

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ ΚΑΙΝΟΤΟΜΙΑ

**ΔΡΑΣΗ ΕΘΝΙΚΗΣ ΕΜΒΕΛΕΙΑΣ:
«Ερευνώ-Δημιουργώ-Καινοτομώ»**

ΒΕΛΟΣ

**Βελτιστοποίηση Μεταφορών με Αλγόριθμους Ελέγχου
Οπισθοπίεσης**

Π3.1

**Αναφορά υλοποίησης υποσυστημάτων και
δοκιμών**

Ημερομηνία	Μήνας 24, 28/07/2023
Ενότητα Εργασίας	Ε.Ε.3
Είδος	Αναφορά
Κατάσταση έγκρισης	Υποβληθέν
Έκδοση	3
Αριθμός σελίδων	74
Όνομα αρχείου	VELOS_Paradoteo_3.1_v03_final.docx
Σύνοψη	Στο Παραδοτέο "Π3.1 - Αναφορά υλοποίησης υποσυστημάτων και δοκιμών" συνοψίζονται όλες οι εργασίες για την υλοποίηση, δοκιμή και αποσφαλμάτωση των επιμέρους υποσυστημάτων της πλατφόρμας του έργου.



ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Αυτή η σελίδα αφέθηκε σκόπιμα κενή



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο
Περιφερειακής Ανάπτυξης

ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Ιστορικό εκδόσεων

Έκδοση	Ημερομηνία	Περιγραφή	Αναθεώρηση
1	18/09/2021	Πρότυπο Παραδοτέου	
2	28/04/2022	Ενδιάμεση έκδοση	
3	28/07/2023	Τελικό Παραδοτέο και συνολική επιμέλεια κειμένου	

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Λίστα συγγραφέων

Οργανισμός	Όνοματεπώνυμο	Στοιχεία επικοινωνίας
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Φοίβος Μυλωνάς	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87741
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Βασίλειος Καρυώτης	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Παναγιώτης Κουρουθανάσης	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Περιφερειακής Ανάπτυξης, Ιόνιο Πανεπιστήμιο	Στέργιος Παλαμάς	Τζεβελέκη & Φιλοσόφων, Τ.Κ. 31100 Λευκάδα, Ελλάδα Τηλ.: 26450-26160
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Κλειώ Στάμου	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Γεώργιος Βούτος	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Ιωάννης Καθενιώτης	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Ειρήνη Μαθέ	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Τμήμα Πληροφορικής, Ιόνιο Πανεπιστήμιο	Αθανασία Τσακίρη	Πλατεία Τσιριγώτη 7, Τ.Κ. 49132 Κέρκυρα, Ελλάδα Τηλ.: 26610 87757
Ινστιτούτο Ανάπτυξης Επιχειρηματικότητας	Γεώργιος Γκόγκας	Βαλτετσίου και Τριπόλεως, Τ.Κ. 41336

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

		Λάρισα, Ελλάδα Τηλ.:2410 626943
Ινστιτούτο Ανάπτυξης Επιχειρηματικότητας	Κατερίνα Παρίζα	Βαλτετσίου και Τριπόλεως, Τ.Κ. 41336 Λάρισα, Ελλάδα Τηλ.:2410 626943
RMR I.K.E.	Βασιλική Σουμπενιώτη	Ηρ. Πολυτεχνείου 95, Τ.Κ. 41222 Λάρισα, Ελλάδα Τηλ.:2410 669040
RMR I.K.E.	Απόστολος Παναγόπουλος	Ηρ. Πολυτεχνείου 95, Τ.Κ. 41222 Λάρισα, Ελλάδα Τηλ.:2410 669040
RMR I.K.E.	Άγγελος Παπαγεωργίου	Ηρ. Πολυτεχνείου 95, Τ.Κ. 41222 Λάρισα, Ελλάδα Τηλ.:2410 669040
RMR I.K.E.	Κωνσταντίνος Γρηγορίου	ΤΗρ. Πολυτεχνείου 95, Τ.Κ. 41222 Λάρισα, Ελλάδα Τηλ.:2410 669040

Περίληψη

Το έργο ΒΕΛΟΣ εστιάζει στις δραστηριότητες των μικρο-μεσαίων μεταφορικών εταιρειών, και ειδικότερα στις ανάγκες της RMR, η οποία είναι μια μικρο-μεσαία επιχείρηση (Small-Medium Enterprise, SME) που δραστηριοποιείται στις μεταφορές αγαθών εξειδικευμένης διαχείρισης. Μεταξύ άλλων, ένα βασικό πρόβλημα της RMR, και αντίστοιχων μικρο-μεσαίων εταιρειών που δραστηριοποιούνται στον κλάδο των μεταφορών, είναι αυτό της βέλτιστης ανάθεσης πόρων (φορτωτικών, οχημάτων, χρονοδιαγραμμάτων, κλπ.) και χρονοπρογραμματισμού των αναγκαίων μεταφορών και/ή συνοδευτικών ενεργειών (π.χ. πακετάρισμα-packaging) σε ημερήσια/εβδομαδιαία/μηνιαία/ετήσια βάση μεταξύ των σημείων ενδιαφέροντος, ώστε να εξασφαλιστεί ότι το ανεκτέλεστο κάθε μέρας (εκκρεμείς παραδόσεις σε αναμονή-αποθήκη) και ο μέσος χρόνος παράδοσης παραμένουν οριοθετημένα (πιθανά και ελεγχόμενα), ενώ ελαχιστοποιούνται άλλα κριτήρια βέλτιστης λειτουργίας, π.χ. κόστος μεταφοράς, αποθήκευσης, συντήρησης στόλου, χρόνος παράδοσης, κλπ.

Τα παραπάνω ζητήματα είναι πολύ σημαντικά για την RMR, όπως και για κάθε αντίστοιχη μικρο-μεσαία επιχείρηση του κλάδου, καθώς μπορεί να αποβούν κρίσιμα για τη λειτουργία της και τις απαραίτητες οικονομίες κλίμακας που πρέπει να γίνουν σε μια παγκοσμιοποιημένη πλέον αγορά. Αυτή η ανάγκη, η οποία σε άλλες χώρες, π.χ. ΗΠΑ, Βρετανία, κλπ., έχει προκύψει εδώ και δεκαετίες, έχει πλέον παγιωθεί και στην Ελλάδα, τόσο λόγω της ανόδου τα τελευταία χρόνια του ηλεκτρονικού λιανεμπορίου, όσο και της αύξησης της ζήτησης που προέκυψε από την πανδημία COVID-19.

Αντικείμενο του παρόντος Παραδοτέου Π3.1 αποτελεί να τεκμηριώσει την υλοποίηση των υποσυστημάτων του πληροφοριακού συστήματος που αναπτύχθηκε στο πλαίσιο του έργου.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Συνομογραφίες και ακρωνύμια

Συνομογραφία	Επεξήγηση
ΠΣ	Πληροφοριακό Σύστημα
CMS	Content Management System
IS	Information System
ISS	Intelligent Support System
UTAUT	Unified Theory of Acceptance and Use of Technology
IQ	Information Quality
TRA	Theory of Justified Action
TPB	Theory of Planned Behaviour
TAM	Technology Acceptance Model
MM	Motivational Model
MPCU	Model of PC Utilization
IDT	Innovation Diffusion Theory
SCT	Social Cognitive Theory
PC	Personal Computer
PE	Performance expectancy
EE	Effort Expectancy
SI	Social Influence
FC	Facilitating Conditions
PEOU	Perceived Ease of Use
PA	Path Analysis
AI	Artificial Intelligence
ES	Expert Systems
KMS	Knowledge Management System
DMS	Data Management Systems

Περιεχόμενα

Εισαγωγή.....	14
Ενότητα 2 - Υποσύστημα χρονοπρογραμματισμού και δρομολόγησης	16
2.1 Εισαγωγή και επεξεργασία δεδομένων	16
2.2 Δημιουργία πίνακα γειτνίασης και εξακρίβωση της απόστασης των κόμβων .	18
2.3 Διαχωρισμός πακέτων βάση προτεραιότητας και αρχικοποίηση ουρών	20
2.4 Εκτέλεση διαδικασίας Back pressure	20
2.5 Εκτέλεση διαδικασίας διαλογής πακέτων.....	22
2.5 Εκτέλεση διαδικασίας βελτιστοποίησης διαθέσιμου χώρου για την μεταφορά των πακέτων.....	26
3. Υποσύστημα διαδικτυακής πλατφόρμας	28
3.1 UseCase 1: Σύνδεση και αποσύνδεση από το σύστημα.....	28
3.2 UseCase 2: Εισαγωγή νέας παραγγελίας και πακέτων	32
3.3 UseCase 3: Εμφάνιση όλων των παραγγελιών και λεπτομέρειες για κάθε μία από αυτές.....	39
3.4 UseCase 4: Εμφάνιση όλων των δεδομένων της βάσης δεδομένων	42
3.6 UseCase 6: Εμφάνιση δικτύου διανομών και λεπτομέρειες για τους κόμβους της εταιρείας.....	46
3.7 UseCase 7: Εμφάνιση της τελευταίας επανάληψης του αλγορίθμου.....	53
3.8 UseCase 8: Εμφάνιση όλων των προηγούμενων εκτελέσεων του αλγορίθμου	56
4. Υποσύστημα κινητής εφαρμογής	58
4.1 Μεθοδολογία ανάπτυξης.....	58
Ανάλυση Απαιτήσεων.....	58
Σχεδιασμός Εφαρμογής.....	59
Ανάπτυξη (Development).....	60
Επαναληπτικές Διορθώσεις.....	60
Έκδοση και Κυκλοφορία.....	61
4.2 Εργαλεία ανάπτυξης.....	62

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Adobe XD	62
React Native	63
Expo.....	64
4.3 Λειτουργικότητα κινητής εφαρμογής.....	65
4. Σύνοψη - Συμπεράσματα.....	73
Αναφορές	74

Κατάλογος Εικόνων

Σχήμα 1: Εισαγωγή και τροποποίηση δεδομένων	17
Σχήμα 2: Δημιουργία πίνακα γειτνίασης	18
Σχήμα 3: Υπολογισμός Hop Count	19
Σχήμα 4: Ανάθεση Hop Count στα πακέτα	20
Σχήμα 5: Διαχωρισμός πακέτων σε λίστες βάσει προτεραιότητας	20
Σχήμα 6: Εκτέλεση διαδικασίας Back Pressure	21
Σχήμα 7: Ταξινομημένη λίστα DeltaQ	22
Σχήμα 8: Επιλογή γείτονα κόμβου για το επιλεγμένο μονοπάτι	23
Σχήμα 9: Επιλογή γείτονα κόμβου για αριθμό Hops>1	25
Σχήμα 10: Κλήση συνάρτησης Knapsack.....	26
Σχήμα 11: Η συνάρτηση Knapsack για το πρόβλημα της βέλτιστης κατανομής του χώρου	27
Σχήμα 12: Η οθόνη σύνδεσης του συστήματος	29
Σχήμα 13: Αρνητικό μήνυμα σε περίπτωση λάθους.....	30
Σχήμα 14: Κεντρικό μενού συστήματος	31
Σχήμα 15: Κουμπί αποσύνδεσης	31
Σχήμα 16: Επιλογή "Εισαγωγή νέας παραγγελίας" στο κεντρικό μενού .	32
Σχήμα 17: Φόρμα εισαγωγής νέας παραγγελίας	33
Σχήμα 18: Φόρμα εισαγωγής νέας παραγγελίας με συμπληρωμένα τα στοιχεία.....	34
Σχήμα 19: Μήνυμα επιβεβαίωσης εισαγωγής παραγγελίας	34
Σχήμα 20: Μήνυμα επιτυχημένης καταχώρησης παραγγελίας.....	35
Σχήμα 21: Οθόνη εισαγωγής πακέτων σε μία παραγγελία	35
Σχήμα 22: Οθόνη εισαγωγής πακέτων με συμπληρωμένη την φόρμα ...	36
Σχήμα 23: Μήνυμα επιβεβαίωσης εισαγωγής πακέτων στο σύστημα	37
Σχήμα 24: Μήνυμα επιβεβαίωσης εισαγωγής πακέτου	38
Σχήμα 25: Επιλογή "Τα δέματα μου" στο πλαϊνό μενού	39
Σχήμα 26: Πίνακας με όλα δέματα που έχει μία εταιρεία	40

Σχήμα 27: Επιλογή δέματος	40
Σχήμα 28: Λεπτομέρειες για την τοποθεσία και διαδρομή του δέματος..	41
Σχήμα 29: Επιλογή "Όλα τα δεδομένα" στο πλαϊνό μενού	42
Σχήμα 30: Εμφάνιση των πινάκων του συστήματος με τα δεδομένα της βάσης	43
Σχήμα 31: Εμφάνιση του πίνακα "Κεντρα" από την βάση δεδομένων ...	43
Σχήμα 32: Επιλογή "Στατιστικά" για την εμφάνιση στατιστικών στοιχείων	44
Σχήμα 33: Συγκεντρωτικά στατιστικά στοιχεία	45
Σχήμα 34: Επιλογή "Δίκτυο Διανομής" στο πλαϊνό μενού.....	46
Σχήμα 35: Κεντρική σελίδα δίκτυου διανομής	47
Σχήμα 36: Λεπτομέρειες για επιλεγμένο κόμβο	48
Σχήμα 37: Λεπτομέρειες για την κινητικότητα στον κόμβο	48
Σχήμα 38: Συνδέσεις στον κόμβο.....	49
Σχήμα 39: Τρέχουσες μετακινήσεις στο επιλεγμένο κέντρο διανομών ...	50
Σχήμα 40: Διαθέσιμα δέματα στο κέντρο διανομών που έχει επιλεχθεί..	51
Σχήμα 41: Παραγγελίες προς παράδοση στον επιλεγμένο κόμβο.....	51
Σχήμα 42: Επιλογή "Τελευταία επανάληψη του αλγορίθμου".....	53
Σχήμα 43: Οι αποδεκτές και μη αποδεκτές προτάσεις του αλγορίθμου ..	54
Σχήμα 44: Λεπτομέρειες μεταφοράς	55
Σχήμα 45: Επιβεβαίωση ή απόρριψη πρότασης αλγορίθμου	55
Σχήμα 46: Ημερολόγιο εκτέλεσης αλγορίθμου	56
Σχήμα 47: Μετακινήσεις που έγιναν στην προηγούμενη εκτέλεση του αλγορίθμου.....	57
Σχήμα 48: Οθόνη σύνδεσης χρήστη	66
Σχήμα 49: Πρόγραμμα μεταφοράς οδηγού φορτηγού	67
Σχήμα 50: Λεπτομέρειες πακέτου μεταφοράς.....	68
Σχήμα 51: Μενού χρήστη	69
Σχήμα 52: Λίστα αιτημάτων για παράδοση	70
Σχήμα 53: Λεπτομέρειες αιτήματος προς παράδοση - Χαρακτηριστικά πακέτου	71
Σχήμα 54: Μήνυμα έλλειψης δρομολογίων	72



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο
Περιφερειακής Ανάπτυξης

ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Κατάλογος Πινάκων

Εισαγωγή

Το έργο ΒΕΛΟΣ εστιάζει στις δραστηριότητες των μικρο-μεσαίων μεταφορικών εταιρειών, και ειδικότερα στις ανάγκες της RMR, η οποία είναι μια μικρο-μεσαία επιχείρηση (Small-Medium Enterprise, SME) που δραστηριοποιείται στις μεταφορές αγαθών εξειδικευμένης διαχείρισης. Μεταξύ άλλων, ένα βασικό πρόβλημα της RMR, και αντίστοιχων μικρο-μεσαίων εταιρειών που δραστηριοποιούνται στον κλάδο των μεταφορών, είναι αυτό της βέλτιστης ανάθεσης πόρων (φορτωτικών, οχημάτων, χρονοδιαγραμμάτων, κλπ.) και χρονοπρογραμματισμού των αναγκαίων μεταφορών και/ή συνοδευτικών ενεργειών (π.χ. πακετάρισμα-packaging) σε ημερήσια/εβδομαδιαία/μηνιαία/ετήσια βάση μεταξύ των σημείων ενδιαφέροντος, ώστε να εξασφαλιστεί ότι το ανεκτέλεστο κάθε μέρας (εκκρεμείς παραδόσεις σε αναμονή-αποθήκη) και ο μέσος χρόνος παράδοσης παραμένουν οριοθετημένα (πιθανά και ελεγχόμενα), ενώ ελαχιστοποιούνται άλλα κριτήρια βέλτιστης λειτουργίας, π.χ. κόστος μεταφοράς, αποθήκευσης, συντήρησης στόλου, χρόνος παράδοσης, κλπ.

Τα παραπάνω ζητήματα είναι πολύ σημαντικά για την RMR, όπως και για κάθε αντίστοιχη μικρο-μεσαία επιχείρηση του κλάδου, καθώς μπορεί να αποβούν κρίσιμα για τη λειτουργία της και τις απαραίτητες οικονομίες κλίμακας που πρέπει να γίνουν σε μια παγκοσμιοποιημένη πλέον αγορά. Αυτή η ανάγκη, η οποία σε άλλες χώρες, π.χ. ΗΠΑ, Βρετανία, κλπ., έχει προκύψει εδώ και δεκαετίες, έχει πλέον παγιωθεί και στην Ελλάδα, τόσο λόγω της ανόδου τα τελευταία χρόνια του ηλεκτρονικού λιανεμπορίου, όσο και της αύξησης της ζήτησης που προέκυψε από την πανδημία COVID-19.

Το παρόν παραδοτέο εστιάζει στην τεκμηρίωση της ανάπτυξης των επιμέρους τμημάτων του πληροφοριακού συστήματος, ήτοι της διαδικτυακής πλατφόρμας,

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

του τρόπου υλοποίησης του αλγορίθμου χρονοπρογραμματισμού και δρομολόγησης, καθώς και του υποσυστήματος κινητής εφαρμογής. Ειδικά αναφορικά με την τεκμηρίωση του αλγορίθμου, η ανασκόπηση του κώδικα θα γίνει μέσω επεξήγησης διαφόρων σημαντικών σημείων του ενώ παράλληλα θα αναφέρονται οι λόγοι που χρησιμοποιήθηκαν οι συγκεκριμένες τεχνικές. Η τεκμηρίωση των υποσυστημάτων του πληροφοριακού συστήματος θα πραγματοποιηθεί διαδραστικά, δηλαδή θα επεξηγηθεί η χρήση του διαδικτυακού κομματιού του συστήματος με περιπτώσεις χρήσης, ούτως ώστε να γίνουν κατανοητές οι διάφορες λειτουργίες του.

Ενότητα 2 - Υποσύστημα χρονοπρογραμματισμού και δρομολόγησης

Η ενότητα παρουσιάζει τον τρόπο υλοποίησης του αλγορίθμου χρονοπρογραμματισμού και δρομολόγησης. Η αναλυτική δομή του αλγορίθμου αποτυπώνεται στα παραδοτέα 1.2 και 2.1. Το υποσύστημα που υλοποιεί τον αλγόριθμο δημιουργήθηκε αρχικά στην γλώσσα matlab για ερευνητικούς σκοπούς. Αργότερα για την προσαρμογή του αλγορίθμου στο σύστημα χρησιμοποιήθηκε η γλώσσα python και βιβλιοθήκες όπως η math, random κτλ.

2.1 Εισαγωγή και επεξεργασία δεδομένων

Αρχικά ο αλγόριθμος συλλέγει τα απαραίτητα δεδομένα που χρειάζεται από την βάση δεδομένων. Ωστόσο, αυτά τα δεδομένα χρειάζονται επεξεργασία για να βρεθούν στην μορφή που τα χρειάζεται ο αλγόριθμος. Για παράδειγμα, το βάρος ενός αντικειμένου είναι πάντα ένας δεκαδικός αριθμός, αλλά ο αλγόριθμος χρειάζεται το βάρος να είναι ένας ακέραιος αριθμός. Έτσι πρέπει να πολλαπλασιαστεί το βάρος με το 100 για να γίνει ακέραιος αριθμός χωρίς να χάνεται σημαντική πληροφορία κατά την μετατροπή. Ακολουθεί ο κώδικας:

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

```
query1 = "SELECT DISTINCT item.item_id, item.request_id, item.Current_hub_ID,
item.next_hub_ID ,item.Delivery_hub_ID, item.Volumetric_Weight,
request.Priority, item.order_hub_id \
FROM item \
INNER JOIN request ON item.request_id = request.request_id \
WHERE item.Current_hub_id != item.Delivery_hub_ID \
OR request.Status_Request = 'In_transit'"

cursor.execute(query1)
item_rows = cursor.fetchall() ##fetch all the required data from the
database to start the data forming

items = []
for row in item_rows:
    items.append(row)

item_list = []
for item in items:
    if item[3]:
        item_dict = {
            'item_id': item[0],
            'request_id': item[1],
            'starting_location_id': item[3],
            'next_location_id': None,
            'ending_location_id': item[4],
            'weight': int(item[5]*100),
            'hop': None,
            'priority': item[6],
            'item_status': 'In_transit',
            'order_hub_id': item[7]
        }
        item_list.append(item_dict)
    else:
        item_dict = {
            'item_id': item[0],
            'request_id': item[1],
            'starting_location_id': item[2],
            'next_location_id': None,
            'ending_location_id': item[4],
            'weight': int(item[5]*100),
            'hop': None,
            'priority': item[6],
            'item_status': 'In_transit',
            'order_hub_id': item[7]
        }
        item_list.append(item_dict)

with open('items.json', 'w') as f:
    json.dump(item_list,f)
```

Σχήμα 1: Εισαγωγή και τροποποίηση δεδομένων

Με παρόμοιο τρόπο γίνονται όλες οι τροποποιήσεις για όλα τα απαραίτητα δεδομένα

2.2 Δημιουργία πίνακα γειτνίασης και εξακρίβωση της απόστασης των κόμβων

Έχοντας συλλέξει όλα τα απαραίτητα δεδομένα χρειάζεται να δημιουργηθεί ένα πίνακας γειτνίασης μέσω του οποίου ο αλγόριθμος μπορεί να καταλάβει τις συνδέσεις που υπάρχουν μεταξύ όλων των κόμβων του δικτύου. Στον πίνακα γειτνίασης, όταν υπάρχει σύνδεση μεταξύ δύο κόμβων, αναγράφεται και η χωρητικότητα της σύνδεσης όπως φαίνεται παρακάτω.

```
##### MAKE THE ADJACENCY MATRIX #####
with open('Connections.json', 'r') as f:
    Connections = json.load(f)

# Find number of nodes
nodes = set()
for d in Connections:
    nodes.add(d['start_id'])
    nodes.add(d['connection_id'])
num_nodes = len(nodes)                                ##Creation of an adjacency
matrix to map out the nodes

# Initialize adjacency matrix
adj_matrix = [[0 for j in range(num_nodes)] for i in range(num_nodes)]

# Fill adjacency matrix
for d in Connections:
    i = d['start_id'] - 1
    j = d['connection_id'] - 1
    adj_matrix[i][j] = int(d['max_connection_capacity'])
```

Σχήμα 2: Δημιουργία πίνακα γειτνίασης

Έχοντας τον πίνακα γειτνίασης ο αλγόριθμος μπορεί πλέον να βρει την απόσταση μεταξύ κάθε κόμβου με κάθε άλλο ανεξάρτητα από την ύπαρξη σύνδεσης μεταξύ των δύο. Αυτό συμβαίνει διότι πολύ συχνά στο δίκτυο διανομών μία παραγγελία πρέπει να μεταφερθεί από έναν κόμβο σε κάποιον που δεν συνδέεται άμεσα. Για την χαρτογράφηση αυτής της απόστασης ο αλγόριθμος εκτελεί τον αλγόριθμο εύρεση κοντινότερου μονοπατιού Dijkstra και σημειώνει την απόσταση σε hops. Τα hops είναι η απόσταση σε "στάσεις" που πρέπει να γίνουν για να συνδεθούν δύο κόμβοι. Δηλαδή, αν δύο κόμβοι είναι γείτονες και συνδέονται απευθείας μεταξύ τους τότε έχουν απόσταση ενός hop, από την άλλη αν έχουν έναν ενδιάμεσο κόμβο

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

ανάμεσα τους τότε η απόσταση τους είναι δύο hops. Χρησιμοποιείται αυτή η μονάδα μέτρησης στον αλγόριθμο καθώς είναι πολύ σημαντικό να ξέρει ο αλγόριθμος το αμέσως επόμενο βήμα ενός κόμβου για κάθε πιθανό τελικό προορισμό

```
def dijkstra(adj_matrix, start):
    n = len(adj_matrix)
    dist = [float('inf')] * n
    dist[start] = 0
    visited = set()
    heap = [(0, start)]
    prev = {} # dictionary to store previous nodes on the shortest path

    while heap:
        (d, u) = heapq.heappop(heap)
        if u in visited:
            continue #Execute Dijkstra to find
        the shortest path and save it as a hop count
        visited.add(u)
        for v in range(n):
            if adj_matrix[u][v] != 0:
                if dist[u] + 1 < dist[v]:
                    dist[v] = dist[u] + 1
                    prev[v] = u # update prev dictionary
                    heapq.heappush(heap, (dist[v], v))

    shortest_paths = {}
    for v in range(n):
        if v != start:
            path = []
            u = v
            while u != start:
                path.append(u)
                u = prev[u]
            path.append(start)
            path.reverse()
            path = [node+1 for node in path] # add 1 to each node in the
            path
            shortest_paths[v] = {"start": start+1, "end": v+1, "length":
            dist[v], "path": path}

    return shortest_paths
```

Σχήμα 3: Υπολογισμός Hop Count

Έχοντας βρει όλα τα "Hops counts" για όλους τους πιθανούς συνδυασμούς κόμβων τότε αντιστοιχίζονται στα διαθέσιμα πακέτα τα αντίστοιχα "Hops counts" βάση της τρέχουσας θέσης τους και τελικό προορισμό τους. Αυτό συμβαίνει διότι αργότερα θα γίνει διαλογή για των πακέτων βάση αυτού του αριθμού, ούτως ώστε να εξυπηρετηθούν πρώτα τα πακέτα με το μικρότερο "Hop count". Ο κώδικας που υλοποιεί την παραπάνω διαδικασία αποτυπώνεται στο παρακάτω σχήμα.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

```
package_list = []
count = 0
for i, paths in enumerate(all_shortest_paths):

    for path in paths.values():

        for package in packages:
            if package['starting_location_id'] == path['start'] and
                package['ending_location_id'] == path['end']:
                package['hop'] = path['length']

for row in packages:
    package_list.append(row)
```

Σχήμα 4: Ανάθεση Hop Count στα πακέτα

2.3 Διαχωρισμός πακέτων βάση προτεραιότητας και αρχικοποίηση ουρών

Ο αλγόριθμος προδιαγράφει διαχωρισμό των πακέτων/ μεταφορών με βάση την προτεραιότητα που τους έχει ανατεθεί από την εταιρεία logistics. Επομένως όπως φαίνεται παρακάτω δημιουργούνται δύο λίστες, μία με προτεραιότητα και μία χωρίς και εισάγονται τα διαθέσιμα δέματα αντίστοιχα.

```
#####PRIORITY PACKAGE SELECTION#####
package_list_priority = []
package_list_normal = []

for package in package_list:
    if package['priority'] == 'High':
        package_list_priority.append(package)
    else:
        package_list_normal.append(package)
```

Σχήμα 5: Διαχωρισμός πακέτων σε λίστες βάσει προτεραιότητας

2.4 Εκτέλεση διαδικασίας Back pressure

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Η εκτέλεση της διαδικασίας back pressure είναι η ραχοκοκαλιά του αλγορίθμου. Βάση αυτής της διαδικασίας παίρνονται οι απαραίτητες αποφάσεις για την σειρά με την οποία θα πρέπει να εκτελεστούν οι επόμενες κινήσεις του αλγορίθμου.

Στη συνέχεια φαίνεται η εκτέλεση αυτής της διαδικασίας. Αρχικά δημιουργούμε μία λίστα στην οποία θα βάλουμε τα αποτελέσματα των παραπάνω πράξεων με όνομα DeltaQ_List. Σε αυτήν θα βάλουμε όλα τα DeltaQ[i] που θα δημιουργηθούν από τις παραπάνω πράξεις.

```
for path in all_paths_list:
    if nodei['hub_id'] == path['start'] and path['hops'] == 1 and nodej['hub_id'] == path['end'] :
        if nodei['available_capacity'] > 0:
            numerator = (weighti / nodei['max_capacity']) + ((2 - nodei['available_capacity'] /
nodei['max_capacity']) * ((weighti / nodei['available_capacity']) ** mexp))
        else:
            nodei['available_capacity'] = 1
            numerator = (weighti / nodei['max_capacity']) + ((2 - nodei['available_capacity'] /
nodei['max_capacity']) * ((weighti / nodei['available_capacity']) ** mexp))
        if nodei['available_capacity'] > 0:
            denominator = 1 + (weighti / nodei['available_capacity']) ** (mexp - 1)
        else:
            nodei['available_capacity'] = 1
            denominator = 1 + (weighti / nodei['available_capacity']) ** (mexp - 1)
        pressure = round(min(1, numerator / denominator),4)
        result = max(0, pressure)

        deltaQ = {'start':nodei['hub_id'], 'end':nodej['hub_id'], 'hop_count': path['hops'],
'DeltaQ': result}
        DeltaQ_list.append(deltaQ)

    elif nodei['hub_id'] == path['start'] and path['hops'] > 1 and nodej['hub_id'] == path['end']
:
        if nodei['available_capacity'] > 0:
            numeratori = (weighti / nodei['max_capacity']) + ((2 - nodei['available_capacity'] /
nodei['max_capacity']) * ((weighti / nodei['available_capacity']) ** mexp))
            denominatori = 1 + (weighti / nodei['available_capacity']) ** (mexp - 1)
        else:
            nodei['available_capacity'] = 1
            numeratori = (weighti / nodei['max_capacity']) + ((2 - nodei['available_capacity'] /
nodei['max_capacity']) * ((weighti / nodei['available_capacity']) ** mexp))
            denominatori = 1 + (weighti / nodei['available_capacity']) ** (mexp - 1)
        pressurei = round(min(1, numeratori / denominatori),4)
        if nodej['available_capacity'] > 0:
            numeratorj = (weightj / nodej['max_capacity']) + ((2 - nodej['available_capacity'] /
nodej['max_capacity']) * ((weightj / nodej['available_capacity']) ** mexp))
            denominatorj = 1 + (weightj / nodej['available_capacity']) ** (mexp - 1)
        else:
            nodej['available_capacity'] = 1
            numeratorj = (weightj / nodej['max_capacity']) + ((2 - nodej['available_capacity'] /
nodej['max_capacity']) * ((weightj / nodej['available_capacity']) ** mexp))
            denominatorj = 1 + (weightj / nodej['available_capacity']) ** (mexp - 1)
        pressurej = round(min(1, numeratorj / denominatorj),4)
        result = round(max(0, pressurei - pressurej), 4)

        deltaQ = {'start':nodei['hub_id'], 'end':nodej['hub_id'], 'hop_count': path['hops'],
'DeltaQ': result}
        DeltaQ_list.append(deltaQ)
```

Στο αρχικό κομμάτι κώδικα στην Εικόνα 6, για αρχή υπολογίζεται το DeltaQ για συνδυασμούς κόμβων που απέχουν 1 Hop μεταξύ τους, έπειτα υπολογίζεται για συνδυασμούς κόμβων με Hop Count>1. Όλα τα DeltaQ αποθηκεύονται στην παραπάνω λίστα. Ωστόσο, ο αλγόριθμος πρέπει να εκτελέσει τους συνδυασμούς με το μεγαλύτερο δυνατό DeltaQ. Επομένως, ταξινομείται η προηγούμενη λίστα με φθίνουσα σειρά.

```
sorted_DeltaQ_list = sorted(DeltaQ_list, key=lambda x: x['DeltaQ'], reverse=True)
```

Σχήμα 7: Ταξινομημένη λίστα DeltaQ

2.5 Εκτέλεση διαδικασίας διαλογής πακέτων

Έχοντας εκτελέσει όλα τα προηγούμενα βήματα ο αλγόριθμος έχει πλέον την πληροφορία για να μπορέσει να εκτελέσει την διαδικασία διαλογής πακέτων. Ξεκινώντας με τον συνδυασμό κόμβων με το μεγαλύτερο DeltaQ αναγνωρίζει την διαθέσιμη χωρητικότητα της σύνδεσης και βλέπει τον αριθμό Hop. Αν ο αριθμός Hop στον συνδυασμό κόμβων είναι 1 τότε μπορεί κατευθείαν να βρει την διαθέσιμη χωρητικότητα της σύνδεσης και να εξετάσει αν γίνεται να μεταφερθούν πακέτα. Αν ο αριθμός Hop είναι παραπάνω από 1 τότε πρέπει να ανατρέξει στην διαδρομή που

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

βρήκε παραπάνω και να βρει την βέλτιστη διαδρομή για τον συνδυασμό, όπου θα επαναλάβει την παραπάνω διαδικασία. Ακολουθεί ο κώδικας στα παρακάτω σχήματα.

```
if current_path['hop_count'] == 1 and current_path['end_node_available_capacity'] > 0:
    for connection in Connections:
        if current_path['end_node_available_capacity'] > connection['available_capacity']:
            if current_path['start'] == connection['start_id'] and current_path['end'] ==
            connection['connection_id']:
                selected_path = {'start':connection['start_id'], 'end':connection['connection_id'],
                'max_connection_capacity':connection['max_connection_capacity'],
                'current_capacity':connection['current_capacity'], 'available_capacity':connection['available_capacity']}
            elif current_path['end_node_available_capacity'] <= connection['available_capacity']:
                if connection['available_capacity'] > 0:
                    if current_path['start'] == connection['start_id'] and current_path['end'] ==
                    connection['connection_id']:
                        selected_path = {'start':connection['start_id'],
                        'end':connection['connection_id'], 'max_connection_capacity':connection['max_connection_capacity'],
                        'current_capacity':connection['current_capacity'],
                        'available_capacity':current_path['end_node_available_capacity']}

        for package in package_list_priority:
            if package['starting_location_id'] == selected_path['start'] and package['ending_location_id']
            == selected_path['end']:
                selected_packages.append(package)
                #print(len(selected_packages))
                if selected_packages:
                    items_to_be_delivered = knapsack(selected_packages, selected_path['available_capacity'])
                    #print('Starts from:',selected_path['start'], 'Ends
                    to:',selected_path['end'],'edge_capacity:',selected_path['available_capacity'])
                    total_weight = sum(item['weight'] for item in items_to_be_delivered)
                    #print('weight:',total_weight)
                    #print('edge capacity:',selected_path['available_capacity'])
                    for item in items_to_be_delivered:
                        if node_available_capacity - total_weight > 0:
                            for nodei in Hubs_list:
                                if nodei['hub_id'] == item['starting_location_id']:
                                    #print(nodei)
                                    nodei['current_capacity'] = round(nodei['current_capacity'] - item['weight'],
                                    2)
                                    nodei['available_capacity'] = round(nodei['available_capacity'] +
                                    item['weight'], 2)
                                    #print(nodei)
                                    #print(item)

                                for nodej in Hubs_list:
                                    if nodej['hub_id'] == item['ending_location_id']:
                                        item['next_location_id'] = nodej['hub_id']
                                        nodej['current_capacity'] = round(nodej['current_capacity'] + item['weight'],
                                        2)
                                        nodej['available_capacity'] = round(nodej['available_capacity'] -
                                        item['weight'], 2)
                                        #print(nodej)
                                        #print(item)
                                    selected_path['available_capacity'] = round(selected_path['available_capacity'] -
                                    item['weight'], 2)
                                    selected_path['current_capacity'] = round(selected_path['current_capacity'] +
                                    item['weight'], 2)

                                #print('edge_capacity:',selected_path['available_capacity'])
                                for connection in Connections:
                                    if selected_path['start'] == connection['start_id'] and selected_path['end'] ==
                                    connection['connection_id']:
                                        connection['available_capacity'] = selected_path['available_capacity']
                                        connection['current_capacity'] = selected_path['current_capacity']
                                        #print(connection)
                                        #print('Starts from:',connection['start_id'],'Ends
                                        to:',connection['connection_id'],'edge_capacity:',connection['available_capacity'])
                                        final_items_list.append(item)
                                        #NA BAAQ OAA TA ITEMS SE EXHOPISTH ΛΙΣΤΑ ΠΙΝ ΑΔΕΙΑΖΟ ΑΥΘΗΝ
                                        #print('Next iteration', '\n')

                                selected_packages = [] # reset the list for the next iteration
```

Σχήμα 8: Επιλογή γείτονα κόμβου για το επιλεγμένο μονοπάτι



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο
Περιφερειακής Ανάπτυξης

ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

```
elif current_path['hop_count'] > 1 and current_path['end_node_available_capacity'] > 0:

    for package in package_list_priority:
        if package['starting_location_id'] == current_path['start'] and package['ending_location_id'] == current_path['end']:
            package['intermediate_location_id'] = None
            selected_2_hop_packages.append(package)
            #print(package)

    #print(current_path)
    for path in all_paths_list:
        if current_path['start'] == path['start'] and current_path['end'] == path['end']:
            next_node = path['path'][path['path'].index(path['start']) + 1]
            #print('start_node:', path['start'], 'end_node:', path['end'], 'next_node:', next_node)
            actual_path = {'start': path['start'], 'actual_end': next_node, 'end': path['end'],
                           'max_connection_capacity': None, 'current_capacity': None, 'available_capacity': None,
                           'end_node_available_capacity': None}

            for package in selected_2_hop_packages:
                if package['starting_location_id'] == actual_path['start'] and package['ending_location_id'] == actual_path['end']:
                    package['intermediate_location_id'] = actual_path['actual_end']
                    #print(package)
                    selected_packages.append(package)

            for nodei in Hubs_list:
                for nodej in Hubs_list:
                    if nodei['hub_id'] == actual_path['start'] and nodej['hub_id'] == actual_path['actual_end']:
                        actual_path['end_node_available_capacity'] = nodej['available_capacity']

            for connection in Connections:
                if connection['available_capacity'] > 0:
                    if actual_path['end_node_available_capacity'] > connection['available_capacity']:
                        if actual_path['start'] == connection['start_id'] and actual_path['actual_end'] == connection['connection_id']:
                            actual_path['max_connection_capacity'] = connection['max_connection_capacity']
                            actual_path['current_capacity'] = connection['current_capacity']
                            actual_path['available_capacity'] = connection['available_capacity']

                    elif actual_path['end_node_available_capacity'] <= connection['available_capacity']:
                        if actual_path['end_node_available_capacity'] > 0:
                            if actual_path['start'] == connection['start_id'] and actual_path['actual_end'] == connection['connection_id']:
                                actual_path['max_connection_capacity'] = connection['max_connection_capacity']
                                actual_path['current_capacity'] = connection['current_capacity']
                                actual_path['available_capacity'] = actual_path['end_node_available_capacity']
```

Σχήμα 9: Επιλογή γείτονα κόμβου για αριθμό Hops>1

Όπως φαίνεται παραπάνω, μία σημαντική υπολειτουργία του αλγορίθμου είναι να βρει το πραγματικό μονοπάτι, `actual_path`, ανεξάρτητα από τον αριθμό hop που έχουν οι συνδυασμοί κόμβων. Έτσι μπορεί να βρει τη διαθέσιμη χωρητικότητα στο

πραγματικό μονοπάτι και τελικά να επιλέξει τα πακέτα όπως φαίνεται στην επόμενη ενότητα.

2.5 Εκτέλεση διαδικασίας βελτιστοποίησης διαθέσιμου χώρου για την μεταφορά των πακέτων

Στο επόμενο κομμάτι κώδικα φαίνεται η κλήση μίας συνάρτησης που διαχειρίζεται τον χώρο κάθε σύνδεσης. Δηλαδή την διαθέσιμη χωρητικότητα ανά πάσα στιγμή για οποιαδήποτε ένωση δύο κόμβων.

```
items_to_be_delivered = knapsack(selected_packages, actual_path['available_capacity'])
```

Σχήμα 10: Κλήση συνάρτησης Knapsack

Παραπάνω φαίνεται ο τρόπος κλήσης της συνάρτησης Knapsack η οποία δέχεται το πραγματικό μονοπάτι και μία λίστα με πακέτα που μπορούν να μεταφερθούν σε αυτό το μονοπάτι. Γενικά, τα προβλήματα Knapsack είναι προβλήματα δυναμικού προγραμματισμού και ασχολούνται με πληθώρα προβλημάτων, ένα από αυτά είναι η βέλτιστη διανομή του χώρου για την μεταφορά των περισσότερων δυνατών πακέτων. Συγκεκριμένα, η συνάρτηση φτιάχνει πίνακες με όλους τους δυνατούς συνδυασμούς πακέτων που μπορεί να υπάρξουν και επιλέγει αυτόν που ταιριάζει καλύτερα στον εναπομείναντα χώρο. Ακολουθεί ο κώδικας:

```
def knapsack(items, capacity):
    act_capacity = math.ceil(capacity)
    n = len(items)
    table = [[0 for _ in range(act_capacity+1)] for _ in range(n+1)]
    selected_items = [[[[] for _ in range(act_capacity+1)] for _ in range(n+1)]

    for i in range(1, n+1):
        weight = int(items[i-1]['weight'])
        for j in range(1, act_capacity+1):
            if weight <= j:
                if weight + table[i-1][j-weight] > table[i-1][j]:
                    table[i][j] = weight + table[i-1][j-weight]
                    selected_items[i][j] = selected_items[i-1][j-weight] + [items[i-1]]
                else:
                    table[i][j] = table[i-1][j]
                    selected_items[i][j] = selected_items[i-1][j]
            else:
                table[i][j] = table[i-1][j]
                selected_items[i][j] = selected_items[i-1][j]

    max_weight = 0
    max_items = []
    for items in selected_items[n]:
        if sum(item['weight'] for item in items) > max_weight:
            max_weight = sum(item['weight'] for item in items)
            max_items = items

    return max_items
```

Σχήμα 11: Η συνάρτηση Knapsack για το πρόβλημα της βέλτιστης κατανομής του χώρου

Με την υλοποίηση της παραπάνω συνάρτησης πλέον ο αλγόριθμος έχει επιλέξει τις μεταφορές που πρέπει να γίνουν και τα πακέτα που πρέπει να μεταφερθούν. Να σημειωθεί ότι όλη η διαδικασία επαναλαμβάνεται για την ουρά πακέτων που δεν έχουν προτεραιότητα με όση διαθέσιμη χωρητικότητα έχει μείνει στις συνδέσεις.

Έπειτα, ο αλγόριθμος μορφοποιεί τα δεδομένα που έφτιαξε στους τύπους και τις τεχνολογίες της βάσης δεδομένων και αποθηκεύει τις αλλαγές. Έτσι, ολοκληρώνει την λειτουργία του.

3. Υποσύστημα διαδικτυακής πλατφόρμας

Σε αυτό το κεφάλαιο θα εξηγηθούν με use cases όλες οι λειτουργίες της διαδικτυακής πλατφόρμας που αποτελεί το βασικό λειτουργικό τμήμα του πληροφοριακού συστήματος. Τα use cases είναι παρόμοια με αυτά του προηγούμενου παραδοτέου για την υλοποίηση του συστήματος και του API. Σχετικά με τον τρόπο υλοποίησης της διαδικτυακής πλατφόρμας επισημαίνονται τα εξής:

- Το υποσύστημα βάσης δεδομένων δημιουργήθηκε στο DBMS MariaDB. Χρησιμοποιήθηκε η γλώσσα SQL για την δημιουργία των πινάκων και των κατάλληλων εργαλείων για την σωστή λειτουργία της βάσης.
- Το υποσύστημα API δημιουργήθηκε με το framework της JavaScript Nodejs και την βιβλιοθήκη Express. Έτσι ήταν δυνατόν να επεκταθεί γρήγορα το API και να κρατηθεί το πρότυπο του RestAPI.
- Το υποσύστημα του Web App δημιουργήθηκε με το framework της JavaScript, ReactJS. Με την βοήθεια της ReactJS και μίας βιβλιοθήκης της CSS, την Tailwind ήταν δυνατή η γρήγορα προσαρμογή του συστήματος στο συμφωνημένο design.

3.1 UseCase 1: Σύνδεση και αποσύνδεση από το σύστημα

Για να μπορέσει κανείς να χρησιμοποιήσει το σύστημα πρέπει να έχει τα κατάλληλα πειστήρια. Έχοντας αυτά και πηγαίνοντας στην ιστοσελίδα <http://app.velos.di.ionio.gr/> θα έχει την επιλογή να συνδεθεί στο σύστημα όπως φαίνεται παρακάτω:

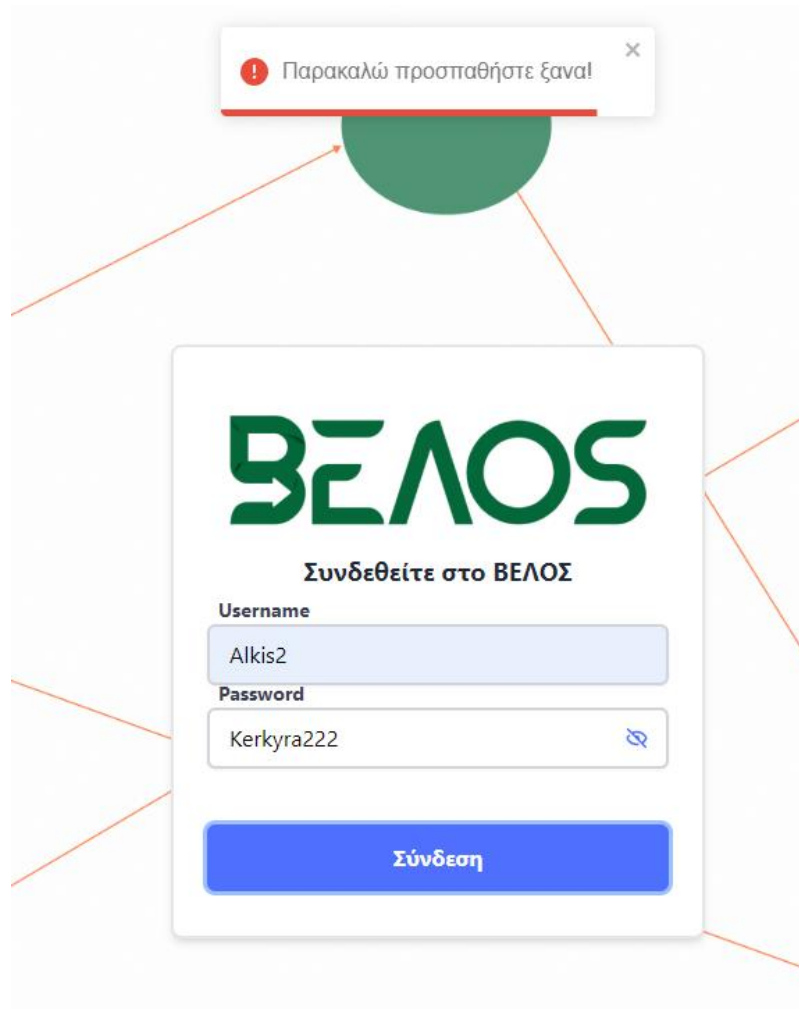
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 12: Η οθόνη σύνδεσης του συστήματος

Εάν τα στοιχεία που έχει πληκτρολογήσει είναι σωστά τότε θα μεταφερθεί στο κεντρικό μενού του συστήματος αλλιώς θα δει το κατάλληλο αρνητικό μήνυμα.

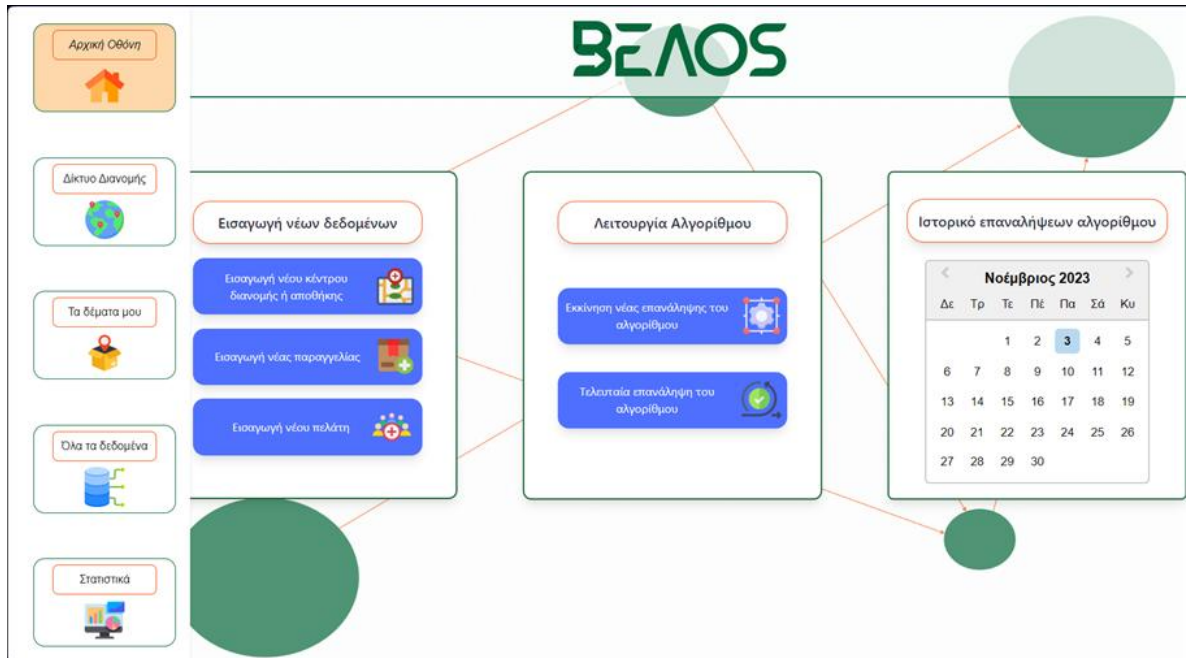
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 13: Αρνητικό μήνυμα σε περίπτωση λάθους

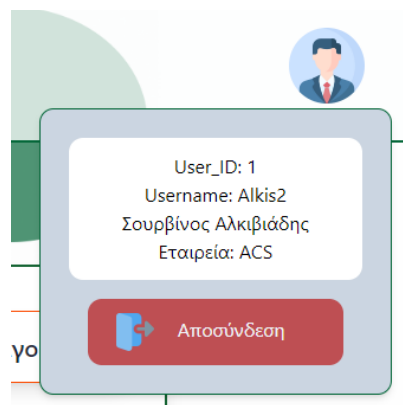
Με την επιτυχημένη σύνδεση στο σύστημα ο χρήστης βλέπει το παρακάτω κεντρικό μενού επιλογών.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 14: Κεντρικό μενού συστήματος

Στο συγκεκριμένο Use Case ο χρήστης κάνει αποσύνδεση πατώντας το πάνω αριστερά εικονίδιο για να εμφανιστεί το παρακάτω μήνυμα.

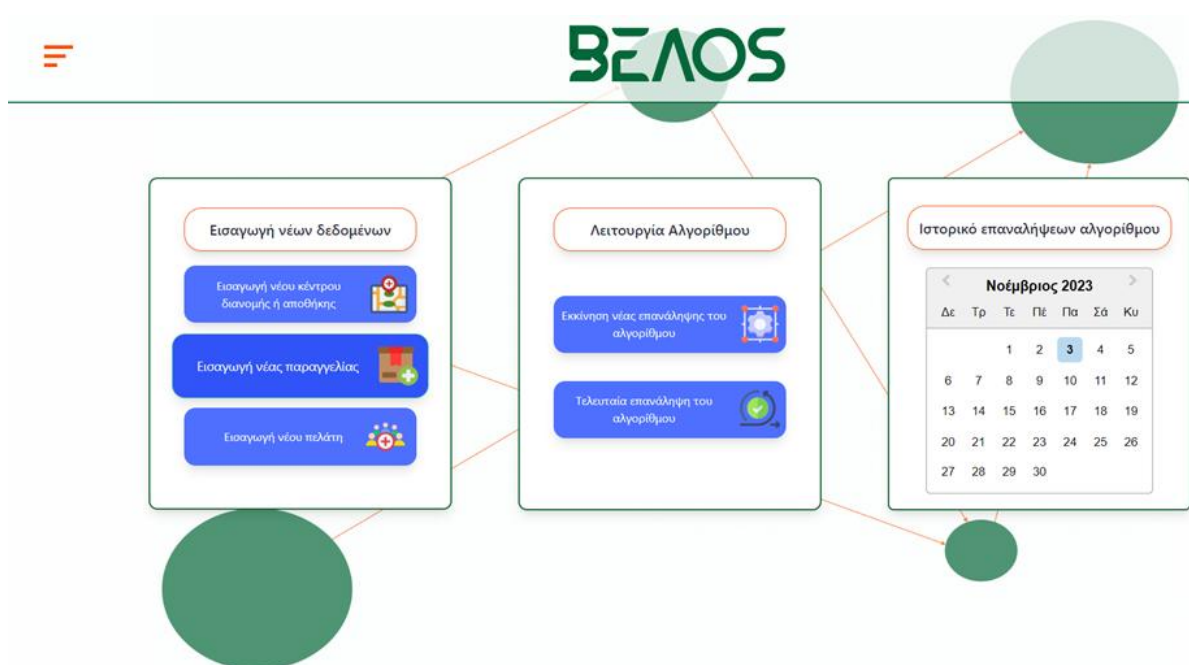


Σχήμα 15: Κουμπί αποσύνδεσης

Το σενάριο τελειώνει με τον χρήστη να πατάει το κουμπί της αποσύνδεσης.

3.2 UseCase 2: Εισαγωγή νέας παραγγελίας και πακέτων

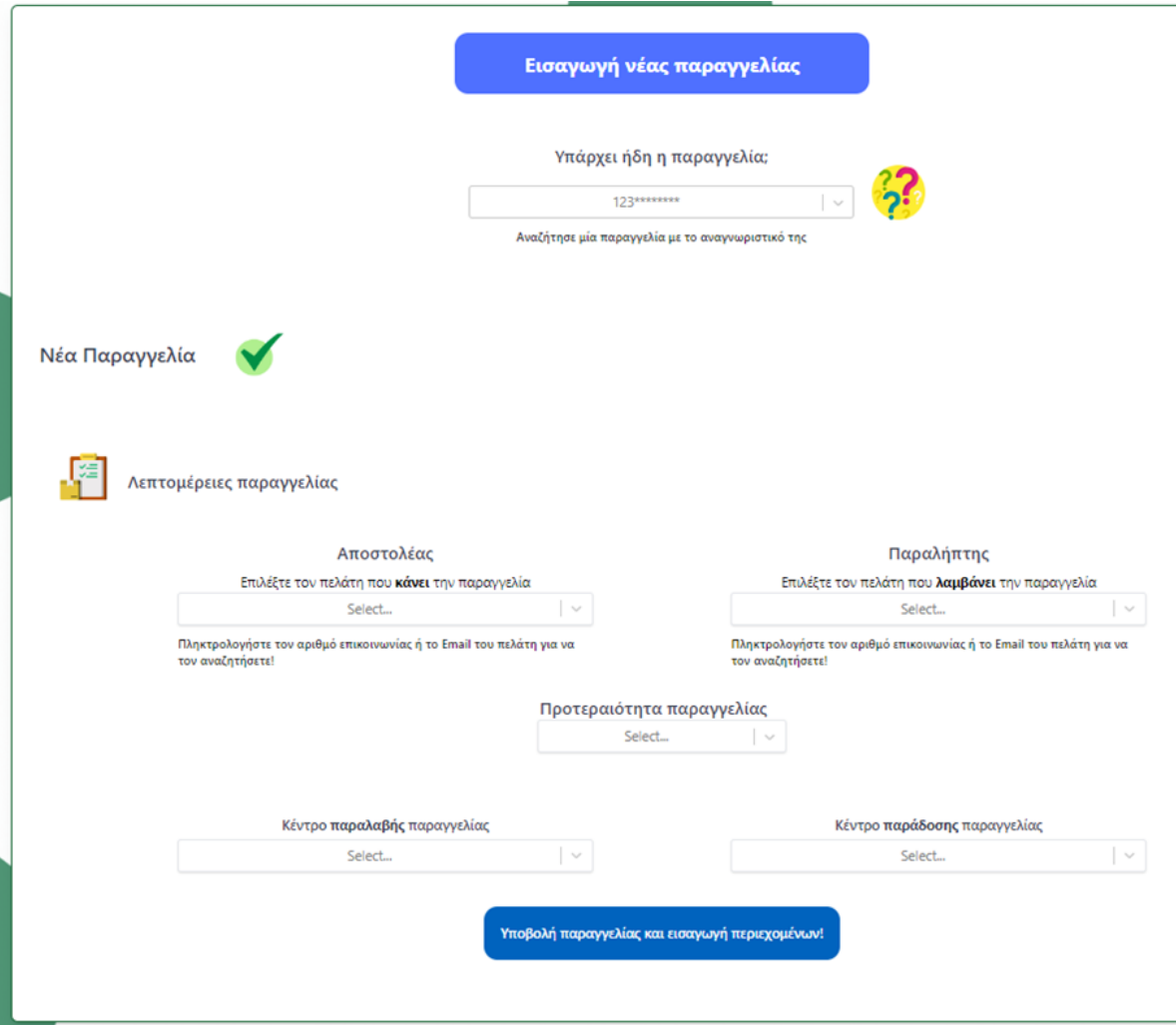
Σε αυτό το σενάριο χρήσης ο χρήστης θέλει να εισάγει μία νέα παραγγελία/ μεταφορά και μετά συγκεκριμένα επιμέρους πακέτα σε αυτήν. Ξεκινώντας ο χρήστης πατάει από το αρχικό μενού την επιλογή, στο πρώτο υπο-μενού "Εισαγωγή νέων Δεδομένων", Εισαγωγή νέας παραγγελίας,



Σχήμα 16: Επιλογή "Εισαγωγή νέας παραγγελίας" στο κεντρικό μενού

Εκεί ο χρήστης θα δει την φόρμα εισαγωγής νέας παραγγελίας όπως φαίνεται παρακάτω.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Εισαγωγή νέας παραγγελίας

Υπάρχει ήδη η παραγγελία:

123*****

Αναζητήστε μία παραγγελία με το αναγνωριστικό της

Νέα Παραγγελία ✓

Λεπτομέρειες παραγγελίας

Αποστολέας
Επιλέξτε τον πελάτη που **κάνει** την παραγγελία
Select...
Πληκτρολογήστε τον αριθμό επικοινωνίας ή το Email του πελάτη για να τον αναζητήσετε!

Παραλήπτης
Επιλέξτε τον πελάτη που **λαμβάνει** την παραγγελία
Select...
Πληκτρολογήστε τον αριθμό επικοινωνίας ή το Email του πελάτη για να τον αναζητήσετε!

Προτεραιότητα παραγγελίας
Select...

Κέντρο παραλαβής παραγγελίας
Select...

Κέντρο παράδοσης παραγγελίας
Select...

Υποβολή παραγγελίας και εισαγωγή περιχομένων!

Σχήμα 17: Φόρμα εισαγωγής νέας παραγγελίας

Ο χρήστης σε αυτήν τη σελίδα μπορεί αρχικά να εξακριβώσει αν υπάρχει η παραγγελία που πάει να εισάγει στο σύστημα. Ενώ αν δεν υπάρχει μπορεί να ξεκινήσει να βάζει τα κατάλληλα δεδομένα. Αρχικά επιλέγει τον πελάτη που στέλνει την παραγγελία και μετά τον πελάτη που θα την παραλάβει. Έπειτα εισάγει την προτεραιότητα της παραγγελίας, το κέντρο διανομών στο οποίο παραλαμβάνεται και το κέντρο διανομών στο οποίο πρέπει να παραδοθεί. Με την εισαγωγή όλων των παραπάνω στοιχείων ο χρήστης βλέπει το παρακάτω.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Νέα Παραγγελία

Λεπτομέρειες παραγγελίας

Αποστολέας

Επιλέξτε τον πελάτη που **κάνει** την παραγγελία

Μολαμάτη Μπραζιώτης, 6945608321, maure98@example... | v

Πληκτρολογήστε τον αριθμό επικοινωνίας ή το Email του πελάτη για να τον αναζητήσετε!

Παραλήπτης

Επιλέξτε τον πελάτη που **λαμβάνει** την παραγγελία

Άρτεμις Μοσχολάκη, 6945032168, grigorios.diamantis@ex... | v

Πληκτρολογήστε τον αριθμό επικοινωνίας ή το Email του πελάτη για να τον αναζητήσετε!

Προτεραιότητα παραγγελίας

Υψηλή | v

Κέντρο παραλαβής παραγγελίας

#1, Αθήνα Κέντρο, ACS | v

Κέντρο παράδοσης παραγγελίας

#4, Ιωάννινα Κέντρο, Μάργαρας και υιοί | v

Υποβολή παραγγελίας και εισαγωγή περιχομένων!

Σχήμα 18: Φόρμα εισαγωγής νέας παραγγελίας με συμπληρωμένα τα στοιχεία

Πατώντας υποβολή φαίνεται ένα μήνυμα επιβεβαίωσης στο οποίο ο χρήστης μπορεί να εξακριβώσει αν υπήρξαν λάθη κατά την διάρκεια της εισαγωγής. Αν δεν υπήρξαν μπορεί να πατήσει επιβεβαίωση

Επιβεβαίωση εισαγωγής νέας παραγγελίας

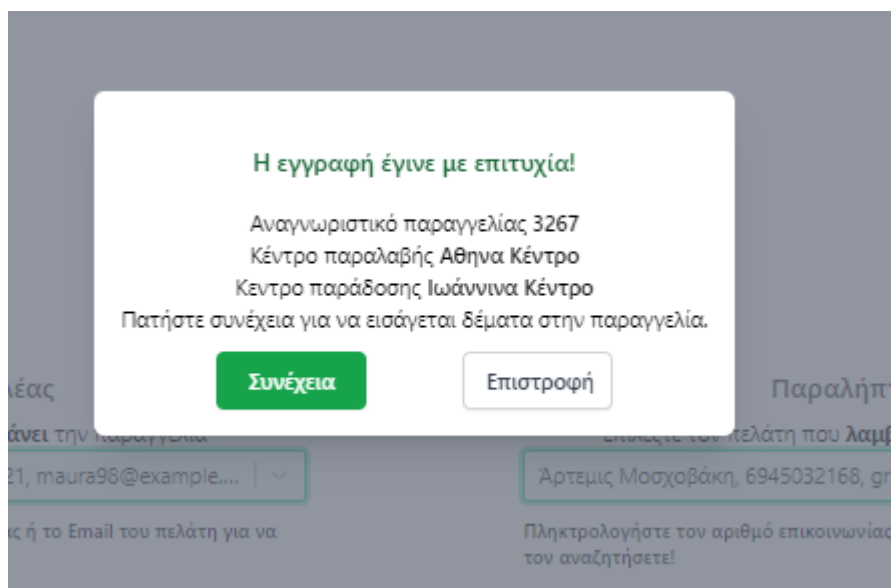
Παραδότης	Παραλαβέας	Προτεραιότητα	Κέντρο παραλαβής	Κέντρο παράδοσης
Μολαμάτη Μπραζιώτης, 6945608321, maure98@example.org	Άρτεμις Μοσχολάκη, 6945032168, grigorios.diamantis@example.net	Υψηλή	#1, Αθήνα Κέντρο, ACS	#4, Ιωάννινα Κέντρο, Μάρ

Επιστροφή **Επιβεβαίωση**

Σχήμα 19: Μήνυμα επιβεβαίωσης εισαγωγής παραγγελίας

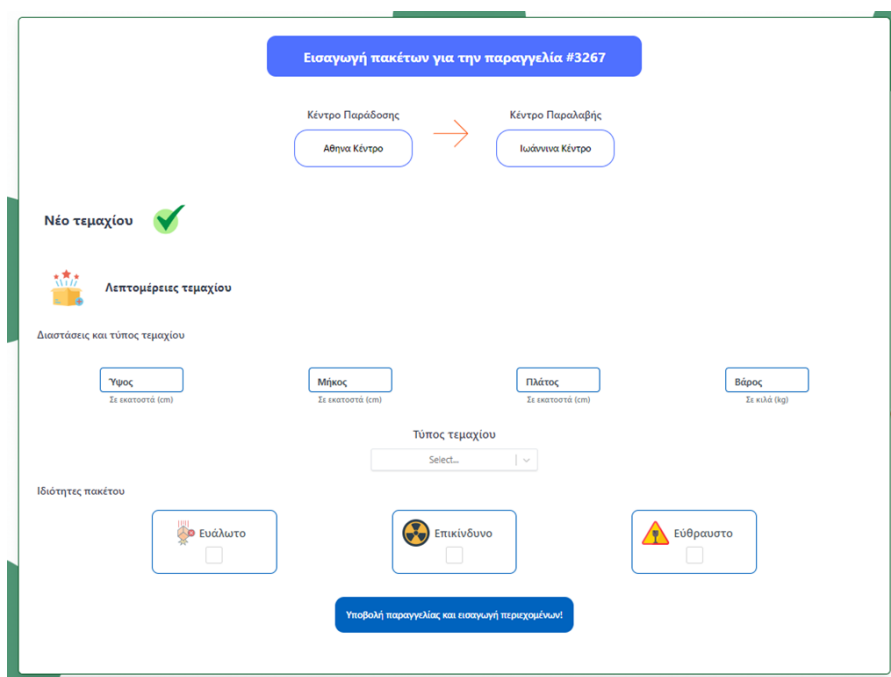
Με την επιτυχημένη εισαγωγή των στοιχείων ο χρήστης δέχεται από το σύστημα μήνυμα επιτυχίας όπως φαίνεται παρακάτω.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 20: Μήνυμα επιτυχημένης καταχώρησης παραγγελίας

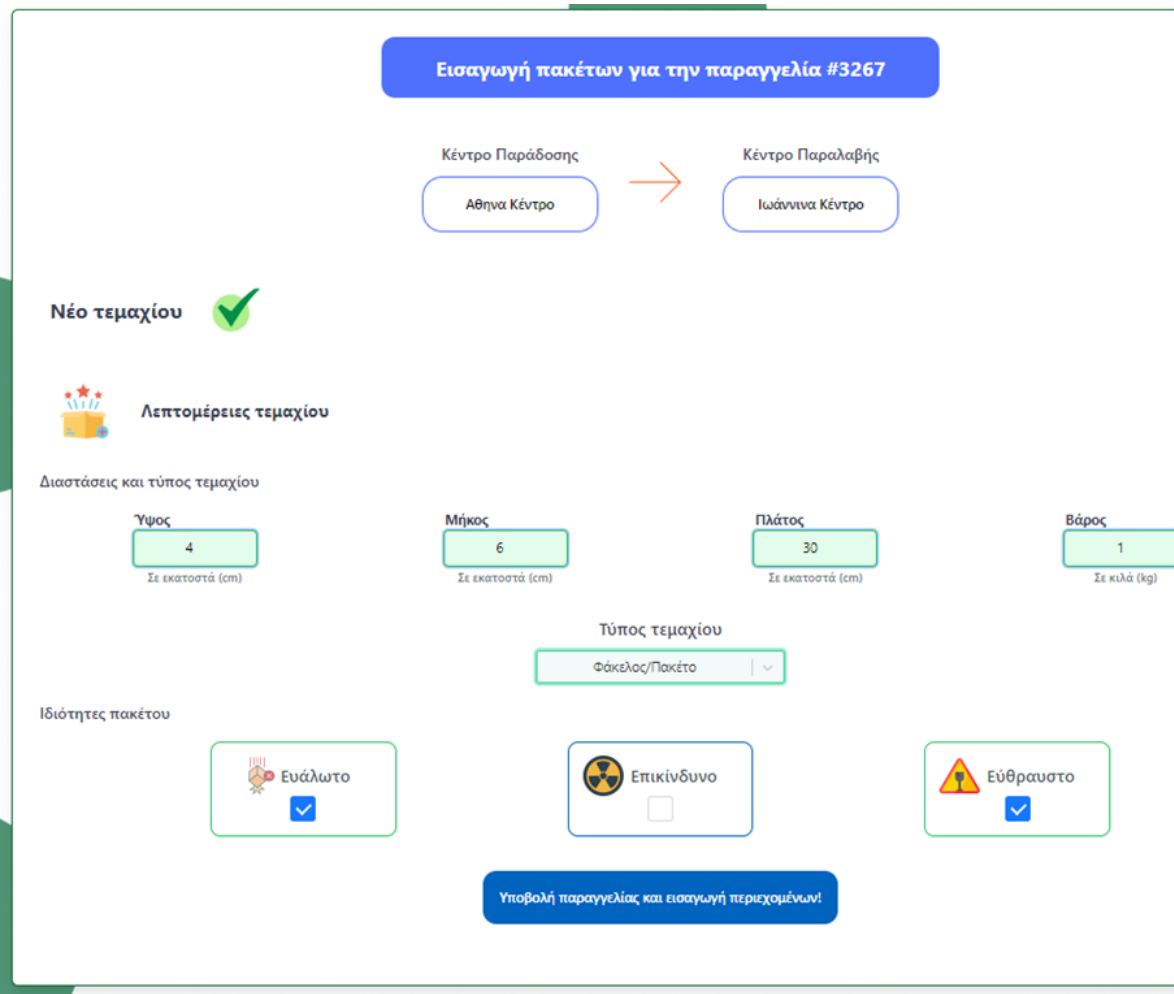
Πατώντας συνέχεια μπορεί να εισάγει πακέτα στην παραγγελία καθώς το σύστημα κρατάει τον αριθμό της νέας παραγγελίας και εμφανίζει την επόμενη οθόνη



Σχήμα 21: Οθόνη εισαγωγής πακέτων σε μία παραγγελία

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Βάζοντας τα απαραίτητα δεδομένα στην παραπάνω φόρμα μπορεί ο χρήστης να



Εισαγωγή πακέτων για την παραγγελία #3267

Κέντρο Παράδοσης → Κέντρο Παραλαβής
Αθηνά Κέντρο → Ιωάννινα Κέντρο

Νέο τεμαχίου ✓

Λεπτομέρειες τεμαχίου

Διαστάσεις και τύπος τεμαχίου

Υψος 4 Σε εκατοστά (cm)	Μήκος 6 Σε εκατοστά (cm)	Πλάτος 30 Σε εκατοστά (cm)	Βάρος 1 Σε κιλά (kg)
-------------------------------	--------------------------------	----------------------------------	----------------------------

Τύπος τεμαχίου
Φάκελος/Πακέτο

Ιδιότητες πακέτου

<input checked="" type="checkbox"/> Ευάλωτο	<input type="checkbox"/> Επικίνδυνο	<input checked="" type="checkbox"/> Εύθραυστο
---	-------------------------------------	---

Υποβολή παραγγελίας και εισαγωγή περιεχομένων!

Σχήμα 22: Οθόνη εισαγωγής πακέτων με συμπληρωμένη την φόρμα

πατήσει υποβολή

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

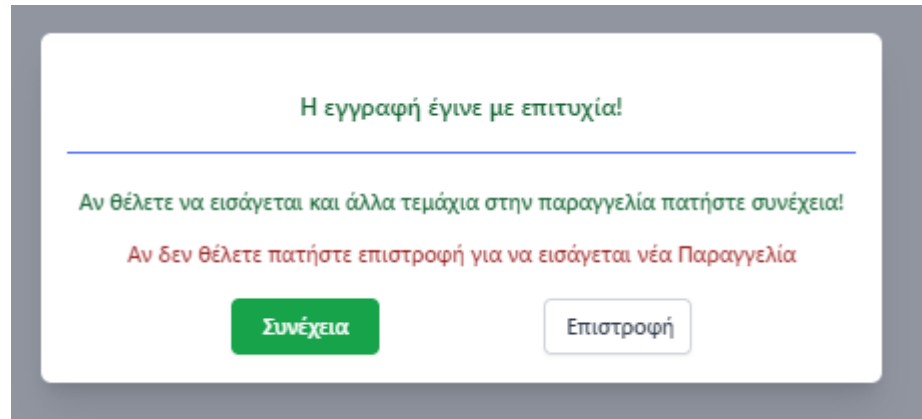
Επιβεβαίωση εισαγωγής νέας παραγγελίας

Μέρος της παραγγελίας	Τύπος τεμαχίου	Ύψος	Μήκος	Πλάτος	Βάρος	Ογκομετρικό βάρος	Είναι Ευάλωτο:	Είναι Επικίνδυνο:	Είναι Εύθραστο:	Κέντρο Παραλαβής	Κέντρο
#3267	Φάκελος/Πακέτο	4cm	6cm	30cm	1kg	1,44m ³ /kg	Ναι	Όχι	Ναι	Αθήνα Κέντρο	Ιωάν

Σχήμα 23: Μήνυμα επιβεβαίωσης εισαγωγής πακέτων στο σύστημα

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

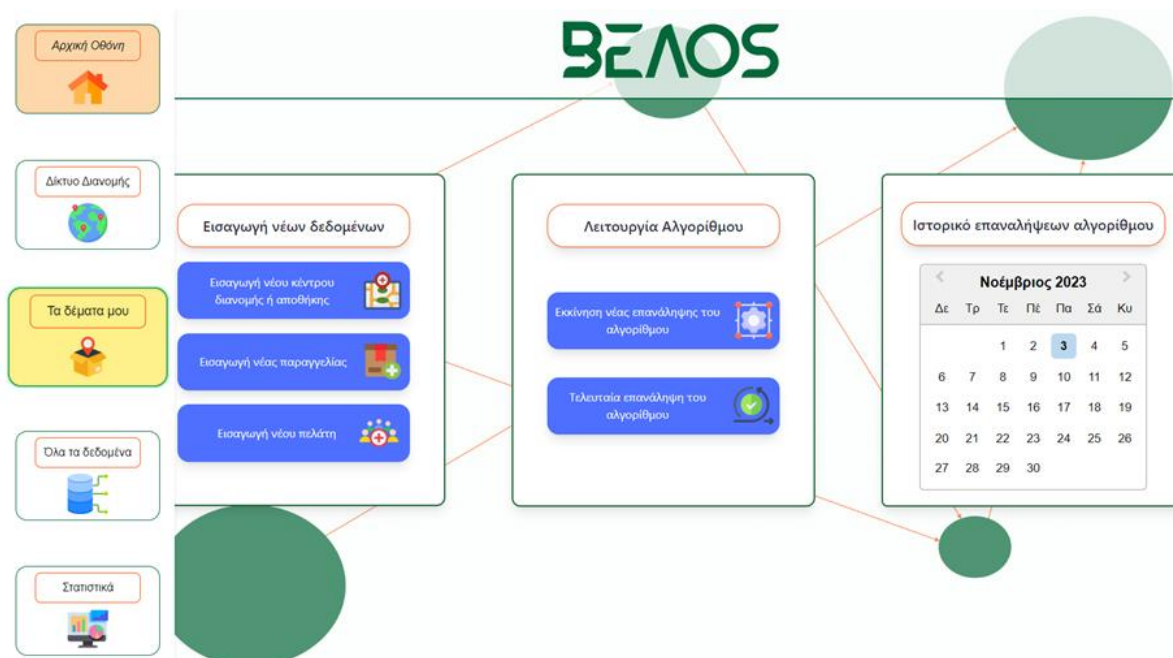
Με το παραπάνω μήνυμα επιβεβαίωσης ο χρήστης μπορεί είτε να πατήσει επιστροφή όταν υπάρχει λάθος είτε να πατήσει επιβεβαίωση με το αντίστοιχο μήνυμα επιτυχίας ή αποτυχίας ανά περίπτωση.



Σχήμα 24: Μήνυμα επιβεβαίωσης εισαγωγής πακέτου

3.3 UseCase 3: Εμφάνιση όλων των παραγγελιών και λεπτομέρειες για κάθε μία από αυτές

Για το παραπάνω σενάριο ο χρήστης πρέπει να επιλέξει την τρίτη επιλογή στο πλαϊνό μενού της αρχικής σελίδας του συστήματος όπως φαίνεται παρακάτω:



Σχήμα 25: Επιλογή "Τα δέματα μου" στο πλαϊνό μενού

Πατώντας την επιλογή ο χρήστης βλέπει την επόμενη οθόνη με όλες τις παραγγελίες/ δέματα που διαχειρίζεται η εταιρεία του.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΒΕΛΟΣ

ID Πακέτου/ Παλέτας	ID Μεταφοράς	ID Τρέχοντος κόμβου	Κατάσταση Παραγγελίας	ID Κόμβου έναρξης	ID Επόμενου κόμβου	ID Τελικού κόμβου	Ημερομηνία εκτέλεσης
29	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
57	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
58	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
59	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
89	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
90	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
146	82	2	Last_mile	4	5	5	11/06/2023 12:55:42
203	98	6	Last_mile	5	1	1	11/06/2023 12:55:42
204	98	6	Last_mile	5	1	1	11/06/2023 12:55:42
205	98	6	Last_mile	5	1	1	11/06/2023 12:55:42
226	82	2	Last_mile	8	5	5	11/06/2023 12:55:42
227	82	2	Last_mile	8	5	5	11/06/2023 12:55:42
247	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
248	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
249	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
352	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
354	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
365	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
366	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
438	82	2	Last_mile	4	5	5	11/06/2023 12:55:42
733	98	6	Last_mile	5	1	1	11/06/2023 12:55:42
781	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
782	80	1	Last_mile	6	9	9	11/06/2023 12:55:42
783	82	2	Last_mile	8	5	5	11/06/2023 12:55:42
815	82	2	Last_mile	8	5	5	11/06/2023 12:55:42
816	82	2	Last_mile	8	5	5	11/06/2023 12:55:42

Σχήμα 26: Πίνακας με όλα δέματα που έχει μια εταιρεία

Περνώντας τον κέρσορα του ποντικιού πάνω από ένα κελί του πίνακα ο χρήστης μπορεί να μάθει περισσότερες λεπτομέρειες για αυτό το δέμα.

ID Πακέτου/ Παλέτας	ID Μεταφοράς	ID Τρέχοντος κόμβου	Κατάσταση Παραγγελίας	ID Κόμβου έναρξης	ID Επόμενου κόμβου	ID Τελικού κόμβου	Ημερομηνία εκτέλεσης
29	82	2	Last_mile	3	5	5	11/06/2023 12:55:42
57	82	2	Last_mile	3	5	5	11/06/2023 12:55:42

Σχήμα 27: Επιλογή δέματος

Πατώντας το πορτοκαλί εικονίδιο στο τέλος του κελιού ο χρήστης μεταφέρεται σε μία νέα οθόνη στην οποία μπορεί να δει αναλυτικά τις λεπτομέρειες ενός δέματος, όπως το που βρίσκεται, την πορεία που είχε κ.ο.κ.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 28: Λεπτομέρειες για την τοποθεσία και διαδρομή του δέματος

Στην πάνω εικόνα φαίνονται στα δεξιά μία χρονοσειρά που απεικονίζει την διαδρομή του πακέτου χρονικά ενώ αριστερά υπάρχει μία απεικόνιση της διαδρομής του πακέτου στο δίκτυο διανομής της εταιρείας με τη χρήση χάρτη.

3.4 UseCase 4: Εμφάνιση όλων των δεδομένων της βάσης δεδομένων

Στο επόμενο σενάριο χρήσης, ο χρήστης μπορεί να επιλέξει την τέταρτη επιλογή του πλαϊνού μενού για να δει όλα τα δεδομένα που υπάρχουν στην βάση.



Σχήμα 29: Επιλογή "Όλα τα δεδομένα" στο πλαϊνό μενού

Στην παρακάτω εικόνα φαίνονται όλα τα καταχωρημένα δέματα που υπάρχουν στην βάση δεδομένων

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΒΕΛΟΣ

Τεμάχια

Παραγγελίες

Πελάτες

Μεταφορές

Κέντρα

Εντοπισμός

Τεμάχια														
ID Πακέτου/ Παλέτας	ID Παραγγελίας	Τύπος τεμαχίου	Ύψος	Μήκος	Πλάτος	Βάρος	Είναι Ευάλωτο;	Είναι Επικίνδυνο;	Είναι Εύφραστο;	Εναρκτηρίος κόμβος	Τρέχον κόμβος	Επόμενος κόμβος	Τελικ κόμβος	
1	2551	Parcel	15	5	3	3.49	Όχι	Ναι	Ναι	Ιωάννινα Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
2	2551	Pallet	5	1	10	5.73	Ναι	Όχι	Ναι	Ιωάννινα Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
3	2551	Parcel	17	10	6	3.47	Ναι	Όχι	Ναι	Ιωάννινα Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
4	2552	Parcel	3	8	2	6.67	Όχι	Όχι	Όχι	Πάτρα Κέντρο	Πρέβεζα Κέντρο	---	Πρέβεζα Κέντρο	
5	2553	Parcel	8	9	12	9.35	Όχι	Όχι	Όχι	Λάρισα Κέντρο	Πρέβεζα Κέντρο	---	Πρέβεζα Κέντρο	
6	2553	Parcel	6	5	2	5.94	Ναι	Ναι	Ναι	Λάρισα Κέντρο	Πρέβεζα Κέντρο	---	Πρέβεζα Κέντρο	
7	2553	Pallet	10	7	5	2.78	Ναι	Όχι	Ναι	Λάρισα Κέντρο	Πρέβεζα Κέντρο	---	Πρέβεζα Κέντρο	
8	2554	Parcel	2	8	6	7.12	Όχι	Όχι	Όχι	Κατερίνη Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
9	2554	Parcel	7	6	11	3.67	Ναι	Όχι	Όχι	Κατερίνη Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
10	2554	Parcel	16	10	1	5.72	Όχι	Ναι	Όχι	Κατερίνη Κέντρο	Αθίνα Κέντρο	---	Αθίνα Κέντρο	
11	2555	Pallet	12	7	10	1.78	Ναι	Ναι	Ναι	Λάρισα Κέντρο	Κέρκυρα	---	Κέρκυρα	

Σχήμα 30: Εμφάνιση των πινάκων του συστήματος με τα δεδομένα της βάσης

Πατώντας το πλαινό μενού της οθόνης ο χρήστης μπορεί να περιηγηθεί σε όλους τους πίνακες της βάσης δεδομένων. Για παράδειγμα, πατώντας το κουμπι "Κέντρα" εμφανίζεται η επόμενη οθόνη.

Τεμάχια

Παραγγελίες

Πελάτες

Μεταφορές

Κέντρα

Κέντρα διανομών							
ID Κέντρου	Όνομασία Κέντρου	Εταιρεία	Διεύθυνση	Πόλη	Ταχυδρομικός κώδικας	Μίσσηση χωρητικότητα	Τρέχουσα χωρητικότητα
1	Αθίνα Κέντρο	ACS	Βεργίου 152	Αθίνα	TK 81023	1247	83.67
2	Θεσσαλονίκη Κέντρο	RMR	Πλ. Κρακελλου 98-08	Θεσσαλονίκη	TK 76901	1027	210.26
3	Πάτρα Κέντρο	Ionian University	Κακοπευρίου 05	Πάτρα	632 00	1138	45.97
4	Ιωάννινα Κέντρο	Μάργαρης και υιοί	Εμμανουηλίδου 414	Ιωάννινα	839 88	1320	45.03
5	Κέρκυρα Κέντρο	ACS	Βάρδας 50-50	Κέρκυρα	42730	1262	63.54
6	Λάρισα Κέντρο	RMR	Κακοπευρίου 05	Λάρισα	632 00	1138	132.05
7	Κοζάνη Κέντρο	Elta Courier	Πηγαδίων 29-62	Κοζάνη	302 71	1482	66.61
8	Πρέβεζα Κέντρο	Χριστοδούλου Παπαδύμα και Αβραμίδης	Πλ. Βύσσας 060-622	Πρέβεζα	205 51	1400	38.06
9	Κατερίνη Κέντρο	FeDEX	Πλ. Κρακελλου 98-08	Κατερίνη	TK 76901	1027	59.07

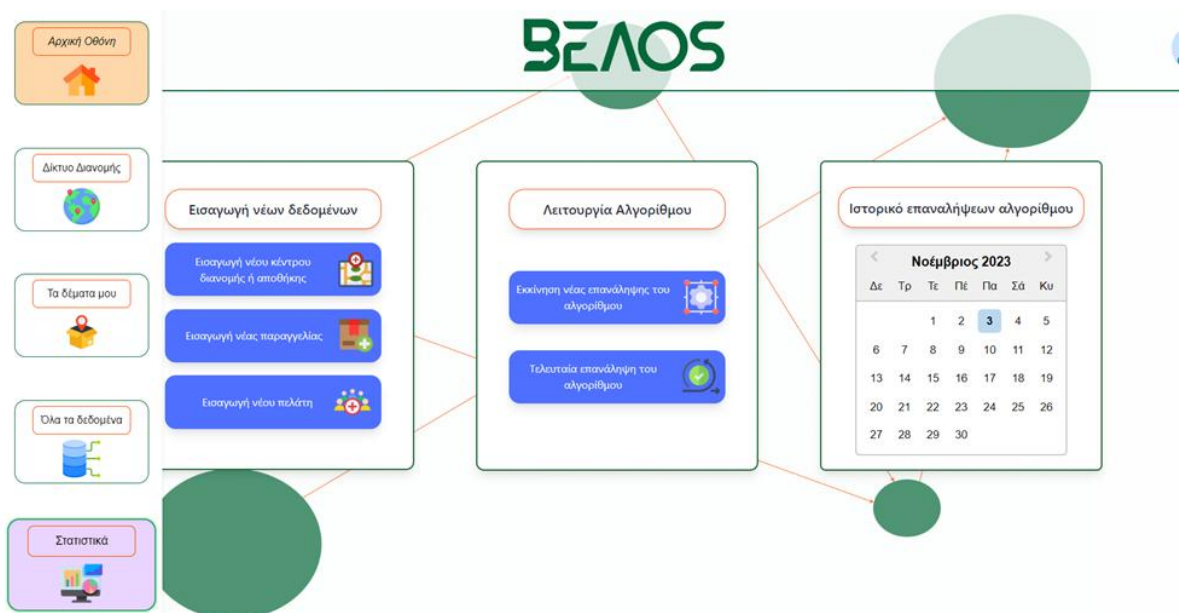
Σχήμα 31: Εμφάνιση του πίνακα "Κέντρα" από την βάση δεδομένων

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Με παρόμοιο τρόπο ο χρήστης μπορεί να δει τα δεδομένα όλης της βάσης με την ίδια μορφοποίηση

3.5 UseCase 5: Εμφάνιση στατιστικών στοιχείων για την εταιρεία μου

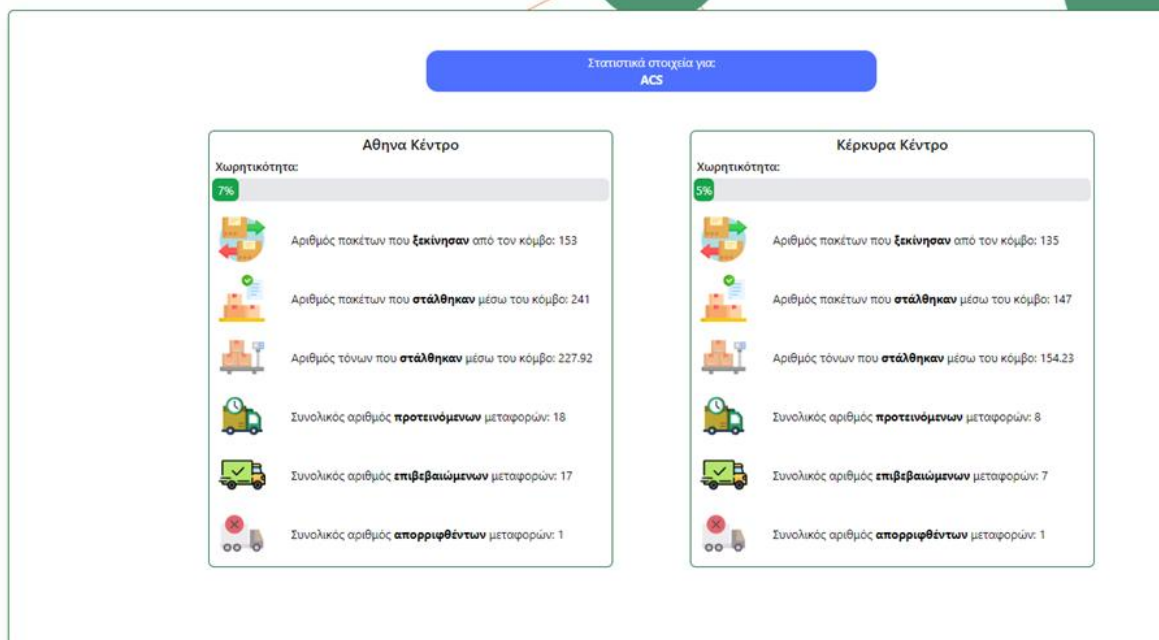
Στο επόμενο σενάριο χρήσης, ο χρήστης μπορεί να επιλέξει την τελευταία επιλογή του πλαϊνού μενού για να δει τα στατιστικά στοιχεία που δημιουργεί το σύστημα.



Σχήμα 32: Επιλογή "Στατιστικά" για την εμφάνιση στατιστικών στοιχείων

Πατώντας το κουμπί στατιστικά, ο χρήστης μεταφέρεται στην οθόνη που υπάρχουν τα στατιστικά στοιχεία που δημιουργεί το σύστημα όπως φαίνονται παρακάτω.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 33: Συγκεντρωτικά στατιστικά στοιχεία

Παραπάνω φαίνονται σημαντικές πληροφορίες για τα κέντρα διανομών που αντιστοιχούν στον χρήστη. Ενώ φαίνονται και γενικά στατιστικά όπως οι προτεινόμενες μεταφορές, οι επιβεβαιωμένες διαφορές όπως και οι μεταφορές που έχουν απορριφθεί.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

3.6 UseCase 6: Εμφάνιση δικτύου διανομών και λεπτομέρειες για τους κόμβους της εταιρείας

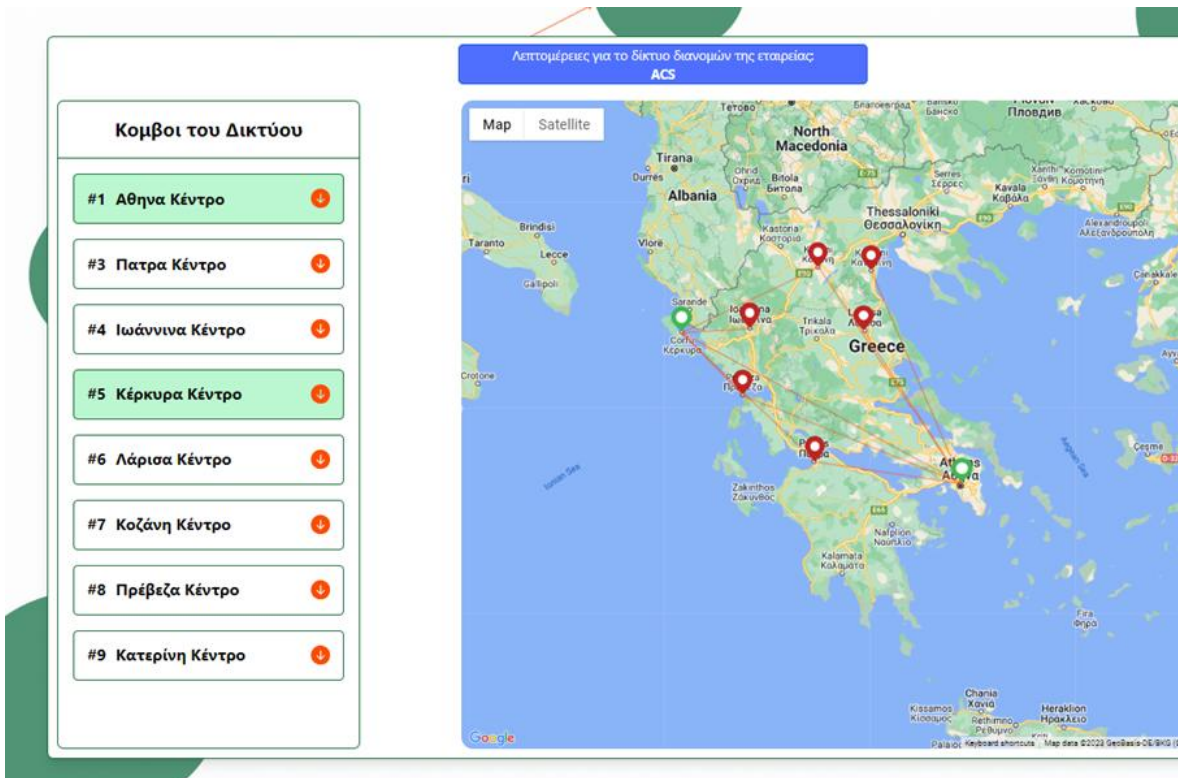
Για να δει ο χρήστης πληροφορίες για το δίκτυο διανομών της εταιρείας στην οποία ανήκει, πρέπει να πατήσει το δεύτερο κουμπί στο πλαϊνό μενού που ονομάζεται "Δίκτυο Διανομής", με το ίδιο τρόπο που φαίνεται παρακάτω:



Σχήμα 34: Επιλογή "Δίκτυο Διανομής" στο πλαϊνό μενού

Επιλέγοντας το προηγούμενο κουμπί ο χρήστης μεταφέρεται στην κεντρική σελίδα που αφορά τα κέντρα διανομών.

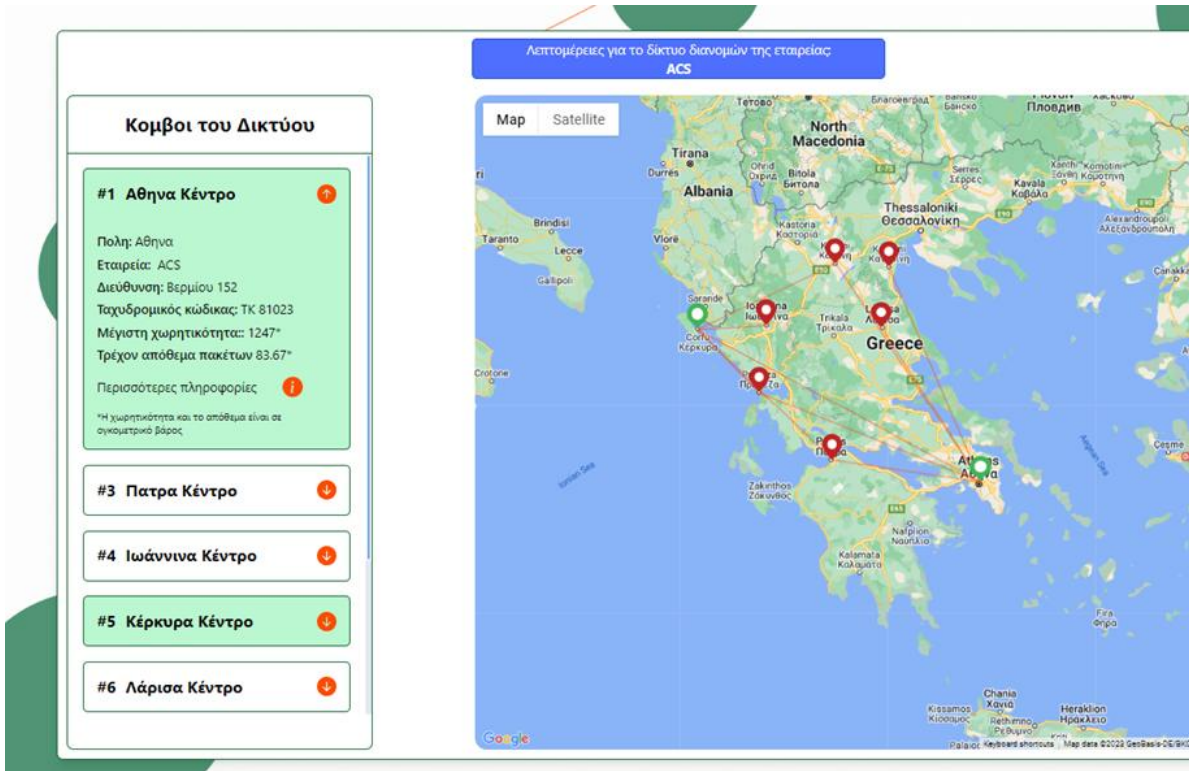
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 35: Κεντρική σελίδα δικτύου διανομής

Στην παραπάνω εικόνα (35) φαίνεται το δίκτυο διανομών μίας εταιρείας. Οι πινέζες στον χάρτη που είναι πράσινες έχουν το ίδιο χρώμα και στο πλαίσιο μενού της σελίδας. Αυτό συμβαίνει διότι οι πράσινες πινέζες αντιπροσωπεύουν τα κέντρα διανομών που ανήκουν στην εταιρεία του χρήστη. Πατώντας σε ένα από αυτά τα κέντρα μπορεί κανείς να δει παραπάνω πληροφορίες για αυτό όπως φαίνεται παρακάτω.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 36: Λεπτομέρειες για επιλεγμένο κόμβο

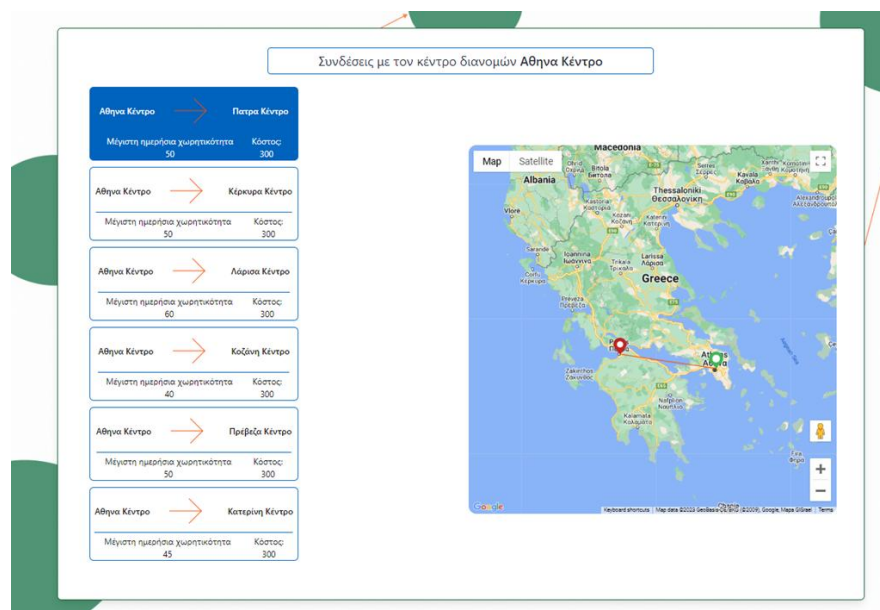
Πατώντας την επιλογή περισσότερες πληροφορίες ο χρήστης μπορεί να δει λεπτομέρειες για τις κινήσεις και την κινητικότητα στον κόμβο όπως φαίνεται παρακάτω.



Οι Σχήμα 37: Λεπτομέρειες για την κινητικότητα στον κόμβο

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

λεπτομέρειες χωρίζονται σε δύο υπο-μενού. Ένα για τις παραγγελίες και ένα για την κινητικότητα. Ξεκινώντας με την κινητικότητα με την πρώτη επιλογή, ο χρήστης μπορεί να δει όλες τις συνδέσεις που έχει ο κόμβος που έχει επιλέξει με τους γείτονες του. Με την δεύτερη επιλογή ο χρήστης μπορεί να δει όλες τις μετακινήσεις που είτε βρίσκονται σε εξέλιξη είτε μόλις έχουμε προταθεί από τον αλγόριθμο. Ακολουθούν φωτογραφίες που δείχνουν τα αντίστοιχα.



Σχήμα 38: Συνδέσεις στον κόμβο

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Τρέχουσες μετακινήσεις για το κέντρο διανομών Αθήνα Κέντρο

Αφίξεις				Αναχωρήσεις			
ID Μεταφοράς 98	ID οδηγού 9	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Φόνοι από: Λάρισα Κέντρο	ID Μεταφοράς 75	ID οδηγού 5	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Πάτρα Κέντρο
ID Μεταφοράς 112	ID οδηγού 5	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Φόνοι από: Κατάρνη Κέντρο	ID Μεταφοράς 76	ID οδηγού 1	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Κέρκυρα Κέντρο
				ID Μεταφοράς 77	ID οδηγού 6	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Λάρισα Κέντρο
				ID Μεταφοράς 78	ID οδηγού 3	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Κοζάνη Κέντρο
				ID Μεταφοράς 79	ID οδηγού 8	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Πάτρα Κέντρο
				ID Μεταφοράς 80	ID οδηγού 4	Ημερομηνία Έκδοσης 11/06/2023 12:55:41	Αναχωρεί για: Κατάρνη Κέντρο

Σχήμα 39: Τρέχουσες μετακινήσεις στο επιλεγμένο κέντρο διανομών

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Συνεχίζοντας με το δεύτερο υπο-μενού η πρώτη επιλογή εμφανίζει όλα τα δέματα που υπάρχουν στο κέντρο διανομών που έχει επιλεχθεί. Ενώ η δεύτερη επιλογή δείχνει όλες τις παραγγελίες που είναι έτοιμες να παραδοθούν άμεσα στο κέντρο διανομών.

Τεμάχια στον κόμβο Αθήνα Κέντρο

Τεμάχια υπό επεξεργασία

Τεμάχια από τρίτους

Τεμάχιο	Παραγγελία	Αρχικός προορισμός	Τελικός προορισμός
#89	#2593	Λάρισα Κέντρο	Κατερίνη Κέντρο
#90	#2593	Λάρισα Κέντρο	Κατερίνη Κέντρο
#352	#2721	Λάρισα Κέντρο	Κατερίνη Κέντρο
#354	#2721	Λάρισα Κέντρο	Κατερίνη Κέντρο
#781	#2926	Λάρισα Κέντρο	Κατερίνη Κέντρο
#782	#2926	Λάρισα Κέντρο	Κατερίνη Κέντρο
#942	#3005	Λάρισα Κέντρο	Κατερίνη Κέντρο
#1060	#3058	Αθήνα Κέντρο	Πάτρα Κέντρο
#091	#3058	Αθήνα Κέντρο	Πάτρα Κέντρο
#1118	#3091	Αθήνα Κέντρο	Κοζάνη Κέντρο
#1212	#3133	Αθήνα Κέντρο	Κίρκερα Κέντρο

Λεπτομέρειες για το πακέτο #89

Διαστάσεις παλέτας

Μήκος: 4 cm
Πλάτος: 8 cm
Ύψος: 14 cm
Βάρος: 7.1 kg

Χαρακτηριστικά παλέτας

Επικίνδυνα: Ναι
Εύθραστο: Ναι
Ευάλυτο: Ναι

Μέρος της Παραγγελίας #2593

Αρχικός κόμβος: Λάρισα Κέντρο
Τελικός κόμβος: Κατερίνη Κέντρο

Χρονολόγιο παλέτας #89

Εναρξη παραγγελίας Κόμβος έναρξης: Λάρισα Κέντρο

28/05/2023 13:20:47

Μετακίνηση: #2 Κόμβος έναρξης: Λάρισα Κέντρο Κόμβος αφής: Αθήνα Κέντρο ID Μετακίνησης: 59 Κατάσταση: Καθ'όσον

30/05/2023 13:22:09

Μετακίνηση: #3 Κόμβος έναρξης: Αθήνα Κέντρο Κόμβος αφής: Κατερίνη Κέντρο ID Μετακίνησης: 80 Κατάσταση: Προς παράδοση

11/06/2023 12:55:42

Σχήμα 40: Διαθέσιμα δέματα στο κέντρο διανομών που έχει επιλεχθεί

Παραγγελίες προς παράδοση για το κέντρο διανομών Αθήνα Κέντρο

ID παραγγελίας:	Προτεραιότητα:	Ημ/νία Εναρξης:
2551	Κανονική	28/05/2023 13:20:47
2554	Κανονική	28/05/2023 13:20:47
2556	Υψηλή	28/05/2023 13:20:47
2570	Υψηλή	28/05/2023 13:20:47
2571	Κανονική	28/05/2023 13:20:47

Λεπτομέρειες για την παραγγελία #2551

Συνολικό σακο/κο βάρος: 14.04
Αποδέκτης παραγγελίας 28

Αναλυτικά τα περιεχόμενα της παραγγελίας

Λεπτομέρειες για το τεμάχιο #1

Διαστάσεις πακέτου

Μήκος: 3 cm
Πλάτος: 5 cm
Ύψος: 15 cm
Βάρος: 3.49 kg

Χαρακτηριστικά πακέτου

Επικίνδυνα: Ναι
Εύθραστο: Ναι
Ευάλυτο: Όχι

Μέρος της Παραγγελίας #2551

Αρχικός κόμβος: Ιωάννινα Κέντρο
Τελικός κόμβος: Αθήνα Κέντρο

Λεπτομέρειες για το τεμάχιο #2

Διαστάσεις παλέτας

Μήκος: 10 cm
Πλάτος: 1 cm
Ύψος: 5 cm

Χαρακτηριστικά παλέτας

Επικίνδυνα: Όχι
Εύθραστο: Ναι

Μέρος της Παραγγελίας #2551

Αρχικός κόμβος: Ιωάννινα Κέντρο

Σχήμα 41: Παραγγελίες προς παράδοση στον επιλεγμένο κόμβο



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο
Περιφερειακής Ανάπτυξης

ΕΠΑνεΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

3.7 UseCase 7: Εμφάνιση της τελευταίας επανάληψης του αλγορίθμου

Συνεχίζοντας στο κύριο μενού του συστήματος, ο χρήστης μπορεί να επιλέξει την δεύτερη επιλογή στο μεσαίου υπο-μενού που λέγεται "Τελευταία επανάληψη του αλγορίθμου"



Σχήμα 42: Επιλογή "Τελευταία επανάληψη του αλγορίθμου"

Ο χρήστης μεταφέρεται στην οθόνη που φαίνονται οι αποδεκτές και οι μη αποδεκτές μετακινήσεις που έχει ο ίδιος αποφασίσει.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Αναχωρήσεις

Μεταφορές που αναχωρούν από τον κόμβο: Αθίνα Κέντρο

ID Μεταφοράς	ID Οδηγού	Ποσότητα σε τεμάχια:	Συνολικό βάρος φορτίου:	Κόστος μεταφοράς:	Ημερομηνία έκδοσης:	Από:	Προς:
75	5	9	6.67	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Πάτρα Κέντρο
76	1	16	18.21	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Κέρκυρα Κέντρο
77	6	2	0.5	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Λάρισα Κέντρο
78	3	5	9.97	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Κοζάνη Κέντρο
79	8	5	5.38	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Πρέβεζα Κέντρο
80	4	9	10.31	300	11/06/2023 12:55:41	Αθίνα Κέντρο	Κατερίνη Κέντρο

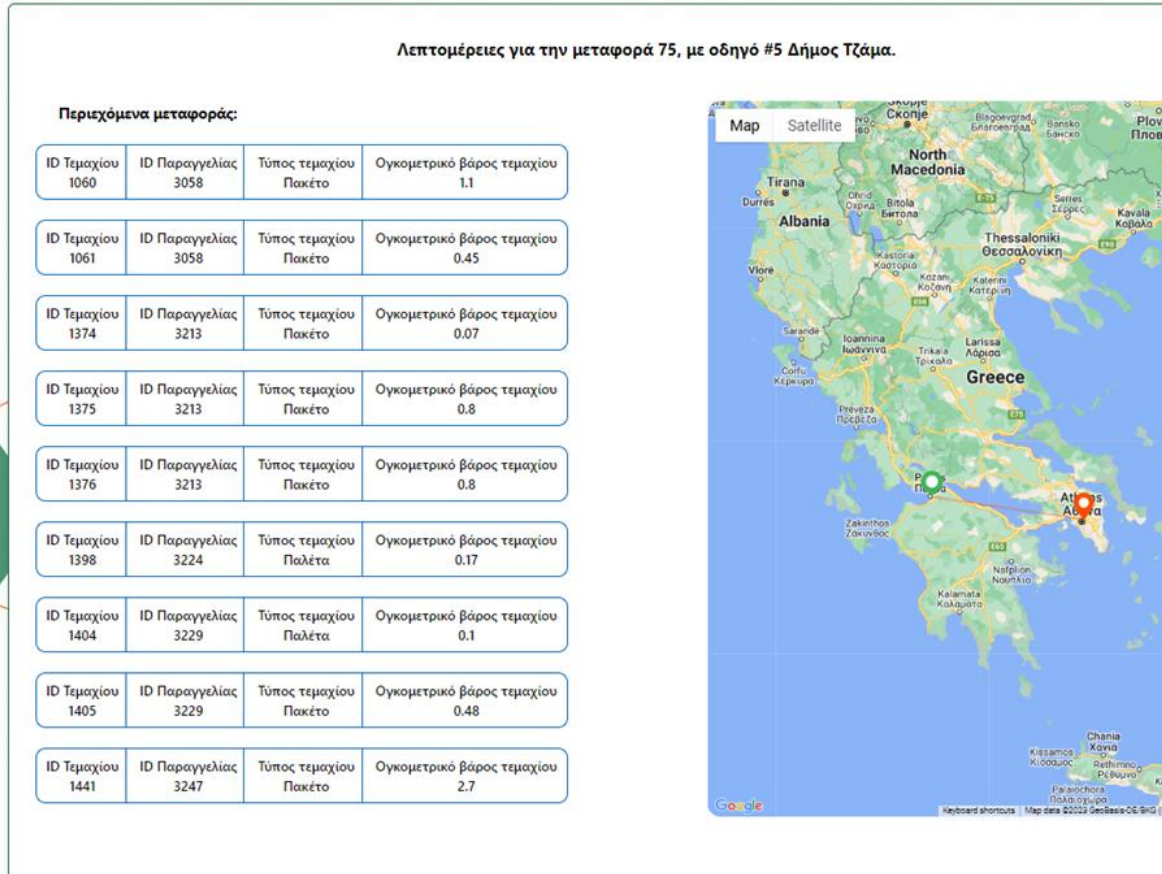
Μεταφορές που αναχωρούν από τον κόμβο: Κέρκυρα Κέντρο

ID Μεταφοράς	ID Οδηγού	Ποσότητα σε τεμάχια:	Συνολικό βάρος φορτίου:	Κόστος μεταφοράς:	Ημερομηνία έκδοσης:	Από:	Προς:
94	8	5	3.68	300	11/06/2023 12:55:41	Κέρκυρα Κέντρο	Πάτρα Κέντρο
95	9	12	7.87	300	11/06/2023 12:55:41	Κέρκυρα Κέντρο	Ιωάννινα Κέντρο
96	3	3	1.87	300	11/06/2023 12:55:41	Κέρκυρα Κέντρο	Κοζάνη Κέντρο
97	3	3	5.43	300	11/06/2023 12:55:41	Κέρκυρα Κέντρο	Πρέβεζα Κέντρο

Σχήμα 43: Οι αποδεκτές και μη αποδεκτές προτάσεις του αλγορίθμου

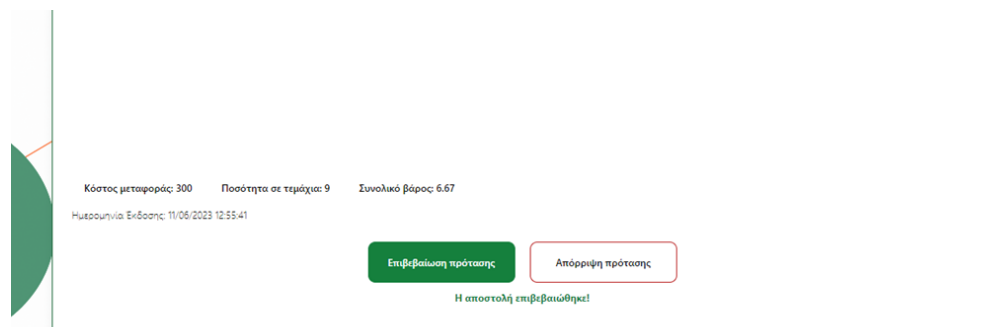
Με πράσινο είναι οι προτάσεις που είναι αποδεκτές ενώ με κόκκινο είναι οι μη αποδεκτές. Πατώντας σε μία από αυτές μπορεί κανείς να δει περισσότερες πληροφορίες για την μεταφορά καθώς και τα περιεχόμενα της. Όπως φαίνεται παρακάτω

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 44: Λεπτομέρειες μεταφορές

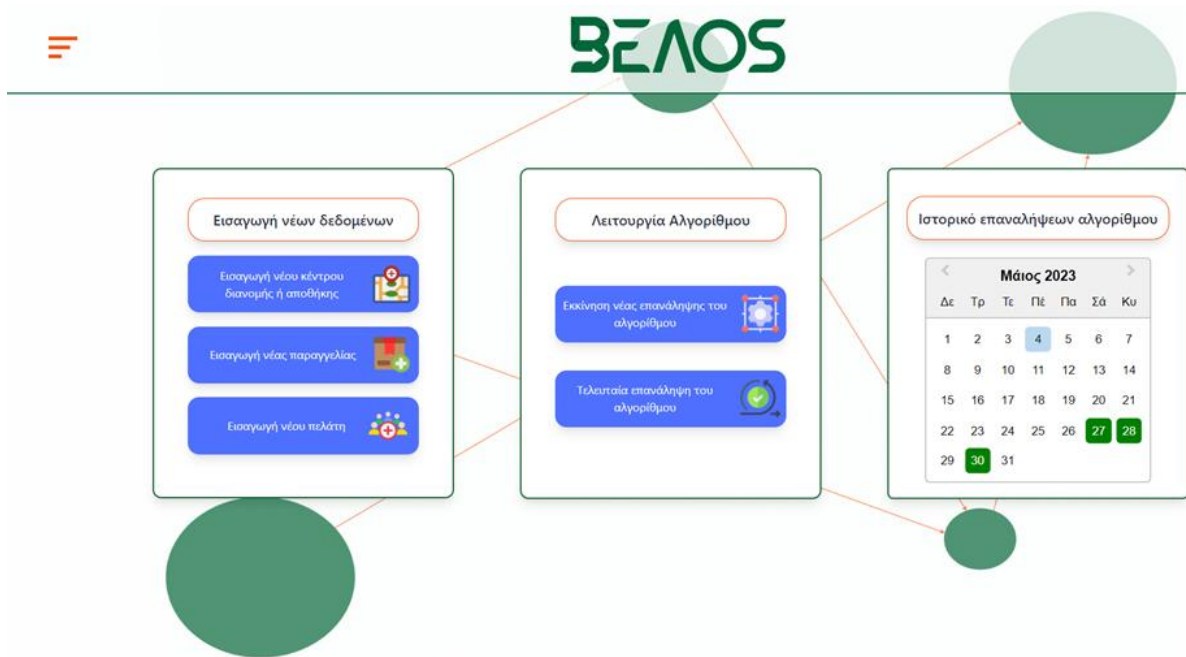
Στη συνέχεια ο χρήστης έχει την επιλογή να επιβεβαιώσει ή να απορρίψει την πρόταση του αλγορίθμου. Ενώ, αν η επιλογή αυτή έχει ήδη γίνει μπορεί απλά να δει λεπτομέρειες.



Σχήμα 45: Επιβεβαίωση ή απόρριψη πρότασης αλγορίθμου

3.8 UseCase 8: Εμφάνιση όλων των προηγούμενων εκτελέσεων του αλγορίθμου

Στο κεντρικό μενού, στα αριστερά υπάρχει ένα ημερολόγιο. Η λειτουργία αυτού είναι να δείχνει πότε έγιναν εκτελέσεις του αλγορίθμου και να δίνει την δυνατότητα στον χρήστη να ανατρέξει σε αυτές όταν και όποτε θέλει.



Σχήμα 46: Ημερολόγιο εκτέλεσης αλγορίθμου

Ο χρήστης μπορεί να πατήσει μία από τις ημερομηνίες που βλέπει με πράσινο και να έχει το επόμενο αποτέλεσμα

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Εκτέλεση αλγορίθμου για: 27/05/2023

ID Πακέτου/ Παλέτας	ID Μεταφοράς	Κατάσταση Παραγγελίας	Τρέχον κόμβος	Κόμβος έναρξης	Επόμενος κόμβος	Ημερομηνία εκτέλεσης	Έναρξη αποστολής	Λήξη
#29	#70	Καθ' οδόν	Πατρα Κέντρο	Πατρα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#47	#67	Προς παράδοση	Θεσσαλονικη Κέντρο	Πρέβεζα Κέντρο	Λάρισα Κέντρο	27/05/2023 13:22:09	---	
#48	#67	Προς παράδοση	Θεσσαλονικη Κέντρο	Πρέβεζα Κέντρο	Λάρισα Κέντρο	27/05/2023 13:22:09	---	
#50	#73	Προς παράδοση	Λάρισα Κέντρο	Κέρκυρα Κέντρο	Αθίνα Κέντρο	27/05/2023 13:22:09	---	
#57	#70	Καθ' οδόν	Πατρα Κέντρο	Πατρα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#58	#70	Καθ' οδόν	Πατρα Κέντρο	Πατρα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#59	#70	Καθ' οδόν	Πατρα Κέντρο	Πατρα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#76	#74	Καθ' οδόν	Πρέβεζα Κέντρο	Πρέβεζα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#77	#74	Καθ' οδόν	Πρέβεζα Κέντρο	Πρέβεζα Κέντρο	Θεσσαλονικη Κέντρο	27/05/2023 13:22:09	---	
#89	#64	Προς παράδοση	Αθίνα Κέντρο	Λάρισα Κέντρο	Κατερίνη Κέντρο	27/05/2023 13:22:09	---	
#90	#64	Προς παράδοση	Αθίνα Κέντρο	Λάρισα Κέντρο	Κατερίνη Κέντρο	27/05/2023 13:22:09	---	
#101	#65	Προς παράδοση	Θεσσαλονικη Κέντρο	Κατερίνη Κέντρο	Ιωάννινα Κέντρο	27/05/2023 13:22:09	---	
#102	#65	Προς παράδοση	Θεσσαλονικη Κέντρο	Κατερίνη Κέντρο	Ιωάννινα Κέντρο	27/05/2023 13:22:09	---	
#104	#73	Προς παράδοση	Λάρισα Κέντρο	Πατρα Κέντρο	Αθίνα Κέντρο	27/05/2023 13:22:09	---	

Σχήμα 47: Μετακινήσεις που έγιναν στην προηγούμενη εκτέλεση του αλγορίθμου

Εδώ ο χρήστης μπορεί να δει με πολύ μεγάλη λεπτομέρειες όλες τις κινήσεις που έγιναν και να ανατρέξει σε όλα τα δέματα ή μεταφορές. Με αυτόν το τρόπο ο χρήστης έχει πλήρη επίγνωση για το τι συμβαίνει στο σύστημα και στην πραγματική αλυσίδα εφοδιασμού

4. Υποσύστημα κινητής εφαρμογής

Στο πλαίσιο του έργου ΒΕΛΟΣ αναπτύχθηκε εφαρμογή για κινητά τηλέφωνα. Η εφαρμογή αφορά τους couriers και τους οδηγούς φορτηγών και στοχεύει στη βελτίωση της διαχείρισης των δρομολογίων τους παρέχοντάς τους τη δυνατότητα να παρακολουθούν τις διαδρομές τους σε πραγματικό χρόνο. Μέσω της εφαρμογής, οι χρήστες της έχουν πρόσβαση σε λεπτομερείς πληροφορίες σχετικά με τα δέματα και τα πακέτα που πρέπει να παραδώσουν και μπορούν να αλληλοεπιδράσουν με το πληροφοριακό σύστημα που έχει αναπτυχθεί, ενημερώνοντας το σχετικά με την εξέλιξη της παράδοσης κάθε δέματος ή πακέτου.

4.1 Μεθοδολογία ανάπτυξης

Ανάλυση Απαιτήσεων

Στο στάδιο της Ανάλυσης Απαιτήσεων, πραγματοποιήθηκε λεπτομερείς εξέταση των αναγκών και απαιτήσεων που πρέπει να καλύψει η εφαρμογή. Αποτέλεσε το θεμέλιο για την επιτυχή ανάπτυξη της. Οι κύριες διαδικασίες που ακολουθήθηκαν είναι οι εξής:

- **Προσδιορισμός των Χρηστών:** Καθορίστηκαν οι διάφοροι χρήστες της εφαρμογής και έγινε ανάλυση της χρήσης της εφαρμογής για τον καθένα από αυτούς. Συγκεκριμένα στην δική μας εφαρμογή οι χρήστες είναι δύο, ο courier και ο οδηγός φορτηγού.
- **Περιγραφή των Λειτουργιών:** Καθορίστηκαν οι βασικές λειτουργίες που θα προσφέρει η εφαρμογή. Αυτό συμπεριλαμβάνει τις διάφορες λειτουργίες που οι χρήστες θα μπορούν να εκτελούν, όπως επεξεργασία δεδομένων, αναζήτηση, αλληλεπίδραση με το σύστημα κ.λπ.
- **Καταγραφή Απαιτήσεων:** Καταγράψαμε τις απαιτήσεις της εφαρμογής.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

- **Καθορισμός Στόχων:** Καθορίσαμε τους στόχους της εφαρμογής, δηλαδή τι πρέπει να επιτύχει για τους χρήστες της. Οι στόχοι αυτοί σχετίζονται με την απόδοση, την ασφάλεια, την ευχρηστία και άλλες διαφορετικές πτυχές.

Σχεδιασμός Εφαρμογής

Στο στάδιο του Σχεδιασμού, επικεντρωθήκαμε στην ανάπτυξη της διεπαφής του χρήστη (UI) και της λειτουργικότητας της εφαρμογής. Στο πλαίσιο αυτής της διαδικασίας, ακολουθήθηκαν τα παρακάτω βήματα:

- **Δημιουργία Wireframes:** Τα wireframes είναι απλά σκίτσα ή διαγράμματα που παρουσιάζουν τη δομή της διεπαφής χρήστη, χωρίς να περιλαμβάνουν λεπτομέρειες. Σκοπός τους είναι να καθορίσουν τη διάταξη των στοιχείων και τη ροή της εφαρμογής.
- **Δημιουργία Mockups:** Τα mockups είναι πιο λεπτομερείς από τα wireframes και περιλαμβάνουν πραγματικές γραφικές αναπαραστάσεις των στοιχείων της διεπαφής. Περιλαμβάνουν τον σχεδιασμό των κουμπιών, των γραφικών στοιχείων και την γενική εμφάνιση της εφαρμογής.
- **Δημιουργία Προτύπων Σχεδίασης:** Τα πρότυπα σχεδίασης (design templates) αναπαριστούν το συνολικό στίλ και τη γραφική ταυτότητα της εφαρμογής. Συμπεριλαμβάνουν το χρώμα, τα γραφικά, τα εικονίδια και άλλα γραφικά στοιχεία που θα χρησιμοποιούνται στη διεπαφή.

Η εφαρμογή που αναπτύχθηκε στο πλαίσιο του Έργου ΒΕΛΟΣ υλοποιήθηκε με την χρήση του εργαλείου **Adobe XD**. Ο κύριος στόχος ήταν η δημιουργία μιας διεπαφής που θα ήταν εξαιρετικά φιλική προς τον χρήστη, απλή στη χρήση, και χωρίς περιττή πληροφορία.

Για τον σκοπό αυτό, επιλέχθηκαν φωτεινά χρώματα με αρκετή αντίθεση, προκειμένου να διασφαλιστεί η ευανάγνωστη παρουσίαση του περιεχομένου.

Επίσης, χρησιμοποιήθηκαν εικονίδια για να βελτιωθεί η κατανόηση του περιεχομένου. Παράλληλα, εφαρμόστηκαν διάφορα στοιχεία όπως dropdown, popups, modals κ.λπ. σε κατάλληλα σημεία, με σκοπό να παρουσιάζονται μόνο τα απολύτως απαραίτητα στοιχεία στις οθόνες της εφαρμογής. Αυτό εξασφαλίζει την αποφυγή της σύγχυσης και την ευκολότερη πλοήγηση για τους χρήστες.

Ανάπτυξη (Development)

Στο στάδιο της Ανάπτυξης, υλοποιήθηκε η ανάπτυξη της εφαρμογής βασιζόμενοι στον σχεδιασμό και τις αναλύσεις που πραγματοποιήθηκαν στα προηγούμενα στάδια και περιλαμβάνει τα εξής βήματα:

- **Κωδικοποίηση (Coding):** Αποτελεί την κωδικοποίηση της εφαρμογής, δημιουργώντας τον πηγαίο κώδικα που θα χρησιμοποιηθεί για τη λειτουργία της. Ο κώδικας της εν λόγω εφαρμογής έχει γραφτεί με χρήση της γλώσσας προγραμματισμού Javascript και πιο συγκεκριμένα του framework React Native για το frontend ενώ για το backend έγινε χρήση της γλώσσας PHP για τη δημιουργία των απαραίτητων APIs.
- **Δοκιμές (Testing):** Καθώς ο κώδικας αναπτύσσεται, πραγματοποιούνται δοκιμές για την εντόπιση πιθανών σφαλμάτων και προβλημάτων απόδοσης. Αυτό περιλαμβάνει τόσο αυτοματοποιημένες δοκιμές όσο και δοκιμές από ανθρώπους.
- **Ασφάλεια και Έλεγχος Ποιότητας:** Κατά την ανάπτυξη, ληφθήκαν τα κατάλληλα μέτρα για την ασφάλεια της εφαρμογής και τον έλεγχο της ποιότητας του κώδικα.

Επαναληπτικές Διορθώσεις

Το στάδιο αυτό αποτέλεσε κρίσιμο μέρος της διαδικασίας ανάπτυξης της εφαρμογής. Κατά τη διάρκεια αυτού του σταδίου, εξετάστηκαν τα αποτελέσματα

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

των δοκιμών και έγιναν όπου κρίθηκε απαραίτητο αναθεωρήσεις και διορθώσεις στον κώδικα και τον σχεδιασμό της εφαρμογής. Αυτό το στάδιο περιλαμβάνει τα παρακάτω βήματα:

- **Ανάλυση Αποτελεσμάτων Δοκιμών:** Πραγματοποιήθηκαν δοκιμές της εφαρμογής για τον έλεγχο της απόδοσης, της λειτουργικότητας, και της ασφάλειας. Τα αποτελέσματα αυτών των δοκιμών αξιολογήθηκαν προσεκτικά.
- **Εντοπισμός Σφαλμάτων (Bug Tracking):** Εντοπίστηκαν και καταγράφηκαν τυχόν σφάλματα, προβλήματα ασφαλείας ή ανεπιθύμητη συμπεριφορά στην εφαρμογή.
- **Διόρθωση Σφαλμάτων (Bug Fixing):** Ο κώδικας της εφαρμογής διορθώθηκε για να αντιμετωπιστούν τα εντοπισμένα σφάλματα.
- **Βελτιστοποίηση Απόδοσης (Performance Optimization):** Πραγματοποιήθηκαν βελτιστοποιήσεις στον κώδικα για να βελτιωθεί η ταχύτητα της εφαρμογής.
- **Διορθώσεις στον Σχεδιασμό (Design Revisions):** Πραγματοποιήθηκαν επιπλέον διορθώσεις ώστε να βελτιωθεί / διασφαλιστεί η ομαλή εμπειρία των χρηστών.
- **Επαναληπτικές δοκιμές (Retesting):** Οι δοκιμές επαναλήφθηκαν για να επιβεβαιωθεί ότι οι διορθώσεις λειτουργούν σωστά και δεν προκαλούν νέα προβλήματα.

Έκδοση και Κυκλοφορία

Το στάδιο αυτό αποτελεί τη φάση όπου η εφαρμογή θεωρείται έτοιμη για διάθεση. Περιλαμβάνει τα παρακάτω βήματα:

- **Προετοιμασία για Κυκλοφορία:** Πριν από την κυκλοφορία της εφαρμογής, είναι απαραίτητο να διασφαλιστεί ότι όλες οι λειτουργίες λειτουργούν σωστά και ότι τα προβλήματα απόδοσης έχουν επιλυθεί.

- **Υποβολή σε Καταστήματα Εφαρμογών:** Η εφαρμογή υποβάλλεται στα καταστήματα εφαρμογών, όπως το App Store για το iOS και το Google Play για το Android. Κατά τη διαδικασία αυτή, πρέπει να παρέχονται όλες τις απαραίτητες πληροφορίες και προϋποθέσεις της καθεμίας πλατφόρμας.
- **Ενημέρωση και Αναβαθμίσεις:** Με την πάροδο του χρόνου, μπορεί να απαιτούνται αναβαθμίσεις και ενημερώσεις για την εφαρμογή, βασισμένες στα αιτήματα και τις ανάγκες των χρηστών.

4.2 Εργαλεία ανάπτυξης

Για τη δημιουργία της παρούσας εφαρμογής έγινε χρήση διάφορων εργαλείων και τεχνολογιών προκειμένου να επιτευχθεί το επιθυμητό αποτέλεσμα. Κύριος στόχος ήταν η δημιουργία μιας καθαρής, φιλικής προς στο χρήστη και γρήγορης εφαρμογής, ενώ παράλληλα να είναι εύκολα επεκτάσιμη και συντηρήσιμη. Έτσι οι κύριες τεχνολογίες που επιλέχθηκαν είναι το Adobe XD για τη δημιουργία των γραφικών στοιχείων και των μακετών που θα μεταφραστούν σε κώδικα, ενώ για την κωδικοποίηση χρησιμοποιήθηκε η React Native για το front end και η PHP για το back end. Παρακάτω βλέπουμε πιο αναλυτικά τα συγκεκριμένα εργαλεία.

Adobe XD

Για τη δημιουργία γραφικών στοιχείων και μακετών, επιλέξαμε το Adobe XD. Αυτό το εργαλείο επιτρέπει τον σχεδιασμό της διεπαφής του χρήστη με μεγάλη ακρίβεια και λεπτομέρεια. Οι λόγοι που καθιστούν το Adobe XD απαραίτητο στην διαδικασία ανάπτυξης περιλαμβάνουν:

- **Σχεδιασμός Διεπαφής Χρήστη:** Το Adobe XD παρέχει ένα ευέλικτο περιβάλλον σχεδιασμού που επιτρέπει στους σχεδιαστές να δημιουργούν τη διεπαφή χρήστη με ευκολία. Αυτό συμπεριλαμβάνει τη δημιουργία και τον

συγχρονισμό γραφικών στοιχείων, τον καθορισμό της δομής της σελίδας και την προεπισκόπηση του πώς θα αντιδρά η διεπαφή σε διάφορες συσκευές.

- **Οπτικοποίηση** : Μας βοηθά να δούμε πώς θα φαίνεται η εφαρμογή σας πριν από την πραγματική ανάπτυξη. Αυτό επιτρέπει τον προσδιορισμό και την αντιμετώπιση οποιωνδήποτε ανεπιθύμητων ή μη λειτουργικών στοιχείων στη διεπαφή πριν από την κατασκευή της.
- **Εύκολη Μετάφραση σε Κώδικα**: Οι σχεδιασμοί που δημιουργούνται στο Adobe XD μπορούν να μεταφραστούν εύκολα σε κώδικα που μπορεί να χρησιμοποιηθεί για την ανάπτυξη της εφαρμογής. Αυτή η δυνατότητα εξοικονομεί χρόνο και διασφαλίζει την συνοχή μεταξύ του σχεδιασμού και του τελικού προϊόντος.

React Native

Στον τομέα της ανάπτυξης του front end, υιοθετήθηκε η React Native, η οποία είναι ένα πλαίσιο ανάπτυξης που ξεχωρίζει για την ικανότητά του να δημιουργεί εφαρμογές κινητών ανεξαρτήτως λειτουργικού συστήματος, προσφέροντας μια εμπειρία παρόμοια με αυτήν που προσφέρουν οι φυσικές εφαρμογές. Η επιλογή της React Native συνέβαλε σημαντικά στην ανάπτυξη της εφαρμογής με τους εξής τρόπους:

- **Ανεξαρτησία πλατφόρμας**: Ένα από τα κύρια πλεονεκτήματα της React Native είναι η δυνατότητά της να δημιουργεί εφαρμογές που λειτουργούν τόσο σε συσκευές iOS όσο και Android. Αυτό εξασφαλίζει ότι οι χρήστες οποιουδήποτε λειτουργικού συστήματος μπορούν να απολαμβάνουν την εφαρμογή, ενισχύοντας την προσβασιμότητά της.
- **Φυσική Εμπειρία Χρήστη**: Χρήση γνωστών γλωσσών προγραμματισμού, όπως η JavaScript, για τη δημιουργία εφαρμογών. Αυτό επιτρέπει στους προγραμματιστές να αναπτύσσουν εφαρμογές με μια φυσική εμπειρία

χρήστη, επιτρέποντας τη χρήση γνωστών στοιχείων και λειτουργιών που χρησιμοποιούνται συχνά σε εφαρμογές.

- **Αποδοτικότητα:** Ο κώδικας που γράφεται σε React Native μπορεί να επαναχρησιμοποιηθεί, μειώνοντας τον χρόνο ανάπτυξης και τη συντήρηση. Αυτό είναι σημαντικό για την διατήρηση της εφαρμογής και την επέκτασή της στο μέλλον.

Expo

Το Expo είναι ένα πλαίσιο ανοικτού κώδικα που επιτρέπει την ανάπτυξη εφαρμογών που λειτουργούν απροβλημάτιστα σε Android και iOS. Συνδυάζει τα πλεονεκτήματα των κινητών και των διαδικτυακών τεχνολογιών, προσφέροντας ένα στιβαρό υπόβαθρο για τη δημιουργία και την επέκταση εφαρμογών.

- **Ανάπτυξη:** Η δημιουργία μιας εφαρμογής με το Expo απαιτεί την χρήση JavaScript ή TypeScript για τον κώδικα. Παρέχει μια τυπική δομή έργου που μπορεί να προσαρμοστεί σύμφωνα με τις εκάστοτε ανάγκες. Επιπλέον, παρέχει έτοιμα πακέτα ανάπτυξης που μειώνουν το χρόνο ανάπτυξης και επιτρέπουν ευελιξία παρόμοια με αυτήν της διαδικτυακής ανάπτυξης, διασφαλίζοντας παράλληλα την σταθερότητα.
- **Πλοήγηση:** Η δημιουργία εφαρμογών που περιλαμβάνουν πολύπλοκη πλοήγηση μπορεί να είναι δύσκολη, περιλαμβάνοντας τη δημιουργία διάφορων οθονών, τη σύνδεσή τους με μοναδικές διευθύνσεις URL, τη μετάβαση ανάμεσα στις οθόνες και την προβολή στοιχείων διεπαφής χρήστη που σχετίζονται με την πλοήγηση. Το Expo Router απλοποιεί αυτήν τη διαδικασία, αυτοματοποιώντας τη μετατροπή κάθε αρχείου στον κατάλογο της εφαρμογής σας σε διαδρομή και προσφέροντας αυτόματο βαθύ σύνδεσμο.
- **Κυκλοφορία:** Το Expo παρέχει μια αυτοματοποιημένη σχεδόν υπηρεσία κατασκευής που διευκολύνει τη διαδικασία τόσο για τα Android όσο και για

τα iOS. Επίσης, παρέχει εργαλεία για την αυτοματοποίηση της αποστολής και της υποβολής των δυαδικών αρχείων της εφαρμογής στα καταστήματα εφαρμογών. Επιπλέον, απλοποιεί τη διόρθωση σφαλμάτων και διευκολύνει την ταχεία αναβάθμιση της εφαρμογής ανάμεσα στις υποβολές στα καταστήματα εφαρμογών, εξασφαλίζοντας μια ομαλή και ευέλικτη διαδικασία κυκλοφορίας.

Παράλληλα με το Expro, χρησιμοποιήθηκε επίσης το Expro Go, μια δωρεάν εφαρμογή που είναι διαθέσιμη τόσο στο Android όσο και στο iOS μέσω των αντίστοιχων καταστημάτων εφαρμογών. Μέσω αυτής της εφαρμογής, μπορούμε να δοκιμάσουμε το έργο μας χωρίς την ανάγκη να εγκαταστήσουμε επιπλέον προσομοιωτές στη συσκευή μας. Αντίθετα, μπορούμε να δούμε και να χρησιμοποιήσουμε την εφαρμογή ή το έργο μας απευθείας μέσω του κινητού μας τηλεφώνου. Αυτό διευκολύνει τον έλεγχο της εφαρμογής και τη δοκιμή της λειτουργικότητας της, επιτρέποντάς μας να αξιολογήσουμε την απόδοση της στην πραγματική συσκευή στην οποία θα εκτελείται, χωρίς την ανάγκη για πολύπλοκες εγκαταστάσεις. Η σύνδεση του έργου με το κινητό τηλέφωνο γίνεται μέσω LAN και χρήση QR code.

4.3 Λειτουργικότητα κινητής εφαρμογής

Σε αυτήν την ενότητα, θα εξετάσουμε τις λειτουργίες της εφαρμογής ανά οθόνη όπως ακριβώς τις βλέπει ο κάθε χρήστης.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 48: Οθόνη σύνδεσης χρήστη

Η παραπάνω οθόνη είναι αυτή που βλέπει ο χρήστης όταν ανοίξει την εφαρμογή. Αποτελείται από μία απλή φόρμα δύο πεδίων όπου ο χρήστης συμπληρώνει το όνομα χρήστη και τον κωδικό αντίστοιχα που του έχει δοθεί από τον διαχειριστή του συστήματος. Σε περίπτωση λάθους εισαγωγής στοιχείων προβάλλεται και το αντίστοιχο μήνυμα.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Σχήμα 49: Πρόγραμμα μεταφοράς οδηγού φορτηγού

Στην παραπάνω οθόνη μπορούμε να δούμε το πρόγραμμα μεταφορών ενός οδηγού φορτηγού όπως αυτό εμφανίζεται. Οι πληροφορίες που αναγράφονται είναι ο προορισμός, η συνολική ποσότητα των δεμάτων και η ημερομηνία δημιουργίας του δρομολογίου. Κάτω από τις πληροφορίες της μεταφοράς βρίσκεται και το κουμπί εκκίνησης/ τερματισμού του δρομολογίου, το οποίο πατώντας το εκτελούνται δύο διεργασίες. Η πρώτη εκτελείται στο παρασκήνιο και είναι αυτή της ενημέρωσης της βάσης δεδομένων και του διαχειριστικού συστήματος ότι ο οδηγός έχει εκκινήσει / τερματίσει το συγκεκριμένο δρομολόγιο, η δεύτερη είναι της ανανέωσης της

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

οθόνης του χρήστη με τη επόμενη στη λίστα μεταφορά (όταν αυτός πατήσει τερματισμό).

Στο κάτω μέρος της οθόνης βλέπουμε το σύνολο των δεμάτων που περιέχονται στη μεταφορά, τα οποία εμφανίζονται σε μορφή scrollable λίστας.

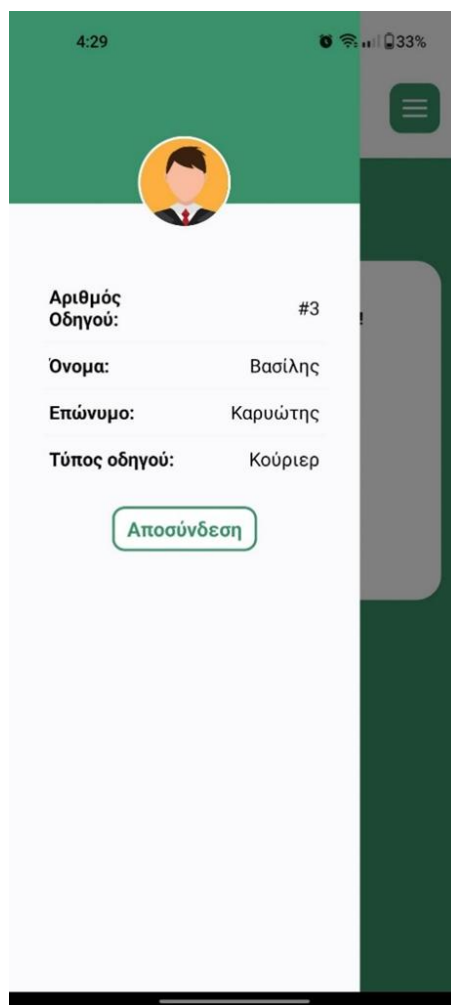


Σχήμα 50: Λεπτομέρειες πακέτου μεταφοράς

Αυτή η οθόνη είναι στην ουσία συνέχεια της προηγούμενης καθώς εμφανίζεται στον χρήστη όταν αυτός επιλέξει ένα πακέτο από τη λίστα με τα δέματα. Σε αυτή την οθόνη αναγράφονται όλα τα απαραίτητα χαρακτηριστικά του δέματος όπως ο

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

κωδικός του, οι διαστάσεις, το βάρος του καθώς και το είδος του (εύθραυστο, επικίνδυνο, ευάλωτο). Το εικονίδιο επάνω αριστερά (κίτρινο βέλος) πατώντας το ο χρήστης μεταφέρεται ξανά στην προηγούμενη οθόνη.

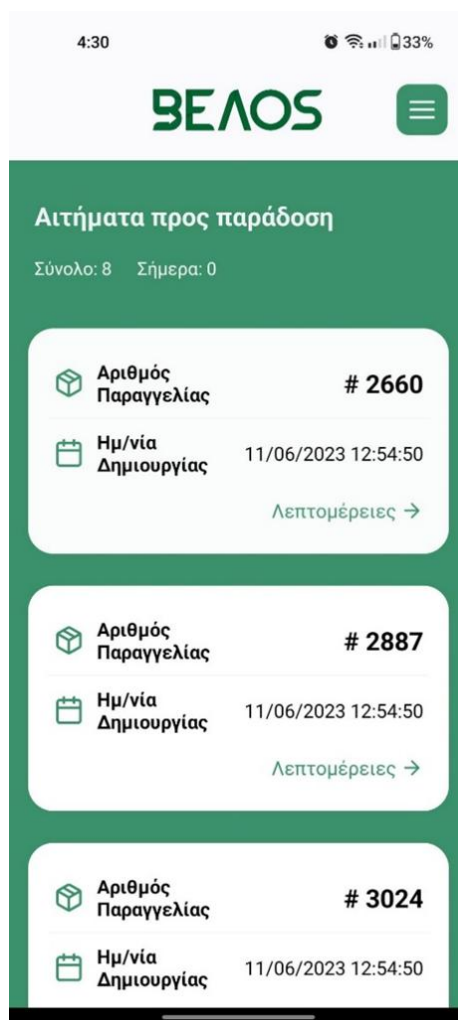


Σχήμα 51: Μενού χρήστη

Πατώντας το πράσινο εικονίδιο πάνω αριστερά σε οποιαδήποτε οθόνη εμφανίζεται το μενού χρήστη. Εκεί ο χρήστης μπορεί να δει τις λεπτομέρειες του λογαριασμού του καθώς και το κουμπί αποσύνδεσης με το οποίο μεταφέρεται στην αρχική οθόνη σύνδεσης.

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Η παρακάτω οθόνη είναι η πρώτη που βλέπει ο courier αφού συνδεθεί. Σε αυτή μπορεί να βρει τη λίστα με όλα τα αιτήματα προς παράδοση. Σε κάθε καρτέλα αναγράφεται ο αριθμός παραγγελίας και η ημερομηνία δημιουργίας, ενώ πιο πάνω φαίνεται ο συνολικός αριθμός των παραγγελιών καθώς επίσης και των ημερήσιων παραγγελιών.

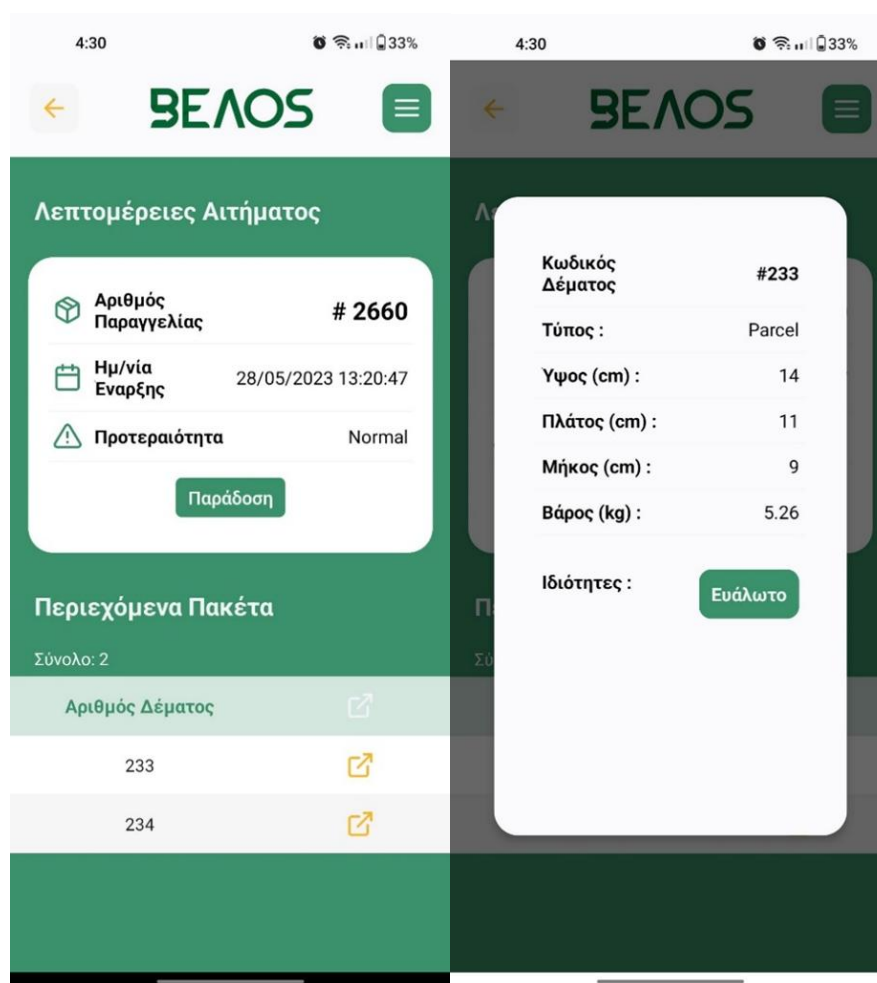


Σχήμα 52: Λίστα αιτημάτων για παράδοση

Επιλέγοντας μία από τις καρτέλες αιτημάτων της προηγούμενης οθόνης, προβάλλονται στον χρήστη οι λεπτομέρειες του αιτήματος. Στο πάνω μέρος της οθόνης βρίσκεται η καρτέλα με τα απαραίτητα στοιχεία του αιτήματος, καθώς

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

επίσης και το κουμπί της παράδοσης. Πατώντας το, ενημερώνεται η βάση δεδομένων και κατ' επέκταση το διαχειριστικό σύστημα σχετικά με την παράδοση, ενώ παράλληλα μεταφέρει τον χρήστη στην πλέον ανανεωμένη οθόνη με τη λίστα των αιτημάτων.



Σχήμα 53: Λεπτομέρειες αιτήματος προς παράδοση - Χαρακτηριστικά πακέτου

Στο κάτω μέρος της οθόνης βρίσκεται η κυλιόμενη λίστα με τα πακέτα που περιέχει το συγκεκριμένο αίτημα. Αν ο χρήστης πατήσει σε κάποιο από αυτά τότε

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

εμφανίζεται ένα modal στην οθόνη του με τα απαραίτητα στοιχεία του πακέτου. Πατώντας οπουδήποτε στην οθόνη επιστρέφει στην προηγούμενη.

Τέλος στη σπάνια περίπτωση που ο χρήστης δεν έχει κάποιο διαθέσιμο δρομολόγιο για την τρέχουσα ημέρα η πρώτη οθόνη που βλέπει αφού συνδεθεί είναι η παρακάτω που τον ενημερώνει σχετικά.



Σχήμα 54: Μήνυμα έλλειψης δρομολογίων

4. Σύνοψη - Συμπεράσματα

Το παραδοτέο παρουσίασε τη λειτουργικότητα και τη μεθοδολογία ανάπτυξης των βασικών υποσυστημάτων που υλοποιήθηκαν στο πλαίσιο του έργου, ήτοι του υποσυστήματος της διαδικτυακής πλατφόρμας, του κινητού υποσυστήματος και του υποσυστήματος χρονοπρογραμματισμού και δρομολόγησης το οποίο εφάρμοσε τον αλγόριθμο οπισθοπίεσης. Όλα τα υποσυστήματα εγκαταστάθηκαν σε server σε φορέα που συμμετέχει το έργο. Η RMR, ως βασικός φορέας – χρήστης, συμμετείχε ενεργά κατά τη διάρκεια υλοποίησης των υποσυστημάτων παρέχοντας ουσιαστικές προτάσεις και διορθώσεις επί της λειτουργικότητας του πληροφοριακού συστήματος.

Το πληροφοριακό σύστημα ελέγχθηκε τόσο σε τεχνικό όσο και σε επιχειρηματικό επίπεδο. Τα αποτελέσματα του τεχνικού ελέγχου αναφορικά με τους τρόπους ολοκλήρωσης των επιμέρους υποσυστημάτων αποτυπώνονται στο παραδοτέο 3.2. Τα αποτελέσματα του επιχειρηματικού ελέγχου αποτυπώνονται στο παραδοτέο 4.3.



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Αναφορές

- [1] <https://velos.di.ionio.gr/>, τελευταία ανάκτηση: 28/07/2023, 12:32