

Εφαρμογή αναζήτησης/κρατήσεων ακτοπλοϊκών εισιτηρίων

Ferry DB

Αλέξανδρος Ματινόπουλος Λόπεθ

AM: 1059295, Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών,
up1059295@upnet.gr

Μητακίδης Ανέστης

AM: 1059517, Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών,
up1059517@upnet.gr

https://github.com/L4Limbo/ferry_db

1 ΠΕΡΙΛΗΨΗ

Στόχος της εργασίας ήταν να δημιουργηθεί η βάση δεδομένων μίας εφαρμογής αναζήτησης και κράτησης ακτοπλοϊκών εισιτηρίων. Προκειμένου να παρουσιαστεί η χρήση της βάσης δεδομένων στην αναφερόμενη εφαρμογή υλοποιήθηκε και ένα CLI το οποίο λειτουργεί ως η γραφική διεπαφή του χρήστη με την εφαρμογή. Επίσης δημιουργήθηκε και ένα CLI το οποίο προορίζεται για την χρήση της εφαρμογής από κάποιον διαχειριστή της. Για να διευκολυνθεί ο προγραμματιστής στη δοκιμή των λειτουργιών και στην ανανέωση και επέκταση τόσο της δομής της βάσης δεδομένων όσο και των δεδομένων που δέχεται έχουν δημιουργηθεί λειτουργίες migration και seeding, οι οποίες δημιουργούν τους πίνακες της βάσης και προσθέτουν κατασκευασμένα δεδομένα σε αυτή, αντίστοιχα. Σημαντικό μέρος της υλοποίησης της βάσης είναι η μείωση των αποθηκευμένων στάσεων των ακτοπλοϊκών γραμμών μέσω τρόπου που αντιγράφει τις διασυνδεδεμένες λίστες (linked-lists) καθώς και ο τρόπος υπολογισμού των διαθέσιμων επιβατών σε κάθε διαδρομή κάθε δρομολογίου αντίστοιχα. Ενδιαφέρον κομμάτι της εφαρμογής είναι ότι ο χρήστης μπορεί να επιλέξει διάφορους τύπους θέσεων, τύπους εισιτηρίων και οχημάτων κατά την κράτηση των εισιτηρίων του.

2 ΜΕΘΟΔΟΛΟΓΙΑ

Αρχικά έγινε ανάλυση του μικρόκοσμου της εφαρμογής και μελέτη ήδη υπαρχόντων εφαρμογών, όπως το ferryscanner, που λύνουν το ίδιο πρόβλημα και ορισμός των προδιαγραφών και των παραδοχών της λύσης που πρέπει να εφαρμοστεί στο πρόβλημα. Στην συνέχεια σχεδιάστηκε ο μικρόκοσμος του προβλήματος και υλοποιήθηκε το Entity Relational Diagram (ERD) μέσω την εφαρμογής ERDMaker του προβλήματος στο οποίο εφαρμόστηκαν βελτιώσεις και αλλαγές μέχρι να προκύψει η τελική του μορφή στην οποία λαμβάνονται υπόψη όλες οι προδιαγραφές και οι παραδοχές που έγιναν.

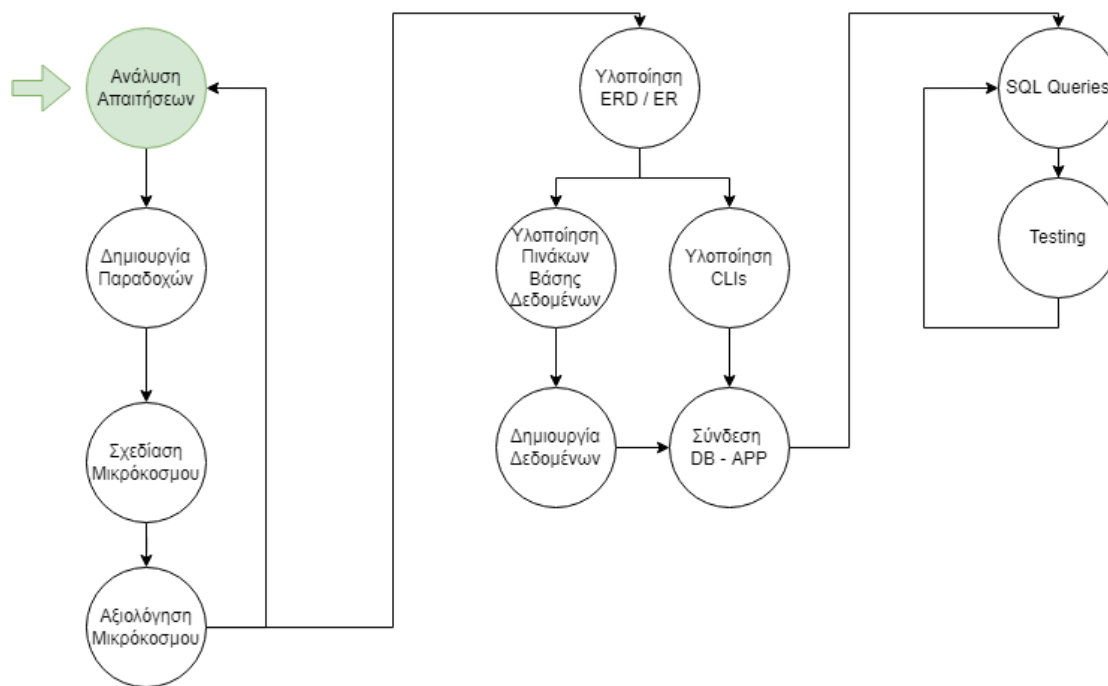
Μετά την υλοποίηση του ERD έγινε η μετατροπή του στο Relational Schema της βάσης μέσω της εφαρμογής DB DESIGNER.

Με το πέρας της ενδιάμεσης παρουσίασης της εργασίας έγινε επαναπροσδιορισμός κάποιων εκ των απαιτήσεων της λύσης για να ολοκληρωθεί η σχεδίαση του μικρόκοσμου της εφαρμογής. Στην συνέχεια, αποφασίστηκαν οι τεχνολογίες και τα εργαλεία που θα χρησιμοποιηθούν για την υλοποίησή της. Η βάση δεδομένων αποφασίστηκε να υλοποιηθεί σε SQLite, ενώ η εφαρμογή και οι λειτουργίες που αξιοποιούν την βάση δεδομένων επιλέχθηκε να υλοποιηθούν μέσω της Python.

Το επόμενο βήμα ήταν η εύρεση και η δημιουργία δεδομένων τα οποία θα χρησιμοποιούνταν για την επίδειξη των λειτουργιών της βάσης και της εφαρμογής. Αποφασίστηκε τα δεδομένα που θα χρησιμοποιηθούν να αποτελούνται από όλα τα λιμάνια της Ελλάδας, ενώ τα υπόλοιπα θα είναι κατασκευασμένα δεδομένα με τιμές που προσπαθούν να αντιγράψουν τις πραγματικές.

Για διευκόλυνση και επιτάχυνση της υλοποίησης της εφαρμογής έγινε διαχωρισμός των εργασιών σε δημιουργία της βάσης δεδομένων και προσθήκη των κατασκευασμένων δεδομένων σε αυτή και σε υλοποίηση των γραφικών διεπαφών του χρήστη και του διαχειριστή της εφαρμογής.

Τέλος, μετά την υλοποίηση των παραπάνω έγινε ενοποίηση εφαρμογής-βάσης δεδομένων και υλοποίηση των SQL ερωτημάτων και βελτιώσεις σε όλη την λύση του προβλήματος.



Εικόνα 1: Μοντέλο Μεθοδολογίας

3 ΑΞΙΟΛΟΓΗΣΗ

Υπάρχουν διάφορα κριτήρια αξιολόγησης μίας βάσης δεδομένων. Αρχικό και κυριότερο κριτήριο είναι η λειτουργίες που μπορεί να υποστηρίξει και αν καλύπτει όλες τις απαιτήσεις που ορίστηκαν πριν το σχεδιασμό της και πρέπει να περιλαμβάνονται στον μικρόκοσμο τον οποίο αντιπροσωπεύει. Για τον σκοπό αυτό πραγματοποιήθηκαν διάφορες δοκιμές κατά τις οποίες έγινε έλεγχος της λειτουργίας δημιουργίας πινάκων, της προσθήκης κατασκευασμένων δεδομένων σε αυτούς και των συναλλαγών μεταξύ των CLIs και της βάσης (CRUD). Δόθηκε βάση αφενός στην υποστήριξη των λειτουργιών του μικρόκοσμου και την αποθήκευση των κατάλληλων δεδομένων, και αφετέρου στην μείωση του πλήθους των αποθηκευμένων δεδομένων που απαιτούνται για την υποστήριξη όλων των λειτουργιών. Επίσης αξιολογήθηκε κατά πόσο η εφαρμογή ανταποκρίνεται στις απαιτήσεις του προβλήματος το οποίο καλείται να επιλύσει, δηλαδή αν είναι σε θέση να εξυπηρετεί πιστά τον σκοπό της. Τέλος, δόθηκε βαρύτητα στο γεγονός ότι οι λειτουργίες της είναι εξειδικευμένες ως όπως το πρόβλημα, και επομένως είναι σε θέση να δώσει αποδοτική λύση ακριβώς μέσα στα όρια του θέματος που ανατέθηκε.

4 ΔΕΔΟΜΕΝΑ

Όσον αφορά στα δεδομένα που χρησιμοποιήθηκαν για την υλοποίηση και την επίδειξη λειτουργιών της εφαρμογής έγινε χρήση όλων των λιμανιών της Ελλάδας που βρίσκονται στο αρχείο https://ec.europa.eu/eurostat/cache/metadata/Annexes/mar_esms_an2.xlsx της ιστοσελίδας eurostat (<https://ec.europa.eu/eurostat/>). Επίσης έγινε δημιουργία ψεύτικων δρομολογίων με στοιχεία που προσπαθούν να αντιγράψουν την πραγματικότητα στα πεδία κόστους, επιβατών, διάρκειας ταξιδιού και δημιουργήθηκαν μέσω των βιβλιοθηκών random και faker της Python, ενώ για τα είδη θέσεων, οχημάτων και εισιτηρίων που υποστηρίζονται χρησιμοποιήθηκαν στοιχεία που βρέθηκαν σε εφαρμογές κράτησης ακτοπλοϊκών εισιτηρίων, και συγκεκριμένα στο ferryscanner.com, anek.gr και minoan.gr. Η δημιουργία των κατασκευασμένων δεδομένων υπάρχει στον φάκελο 'data_generation/' των αρχείων της εργασίας.

5 ΚΥΡΙΕΣ ΕΝΕΡΓΕΙΕΣ

5.1 ΕΝΕΡΓΕΙΕΣ

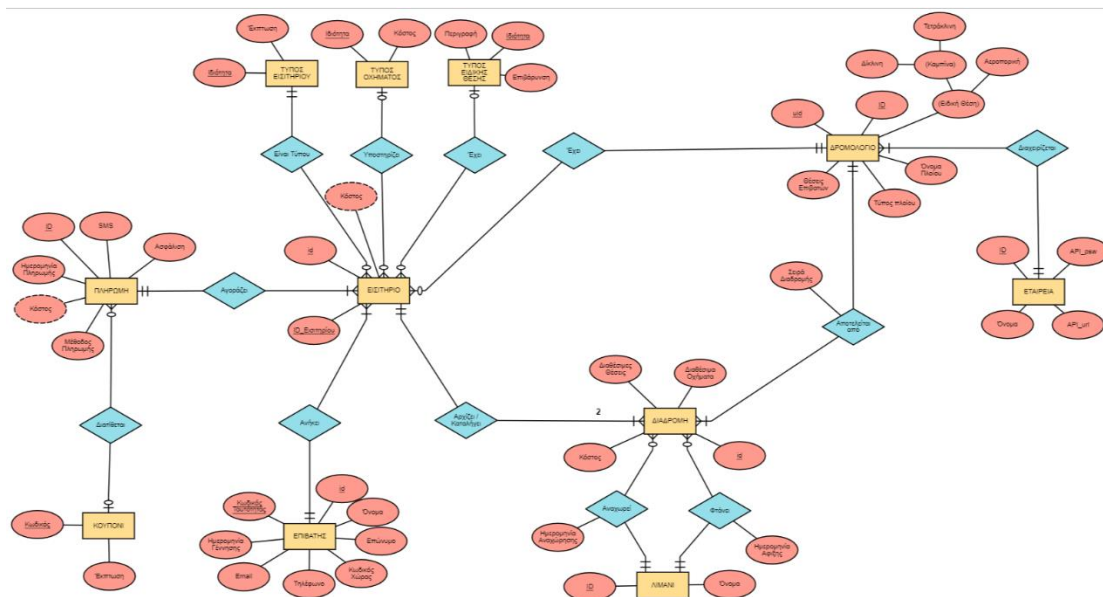
5.1.1 Σχεδίαση Μικρόκοσμου Προβλήματος

Η πρώτη ενέργεια μετά την ανάθεση του θέματος ήταν η ανάλυση του μικρόκοσμου του προβλήματος και ο καθορισμός των παραδοχών και των προδιαγραφών του. Πιο συγκεκριμένα, έγινε μελέτη και χρήση εφαρμογών και σελίδων που λύνουν το ίδιο πρόβλημα όπως ferryscanner.com, anek.gr και minoan.gr και ανασκόπηση των χαρακτηριστικών τους. Ακολούθησε ο ορισμός των παραδοχών του μικρόκοσμου έτσι ώστε να δημιουργηθούν οι τελικές απαιτήσεις της λύσης του προβλήματος.

Οι παραδοχές του μικρόκοσμου φαίνονται παρακάτω:

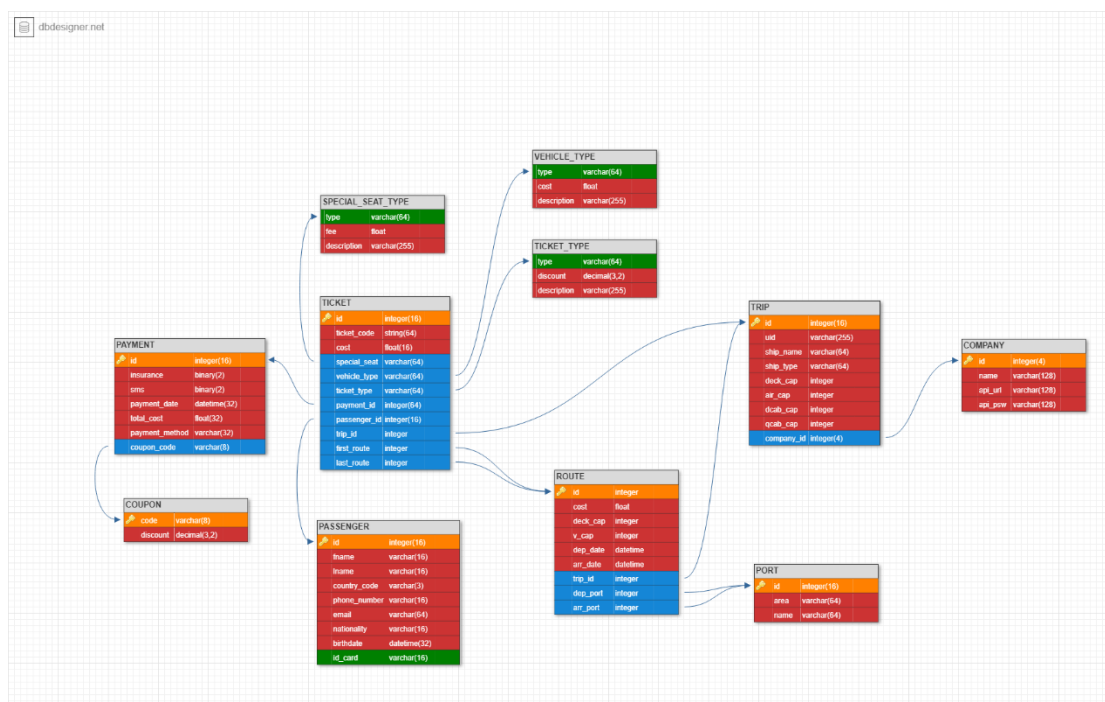
- Αναζήτηση δρομολογίων με βάση τις προτιμήσεις του χρήστη στον προγραμματισμό του ταξιδιού,
- Επιλογή μονής ή διπλής διαδρομής
- Επιλογή τύπου θέσης, εισιτηρίου και οχήματος με βάση την αντίστοιχη διαθεσιμότητα
- Συμπλήρωση στοιχείων επιβατών
- Πληρωμή πακέτου εισιτηρίων χωρίς την είσοδο χρήστη στην εφαρμογή και μέσω εφαρμογής τρίτου για λόγους ασφάλειας
- Υπολογισμός του συνολικού κόστους των εισιτηρίων και εφαρμογή εκπτώσεων/ επιβαρύνσεων με βάση τις επιλογές του επιβάτη
- Προσθήκη δρομολογίων εταιρειών μέσω της χρήσης API των ίδιων και δεδομένων κατάλληλης μορφής
- Δεν υπάρχει η δυνατότητα ακύρωσης πληρωμής/εισιτηρίων από τον χρήστη (Η συγκεκριμένη ενέργεια γίνεται μετά από επικοινωνία με την αρμόδια εταιρεία)
- Δυνατότητα διαγραφής πληρωμής μετά από αίτηση της αρμόδιας εταιρείας
- Δυνατότητα διαγραφής δρομολογίου μετά από αίτηση της αρμόδιας εταιρείας ή μετά την εκτέλεση του δρομολογίου
- Δυνατότητα αύξησης ή μείωσης των θέσεων επιβατών/ οχημάτων ενός δρομολογίου
- Οι ειδικές θέσεις (αριθμημένες, καμπίνες) δεν ελευθερώνονται σε ενδιάμεση στάση του δρομολογίου (λόγω ανάγκης απολύμανσης τους)
- Δυνατότητα ελευθέρωσης θέσεων καταστρώματος και οχημάτων στις ενδιάμεσες στάσεις ενός δρομολογίου

Λαμβάνοντας υπ' όψη όλα τα παραπάνω προέκυψε το ERD της βάσης δεδομένων όπως φαίνεται στην Εικόνα 2.



Εικόνα 2: Entity Relational Diagram

Στην συνέχεια έγινε μετατροπή του ERD στο Relational Schema της βάσης Δεδομένων όπως φαίνεται στην Εικόνα 3.



Εικόνα 3: Relational Schema

5.1.2 Δημιουργία Δεδομένων

Στο κομμάτι της εύρεσης και δημιουργίας των δεδομένων που χρησιμοποιήθηκαν έγινε αναζήτηση και ανάκτηση των λιμανιών που υπάρχουν στην Ελλάδα από την Eurostat (<https://ec.europa.eu/eurostat/>) από το αρχείο https://ec.europa.eu/eurostat/cache/metadata/Annexes/mar_esms_an2.xlsx. Στην συνέχεια μέσω των βιβλιοθηκών της Python pandas, random και faker δημιουργήθηκαν αρχεία csv με

ψεύτικα δεδομένα, επιβατών, πληρωμών, εισιτηρίων και εταιρειών, οι τιμές των οποίων αντιγράφουν στοιχεία της πραγματικότητας.

5.1.3 Υλοποίηση Βάσης Δεδομένων και Είσοδος Δεδομένων

Για την δημιουργία των πινάκων της βάσης δεδομένων δημιουργήθηκε μία μέθοδος migrate η οποία δημιουργεί κενούς τους πίνακες της βάσης και διαγράφει τυχόν παλιές εγγραφές έτσι ώστε να διευκολυνθούν οι προγραμματιστές κατά τις δοκιμές της εφαρμογής και της προσθήκης βελτιώσεων στη βάση.

Για την είσοδο των κατασκευασμένων δεδομένων δημιουργήθηκαν μέθοδοι Seeder κάθε μία από τις οποίες γεμίζει τον αντίστοιχο πίνακα με δεδομένα καθαρίζοντας πρώτα τα ήδη υπάρχοντα χωρίς όμως να ξαναδημιουργείται ο πίνακας όπως γίνεται στο migrate.

5.1.4 Υλοποίηση Διεπαφών

Η διεπαφή του χρήστη υλοποιήθηκε σε Python (έκδοση 3.8.10) με στόχο να μπορεί μέσω αυτής ο χρήστης να μπορεί να κλείσει ακτοπλοικά εισιτήρια προς τον προορισμό που επιθυμεί, με βάση κάποιες απαιτήσεις του και σε συνδυασμό με τις διάφορες δυνατότητες που του δίνονται, μέσω του command line της. Ο χρήστης εισάγει τις παραμέτρους της αναζήτησης των εισιτηρίων (Λιμάνια Αναχώρησης και Άφιξης, Ημερομηνία Αναχώρησης, Ημερομηνία Επιστροφής (εάν επιθυμεί το ταξίδι να είναι με επιστροφή), Αριθμό Επιβατών για τους οποίους επιθυμεί να κλείσει εισιτήρια και αριθμό οχημάτων). Αφού γίνει η απαραίτητη αναζήτηση στη βάση, επιλέγονται τα δρομολόγια που πληρούν τις απαραίτητες προϋποθέσεις. Από αυτά τα δρομολόγια ο χρήστης επιλέγει αυτό/αυτά που τον βολεύουν, και έπειτα του δίνεται η επιλογή να επιλέξει την κατηγορία του εισιτηρίου του, την κατηγορία του οχήματος του και να κλείσει ειδική θέση για τα δρομολόγια που επέλεξε, για κάθε επιβάτη και όχημα. Στη συνέχεια ο χρήστης χρειάζεται να εισάγει τα απαραίτητα προσωπικά στοιχεία του κάθε επιβάτη, και να επιλέξει αν επιθυμεί να εισάγει κάποιο εκπαιδευτικό κουπόνι, ή/και να αγοράσει κάποια επιπλέον υπηρεσία για τους επιβάτες, στις οποίες περιλαμβάνονται η ασφάλιση κατά την διάρκεια του ταξιδιού και η υπενθύμιση μέσω sms για το ταξίδι. Έπειτα ο χρήστης μπορεί να προχωρήσει στην πληρωμή των εισιτηρίων, επιλέγοντας μέθοδο πληρωμής, με βάση την οποία στη συνέχεια θα ανακατευθυνθεί σε έναν ασφαλή ιστότοπο για να την ολοκληρώσει. Όταν η πληρωμή ολοκληρωθεί, η βάση ενημερώνεται για όλες τις απαραίτητες αλλαγές. Η διεπαφή του admin υλοποιήθηκε και αυτή σε Python (έκδοση 3.8.10) με στόχο να μπορεί μέσω αυτής ο διαχειριστής να υλοποιεί ορισμένες σημαντικές λειτουργίες, μέσω του command line. Όταν ανοίγει η διεπαφή, εμφανίζεται στον admin το σύνολο αυτό από εντολές, του οποίου κάποιες ενδεικτικά είναι ενημέρωση στοιχείων πελάτη, εμφάνιση εισιτηρίων ενός δρομολογίου, πρόσθεση δρομολογίου, ακύρωση πληρωμής/εισιτηρίων, αναζήτηση δημοφιλέστερων προορισμών, κλπ.

5.1.5 Σύνδεση Βάσης Δεδομένων – Διεπαφών

Αφού ολοκληρώθηκε και η υλοποίηση των γραφικών διεπαφών έγινε η σύνδεση τους με τη βάση δεδομένων μέσω του κατάλληλου αρχείου στο οποίο γράφονται τα SQL Queries με τα οποία θα γίνεται η δημιουργία, ανάγνωση, τροποποίηση και διαγραφή εγγραφών της βάσης δεδομένων. Πριν την χρήση των δεδομένων που ανακτούνται γίνεται η μετατροπή τους στην κατάλληλη μορφή ώστε να χρησιμοποιηθούν από την εφαρμογή.

5.1.6 Σχεδίαση και Υλοποίηση SQL Queries

Τα SQL Queries που αναπτύχθηκαν περιλαμβάνουν εντολές δημιουργίας (Create), ανάγνωσης (Read), ανανέωσης (Update) και διαγραφής (Delete) δεδομένων. Πιο συγκεκριμένα για το μέρος της υλοποίησης των ενεργειών που κάνει κάποιος χρήστης της εφαρμογής μέσω του αντίστοιχου CLI υλοποιήθηκαν εντολές ανάγνωσης των διαθέσιμων δρομολογίων με βάση τα κριτήρια του χρήστη, εντολές δημιουργίας για την είσοδο των στοιχείων των επιβατών, των εισιτηρίων τους και των πληρωμών τους, ανανέωσης των διαθέσιμων θέσεων επιβατών και οχημάτων από τις αντίστοιχες διαδρομές. Για το μέρος της

διαχείρισης της βάσης δημιουργήθηκαν εντολές δημιουργίας νέων δρομολογίων, ανάγνωσης διάφορων στοιχείων (εισιτηρίων, προορισμών, δρομολογίων), ανανέωσης στοιχείων πινάκων (επιβατών, διαθέσιμων θέσεων) και διαγραφής στοιχείων (πληρωμών, δρομολογίων).

5.2 ΑΝΑΘΕΣΗ ΕΡΓΑΣΙΩΝ

Οι εργασίες ανατέθηκαν όπως φαίνεται παρακάτω:

5.2.1 1^ο Μέλος

Αλέξανδρος Ματινόπουλος Λόπεθ:

- Υλοποίηση user CLI
- Υλοποίηση admin CLI

5.2.2 2^ο Μέλος

Ανέστης Μητακίδης:

- Υλοποίηση migration
- Υλοποίηση 'seeding

5.2.3 Pair Programming

- Ανάλυση απαιτήσεων και καθορισμός παραδοχών του προβλήματος
- Σχεδίαση Μικρόκοσμου
- Υλοποίηση ERD και Relational Schema
- Σύνδεση Βάσης Δεδομένων - Διεπαφών
- Σχεδίαση και Υλοποίηση SQL Queries

6 ΧΡΟΝΟΔΙΑΓΡΑΜΜΑ

	27/10/21 - 03/11/21	09/11/21-10/11/21	10/11/21-17/11/21	17/11/21 - 24/11/21	24/11/21 - 01/12/21	01/12/21-08/12/21	08/12/21 - 15/12/21	21/12/21 - 28/12/21	28/12/21 - 05/01/22	05/01/22 - 16/01/22
Ανάθεση Εργασιών										
Ανάλυση Απαιτήσεων										
Παραδοχές Μικρόκοσμου										
Υλοποίηση ERD										
Βελτιώσεις ERD										
Υλοποίηση Relational Schema										
Δημιουργία User CLI										
Δημιουργία Admin CLI										
Δημιουργία Migrations										
Δημιουργία Δεδομένων										
Δημιουργία Seeding										
Σύνδεση Βάσης - CLI										
Υλοποίηση Functions - SQL Queries										
Παρουσίαση - Αναφορά										

Εικόνα 4: Χρονοδιάγραμμα

7.1 Παράρτημα Α: Παράδειγμα Χρήσης Προγράμματος

Αρχικά γίνεται η εκκίνηση της εφαρμογής app.py σε Interactive Python Shell και η γραφή της λέξης book. Στην συνέχεια ο χρήστης συμπληρώνει τα στοιχεία που του ζητούνται για να κάνει την αναζήτηση διαδρομής που επιθυμεί.

Εικόνα 5: Προτιμήσεις Χρήστη

```
-----
INITIAL TRIP

TRIP 0:
Departure: Mastichari Kos at 2022-01-09 06:30:10
Station 1: Kaloi Limenes Rethymno at 2022-01-09 08:34:10
Station 2: Alexandroupolis at 2022-01-09 09:49:37
Station 3: Eretria Evoias at 2022-01-09 11:35:00
Arrival: Porto Lagos at 2022-01-09 14:35:11
Base Cost: 144.0

TRIP 1:
Departure: Mastichari Kos at 2022-01-09 11:57:43
Station 1: Souda Bay at 2022-01-09 13:10:55
Station 2: Panormos Tinou at 2022-01-09 15:27:06
Station 3: Psahna at 2022-01-09 15:35:12
Arrival: Porto Lagos at 2022-01-05 17:09:09
Base Cost: 106.0
-----
RETURN TRIP

TRIP 0:
Departure: Porto Lagos at 2022-01-13 15:49:27
Arrival: Mastichari Kos at 2022-01-13 18:28:05
Base Cost: 13.0

Select initial trip: 1
Select return trip: 0
```

Η επιλογή των επιθυμητών δρομολογίων γίνεται με την πληκτρολόγηση του αντίστοιχου αριθμού δρομολογίου.

Στην συνέχεια θα ζητηθεί από τον χρήστη να συμπληρώσει τα είδη εισιτηρίων, θέσης και οχήματος που επιθυμεί εφόσον υπάρχουν διαθέσιμα στο αντίστοιχο δρομολόγιο.

```
-----  
  
INITIAL TRIP DETAILS  
  
Ticket types:  
0. Adult  
1. Infant  
2. Child  
3. Student  
4. Senior  
5. Passenger with reduced mobility  
6. Greek military personel  
  
Ticket classes:  
0. Deck (150 available)  
1. Assigned Seat (50 available)  
2. 2 Bed Cabin (Single Bed) (20 available)  
3. 2 Bed Cabin (Whole Bed) (10 available)  
4. 4 Bed Cabin (Single Bed) (40 available)  
5. 4 Bed Cabin (Whole Bed) (10 available)  
  
Vehicle types:  
0. Car (15 slots available)  
1. Motorcycle (30 slots available)  
2. Truck (7 slots available)  
  
Passenger 1 ticket type: Adult  
Passenger 1 ticket class: 2 Bed Cabin (Single Bed)  
Passenger 2 ticket type: Student  
Passenger 2 ticket class: Assigned Seat  
  
Vehicle 1 is assigned to passenger: 1  
Vehicle 1 type: Car
```

Εικόνα 7: Συμπλήρωση τύπων θέσης, εισιτηρίου και οχήματος

RETURN TRIP DETAILS

Ticket types:

- 0. Adult
- 1. Infant
- 2. Child
- 3. Student
- 4. Senior
- 5. Passenger with reduced mobility
- 6. Greek military personel

Ticket classes:

- 0. Deck (150 available)
- 1. Assigned Seat (50 available)
- 2. 2 Bed Cabin (Single Bed) (20 available)
- 3. 2 Bed Cabin (Whole Bed) (10 available)
- 4. 4 Bed Cabin (Single Bed) (40 available)
- 5. 4 Bed Cabin (Whole Bed) (10 available)

Vehicle types:

- 0. Car (15 slots available)
- 1. Motorcycle (30 slots available)
- 2. Truck (7 slots available)

Passenger 1 ticket type: Adult
Passenger 1 ticket class: Deck
Passenger 2 ticket type: Student
Passenger 2 ticket class: Deck

Vehicle 1 is assigned to passenger: 1
Vehicle 1 type: Car

Εικόνα 8: Συμπλήρωση τύπων θέσης, εισιτηρίου και οχήματος

Μετά τον υπολογισμό του κόστους του εισιτηρίου κάθε επιβάτη θα ζητηθούν τα προσωπικά του στοιχεία

Passenger 2
Mastichari Kos - Porto Lagos: Student, Assigned Seat
Cost of trip: 74.2
Porto Lagos - Mastichari Kos: Student, Deck
Cost of trip: 6.5

First Name: Agatha
Last Name: Christie
Nationality (3 letter code): ENG
Date Of Birth (YYYY/MM/DD): 1999-11-04
Email: agachris@outlook.com
Mobile phone: 6958745526
National ID number: FT12548

Εικόνα 9: Συμπλήρωση Στοιχείων Επιβάτη

```
Passenger 1
Mastichari Kos - Porto Lagos: Adult, 2 Bed Cabin (Single Bed)
Cost of trip: 371.0
Porto Lagos - Mastichari Kos: Adult, Deck
Cost of trip: 32.5

First Name: John
Last Name: Thomson
Nationality (3 letter code): ENG
Date Of Birth (YYYY/MM/DD): 1975-05-13
Email: johntho@gmail.com
Mobile phone: 6984572155
National ID number: GB25145
```

Εικόνα 10: Συμπλήρωση Στοιχείων Επιβάτη

Προχωρώντας στην πληρωμή ο χρήστης μπορεί να επιλέξει αν θέλει ταξιδιωτική ασφάλιση και αν θέλει να ειδοποιηθεί με sms.

```
The total cost comes up to 484.2
Coupon code (type no if you dont have one): VENIAM
Travel Insurance (20 euro fee) (y/n): n
SMS Notification (2.5 euro fee) (y/n): n

New Total Cost comes up to: 411.57

Proceed to payment? (y/n)y

Payment Methods:

0. paypal
1. paysafe
2. visa
3. mastercard
4. bitcoin

Payment method: paypal
```

Εικόνα 11: Στοιχεία Πληρωμής

Μετά την καταχώρηση του τρόπου πληρωμής τα στοιχεία του αποθηκεύονται στην βάση δεδομένων και τυπώνονται τα εισιτήριά του.

Passenger: John Thomson
Ticket code: R94187274
Departure: 2022-01-09 11:57:43 from Mastichari Kos
Arrival: 2022-01-05 17:09:09 at Porto Lagos
Ship name: Ronald
Ship type: passenger
Company name: bailey-macdonald.net
Ticket cost: 371.0
Vehicle: Car

Passenger: Agatha Christie
Ticket code: R94187274
Departure: 2022-01-09 11:57:43 from Mastichari Kos
Arrival: 2022-01-05 17:09:09 at Porto Lagos
Ship name: Ronald
Ship type: passenger
Company name: bailey-macdonald.net
Ticket cost: 74.2
Vehicle: None

Passenger: John Thomson
Ticket code: P39636430
Departure: 2022-01-13 15:49:27 from Porto Lagos
Arrival: 2022-01-13 18:28:05 at Mastichari Kos
Ship name: Paula
Ship type: passenger
Company name: bailey-macdonald.net
Ticket cost: 32.5
Vehicle: Car

Passenger: Agatha Christie
Ticket code: P39636430
Departure: 2022-01-13 15:49:27 from Porto Lagos
Arrival: 2022-01-13 18:28:05 at Mastichari Kos
Ship name: Paula
Ship type: passenger
Company name: bailey-macdonald.net
Ticket cost: 6.5
Vehicle: None

Εικόνα 12: Εισιτήρια

Παρακάτω φαίνονται οι αλλαγές που γίνονται στη βάση μετά την πληρωμή.

- Μείωση των διαθέσιμων θέσεων σε κάθε διαδρομή.

Table: ROUTE											
	id	cost	deck_cap	v_cap	dep_date	arr_date	route_seq	dep_port_id	arr_port_id	trip_id	
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	
25	25	17.0	150	30	2022-01-09 11:28:06	2022-01-09 11:36:23	1	207	54	6	
26	26	22.0	150	30	2022-01-09 11:36:23	2022-01-09 15:07:08	2	54	159	6	
27	27	32.0	150	30	2022-01-09 15:07:08	2022-01-09 15:43:18	3	159	163	6	
28	28	37.0	150	30	2022-01-09 15:43:18	2022-01-09 16:00:25	4	163	192	6	
29	29	15.0	150	30	2022-01-12 14:54:17	2022-01-12 17:28:27	1	20	69	7	
30	30	24.0	150	30	2022-01-12 17:28:27	2022-01-12 19:03:54	2	69	259	7	
31	31	30.0	150	30	2022-01-12 19:03:54	2022-01-12 21:53:22	3	259	89	7	
32	32	37.0	150	30	2022-01-12 21:53:22	2022-01-12 23:29:36	4	89	131	7	
33	33	13.0	148	28	2022-01-13 15:49:27	2022-01-13 18:28:05	1	217	172	8	
34	34	19.0	150	30	2022-01-13 18:28:05	2022-01-13 18:48:27	2	172	181	8	
35	35	30.0	150	30	2022-01-13 18:48:27	2022-01-13 20:42:47	3	181	89	8	
36	36	37.0	150	30	2022-01-13 20:42:47	2022-01-13 22:48:38	4	89	142	8	
37	37	17.0	150	30	2022-01-16 03:27:54	2022-01-16 03:58:15	1	178	158	9	
38	38	24.0	150	30	2022-01-16 03:58:15	2022-01-16 05:02:50	2	158	74	9	
39	39	32.0	150	30	2022-01-16 05:02:50	2022-01-16 07:10:03	3	74	234	9	
40	40	37.0	150	30	2022-01-16 07:10:03	2022-01-16 09:07:10	4	234	251	9	
41	41	43.0	150	30	2022-01-16 09:07:10	2022-01-16 10:23:09	5	251	21	9	

Εικόνα 13: Ενημέρωση διαθέσιμων θέσεων

Table: ROUTE										
	id	cost	deck_cap	v_cap	dep_date	arr_date	route_seq	dep_port_id	arr_port_id	trip_id
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	13.0	150	30	2022-01-09 05:28:35	2022-01-09 06:30:10	1	132	172	1
2	2	24.0	150	30	2022-01-09 06:30:10	2022-01-09 08:34:10	2	172	127	1
3	3	30.0	150	30	2022-01-09 08:34:10	2022-01-09 09:49:37	3	127	59	1
4	4	42.0	150	30	2022-01-09 09:49:37	2022-01-09 11:35:00	4	59	75	1
5	5	48.0	150	30	2022-01-09 11:35:00	2022-01-09 14:35:11	5	75	217	1
6	6	59.0	150	30	2022-01-09 14:35:11	2022-01-09 15:03:48	6	217	38	1
7	7	18.0	148	28	2022-01-09 11:57:43	2022-01-05 13:10:55	1	172	246	2
8	8	23.0	148	28	2022-01-09 13:10:55	2022-01-05 15:27:06	2	246	6	2
9	9	30.0	148	28	2022-01-09 15:27:06	2022-01-05 15:35:12	3	6	211	2
10	10	35.0	148	28	2022-01-09 15:35:12	2022-01-05 17:09:09	4	211	217	2
11	11	16.0	150	30	2022-01-05 09:15:43	2022-01-05 09:46:04	1	6	98	3
12	12	22.0	150	30	2022-01-05 09:46:04	2022-01-05 12:04:39	2	98	154	3
13	13	31.0	150	30	2022-01-05 12:04:39	2022-01-05 14:10:46	3	154	254	3
14	14	37.0	150	30	2022-01-05 14:10:46	2022-01-05 15:52:24	4	254	186	3
15	15	47.0	150	30	2022-01-05 15:52:24	2022-01-05 17:02:41	5	186	244	3
16	16	54.0	150	30	2022-01-05 17:02:41	2022-01-05 21:04:06	6	244	70	3
17	17	15.0	150	30	2022-01-07 00:57:27	2022-01-07 04:29:35	1	200	118	4

Εικόνα 14: Ενημέρωση διαθέσιμων θέσεων

- Προσθήκη των εισιτηρίων και της πληρωμής

Table: TICKET											
	id	ticket_code	cost	special_seat	v_type	t_type	payment_id	passenger_id	trip_id	first_route	last_route
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
8	8	50775927	53.0	NULL	NULL	Adult	4	8	3	12	13
9	9	65742167	73.0	NULL	NULL	Adult	5	9	4	19	20
10	10	98707379	72.0	NULL	NULL	Adult	5	10	12	56	57
11	11	69057243	235.0	NULL	NULL	Adult	6	11	19	90	95
12	12	98149315	130.0	NULL	NULL	Adult	6	12	20	99	101
13	13	54806399	107.0	NULL	NULL	Adult	7	13	1	5	6
14	14	18577129	119.0	NULL	NULL	Adult	7	14	19	94	95
15	15	17643702	73.0	NULL	NULL	Adult	8	15	9	37	39
16	16	84017024	150.0	NULL	NULL	Adult	8	16	4	18	21
17	17	17331876	92.0	NULL	NULL	Adult	9	17	4	20	21
18	18	63169393	153.0	NULL	NULL	Adult	9	18	3	11	15
19	19	74193905	120.0	NULL	NULL	Adult	10	19	12	56	58
20	20	39768865	93.0	NULL	NULL	Adult	10	20	11	51	52
21	21	RO2022R94114028	371.0	2 Bed Cabin (Single Bed)	Car	Adult	11	21	2	7	10
22	22	RO2022R94114553	74.2	Assigned Seat	NULL	Student	11	22	2	7	10
23	23	PA2022P39611662	32.5	NULL	Car	Adult	11	21	8	33	33
24	24	PA2022P39611654	6.5	NULL	NULL	Student	11	22	8	33	33

Εικόνα 15: Προσθήκη Εισιτηρίων

Table:

PAYMENT

Filter in any c

	id	insurance	sms	payment_date	total_cost	payment_method	coupon_code
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	0	0	2022-01-14	401.0	paypal	NULL
2	2	0	0	2022-01-14	274.0	paypal	NULL
3	3	0	0	2022-01-14	140.0	paypal	NULL
4	4	0	0	2022-01-14	111.0	paypal	NULL
5	5	0	0	2022-01-14	145.0	paypal	NULL
6	6	0	0	2022-01-14	365.0	paypal	NULL
7	7	0	0	2022-01-14	226.0	paypal	NULL
8	8	0	0	2022-01-14	223.0	paypal	NULL
9	9	0	0	2022-01-14	245.0	paypal	NULL
10	10	0	0	2022-01-14	213.0	paypal	NULL
11	11	0	0	2022-01-16	411.57	paypal	VENIAM

Εικόνα 16: Προσθήκη Πληρωμής

- Προσθήκη των στοιχείων των επιβατών.

Table:

PASSENGER

Filter in any column

	id	fname	lname	country_code	phone_number	email	birthdate	id_card
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
6	6	Margaret	Tina	PK	+44808 1570204	dixonanthony@example.org	1987-05-09	XX88191768
7	7	Julie	Oliver	TV	+44131 4960009	dnash@example.net	1961-04-02	MM85673250
8	8	Mr	Holland	SE	+44(0)161 4960695	wyoung@example.net	1956-10-08	BB00290791
9	9	Sylvia	Denise	OM	01154960141	neilbarnes@example.org	1922-05-23	KK16342163
10	10	Andrew	Smith	NULL	+448081570366	nphillips@example.com	1957-06-21	XX78841659
11	11	Mr	Hall	TN	+441184960611	kylewilliams@example.com	1994-11-22	OO82757892
12	12	Barry	Lauren	IR	+44(0)1214960044	martin00@example.net	2004-02-12	GG20414320
13	13	Mr	Clark	AU	+44(0)1914960800	msavage@example.net	1919-08-23	QQ00193856
14	14	Gregory	Carter	US	01214960695	ybailey@example.net	1917-12-09	OO83947131
15	15	Fiona	Hunt-Gibbs	CY	+44117 4960590	rickyrobertson@example.net	1922-06-09	EE99254087
16	16	Ryan	Gail	IQ	+44(0)113 4960793	melanieahmed@example.org	1950-10-26	VV87035872
17	17	William	Zoe	TV	+44292018833	brownfiona@example.net	1939-01-13	RR67964727
18	18	Dr	Walsh-Curtis	PW	+44(0)117 4960202	poolegail@example.com	1929-04-28	LL01270747
19	19	Robert	Osborne	SC	020 7946 0097	taylorvictoria@example.com	1929-08-25	SS43857111
20	20	Hannah	Jones	RW	0161 496 0101	turnerjason@example.net	1978-12-15	KK24434539
21	21	John	Thomson	ENG	6984572155	johntho@gmail.com	2022-05-13	GB25145
22	22	Agatha	Christie	ENG	6958745526	agachris@outlook.com	1999-11-04	FT12548

Εικόνα 17: Προσθήκη Επιβατών

7.1.2 Παράδειγμα Χρήσης Διαχειριστή

Αρχικά γίνεται η εκκίνηση της εφαρμογής admin.py σε Interactive Python Shell.

Στην συνέχεια εμφανίζονται στην οθόνη του διαχειριστή οι διαθέσιμες εντολές διαχείρισης μαζί με τα ορίσματά τους τις οποίες μπορεί να τρέξει μέσω του Python Shell.

```
<><><><><><><><><><><><><><><><><><><><><><><>
Welcome to the Admin Interface.

Available Commands:
1. updatePassenger(fname,lname,country_code,phone_number,email,bdate,id_card)
2. addExtraCapacity(trip_id, deck_extra, veh_extra, air_extra, dcab_extra, qcab_extra)
3. readTicketsOfCompany(api_url)
4. readTicketsOfTrip(trip_uid)
5. getTopTenDestinations()
6. getTopTenDepartures()
7. deletePayment(payment_id)
8. addTrip(uid, ship_name, ship_type, deck_cap, air_cap, dcab_cap, qcab_cap, company_id, routes)
9. getOwnerOfTicket(ticket_code)

>>>

<><><><><><><><><><><><><><><><><><><><><><><>
Welcome to the Admin Interface.

Available Commands:
1. updatePassenger(fname,lname,country_code,phone_number,email,bdate,id_card)
2. addExtraCapacity(trip_id, deck_extra, veh_extra, air_extra, dcab_extra, qcab_extra)
3. readTicketsOfCompany(api_url)
4. readTicketsOfTrip(trip_uid)
5. getTopTenDestinations()
6. getTopTenDepartures()
7. deletePayment(payment_id)
8. addTrip(uid, ship_name, ship_type, deck_cap, air_cap, dcab_cap, qcab_cap, company_id, routes)
9. getOwnerOfTicket(ticket_code)

>>> getOwnerOfTicket('87072358')
[('87072358', 77.0, 'Dr', 'Preston', 'NN98987177', 'hobbsslydia@example.org', '+44(0)1214960235', 'Arkii Dodekanissou ', 'Argostoli')]
>>>
>>>
>>> readTicketsOfTrip('R82259372')

Tickets of trip R82259372

(5, '61303621', 71.0, None, None, 'Adult', 3, 5, 6)
(6, '15099167', 69.0, None, None, 'Adult', 3, 6, 6)
>>>
>>>
>>> getTopTenDestinations()

Top 10 Destination Ports

(2, 'Porto Lagos')
(2, 'Mastichari Kos')
(2, 'Korinthos')
(2, 'Agistri Aiginas')
(1, 'Thirasia')
(1, 'Thassos')
(1, 'Spetses')
(1, 'Skopelos')
(1, 'Schinoussa Kyklades')
(1, 'Poros Kefallinias')
>>>
```

Εικόνα 18: Εντολές Διαχείρισης

7.2 Παράρτημα Β: Οδηγίες Εγκατάστασης

ReadME.md

```
# Ferry DB App... developed for University Databases Course
```

Project Setup

- * Clone Repo

* In Terminal: Run the follow commands in terminal to install the modules that are used

- * Windows

Download Python 3.8.10 from <https://www.python.org/downloads/> and run installation.

Once Python is installed run the follow commands in a new Terminal.

...

```
pip install sqlite3
```

```
pip install faker
```

```
pip install pandas
```

...

* Linux

Install Python:

```

```
sudo apt install python3.8
```

```

Confirm the installation:

```

```
python3.8 --version
```

```

Once Python is installed run the follow commands in a new Terminal.

```

```
pip3 install sqlite3
```

```
pip3 install faker
```

```
pip3 install pandas
```

```

* Once you have all dependencies installed open ````/src/seed.py```` and in an interactive Python Shell run the ````migrate.migrate()```` method to create the database tables. When tables are created in the same Shell run ````Seed()```` to insert fake data in the database.

Now you are ready to open ````/src/app.py```` in an interactive Python Shell.

TIP: Type ````book```` to Start!

If you want to run the admin section open ````/src/admin.py```` in an interactive Python Shell.

Note

In the following files check if the ````dbfile=```` variable points right at the path that your database want to be created.

````/src/db.py````

````/src/seed.py````

````/src/migrate.py````

````/src/app.py````

````/src/admin.py````

## # Contributors

\* Mitakidis Anestis, Undergrad Student of ECE at the University of Patras.

: <https://github.com/L4Limbo>

\* Alejandro Matinopoulos Lopez, Undergrad Student of ECE at the University of Patras.

: <https://github.com/alematlop>

## # Useful links

\* Github Repo: [https://github.com/L4Limbo/ferry\\_db](https://github.com/L4Limbo/ferry_db)