['user'] == 1) {
    returntostart("../index.php");
}
?>

```

Εικόνα 33: Έλεγχος πρόσβασης με SESSION

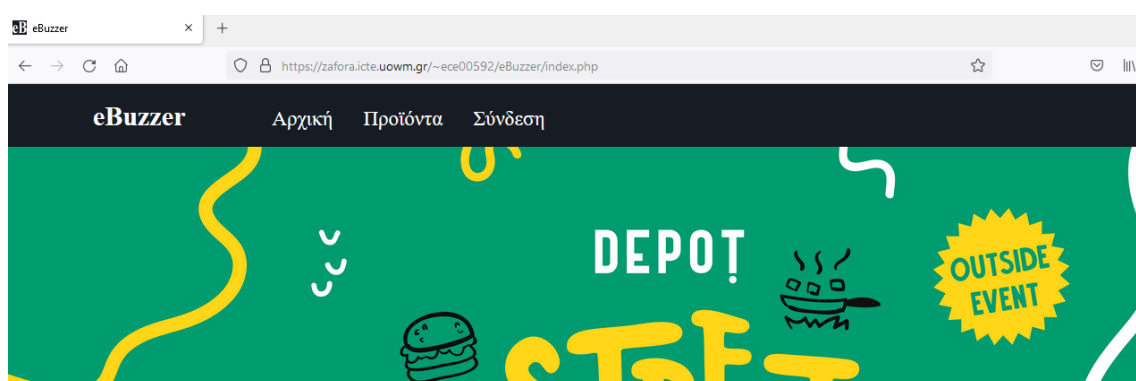
## 3.8 Σύνοψη κεφαλαίου

Σε αυτό το κεφάλαιο έγινε ανάλυση του ΠΣ eBuzzer από την σύλληψη της ιδέας, τις απαιτήσεις, τον σχεδιασμό της βάσης, την υποδομή, τα αρχεία έως και τον κώδικα του. Σε πολλά σημεία χρησιμοποιήθηκαν εικόνες για να γίνουν πιο κατανοητά όλα τα σημεία του ΠΣ που αναλύθηκαν. Παρουσιάστηκαν: το σχεδιάγραμμα του κυκλώματος, οι πίνακες της βάσης δεδομένων, η υποδομή που απαιτείται για να το υποστηρίξει, τα αρχεία του ΠΣ και ο κώδικα του. Τέλος, αναλύθηκαν οι τεχνικές ασφάλειας που χρησιμοποιήθηκαν και ο τρόπος διασύνδεσης του ΠΣ με την πλακέτα esp8266. Στο κεφάλαιο που ακολουθεί θα γίνει η παρουσίαση της διεπαφής του χρήστη και οι λειτουργίες της.

## 4ο Κεφάλαιο - Διεπαφή χρήστη και λειτουργίες

### 4.1 Εγγραφή, σύνδεση και αποσύνδεση στο ΠΣ

Όπως αναφέραμε και παραπάνω το σύστημα υποστηρίζει διαβαθμισμένη πρόσβαση χρηστών και αποτρέπει σε μη εξουσιοδοτημένους χρήστες την πρόσβαση τους σε αυτό. Ως επισκέπτης και χωρίς εξουσιοδοτημένη πρόσβαση μπορεί να χρησιμοποιήσει το ΠΣ για να ενημερωθεί για τα προϊόντα που υπάρχουν. Στην Εικόνα 34 φαίνεται η διεπαφή του χρήστη χωρίς εξουσιοδοτημένη πρόσβαση. Εδώ να αναφέρουμε ότι η εικόνα του παρασκηνίου του ΠΣ είναι από το depotcardiff.com [60].



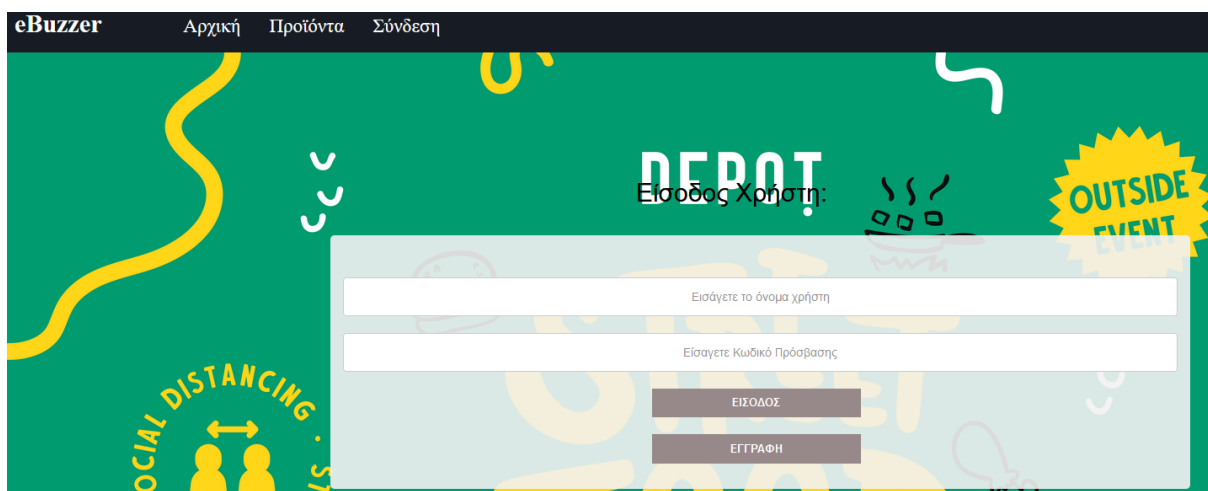
Εικόνα 34: eBuzzer χωρίς εξουσιοδοτημένη πρόσβαση

Αν ο επισκέπτης θελήσει να κάνει εγγραφή τότε όπως φαίνεται και στην Εικόνα 35 επιλέγει το μενού εγγραφή και καταχωρεί τα στοιχεία του μέσω της φόρμας εγγραφής. Η φόρμα όταν ανοίγει έχει εφέ ζουμ το οποίο διαρκεί για τέσσερα δευτερόλεπτα μέχρι να φτάσει το προβλεπόμενο μέγεθος που ορίζεται από στο style.css. Γίνονται οι απαραίτητοι έλεγχοι στην βάση δεδομένων (αν υπάρχει ή δεν υπάρχει ο χρήστης), στο username να είναι έως είκοσι χαρακτήρες, στον κωδικό πρόσβασης να είναι τουλάχιστον οκτώ χαρακτήρες, στο όνομα να είναι έως είκοσι χαρακτήρες, στο επίθετο να είναι έως τριάντα χαρακτήρες, στο email να έχει τη σωστή μορφή και έως τριάντα χαρακτήρες, στον αριθμό τηλεφώνου να είναι δέκα ψηφία και τέλος η διεύθυνση να είναι μέχρι τριάντα χαρακτήρες. Σε περίπτωση που δεν υπάρξει κάποιο πρόβλημα κατά των έλεγχο των στοιχείων τότε ολοκληρώνεται η εγγραφή του χρήστη, αλλιώς εμφανίζεται κατάλληλο μήνυμα, όπως φαίνεται στην Εικόνα 36.

Εικόνα 35: Φόρμα εγγραφής

Εικόνα 36: Μήνυμα λάθους στη φόρμα εγγραφής

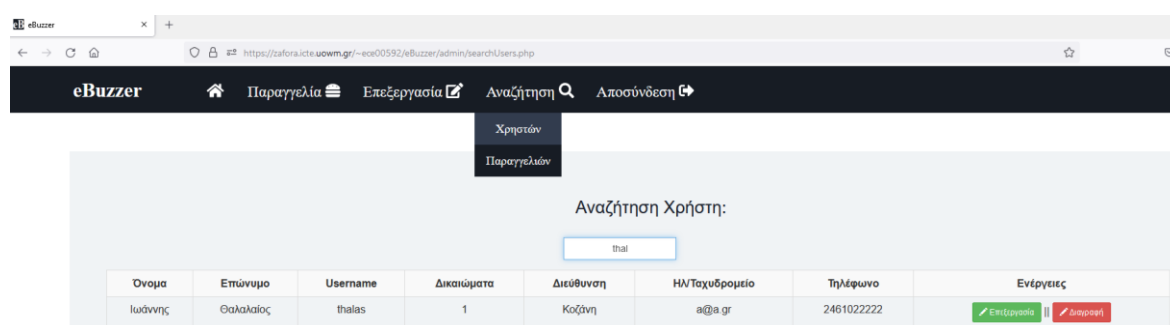
Ο χρήστης αφού ολοκληρώσει την εγγραφή του μπορεί να συνδεθεί στο σύστημα με τα στοιχεία που καταχώρησε μέσω της εγγραφής του. Όπως φαίνεται και στην πληκτρολογώντας το όνομα χρήστη και τον κωδικό πρόσβασης μπορεί να κάνει είσοδο. Οποιαδήποτε στιγμή θελήσει ο χρήστης μπορεί να αλλάξει τα στοιχεία που συμπλήρωσε κατά την εγγραφή του. Με την ολοκλήρωση της εγγραφής του ο χρήστης έχει δικαιώματα απλού χρήστη στο ΠΣ. Εκτός από τον απλό χρήστη όπως προαναφέραμε υπάρχει και ο χρήστης διαχειριστής παρουσίαση των λειτουργιών του θα γίνει στην επόμενη ενότητα.



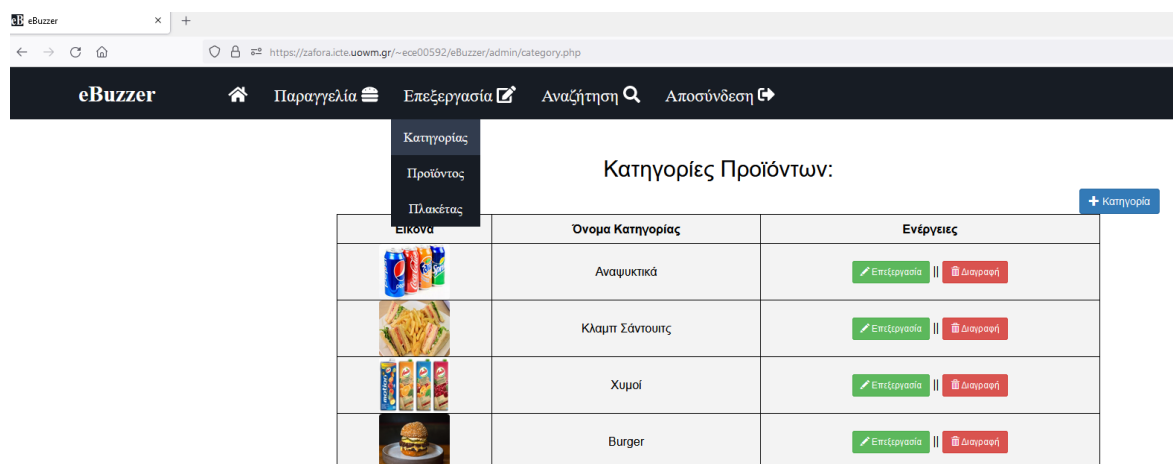
Εικόνα 37: Σύνδεση χρήστη

## 4.2 Λειτουργίες χρήστη διαχειριστή

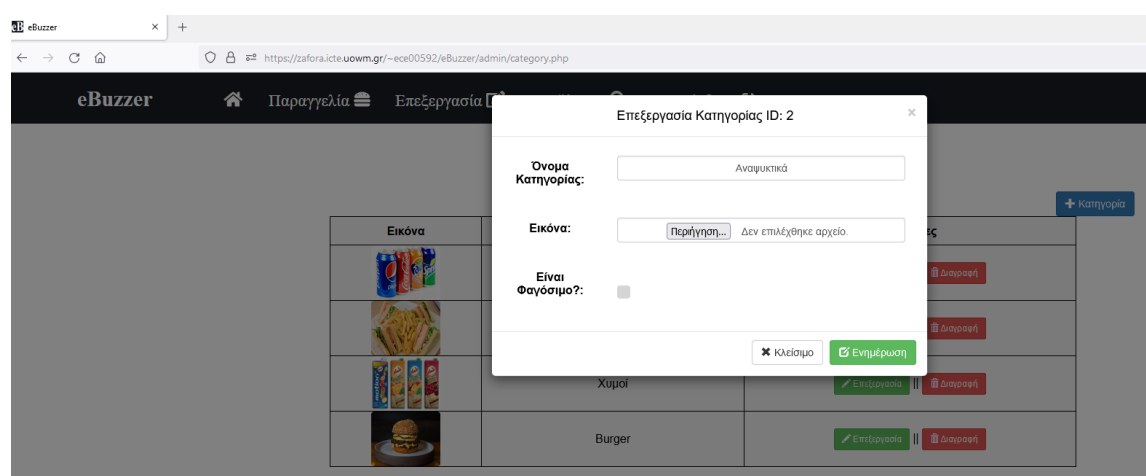
Ο χρήστης διαχειριστής όπως αναφέραμε έχει αυξημένες δυνατότητες και λειτουργίες στο ΠΣ. Αρχικά θα δούμε ότι ο διαχειριστής μπορεί να αναζητήσει, να επεξεργαστεί και να διαγράψει οποιοδήποτε χρήστη από το ΠΣ, όπως φαίνεται πιο κάτω στην Εικόνα 38. Εδώ να αναφέρουμε ότι η αναζήτηση είναι δυναμική καθώς ο διαχειριστής πληκτρολογεί όνομα χρήστη για αναζήτηση αλλάζουν τα αποτελέσματα δυναμικά μέσω τεχνικής AJAX. Στην συνέχεια, από το μενού επεξεργασία ο διαχειριστής μπορεί να διαχειρίζεται τις κατηγορίες των προϊόντων, προσθέτοντας νέα κατηγορία, αλλάζοντας τα στοιχεία από μια υπάρχουσα και διαγράφοντας κατηγορία. Οι παραπάνω ενέργειες φαίνονται στις Εικόνα 39 και Εικόνα 40 που ακολουθούν.



Εικόνα 38: Διαχείριση χρηστών



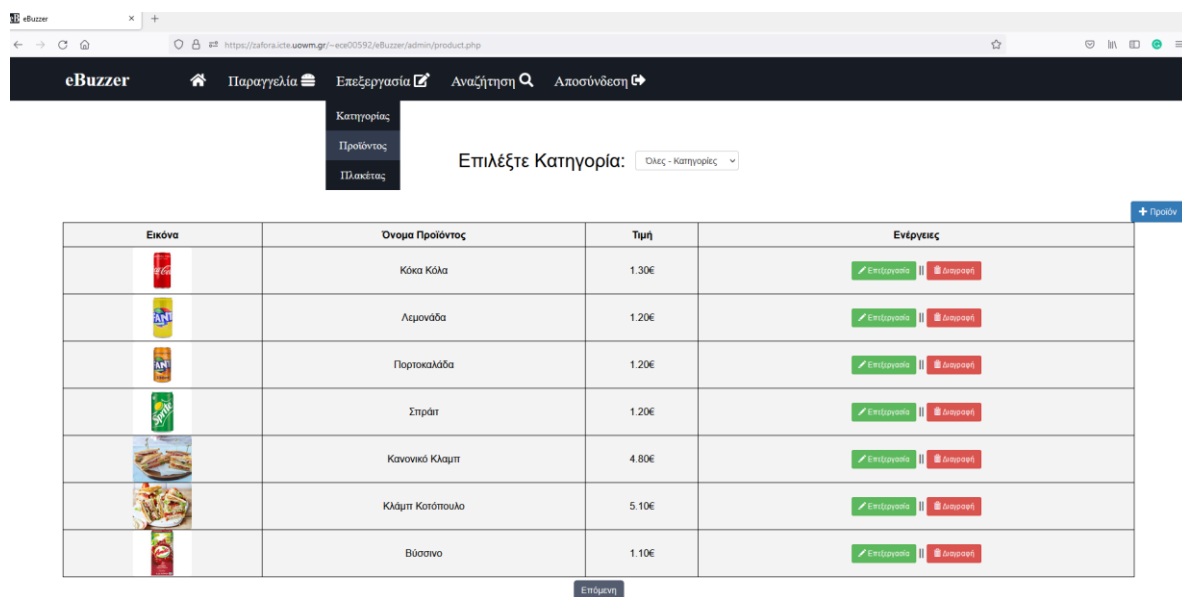
Εικόνα 39: Διαχείριση κατηγοριών



Εικόνα 40: Επεξεργασία κατηγορίας

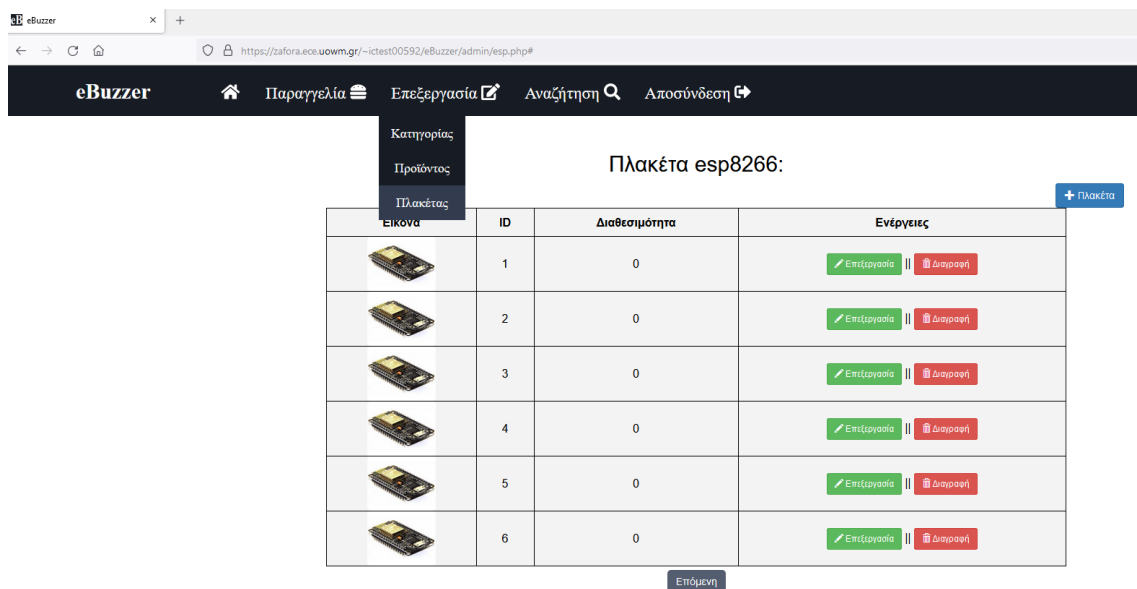
Με ανάλογο τρόπο, που φαίνεται στην Εικόνα 41, γίνεται η διαχείριση των προϊόντων, όπως βλέπουμε ο διαχειριστής μπορεί να εμφανίσει από ένα dropdown μενού τα προϊόντα από όλες τις κατηγορίες ή να επιλέξει κάποια από αυτές. Επίσης, μπορεί να προσθέσει ένα προϊόν, να το επεξεργαστεί ή να το διαγράψει κάνοντας κλικ σε κάποιο από τα αντίστοιχα κουμπιά που υπάρχουν. Επίσης, αν τα προϊόντα είναι περισσότερα από επτά γίνεται σελιδοποίηση και ο διαχειριστής μπορεί να εμφανίζει προϊόντα της επόμενης ή της προηγούμενης σελίδας από τα κουμπιά που υπάρχουν στο κάτω μέρος.





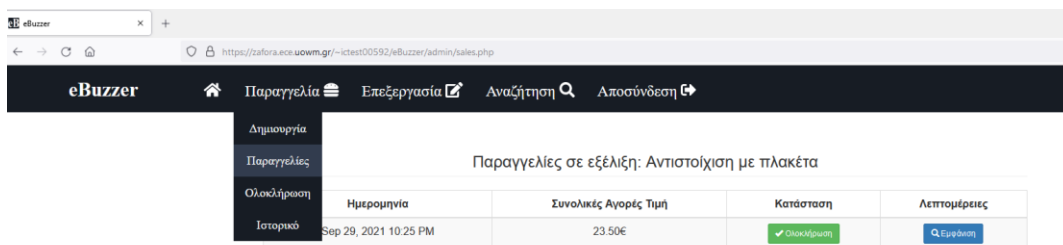
Εικόνα 41: Διαχείριση προϊόντων

Μία άλλη δυνατότητα που έχει ο διαχειριστής είναι να διαχειρίζεται τις πλακέτες esp8266. Στην Εικόνα 42 φαίνονται οι λειτουργίες που μπορεί να ενεργήσει ο διαχειριστής είναι: η καταχώριση μιας νέας πλακέτας esp8266, η επεξεργασία μιας υπάρχουσας και η διαγραφή κάποιας από τις πλακέτες esp8266 που έχει το ΠΣ. Ο διαχειριστής μπορεί να αλλάζει τη διαθεσιμότητα της πλακέτας esp8266 όταν αυτή χορηγείται σε πελάτη κατά το στάδιο της παραγωγής.

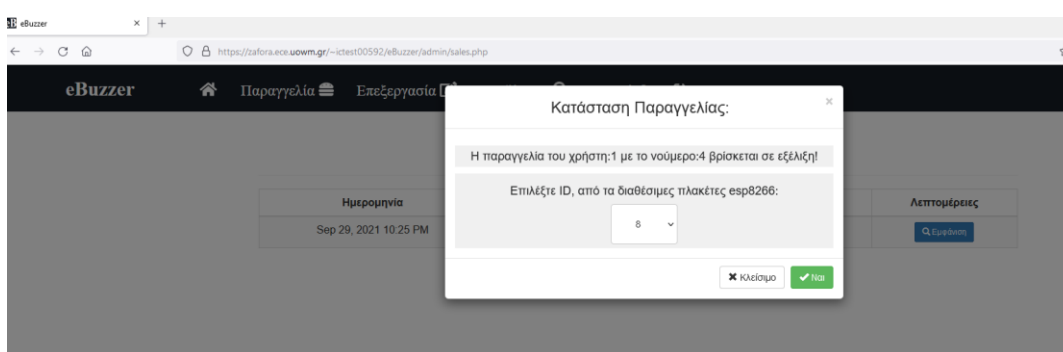


Εικόνα 42: Διαχείριση πλακετών esp

Επίσης, ο διαχειριστής μπορεί να παρακολουθεί τις παραγγελίες, όπως φαίνεται από την Εικόνα 43, ώστε να τις αντιστοιχεί με κάποια διαθέσιμη πλακέτα esp8266, το οποίο φαίνεται στην Εικόνα 44.

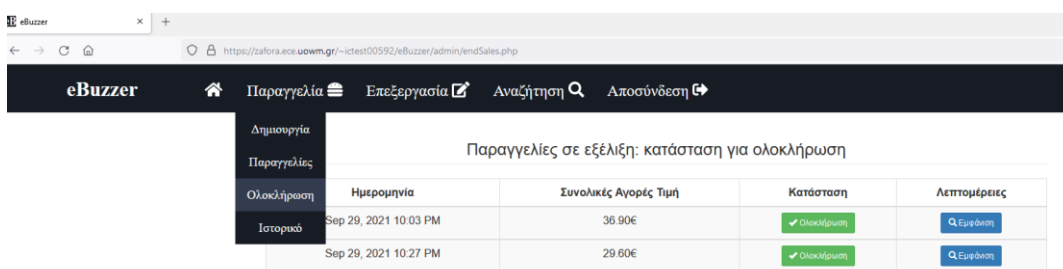


Εικόνα 43: Παραγγελία σε εξέλιξη



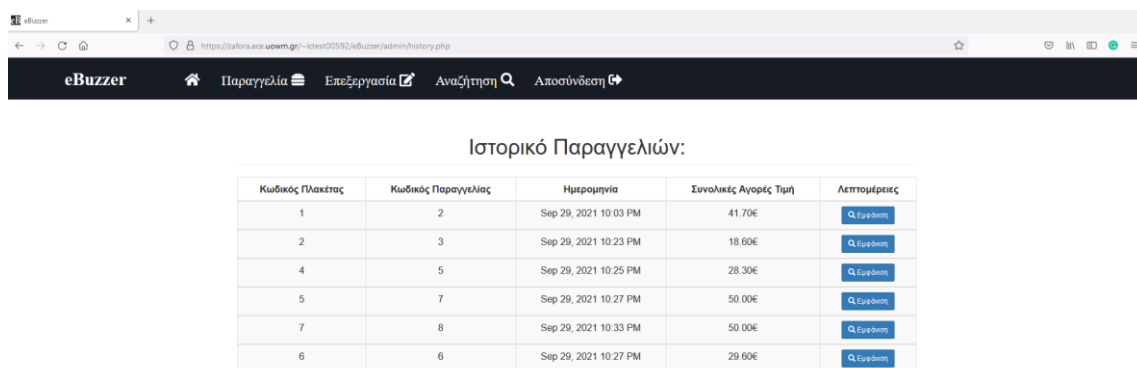
Εικόνα 44: Αντιστοίχιση με πλακέτα esp

Ο διαχειριστής, όπως φαίνεται στην Εικόνα 45, ορίζει την παραγγελία ως έτοιμη και εκείνη την στιγμή ο πελάτης ενημερώνεται μέσω της πλακέτας esp8266 για την ετοιμότητά της.



Εικόνα 45: Ολοκλήρωση παραγγελίας

Τέλος ο διαχειριστής έχει την δυνατότητα να προβάλει όλες τις παραγγελίες των πελατών, όπως φαίνεται στην Εικόνα 46. Στην συνέχεια θα γίνει περιγραφή της διαδικασίας για την διενέργεια της παραγγελίας.

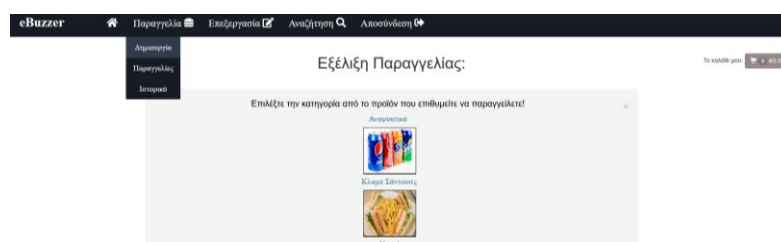


Κωδικός Πλακέτας	Κωδικός Παραγγελίας	Ημερομηνία	Συνολικές Αγορές Τιμή	Λεπτομέρειες
1	2	Sep 29, 2021 10:03 PM	41.70€	<a href="#">Εμφάνιση</a>
2	3	Sep 29, 2021 10:23 PM	18.60€	<a href="#">Εμφάνιση</a>
4	5	Sep 29, 2021 10:25 PM	28.30€	<a href="#">Εμφάνιση</a>
5	7	Sep 29, 2021 10:27 PM	50.00€	<a href="#">Εμφάνιση</a>
7	8	Sep 29, 2021 10:33 PM	50.00€	<a href="#">Εμφάνιση</a>
6	6	Sep 29, 2021 10:27 PM	29.60€	<a href="#">Εμφάνιση</a>

Εικόνα 46: Ιστορικό παραγγελιών

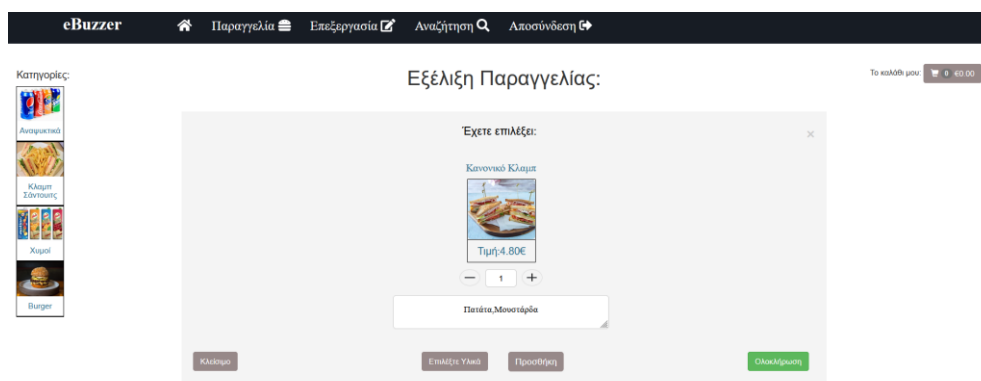
### 4.3 Διενέργεια παραγγελίας μέσω του ΠΣ

Μετά την είσοδό του στο ΠΣ είτε ο απλός χρήστης είτε ο διαχειριστής από το μενού Παραγγελία μπορεί να ξεκινήσει την παραγγελία του η οποία ολοκληρώνεται σε τρία στάδια. Όπως φαίνεται και στην Εικόνα 47 στο πρώτο στάδιο επιλέγει την κατηγορία προϊόντος που θέλει να παραγγείλει, μεταξύ των αναψυκτικών, κλαμπ σάντουιτς, χυμών και burger.



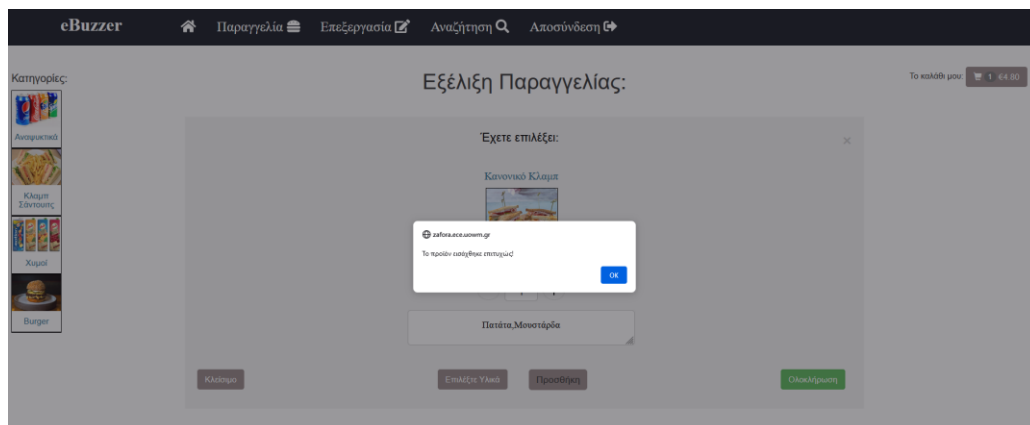
Εικόνα 47: Διενέργεια παραγγελίας 1ο στάδιο

Στην συνέχεια στο δεύτερο στάδιο επιλέγει το προϊόν της κατηγορίας, που για το συγκεκριμένο παράδειγμα επιλέχθηκε η κατηγορία κλαμπ σάντουιτς, με περαιτέρω επιλογές κανονικό κλαμπ και κλαμπ κοτόπουλο. Αφού επιλέξει το κανονικό κλαμπ σάντουιτς ο χρήστης μπορεί να επιλέξει έξτρα υλικά (πατάτες, κέτσαπ, μουστάρδα, ντομάτα, μαρούλι) καθώς και την ποσότητα, όπως φαίνεται στην παρακάτω Εικόνα 48.



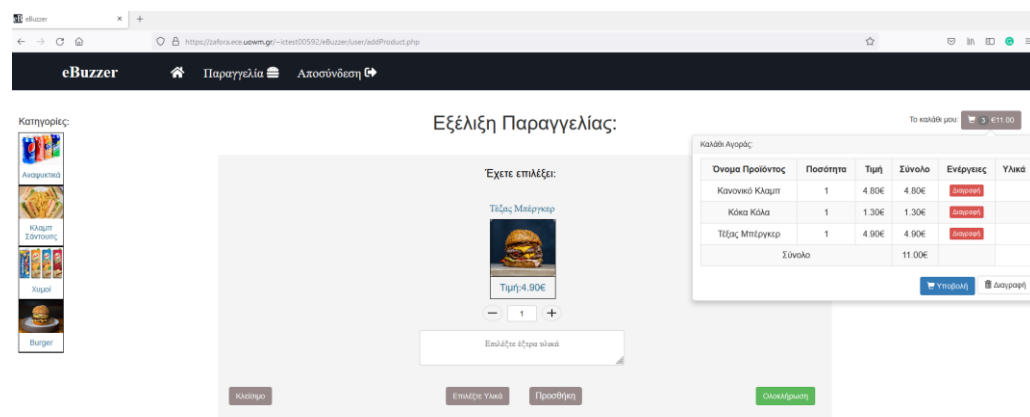
Εικόνα 48: Διενέργεια παραγγελίας 2ο στάδιο

Στην συνέχεια ο χρήστης πατώντας το κουμπί προσθήκη η παραγγελία μπαίνει στο καλάθι με κατάλληλο μήνυμα που φαίνεται στην Εικόνα 49. Επιπλέον, ο χρήστης μπορεί να παραγγείλει όσα προϊόντα επιθυμεί με την ανωτέρω διαδικασία, προσθέτοντάς τα στο καλάθι.

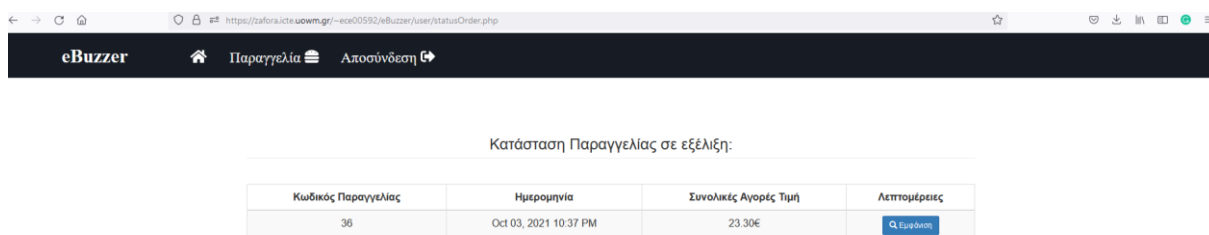


Εικόνα 49: Προσθήκη προϊόντος στο καλάθι

Τέλος στο τρίτο στάδιο ο χρήστης από το κουμπί ολοκλήρωση ή από το κουμπί υποβολή (μέσω του καλαθιού) μπορεί να ολοκληρώσει την παραγγελία του βλέποντας το συνολικό της κόστος, όπως φαίνεται στην Εικόνα 50. Αμέσως μετά ο χρήστης κατευθύνεται στο στάδιο κατάστασης – εξέλιξης παραγγελίας όπως φαίνεται στην Εικόνα 51.



Εικόνα 50: Ολοκλήρωση διενέργειας παραγγελίας



Εικόνα 51: Κατάσταση παραγγελίας

#### **4.4 Σύνοψη κεφαλαίου**

Στο κεφάλαιο αυτό έγινε η παρουσίαση λειτουργίας του ΠΣ με αρκετή λεπτομέρεια. Όλες οι λειτουργίες παρουσιάστηκαν με στιγμιότυπα για την ευκολότερη κατανόησή τους. Στο επόμενο κεφάλαιο ως επίλογο της διπλωματικής εργασίας παρουσιάζονται τα συμπεράσματα, η swot ανάλυση, η εκτίμηση συνεχόμενης λειτουργίας και οι μελλοντικές επεκτάσεις της.

## 5ο Κεφάλαιο - Επίλογος

Στο κεφάλαιο αυτό παρουσιάζονται τα συμπεράσματα, η SWOT (Strengths, Weaknesses, Opportunities, Threats) ανάλυση και οι μελλοντικές επεκτάσεις της παρούσας διπλωματικής εργασίας. Γίνεται καθολική και συνοπτική παρουσίαση όλων όσων υλοποιήθηκαν, τα συμπεράσματα που εξήχθησαν από τη διαδικασία αυτή, ορισμένες ιδέες για μελλοντικές επεκτάσεις, τις μετρικές του ΠΣ και η ανάλυση SWOT του συστήματος.

### 5.1 Σύνοψη του ΠΣ

Στην παρούσα διπλωματική εργασία σχεδιάστηκε και υλοποιήθηκε ένα ΠΣ με στόχο την υποστήριξη της λειτουργίας ενός ταχυφαγείου. Το ΠΣ δύναται να δέχεται παραγγελίες μέσω του Διαδικτύου, είτε απομακρυσμένα, είτε δια ζώσης. Χρησιμοποιώντας την πλακέτα esp8266, ο πελάτης ενημερώνεται για την παραγγελία όταν αυτή είναι έτοιμη. Για την κατασκευή του ΠΣ χρησιμοποιήθηκαν τεχνολογίες ανοιχτού κώδικα (HTML, CSS, JavaScript, jQuery, AJAX, Bootstrap, PHP, MySQL).

Το ΠΣ που υλοποιήθηκε έχει δύο είδη χρηστών τον απλό χρήστη και τον διαχειριστή. Ο απλός χρήστης – πελάτης έχει την δυνατότητα σε μία παραγγελία να επιλέξει πολλά προϊόντα και μάλιστα σε όσα είναι φαγώσιμα να κάνει επιλογή για ορισμένα από τα χαρακτηριστικά τους. Επίσης, μπορεί να διαχειριστεί τις παραγγελίες του, ενώ τέλος υπάρχει δυνατότητα να βλέπει το ιστορικό των ολοκληρωμένων παραγγελιών του. Ο διαχειριστής έχει την πλήρη διαχείριση των χρηστών, παραγγελιών, προϊόντων, κατηγοριών προϊόντων, ιστορικού και πλακετών (esp8266). Επιπλέον οι πλακέτες esp8266, οι οποίες τοποθετούνται μέσα σε ειδικά σχεδιασμένο και εκτυπωμένο από 3D εκτυπωτή κουτί, ενημερώνουν μέσω ηχητικού σήματος και φωτεινής ένδειξης με τον κωδικό παραγγελίας τον πελάτη για την ετοιμότητα της διαζώσεις παραγγελίας του. Τέλος, ο Πίνακας 1 έχει τα δεδομένα με τις μετρικές του ΠΣ.

Μετρικές Συστήματος	Αριθμός
Φάκελοι (find . -mindepth 1 -maxdepth 10 -type d   wc -l)	15
Αριθμός αρχείων (find . -type f   wc -l)	267
Γραμμές κώδικα (find . -name '*.*'   xargs wc -l)	46284
Συνολικό μέγεθος ΠΣ	10.5MB

Πίνακας 1: Μετρικές ΠΣ

## 5.2 Συμπεράσματα

Με την ολοκλήρωση της διπλωματικής εργασίας τα συμπεράσματα που εξήχθησαν είναι αρκετά ενδιαφέροντα και σχετίζονται με την λειτουργικότητα και τα οφέλη που παρέχονται από το ΠΣ. Μειώνεται αισθητά η αναμονή του πελάτη στο στάδιο της παραγγελίας, είτε στο κατάστημα, είτε εξ' αποστάσεως, διότι δεν υπάρχει ουρά αναμονής για τη διενέργεια της παραγγελίας. Το μόνο που μπορεί να επηρεάσει τον χρόνο αναμονής είναι η μη διαθεσιμότητα πλακέτας esp8266 το οποίο σχετίζεται με την μέγιστη δυνατότητα εξυπηρέτησης στις ώρες αιχμής του εστιατορίου. Εξοικονομείται τουλάχιστον ένας υπάλληλος για καταγραφή των παραγγελιών ανεξαρτήτου τρόπου παραγγελίας και περισσότεροι από ένας υπάλληλοι με ρόλο σερβιτόρου για την διανομή των παραγγελιών στο εστιατόριο διότι ο πελάτης παραλαμβάνει την παραγγελία μόλις αυτή είναι έτοιμη μόνος του αφού πρώτα ενημερωθεί από την πλακέτα esp8266. Τέλος γίνεται αποδοτικότερη η καταγραφή και η παρακολούθηση όλων των παραγγελιών που διενεργούνται στο εστιατόριο.

## 5.3 Ανάλυση SWOT

Η ανάλυση SWOT αποτελεί ένα εργαλείο σχεδιασμού στρατηγικής, το οποίο βοηθά να αντλήσουμε πληροφορίες σχετικές με το ΠΣ μας. Οι πληροφορίες αυτές συσχετίζουν τα δυνατά με τα αδύνατα σημεία, όπως επίσης τις ευκαιρίες με τις απειλές. Η συλλογή αυτών των πληροφοριών καθιστά ευκολότερο τον καθορισμό των απαιτήσεων και των μελλοντικών επεκτάσεων του συστήματος.

### 5.3.1 Δυνατά σημεία (strengths)

Το βασικότερο πλεονέκτημα του ΠΣ που υλοποιήθηκε είναι ότι προσφέρει σε ένα εστιατόριο την δυνατότητα να διενεργούνται παραγγελίες χωρίς παρέμβαση τρίτου και χωρίς ουρά αναμονής στο στάδιο της παραγγελίας. Επίσης, προσφέρεται η δυνατότητα παραγγελίας από οποιαδήποτε έξυπνη συσκευή (H/Y, ταμπλέτα, κινητό, τηλεόραση, κλπ.). Τέλος το ΠΣ είναι ανοιχτού λογισμικού και ανοιχτού υλικού, συνεπώς είναι δωρεάν και διαθέσιμο για επεκτάσεις και μετατροπές για όποιον επιθυμεί.

### 5.3.2 Αδύνατα σημεία (weaknesses)

Το βασικότερο μειονέκτημα του ΠΣ που υλοποιήθηκε είναι ότι για την λειτουργία του απαιτείται πρόσβαση στο Διαδίκτυο. Παρόλο που έχουν γίνει δοκιμές σε εικονική λειτουργία του ΠΣ ένα επιπλέον μειονέκτημα αποτελεί το γεγονός ότι δεν έχει δοκιμαστεί το ΠΣ σε πραγματικές συνθήκες για να προσδιοριστούν τυχόν απρόσμενες δυσλειτουργίες.

### 5.3.3 Ευκαιρίες (opportunities)

Σε ένα ανταγωνιστικό πλαίσιο που επικρατεί στον χώρο της εστίασης με εστιατόρια να λειτουργούν με διάφορους τρόπους. Το ΠΣ που υλοποιήθηκε δίνει την ευκαιρία σε πολλές από αυτές να μεταβούν χωρίς κόστος από ένα παραδοσιακό μοντέλο λειτουργίας σε ένα πιο σύγχρονο μοντέλο λειτουργίας. Επίσης οι επιχειρήσεις εστίασης της σημερινής εποχής, λόγω COVID-19, επιβάλλεται να μεταβούν σε ένα εξ' αποστάσεως μοντέλο λειτουργίας, ώστε να μπορέσουν να επιβιώσουν στο ανταγωνιστικό περιβάλλον που επικρατεί και με το ΠΣ αυτό μπορούν να πετύχουν την μετάβαση χωρίς κόστος.

### 5.3.4 Απειλές (threats)

Η κυριότερη απειλή είναι το ανταγωνιστικό περιβάλλον που επικρατεί στον τομέα της εστίασης σε συνδυασμό με τη διαρκή εξέλιξη των τεχνολογιών, τα οποία επιτάσσουν τη συνεχή βελτίωση των ΠΣ ώστε να παραμένουν ανταγωνιστικά στο πέρασμα του χρόνου.

## 5.4 Εκτίμηση συνεχόμενης λειτουργίας (scaling)

Για να γίνει μία εκτίμηση της συνεχόμενης λειτουργίας του ΠΣ θα πρέπει αρχικά να γίνει μία εκτίμηση των εγγραφών που μπορεί να υποστηρίξει η βάση δεδομένων του ΠΣ. Αν μελετήσουμε τους πίνακες που θα αποθηκεύονται όλα τα δεδομένα θα παρατηρήσουμε ότι τα κύρια και τα ξένα κλειδιά τους είναι όλα τύπου int. Επομένως, η μέγιστη τιμή που μπορούν να αποθηκεύσουν είναι 4byte ή 32bit. Επιπλέον, ο Πίνακας 2 δείχνει τις μέγιστες εγγραφές που μπορεί να υποστηρίξει το ΠΣ σε σχέση με τη βάση δεδομένων σε πενταετές πλάνο λειτουργίας. Να σημειωθεί ότι, το πεδίο id\_pur εκτιμήθηκε με μέσο όρο πέντε προϊόντων για κάθε παραγγελία. Άρα μία φυσιολογική επιχείρηση δεν μπορεί να ξεπεράσει τον μέγιστο αριθμό εγγραφών την ημέρα για κανένα από τα παρακάτω πεδία.

Πεδίο	Μέγιστη τιμή int	Μέγιστος ετήσιος αριθμός εγγραφών	Μέγιστος ημερήσιος αριθμός εγγραφών
id_pur	4294967295	171798691,8	470681,3474
order_id	4294967295	858993459	2353406,737
history_id	4294967295	858993459	2353406,737
user_id	4294967295	858993459	2353406,737

Πίνακας 2: Μέγιστες τιμές πενταετούς πλάνου συνεχόμενης λειτουργίας

Επομένως, σε μία φυσιολογική λειτουργία μιας επιχείρησης εστίασης με μέσο όρο τριακόσιες (300) παραγγελίες την ημέρα, πέντε (5) προϊόντα για κάθε παραγγελία και τριακόσιους (300) διαφορετικούς χρήστες την ημέρα θα έχουμε τα δεδομένα που έχει ο Πίνακας 3 που ακολουθεί. Συνεπώς το ΠΣ δεν πλησιάζει τα όριά του. Επίσης, έχει



προγραμματιστεί ετήσια συντήρησή του και μεταφορά όλων των δεδομένων από την ενεργή βάση σε μια εφεδρική βάση, η διαδικασία της οποίας αναλύεται στην παράγραφο 7.3 του παραρτήματος.

Πεδίο	Πενταετής αριθμός εγγραφών	Ετήσιος αριθμός εγγραφών	Ημερήσιος αριθμός εγγραφών
id_pur	2737500	547500	1500
order_id	547500	109500	300
history_id	547500	109500	300
user_id	547500	109500	300

Πίνακας 3: Πενταετές πλάνο φυσιολογικής συνεχόμενης λειτουργίας

## 5.5 Μελλοντικές επεκτάσεις

Δεδομένης της τεχνολογικής εξέλιξης σε μερικά χρόνια το ΠΣ θα έχει ξεπεραστεί και θα πρέπει να γίνουν βελτιώσεις. Παρόλο που το σύστημα είναι πλήρως λειτουργικό και καλύπτει τις ανάγκες που προαναφέραμε, υπάρχει δυνατότητα να βελτιωθεί και να προστεθούν νέες λειτουργίες. Ορισμένες πιθανές μελλοντικές επεκτάσεις του μπορεί να είναι οι εξής:

- Προσθήκη πλακέτας esp8266 στο ΠΣ της για να εξασφαλίσει την ασφαλή και χωρίς αλλοίωση μεταφορά των προϊόντων μέσω ειδικού κουτιού που θα ανοίγει με κωδικό μίας χρήσης.
- Προσθήκη ασύρματης φόρτισης στο esp8266.
- Δημιουργία ειδικής εφαρμογής για Android και iOS.

## 5.6 Σύνοψη κεφαλαίου

Στο κεφάλαιο αυτό συνοψίσαμε τον τρόπο λειτουργίας και σχεδίασης του ΠΣ και αναφερθήκαμε στα συμπεράσματα και τα οφέλη χρήσης του. Επιπλέον έγινε ανάλυση SWOT του συστήματος, εκτιμήθηκε ένα πενταετές πλάνο συνεχόμενης λειτουργίας του ΠΣ και τέλος, έγινε αναφορά σε πιθανές μελλοντικές επεκτάσεις του.

## 6ο Κεφάλαιο - Βιβλιογραφία

- [1] «Ελληνική Στατιστική Αρχή - Στατιστικό Μητρώο Επιχειρήσεων /2018,» [Ηλεκτρονικό]. Διαθέσιμο: [https://www.statistics.gr/el/statistics?p\\_p\\_id=documents\\_WAR\\_publicationsportlet\\_INSTANCE\\_Mr0GiQJSgPHd&p\\_p\\_lifecycle=2&p\\_p\\_state=normal&p\\_p\\_mode=view&p\\_p\\_cacheability=cacheLevelPage&p\\_p\\_col\\_id=column-2&p\\_p\\_col\\_count=4&p\\_p\\_col\\_pos=2&\\_documents\\_WAR\\_publicat.](https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_Mr0GiQJSgPHd&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col_id=column-2&p_p_col_count=4&p_p_col_pos=2&_documents_WAR_publicat.) [Πρόσβαση 17 Αυγούστου 2021].
- [2] "WITEC - Wireless Intelligent Technology," WITEC - Wireless Intelligent Technology, 2016. [Online]. Available: <https://www.witec.gr/covid-19-fb-full/>. [Accessed 21 September 2021].
- [3] "Evoke - Case Studies," Evoke Creative Limited, 2019. [Online]. Available: <https://www.evoke-creative.com/clients/>. [Accessed 21 September 2021].
- [4] "evoke-creative - Case Studies MCDONALDS," Evoke Creative Limited, 2019. [Online]. Available: <https://www.evoke-creative.com/project/mcdonalds/>. [Accessed 21 September 2021].
- [5] "FOODSERVICE EQUIPMENT JOURNAL - SPECIAL REPORT: The self-service revolution causing QSRs to rethink kitchen design," Promedia Publishing Ltd, 5 October 2016. [Online]. Available: <https://www.foodserviceequipmentjournal.com/why-qsr-operators-are-turning-to-self-service-kiosks-to-maximise-output/>. [Accessed 21 September 2021].
- [6] "SIGNALGRYD - Self Service Buzzers," SIGNALGRYD Pte. Ltd., 2021. [Online]. Available: <https://www.signalgryd.com/industries/cafe-restaurant-food-and-beverage/customer-paging-system/>. [Accessed 21 September 2021].
- [7] A. Sonu, D. Supraja, G. Navya and G. Pavani, "Smart and Secure Delivery System: a Novel Idea using IoT and Cloud Computing Technologies," *Journal of Emerging Technologies and Innovative Research*, vol. 6, no. 4, April 2019.
- [8] Θ. Κόλιας, «Ψηφιακή Βιβλιοθήκη και Ιδρυματικό Αποθετήριο Ψηφίδα - Έξυπνη κατοικία, προσέγγιση μέσω του διαδικτύου των πραγμάτων,» Ιούνιος 2021. [Ηλεκτρονικό]. Διαθέσιμο: <https://dspace.lib.uom.gr/handle/2159/25643>. [Πρόσβαση 18 Αυγούστου 2021].
- [9] Κ. Κουτσαντώνη, «Ιδρυματικό Καταθετήριο Αθηνά του Τ.Ε.Ι. Δυτικής Ελλάδας - Διαχείριση καλλιιεργειών με συστήματα του διαδικτύου των πραγμάτων (IoT),» 2021. [Ηλεκτρονικό]. Διαθέσιμο:

- <http://repository.library.teimes.gr/xmlui/handle/123456789/8997>. [Πρόσβαση 18 Αυγούστου 2021].
- [10] Μ.-Σ. Κολοβού, «Ιδρυματικό Αποθετήριο Πανεπιστημίου Πατρών Νημερτής - Η επίδραση του Διαδικτύου των Πραγμάτων στον αθλητισμό,» Ιούλιος 2019. [Ηλεκτρονικό]. Διαθέσιμο: <http://hdl.handle.net/10889/13348>. [Πρόσβαση 18 Αυγούστου 2021].
- [11] Σ. Ζωμένος, «Ιδρυματικό Αποθετήριο Πανεπιστημίου Πατρών - Η χρήση του Διαδικτύου των Πραγμάτων (Internet of Things - IoT) για την [sic] βελτίωση της ανταγωνιστικότητας των επιχειρήσεων : μελέτη περίπτωσης σε super market,» Φεβρουάριος 2021. [Ηλεκτρονικό]. Διαθέσιμο: <http://hdl.handle.net/10889/14592>. [Πρόσβαση 18 Αυγούστου 2021].
- [12] Ε. Γ. Σωτηρόπουλος και Ε. Α. Γούλα, «Ιδρυματικό Αποθετήριο Πανεπιστημίου Θεσσαλίας - Το διαδίκτυο των "πραγμάτων" στη βιομηχανία,» Ιούλιος 2018. [Ηλεκτρονικό]. Διαθέσιμο: <http://hdl.handle.net/11615/49298>. [Πρόσβαση 18 Αυγούστου 2021].
- [13] Ι. Λεμονής και Δ. Σιδηρόπουλος, «Ιδρυματικό Αποθετήριο Ωκεανίς - Διαδίκτυο των πραγμάτων σε επιχειρήσεις εστίασης,» Φεβρουάριος 2020. [Ηλεκτρονικό]. Διαθέσιμο: <http://oceanis.lib.teipir.gr/xmlui/handle/123456789/5232>. [Πρόσβαση 18 Αυγούστου 2021].
- [14] M. Schwartz, Internet of Things with ESP8266, Birmingham, U.K.: Packt Publishing Ltd, 2016.
- [15] "What is Arduino?," Arduino, 2021. [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction>. [Accessed 03 October 2021].
- [16] "Wikipedia - Prusa i3," Wikimedia Foundation, 2003. [Online]. Available: [https://en.wikipedia.org/wiki/Prusa\\_i3](https://en.wikipedia.org/wiki/Prusa_i3). [Accessed 26 September 2021].
- [17] "PrusaPrinters Blog - What's up with Original Prusa i3?," PrusaPrinters, 2021. [Online]. Available: [https://blog.prusaprinters.org/whats-up-with-original-prusa-i3\\_4124/](https://blog.prusaprinters.org/whats-up-with-original-prusa-i3_4124/). [Accessed 26 September 2021].
- [18] "INTRODUCTION & DOWNLOAD," Prusa Research a.s., 2021. [Online]. Available: <https://www.prusa3d.com/prusaslicer/>. [Accessed 26 September 2021].
- [19] "Welcome to PrusaSlicer documentation (PrusaSlicer manual)," Prusa Research a.s., 2021. [Online]. Available: [https://help.prusa3d.com/en/article/general-info\\_1910#\\_ga=2.46187492.2101527527.1632673941-1568838571.1632673941](https://help.prusa3d.com/en/article/general-info_1910#_ga=2.46187492.2101527527.1632673941-1568838571.1632673941). [Accessed 26 September 2021].
- [20] "GitHub - fritzing / fritzing-app," GitHub, Inc., [Online]. Available: <https://github.com/fritzing/fritzing-app>. [Accessed 18 August 2021].

- 
- [21] "HTML History," [Online]. Available: [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp). [Accessed 18 August 2021].
- [22] "HTML Living Standard — Last Updated 12 August 2021," WHATWG (Apple, Google, Mozilla, Microsoft), [Online]. Available: <https://html.spec.whatwg.org/>. [Accessed 18 August 2021].
- [23] "Learn to style HTML using CSS," Mozilla and individual contributors, 2005. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Learn/CSS>. [Accessed 18 August 2021].
- [24] "Cascading Style Sheets home page," W3C, 1994. [Online]. Available: <https://www.w3.org/Style/CSS/>. [Accessed 18 August 2021].
- [25] "Cascading Style Sheets working group," W3C, 1994. [Online]. Available: <https://www.w3.org/Style/CSS/members>. [Accessed 18 August 2021].
- [26] T. Atkins Jr., E. J. Etemad and F. Rivoal, "CSS Snapshot 2020," W3C Working Group, 22 December 2020. [Online]. Available: <https://www.w3.org/TR/css-2020/>. [Accessed 18 August 2021].
- [27] "Wikipedia JavaScript," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/JavaScript>. [Accessed 18 August 2021].
- [28] D. A. Rauschmayer, "How JavaScript Was Created," in *Speaking JavaScript*, Sebastopol, O'Reilly Media, 2015.
- [29] "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine," OpenJS Foundation, 27 May 2009. [Online]. Available: <https://nodejs.org/en/>. [Accessed 18 August 2021].
- [30] "Angular - The modern web developer's platform," Google, 14 September 2016. [Online]. Available: <https://angular.io/>. [Accessed 18 August 2021].
- [31] "React - A JavaScript library for building user interfaces," Facebook Open Source, 29 May 2013. [Online]. Available: <https://reactjs.org/>. [Accessed 18 August 2021].
- [32] "Vue.js - The Progressive JavaScript Framework," Evan You, February 2014. [Online]. Available: <https://v3.vuejs.org/>. [Accessed 18 August 2021].
- [33] "Ember.js," Ember Core Team, 8 December 2011. [Online]. Available: <https://emberjs.com/>. [Accessed 18 August 2021].
- [34] "Build Full-Stack Javascript Apps with Meteor," Meteor Software, 20 January 2012. [Online]. Available: <https://www.meteor.com/>. [Accessed 18 August 2021].
- [35] "jQuery write less, do more.," The jQuery Team, 26 Αυγούστου 2006. [Online]. Available: <https://jquery.com/>. [Accessed 18 August 2021].

- [36] Δ. Π. Καλφούντζος και Α. Μπέης, «Ανάπτυξη website βάσει responsive design (RWD) με χρήση των τεχνολογιών HTML5, CSS3 (Bootstrap), Javascript (jQuery),» 2019. [Ηλεκτρονικό]. Διαθέσιμο: <http://hdl.handle.net/11615/52120>. [Πρόσβαση 18 Αυγούστου 2021].
- [37] "Wikipedia - jQuery," 2003 Wikimedia Foundation. [Online]. Available: <https://en.wikipedia.org/wiki/JQuery>. [Accessed 18 August 2021].
- [38] "jQuery foundation - The jQuery Team," OpenJS Foundation, 2019. [Online]. Available: <https://jquery.org/team/>. [Accessed 18 August 2021].
- [39] "Wikipedia - Ajax (programming)," Wikimedia Foundation, 2003. [Online]. Available: [https://en.wikipedia.org/wiki/Ajax\\_\(programming\)](https://en.wikipedia.org/wiki/Ajax_(programming)). [Accessed 18 August 2021].
- [40] "Wikipedia - Bootstrap (front-end framework)," Wikimedia Foundation, 2003. [Online]. Available: [https://en.wikipedia.org/wiki/Bootstrap\\_\(front-end\\_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)). [Accessed 18 August 2021].
- [41] "getbootstrap.com - Team," Bootstrap team, [Online]. Available: <https://getbootstrap.com/docs/5.1/about/team/>. [Accessed 18 August 2021].
- [42] "Wikipedia - PHP," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/PHP>. [Accessed 18 August 2021].
- [43] "Laravel - The PHP Framework for Web Artisans," Laravel LLC, 2011. [Online]. Available: <https://laravel.com/>. [Accessed 18 August 2021].
- [44] "CodeIgniter - CodeIgniter Rocks," CodeIgniter Foundation, 2019. [Online]. Available: <https://www.codeigniter.com/>. [Accessed 18 August 2021].
- [45] "Symfony - PHP framework for web projects," Symfony SAS, 2005. [Online]. Available: <https://symfony.com/>. [Accessed 18 August 2021].
- [46] "CakePHP - New CakePHP 4.1 Strawberry, Cleaner. Faster, Tasty as usual," Cake Software Foundation, 2005. [Online]. Available: <https://cakephp.org/>. [Accessed 18 August 2021].
- [47] "YiiFramework - Yii is a fast, secure, and efficient PHP framework.," Yii Software, 2008. [Online]. Available: <https://www.yiiframework.com/>. [Accessed 18 August 2021].
- [48] "Laminas - Laminas Project, the enterprise-ready PHP Framework and components.," Technical Steering Committee (TSC) - Linux Foundation, 2020. [Online]. Available: 18.
- [49] "PHP - PHP is a popular general-purpose scripting language that is especially suited to web development," The PHP Group, 2001. [Online]. Available: <https://www.php.net/>. [Accessed 18 August 2021].
- [50] «W3Techs - Usage statistics of PHP for websites,» Q-Success, 2009. [Online]. Available: <https://w3techs.com/technologies/details/pl-php>. [Accessed 18 August 2021].

- 
- [51] "Wikipedia - MySQL," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/MySQL>. [Accessed 18 August 2021].
- [52] "Wikipedia - MariaDB," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/MariaDB>. [Accessed 18 August 2021].
- [53] "Wikipedia - Notepad+," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/Notepad%2B%2B>. [Accessed 21 September 2021].
- [54] "Wikipedia - Scintilla (software)," Wikimedia Foundation, 2003. [Online]. Available: [https://en.wikipedia.org/wiki/Scintilla\\_\(software\)](https://en.wikipedia.org/wiki/Scintilla_(software)). [Accessed 21 September 2021].
- [55] "Introducing WinSCP," WinSCP.net, 2021. [Online]. Available: <https://winscp.net/eng/docs/introduction>. [Accessed 26 September 2021].
- [56] "WinSCP Project History," WinSCP.net, 2021. [Online]. Available: [https://winscp.net/eng/docs/project\\_history](https://winscp.net/eng/docs/project_history). [Accessed 26 September 2021].
- [57] "PuTTY: a free SSH and Telnet client," [Online]. Available: <https://www.chiark.greenend.org.uk/~sgtatham/putty/>. [Accessed 26 September 2021].
- [58] "Wikipedia - PuTTY," Wikimedia Foundation, 2003. [Online]. Available: <https://en.wikipedia.org/wiki/PuTTY>. [Accessed 26 September 2021].
- [59] D. Binkley, D. Davis, D. Lawrie and C. Morrell, "To camelcase or under\_score," in *2009 IEEE 17th International Conference on Program Comprehension*, Vancouver, BC, Canada, 2009.
- [60] «DEPOT - Street Food Social – 15th July,» Depot Cardiff, 2020. [Online]. Available: <https://depotcardiff.com/wp-content/uploads/Street-Food-Social-Outdoors-Cover-2.png>. [Accessed 03 October 2021].

## 7ο Κεφάλαιο - Παράρτημα

### 7.1 Εγκατάσταση εφαρμογής

Για την εγκατάσταση του ΠΣ σε τοπικό σύστημα απαιτείται να εγκατασταθούν τα παρακάτω:

- Εγκατάσταση του λογισμικού Arduino IDE.
- Σύνδεση της κάθε πλακέτας esp8266 μέσω καλωδίου usb στον ηλεκτρονικό υπολογιστή για να γίνει η εισαγωγή του προγράμματος.
- Εισαγωγή του αρχείου main.ino μέσω του λογισμικού Arduino IDE, στην κάθε πλακέτα esp8266, αλλάζοντας τις τιμές των μεταβλητών ssid, password, host, url, request ώστε να είναι σύμφωνες με το ασύρματο δίκτυο και τον εξυπηρετητή του ΠΣ. Ειδικά για την τιμή της μεταβλητής esp\_id αυτή θα πρέπει να αλλάζει για κάθε πλακέτα esp8266.
- Εγκατάσταση του λογισμικού XAMPP, το οποίο περιλαμβάνει: Apache Web Server, PHP, MySQL/MariaDB Server και phpMyAdmin.
- Ενεργοποίηση των υπηρεσιών Apache Web Server και MySQL/MariaDB Server.
- Εγκατάσταση και ρύθμιση του OpenSSL για την υποστήριξη https.
- Εισαγωγή του αρχείου eBuzzer.sql μέσω του phpMyAdmin, ώστε να δημιουργηθεί η βάση δεδομένων και οι πίνακες, όπως αυτοί αναλύθηκαν παραπάνω.
- Αντιγραφή όλων των αρχείων του ΠΣ στον φάκελο htdocs του εξυπηρετητή.
- Προσαρμογή των ρυθμίσεων του αρχείου config.php, που βρίσκεται στο φάκελο include.
- Σύνδεση στο ΠΣ μέσω του <https://localhost/eBuzzer/>

### 7.2 Πλάνο ελέγχου ορθής λειτουργίας ΠΣ

Όσον αφορά τον έλεγχο ορθής λειτουργίας του ΠΣ, με λειτουργίες που εμπλέκονται με μεθόδους POST και GET έχουν γίνει πολλαπλές δοκιμές για την εξασφάλιση της ορθής λειτουργίας τους σύμφωνα με τον σχεδιασμό και τις απαιτήσεις του ΠΣ. Ο έλεγχος πραγματοποιήθηκε με μεθόδους αποσφαλμάτωσης, παράδειγμα χρησιμοποιήθηκε η συνάρτηση var\_dump() και η εντολή echo για έλεγχο όλων των μεταβλητών σε κατάλληλα σημεία του κώδικα για τη διασφάλιση της ορθής λειτουργίας του ΠΣ. Επιπλέον, για τα JavaScript αρχεία χρησιμοποιήθηκε η μέθοδος console.log() για τους ανάλογους ελέγχους

ορθής λειτουργίας του ΠΣ. Τέλος, δημιουργήθηκε αρχείο PHP που παράγει μεγάλο πλήθος εικονικών εγγραφών στο ΠΣ για να ελεγχθεί η ανταπόκρισή του σε μεγάλο φόρτο και όγκο δεδομένων. Παρακάτω ακολουθεί ο πίνακας ελέγχου ορθής λειτουργίας του συστήματος.

<b>Περιγραφή Λειτουργίας</b>	<b>Ενέργεια/Διεργασία</b>	<b>Αποτέλεσμα</b>
Εμφάνιση προϊόντων χωρίς να απαιτείται σύνδεση	Επιλογή από το μενού προϊόντα	Εμφάνιση των προϊόντων
Εγγραφή χρήστη	Συμπλήρωση και υποβολή της φόρμας εγγραφής	Προσθήκη χρήστη στο σύστημα
Αποκλεισμός εγγραφής χρήστη με υπάρχον όνομα χρήστη	Εμφάνιση μηνύματος μη διαθεσιμότητας του ονόματος χρήστη	Αδυναμία ολοκλήρωσης της εγγραφής
Επεξεργασία στοιχείων χρήστη	Επεξεργασία και υποβολή της φόρμας πληροφοριών χρήστη	Εμφάνιση των αλλαγών των πληροφοριών χρήστη
Προσθήκη πλακέτας esp8266	Συμπλήρωση και υποβολή της φόρμας προσθήκης πλακέτας esp8266	Προσθήκη πλακέτας esp8266 στο σύστημα
Επεξεργασία στοιχείων πλακέτας esp8266	Επεξεργασία και υποβολή της φόρμας πληροφοριών πλακέτας esp8266	Εμφάνιση των αλλαγών των πληροφοριών πλακέτας esp8266
Διαγραφή πλακέτας esp8266	Πάτημα του κουμπιού με κόκκινο χρώμα και προειδοποιητικό μήνυμα για τη διαγραφή της πλακέτας esp8266	Διαγραφή πλακέτας esp8266 από το σύστημα
Προσθήκη κατηγορίας προϊόντων	Συμπλήρωση και υποβολή της φόρμας κατηγορίας των προϊόντων	Προσθήκη κατηγορίας προϊόντων στο σύστημα
Επεξεργασία στοιχείων κατηγορίας προϊόντων	Επεξεργασία και υποβολή της φόρμας πληροφοριών της κατηγορίας των προϊόντων	Εμφάνιση των αλλαγών των πληροφοριών της κατηγορίας των προϊόντων
Διαγραφή κατηγορίας προϊόντων	Πάτημα του κουμπιού με κόκκινο χρώμα και προειδοποιητικό μήνυμα για τη διαγραφή της κατηγορίας προϊόντων	Διαγραφή της κατηγορίας προϊόντων από το σύστημα
Προσθήκη προϊόντος	Συμπλήρωση και υποβολή της φόρμας προϊόντος	Προσθήκη προϊόντος στο σύστημα
Επεξεργασία στοιχείων προϊόντος	Επεξεργασία και υποβολή της φόρμας πληροφοριών του προϊόντος	Εμφάνιση των αλλαγών των πληροφοριών του προϊόντος
Διαγραφή προϊόντος	Πάτημα του κουμπιού με κόκκινο χρώμα και προειδοποιητικό μήνυμα για τη διαγραφή του προϊόντος	Διαγραφή του προϊόντος από το σύστημα



Περιγραφή Λειτουργίας	Ενέργεια/Διεργασία	Αποτέλεσμα
Δημιουργία παραγγελίας	Από το μενού Παραγγελία και το υπομενού Δημιουργία, επιλογή κατηγορίας προϊόντος στην συνέχεια επιλογή προϊόντος (ποσότητα, έξτρα υλικά) και προσθήκη στο καλάθι. Τα παραπάνω επαναλαμβάνονται όσες φορές επιλέξει ο χρήστης. Η παραγγελιά ολοκληρώνετε πατώντας το κουμπί ολοκλήρωση ή υποβολή από το καλάθι	Ολοκλήρωση παραγγελίας και εμφάνιση κατάστασης της παραγγελίας
Αντιστοίχιση πλακέτας esp8266 σε δια ζώσης παραγγελία	Επιλογή από το μενού Παραγγελία και το υπομενού Παραγγελίες γίνεται υποβολής της φόρμας αντιστοίχισης	Αντιστοίχιση πλακέτας esp8266 με την παραγγελία και αλλαγή της διαθεσιμότητας της πλακέτας esp8266 σε μη διαθέσιμη
Ολοκλήρωση παραγγελίας	Επιλογή από το μενού Παραγγελία και το υπομενού Ολοκλήρωση γίνεται υποβολής της φόρμας ολοκλήρωσης παραγγελίας	Η παραγγελία ολοκληρώνεται με αλλαγή της κατάστασής της σε ολοκληρωμένη, ταυτόχρονα ενημερώνεται ο χρήστης
Ιστορικό	Επιλογή από το μενού Παραγγελία και το υπομενού Ιστορικό	Εμφάνιση ιστορικού των ολοκληρωμένων παραγγελιών
Αναζήτηση χρηστών	Επιλογή από το μενού Αναζήτηση και το υπομενού Χρηστών και πληκτρολόγηση του ονόματος χρήστη	Εμφανίζεται δυναμικά κατά την πληκτρολόγηση ο χρήστης που ταιριάζει. Παράλληλα παρακάτω εμφανίζονται όλοι οι υπόλοιποι χρήστες με σελιδοποίηση
Διαγραφή χρήστη	Επιλογή από το μενού Αναζήτηση και το υπομενού Χρηστών και πληκτρολόγηση του ονόματος χρήστη προς διαγραφή. Πάτημα του κουμπιού με κόκκινο χρώμα και προειδοποιητικό μήνυμα για τη διαγραφή του χρήστη	Διαγραφή του χρήστη από το σύστημα
Αναζήτηση παραγγελιών	Επιλογή από το μενού Αναζήτηση και το υπομενού Παραγγελιών και πληκτρολόγηση του κωδικού παραγγελίας	Εμφανίζεται δυναμικά κατά την πληκτρολόγηση η παραγγελία που ταιριάζει
Αποσύνδεση χρήστη	Πατώντας το κουμπί αποσύνδεση	Αποσύνδεση από το σύστημα και μεταφορά στην αρχική σελίδα

### 7.3 Συντήρηση βάσης δεδομένων

Η συντήρηση της βάσης δεδομένων γίνεται σύμφωνα με το παρακάτω πλάνο ενεργειών:

- Δημιουργία μηνιαίου αντίγραφου της ενεργής βάσης δεδομένων, είτε μέσω του phpMyAdmin, είτε μέσω του putty και της εντολής mysqldump. Καλό είναι να δίδεται ένα όνομα eBuzzer\_Backup\_30\_09\_2021 για την εφεδρική βάση δεδομένων που θα προκύπτει στο τέλος κάθε μήνα. Η εντολή mysqldump:

```
[ictest00592@zafora ~]$ mysqldump -u root -p ebuzzer > eBuzzer_Backup_30_09_2021
Enter password:
```

- Δημιουργία ετήσιου αντίγραφου της ενεργής βάσης δεδομένων, είτε μέσω του phpMyAdmin, είτε μέσω του putty και της εντολής mysqldump, στο τέλος κάθε οικονομικού έτους. Καλό είναι να δίδεται ένα όνομα eBuzzer\_Backup\_2021 για την εφεδρική βάση δεδομένων που θα προκύπτει. Η εντολή mysqldump:

```
[ictest00592@zafora ~]$ mysqldump -u root -p ebuzzer > eBuzzer_Backup_2021
Enter password:
```

- Διαγραφή των εγγραφών από τους πίνακες orders, purchase και history, είτε μέσω του phpMyAdmin, είτε μέσω του putty και της εντολής mysql, στο τέλος κάθε οικονομικού έτους. Οι εντολές mysql με την σειρά εκτέλεσής τους:

```
[ictest00592@zafora ~]$ mysql -u root -p -e 'truncate table `ebuzzer`.`purchase`'
Enter password:
[ictest00592@zafora ~]$ mysql -u root -p -e 'truncate table `ebuzzer`.`history`'
Enter password:
[ictest00592@zafora ~]$ mysql -u root -p -e 'SET FOREIGN_KEY_CHECKS = 0; truncate table `ebuzzer`.`orders`;SET FOREIGN_KEY_CHECKS = 1'
Enter password:
[ictest00592@zafora ~]$ █
```