



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ

Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Προηγμένα Θέματα Προγραμματισμού

Προγραμματιστική εργασία

Προγράμματα για επιχείρηση πλυντηρίου αυτοκινήτων

ΟΜΑΔΑ		
Ονοματεπώνυμο	ΑΜ	e-mail
Λάμπρος Γραμματικόπουλος	2022201800038	dit18038@uop.gr
Κωνσταντίνος Νικόλαος Κολοτούρος	2022201800090	dit18090@uop.gr

Εαρινό εξάμηνο 2021

Πίνακας περιεχομένων

Σχεδίαση των κλάσεων	3
Client	3
Server	6
Δομή αρχείου	9
Στοιχεία υλοποίησης που δεν συμφωνούν με την εκφώνηση	10

Σχεδίαση των κλάσεων

Client

Οι κλάσεις που χρησιμοποιήθηκαν από τον Client μαζί με την παρουσίαση της σχεδίασης τους φαίνονται παρακάτω:

- **Client**

Αυτή η κλάση αφορά την διεπαφή του προγράμματος υποδοχής οχημάτων. Πιο αναλυτικά, με αυτήν την κλάση εμφανίζεται στα εισερχόμενα οχήματα μία οθόνη στην οποία οι οδηγοί μέσω ορισμένων κουμπιών ενός πληκτρολογίου εισάγουν στο πρόγραμμα τον αριθμό κυκλοφορίας του οχήματός τους (του οποίου η εγκυρότητα ελέγχεται). Έπειτα, πατώντας κατάλληλο κουμπί μεταβιβάζονται στην επόμενη οθόνη.

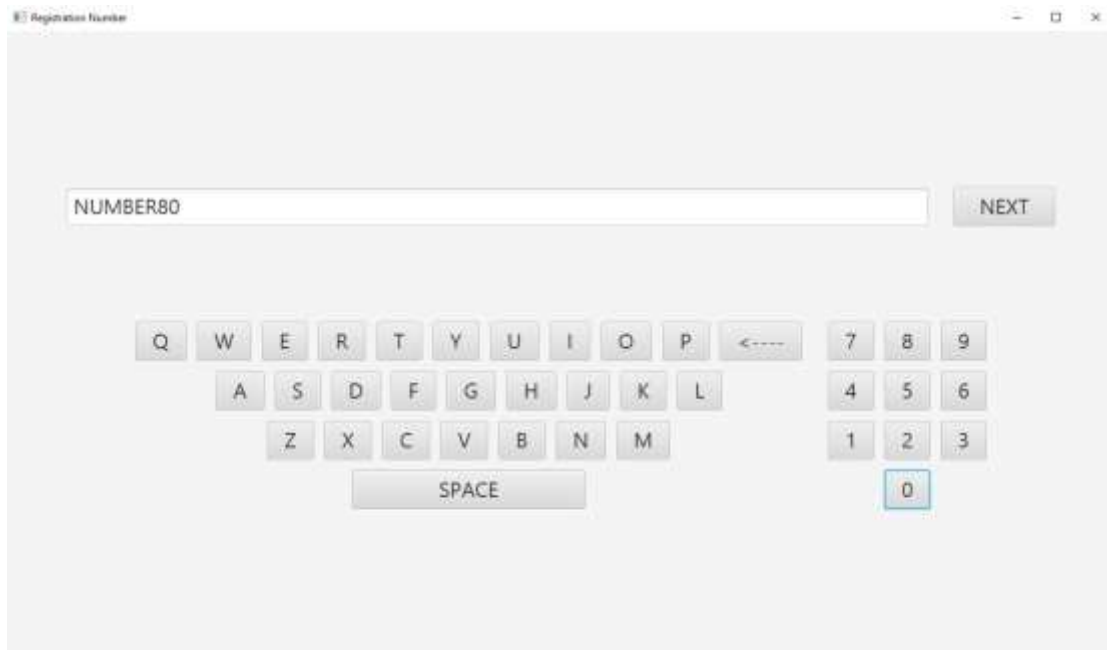
Πεδία της κλάσης:

- **Button** Q, W, E, R, T, Y, U, I, O, P, A, S, D, F, G, H, J, K, L, Z, X, C, V, B, N, M, NEXT, BACKSPACE, SPACE, N_1, N_2, N_3, N_4, N_5, N_6, N_7, N_8, N_9, N_0 → Για την αναπαράσταση των κουμπιών του πληκτρολογίου.
- **Vbox** Vbox, LettersVbox, NumbersVbox → Για την κάθετη αναπαράσταση των δύο μεγάλων Hboxes (hboxTop, hboxBottom) και για την αναπαράσταση των δύο κάθετων στηλών του πληκτρολογίου (γράμματα και αριθμοί).
- **Hbox** LettersHbox1, LettersHbox2, LettersHbox3, LettersHbox4, NumbersHbox1, NumbersHbox2, NumbersHbox3, NumbersHbox4 → Για την αναπαράσταση των 8 γραμμών (4 για γράμματα και 4 για αριθμούς) του πληκτρολογίου.
- **Hbox** hboxTop, hboxBottom → Για την αναπαράσταση του χώρου συμπλήρωσης του αριθμού κυκλοφορίας και του κουμπιού «NEXT» στην ίδια γραμμή (το πρώτο) και για την αναπαράσταση του γραμμάτων και των αριθμών στην ίδια γραμμή (το δεύτερο).
- **TextField** textField → Για την αναπαράσταση του χώρου συμπλήρωσης του αριθμού κυκλοφορίας.

Μέθοδοι της κλάσης:

- **main** → Για την εκτέλεση του προγράμματος.
- **start** → Για την αποικόνιση της σκηνής.

- `getRegistrationNumber` → Για την επιστροφή του συμπληρωμένου αριθμού κυκλοφορίας.



Οθόνη 1

- **services**
Αυτή η κλάση αφορά επίσης την διεπαφή αλλά και την λειτουργικότητα του προγράμματος υποδοχής οχημάτων. Πιο αναλυτικά, με αυτήν την κλάση οι οδηγοί καταχωρούν τον τύπο οχήματός τους και τις υπηρεσίες τις οποίες θα ήθελαν να λάβουν (στις οποίες εφαρμόζονται περιορισμοί ανάλογα με τον επιλεγμένο τύπο οχήματος και την επιλεγμένη υπηρεσία) ενώ παράλληλα ενημερώνονται για το συνολικό κόστος αυτών. Παράλληλα, δίνεται η δυνατότητα ακύρωσης της επιλογής τους με την χρήση κατάλληλου κουμπιού και επιστρέφουν στην πρώτη οθόνη και πάλι. Τέλος, κατά την αποστολή των επιλογών του οδηγού στο ταμείο δημιουργείται μια σύνδεση με αυτό και τα καταχωρημένα δεδομένα αποστέλλονται στο ταμείο.

Πεδία της κλάσης:

- **VBox** vbox, vbox1, vbox2, vbox3, vbox4 → Για την κάθετη αναπαράσταση των υπηρεσιών.
- **Hbox** hboxTop, hboxMiddle, hboxBottom → Για την αναπαράσταση του συνολικού κόστους (το πρώτο), για την αναπαράσταση των διαθέσιμων επιλογών (το δεύτερο) και για την σε σειρά αναπαράσταση των δύο κουμπιών (το τρίτο).
- **RadioButton** rbcar, rbjeep, rbmoto, rb1, rb2, rb3, rb4, rb5, rb6, rb7, rb8, rb9, rb10 → Για την αναπαράσταση των κουμπιών των επιλογών.
- **Label** LBcar, LBjeep, LBmoto, costLabel → Για την αναπαράσταση των επιγραφών του κόστους και του τύπου οχήματος.
- **ToggleGroup** group1, group2, group3, group4 → Για την ομαδοποίηση των επιλογών του τύπου οχήματος και των υπηρεσιών με στόχο την επιβολή περιορισμών.
- **Button** SEND, CANCEL → Για την χρήση των κουμπιών.
- **int** cost → Για τον υπολογισμό του κόστους.
- **int** typeOfVehicle → Για την εύρεση του τύπου οχήματος.
- **boolean** rb1Flag=false, rb2Flag=false, rb3Flag=false, rb4Flag=false, rb5Flag=false, rb6Flag=false, rb7Flag=false, rb8Flag=false, rb9Flag=false, rb10Flag=false → Βοηθητικές για την επιβολή περιορισμών.

Μέθοδοι της κλάσης:

- **services** → Για την αναπαράσταση της σκηνής.
- **checkCode** → Για τον υπολογισμό των κωδικών από την κλάση των κοινών στοιχείων.

Services

Συνολικό κόστος: 210

	● Αυτοκίνητο	● Τζιπ	● Μοτοσυκλέτα
● Πλύσιμο εξωτερικό	7	8	6
● Πλύσιμο εσωτερικό	6	7	-
● Πλύσιμο εξωτ.+εσωτ.	12	14	-
● Πλύσιμο εξωτ.σπέσιαλ	9	10	8
● Πλύσιμο εσωτ.σπέσιαλ	8	9	-
● Πλύσιμο εξωτ. + εσωτ. σπέσιαλ	15	17	-
● Βιολογικός καθαρισμός εσωτ.	80	80	-
● Κέρωμα-Γυάλισμα	80	90	40
● Καθαρισμός κινητήρα	20	20	10
● Πλύσιμο σασί	3	3	-

ΑΠΟΣΤΟΛΗ ΑΚΥΡΩΣΗ

- **Structure**

Αυτή είναι η κλάση των κοινών στοιχείων του Server και του Client, η οποία χρησιμοποιείται και από τους δύο. Σε αυτήν αποθηκεύονται σε μια δομή: `Map<String, String>` treeMap όλες οι υπηρεσίες της επιχείρησης πλυντηρίου αυτοκινήτων. Επίσης, σε αυτήν την κλάση υλοποιούνται ορισμένες μέθοδοι για την αποθήκευση και αναπαράσταση των υπηρεσιών και του κόστους τους.

Μέθοδοι της κλάσης:

- `Structure` → Για τον ορισμό των υπηρεσιών.
- `getValue` → Για την απόκτηση της επιλεγμένης υπηρεσίας και της τιμής της.
- `getCode` → Για την αποθήκευση των υπηρεσιών στην παραπάνω δομή.
- `set` → Για την απόκτηση του μοναδικού κωδικού της επιλεγμένης υπηρεσίας.
- `print` → Για την εκτύπωση των υπηρεσιών (για λόγους debugging).

Server

Οι κλάσεις που χρησιμοποιήθηκαν από τον Server μαζί με την παρουσίαση της σχεδίασης τους φαίνονται παρακάτω:

- **Server**

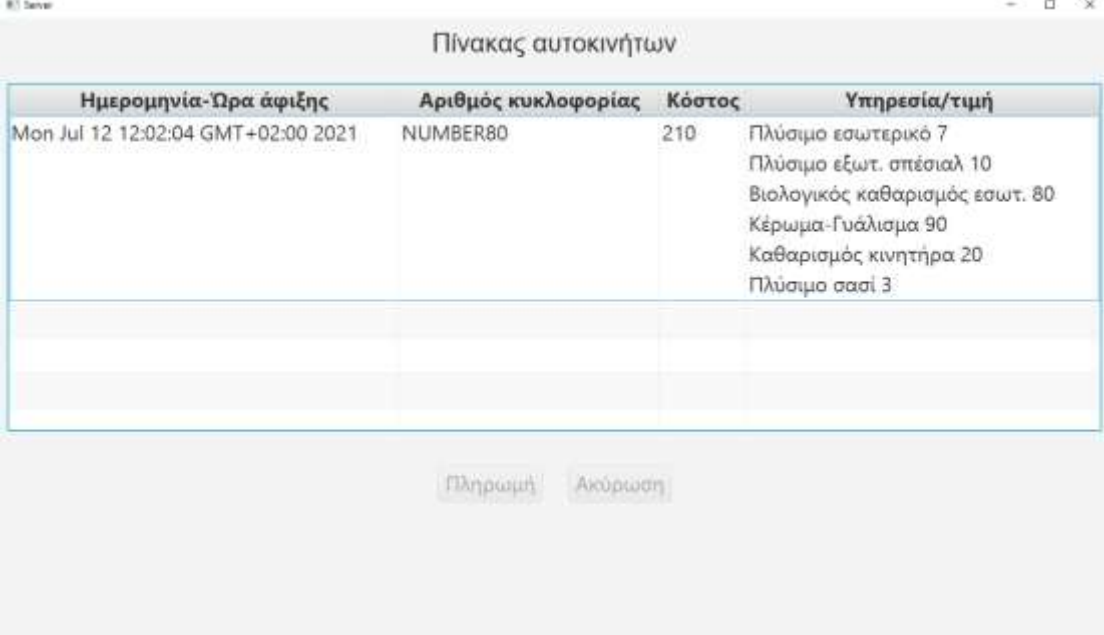
Αυτή η κλάση αφορά την διεπαφή του προγράμματος ταμείου. Πιο συγκεκριμένα, με αυτήν την κλάση εμφανίζονται στο ταμείο σε μία οθόνη σε μορφή πίνακα όλες οι πληροφορίες των εισερχόμενων οχημάτων που στέλνει ο Client (ημερομηνία-ώρα άφιξης, αριθμός κυκλοφορίας, κόστος, υπηρεσία/τιμή). Σε αυτήν την οθόνη ο ταμειάς μπορεί να προβεί στην πληρωμή των υπηρεσιών που έλαβε ένα όχημα και ο πελάτης να λάβει την απόδειξή του ή ακόμα και στην ακύρωση αυτών.

Πεδία της κλάσης:

- `TableView<data>` table → Για την εμφάνιση του πίνακα με τα εισερχόμενα οχήματα.
- `Button` pay, cancel → Για την χρήση των κουμπιών της πληρωμής και της ακύρωσης.

Μέθοδοι της κλάσης:

- **main** → Για την εκτέλεση του προγράμματος.
- **start** → Για την αποικόνιση της σκηνής.



Ημερομηνία-Ωρα άφιξης	Αριθμός κυκλοφορίας	Κόστος	Υπηρεσία/τιμή
Mon Jul 12 12:02:04 GMT+02:00 2021	NUMBER80	210	Πλύσιμο εσωτερικό 7 Πλύσιμο εξωτ. σπέσιαλ 10 Βιολογικός καθαρισμός εσωτ. 80 Κέρωμα-Γυάλισμα 90 Καθαρισμός κινητήρα 20 Πλύσιμο σασί 3

Πληρωμή Ακύρωση

Οθόνη 3

- **data**

Αυτή η κλάση αφορά την **αναπαράσταση** των δεδομένων που στέλνει ο Client.

Πεδία της κλάσης:

- **String** arrivalDateTime → Για την τοπική αποθήκευση της ημερομηνίας και της ώρας άφιξης.
- **String** registrationNumber → Για την τοπική αποθήκευση του αριθμού κυκλοφορίας.
- **int** cost → Για την τοπική αποθήκευση του κόστους.
- **String** value → Για την τοπική αποθήκευση της κάθε υπηρεσίας μαζί με την τιμή της (που λαμβάνεται μέσω της κοινής δομής **Structure**).

Μέθοδοι της κλάσης:

- **data** → Για την αρχικοποίηση των δεδομένων.
- **getArrivalDateTime** → Για την επιστροφή της ημερομηνίας και της ώρας άφιξης.
- **getRegistrationNumber** → Για την επιστροφή του αριθμού κυκλοφορίας.

- `getCost` → Για την επιστροφή του κόστους.
- `getValue` → Για την επιστροφή της κάθε υπηρεσίας μαζί με την τιμή της.

- **Structure**

Αυτή είναι η κλάση των κοινών στοιχείων του Server και του Client, η οποία χρησιμοποιείται και από τους δύο. Σε αυτήν αποθηκεύονται σε μια δομή: `Map<String, String>` `treeMap` όλες οι υπηρεσίες της επιχείρησης πλυντηρίου αυτοκινήτων. Επίσης, σε αυτήν την κλάση υλοποιούνται ορισμένες μεθόδους για την αποθήκευση και αναπαράσταση των υπηρεσιών και του κόστους τους.

Μέθοδοι της κλάσης:

- `Structure` → Για τον ορισμό των υπηρεσιών.
- `getValue` → Για την απόκτηση της επιλεγμένης υπηρεσίας και της τιμής της.
- `getCode` → Για την αποθήκευση των υπηρεσιών στην παραπάνω δομή.
- `set` → Για την απόκτηση του μοναδικού κωδικού της επιλεγμένης υπηρεσίας.
- `print` → Για την εκτύπωση των υπηρεσιών (για λόγους debugging).

- **getData**

Αυτή η κλάση αφορά την λειτουργικότητα του Server και σε αυτήν γίνεται η σύνδεση με τον Client με την μέθοδο `run`.

- **EchoServerThread**

Αυτή η κλάση αφορά επίσης την λειτουργικότητα του Server και σε αυτήν τα δεδομένα που λαμβάνονται από τον Client χωρίζονται σε κατάλληλα πεδία και καταχωρούνται στον πίνακα του ταμείου αλλά και στο βιβλίο εισόδων.

Πεδία της κλάσης:

- `Socket clientSocket` → Για την αναπαράσταση του Socket.
- `ObservableList<data>` `dataList` → Για την εμφάνιση των δεδομένων στον πίνακα του ταμείου.
- `ArrayList<String>` `array` → Για την αποθήκευση των επιλεγμένων υπηρεσιών με βάση τον μοναδικό κωδικό τους.

Μέθοδοι της κλάσης:

- `EchoServerThread` → Για τον αρχικοποίηση του `Socket`.
- `run` → Για τον ορισμό των υπηρεσιών.

- **returnList** → Για την επιστροφή της επιλεγμένης υπηρεσίας και της τιμής της.
 - **getArrivalDateTime** → Για την επιστροφή της ημερομηνίας και της ώρας άφιξης.
 - **getRegistrationNumber** → Για την επιστροφή του αριθμού κυκλοφορίας.
 - **getCost** → Για την επιστροφή του κόστους.
 - **getValue** → Για την επιστροφή της κάθε υπηρεσίας μαζί με την τιμή της.
 - **deleteFromList** → Για την αφαίρεση οχήματος από τον πίνακα του ταμείου.
 - **returnValues** → Για την λήψη της κάθε υπηρεσίας μαζί με την τιμή της από την κοινή δομή **Structure**.
- **saveData**
Αυτή η κλάση αφορά επίσης την λειτουργικότητα του Server και σε αυτήν υλοποιούνται οι μέθοδοι εγγραφής, ενημέρωσης και διαγραφής στο βιβλίο εισόδων.

Μέθοδοι της κλάσης:

- **saveData** → Για την δημιουργία ενός αρχείου που θα αναπαριστά το βιβλίο εισόδων σε περίπτωση που δεν υπάρχει ήδη.
- **writeData** → Για την εγγραφή μιας νέας καταχώρησης που έφτασε ταμείο στο βιβλίο εισόδων.
- **addData** → Για την προσθήκη επιπλέον δεδομένων σε μια καταχώρηση στο βιβλίο εισόδων κατά την πληρωμή της.
- **delete** → Για την διαγραφή από το βιβλίο εισόδων μιας καταχώρησης κατά την ακύρωσή της από το ταμείο.

Δομή αρχείου

Για την δημιουργία του αρχείου (δηλαδή του βιβλίου εισόδου) χρησιμοποιήθηκε η δομή **File**. Ενώ, για την προσπέλαση, εγγραφή, επεξεργασία και διαγραφή δεδομένων από αυτό χρησιμοποιήθηκε η δομή **RandomAccessFile** με εγγραφές μεταβλητού μήκους και για αυτόν τον λόγο χρειάστηκαν απαραίτητοι έλεγχοι και προσωρινή αποθήκευση μέρος του αρχείου σε μία δομή **ArrayList** για την υλοποίηση των παραπάνω λειτουργιών. Αυτό έγινε με δική μας πρωτοβουλία με στόχο την εξοικονόμηση αποθηκευτικού χώρου με αντάλλαγμα στον χρόνο εκτέλεσης του προγράμματος (στους σημερινούς υπολογιστές οι

υπολογισμοί γίνονται με μεγάλες ταχύτητες ακόμα και για μεγάλα αρχεία και δεν υπάρχει πρόβλημα καθυστέρησης εκτέλεσης).

Στοιχεία υλοποίησης που δεν συμφωνούν με την εκφώνηση

Όλα τα στοιχεία υλοποίησης βρίσκονται μέσα στο πλαίσιο της εκφώνησης και συμφωνούν με αυτήν, εκτός από την χρήση αρχείου τυχαίας προσπέλασης με εγγραφές σταθερού μήκους, το οποίο υλοποιήθηκε με αρχείου τυχαίας προσπέλασης με εγγραφές μεταβλητού μήκους (ο λόγος εξηγήθηκε στην παραπάνω ενότητα).