

**Ονομασία Έργου: Smart Park**

**Project-Description**

**Έκδοση: v1.0**



Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΑΛΚΙΒΙΑΔΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

Editor: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Contributor: ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ

Reviewer: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ

Στο GitHub υπάρχουν όλα τα παραδοτέα και τα αρχεία κώδικα:

<https://github.com/EliasPapagelis/Smart-Park-TL>

## Smart Park: Smart and Easy way to park

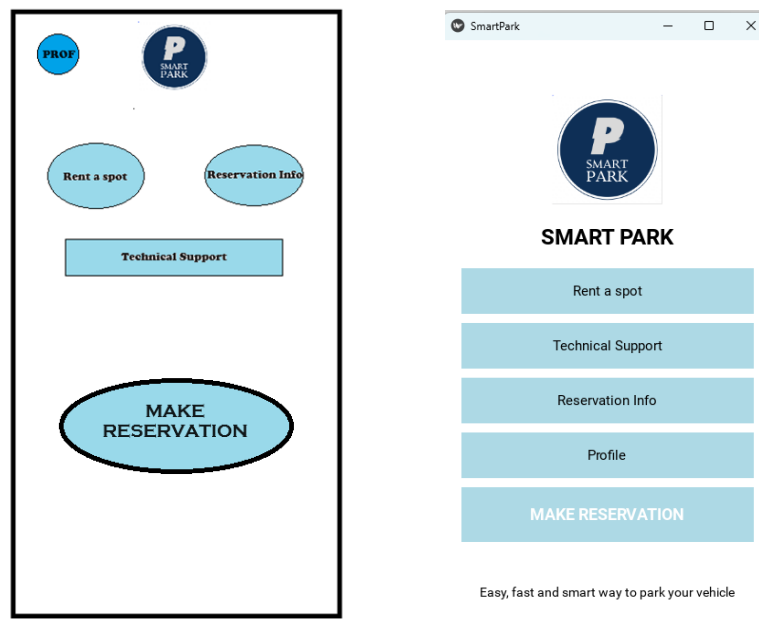
Στις μέρες μας, το πρόβλημα του παρκινγκ όλο και μεγαλώνει καθώς ο αριθμός των οχημάτων στις πόλεις αυξάνεται ενώ η χρήση των μεσών μαζικής μεταφοράς παραμένει σταθερή. Καθημερινά οι πολίτες ψάχνουν απεγνωσμένα μια θέση για παρκινγκ με αποτέλεσμα σε πολλά κομμάτια της πόλης μας να δημιουργείται κυκλοφοριακή συμφόρηση. Για αυτόν τον λόγο κατασκευάσαμε το Smart Park, μια εφαρμογή γρήγορης και έξυπνης εύρεσης παρκινγκ.

Όπως αναφέρθηκε και πιο πάνω το Smart Park είναι μια καινοτόμα εφαρμογή που δίνει την δυνατότητα στον χρήστη να κλείσει διαδικτυακά μια θέση παρκινγκ. Πως γίνεται όμως αυτό; Μέσα από την εφαρμογή μας, ο **χρήστης** μπορεί να δει τις διαθέσιμες θέσεις παρκινγκ για εκείνη την δεδομένη στιγμή αλλά και για την υπόλοιπη μέρα. Έτσι μπορεί πάλι μέσα από την εφαρμογή να κλείσει εξ αποστάσεως εύκολα και γρήγορα μια θέση για συγκεκριμένες ώρες που επιθυμεί για τη στάθμευση του οχήματός του. Με την δημιουργία κράτησης ο χρήστης λαμβάνει ένα QR Code που είναι το «κλειδί» για την είσοδο και έξοδο του στο ειδικά διαμορφωμένο παρκινγκ. Ανάλογα με το όχημα του, ο χρήστης μέσω της εφαρμογής θα έχει την δυνατότητα να κλείσει και ειδικές θέσεις στάθμευσης με περισσότερες παροχές όπως θέσης για αναπήρους ή θέσεις με φορτιστές για ηλεκτρικά οχήματα. Ακόμα, ο χρήστης έχει στην διάθεση του την επιλογή να προβεί σε μια μηναία κράτηση μιας θέσεως. Τέλος, για τον κάθε χρήστη υπάρχει προφίλ όπου μπορεί να επεξεργαστεί τα προσωπικά του στοιχεία αλλά και τα στοιχεία ηλεκτρονικών πληρωμών του, καθώς ένας από τους στόχους της εφαρμογής είναι η γρήγορη και ανέπαφη πληρωμή.

Πέρα από τον χρήστη, την εφαρμογή μας μπορεί την χρησιμοποιήσει και ένας **διαχειριστής** ενός παρκινγκ. Μια από τις δυνατότητες του διαχειριστή είναι να βλέπει ανάλογα τις ημερομηνίες που αυτός επιθυμεί τα έσοδα που κάνει το παρκινγκ, δηλαδή μπορεί να επιλέξει μια συγκεκριμένη εβδομάδα και να ελέγξει τα έσοδα εκείνης της εβδομάδας. Ακόμα ο διαχειριστής μπορεί να δώσει κάποιες προσφορές για τις κρατήσεις του παρκινγκ, όπως για παράδειγμα μικρότερη χρέωση σε κάποιες συγκεκριμένες ώρες της ημέρας. Τέλος ο διαχειριστής έχει και την επιλογή να τροποποιεί τα ωράρια λειτουργίας του παρκινγκ όπως αυτός επιθυμεί, παραδείγματος χάρη να απενεργοποιεί τις νυχτερινές ώρες χρήσης.

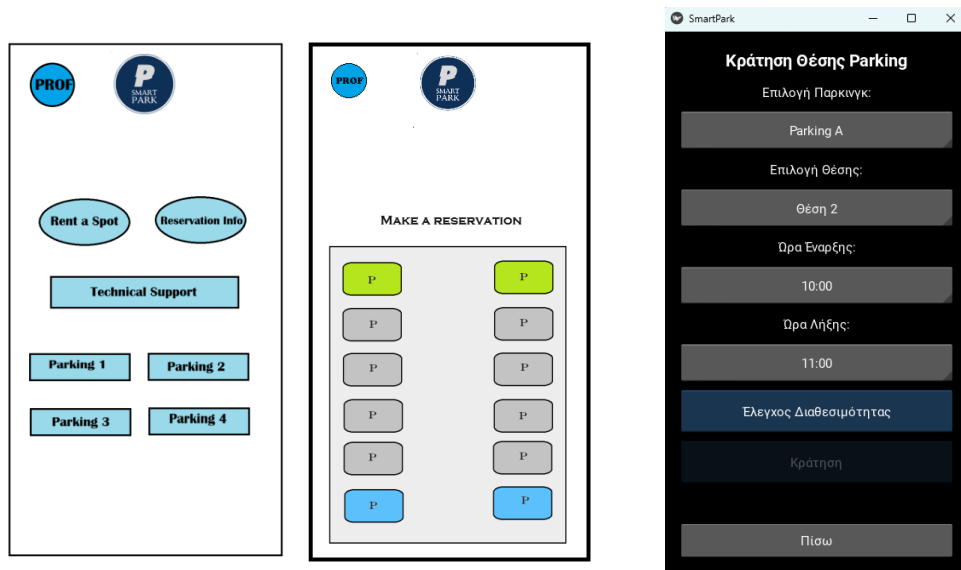
**Από την έκδοση v0.1 στην έκδοση v1.0 προστέθηκαν τα τελικά screen και έγιναν κάποιες επεξηγήσεις πάνω στις αλλαγές σε σχέση με τα αρχικά Mock-Ups**

### Τα Screens της εφαρμογής:



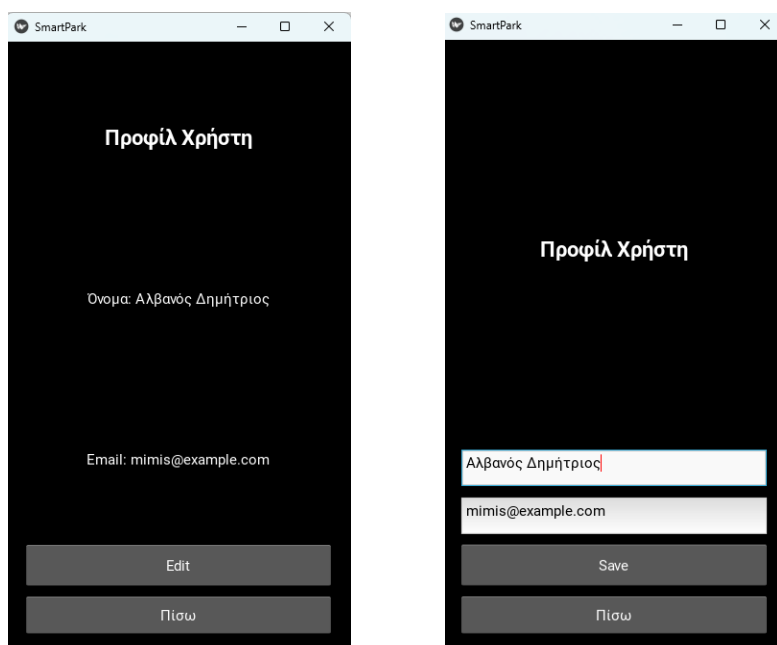
Εικόνα 1: Mock-Up screen αρχικής σελίδας χρηστή και το τελικό screen που υλοποιήθηκε.

Στο αριστερό μέρος της Εικόνας 1 φαίνεται το Mock-Up screen που βασίστηκε η ιδέα υλοποίησης της αρχικής σελίδας. Στην τελική υλοποίηση έχουν παραμείνει τα ίδια κουμπιά, απλά με διαφορετική διάταξη για την διευκόλυνση της κατανόησης του χρήστη. Στο πάνω μέρος φαίνεται το logo της εφαρμογής μας και από κάτω οι επιλογές ενοικίασης και κράτησης θέσεως, οι πληροφορίες μια κράτησης,, η τεχνική υποστήριξη και το προφίλ του χρήστη.



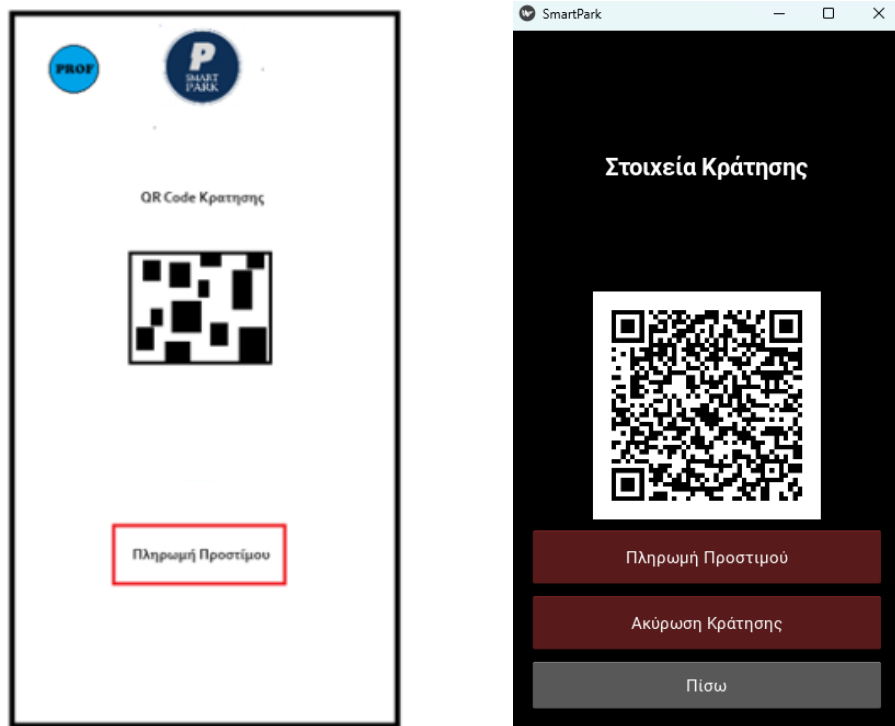
Εικόνα 2: Mock-Up και τελικά screens για την κράτηση θέσεως .

Σαν αρχική ιδέα η κράτηση θέσεων εξελισσόταν σε συνεχόμενες νέες οθόνες και ήταν πιο διαδραστική. Προφανώς κάτι τέτοιο ήταν δύσκολο στην υλοποίηση και για αυτό τα περισσότερα screens συγχωνεύτηκαν σε ένα που δίνει όλες τις επιλογές όπως η επιλογή παρκινγκ, θέσεως και των ωρών της κράτησης.



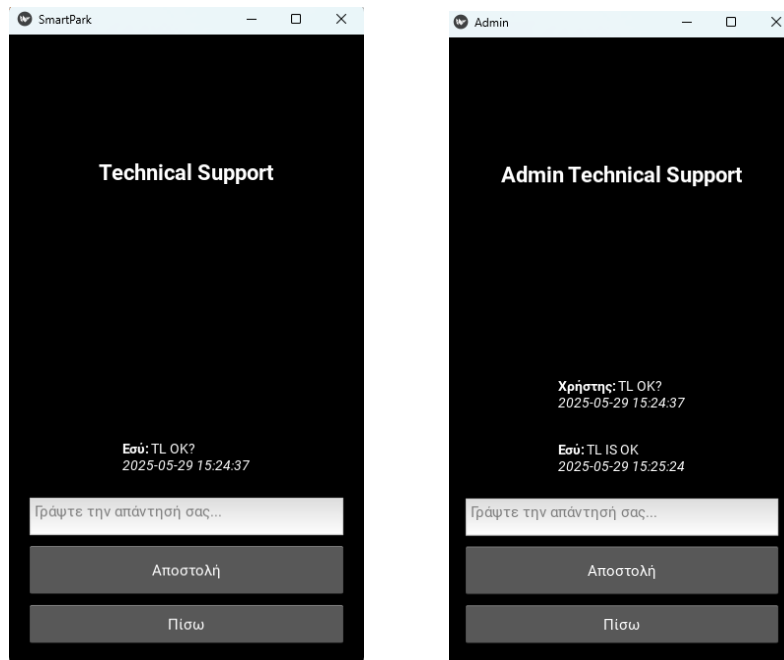
Εικόνα 3: Screens επεξεργασίας προφίλ.

Στα παραπάνω screens φαίνεται η διαδικασία επεξεργασίας προφίλ. Αρχικά δίνονται μερικές πληροφορίες όπως το όνομα του χρήστη και το μείλ του. Πατώντας το κουμπί edit ο χρήστης μπορεί να αλλάξει αυτά τα στοιχεία όπως αυτός επιθυμεί.



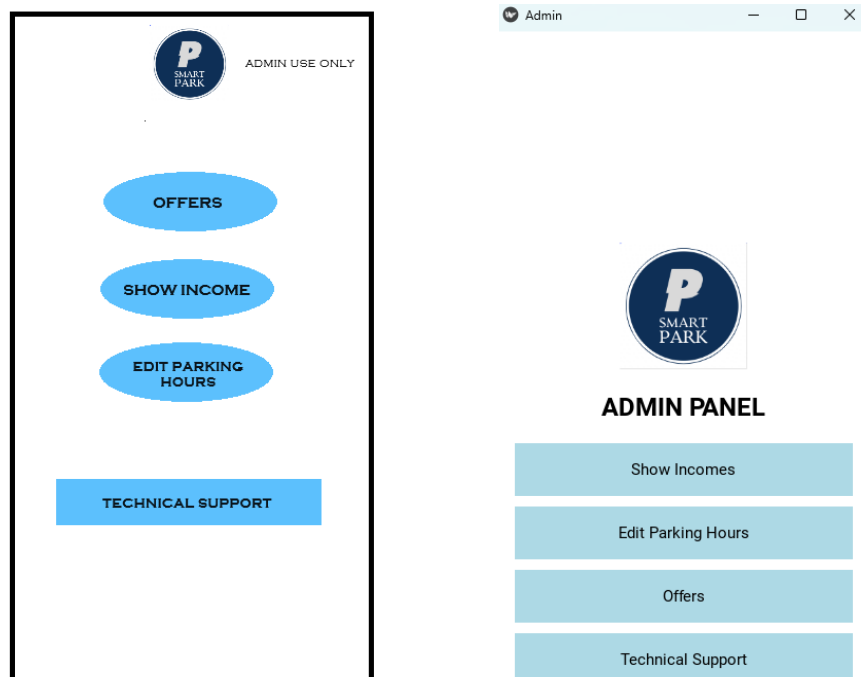
Εικόνα 4: Mock-Up screen σελίδας «Reservation Info» και τελική υλοποίηση.

Στην Εικόνα 4 φαίνεται άλλο ένα screen που δέχτηκε μερικές αλλαγές. Πια, ο χρήστης πέρα από τις αρχικές επιλογές του να δει το qr code της κράτησης και την πληρωμή προστίμου, έχει και την δυνατότητα να ακυρώσει την κράτηση του.



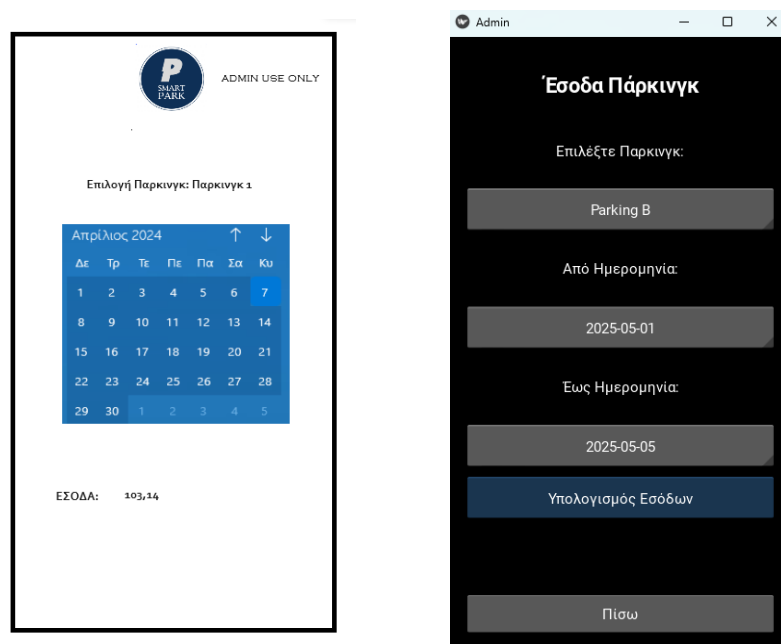
Εικόνα 5: Technical support

Σε αυτή την εικόνα φαίνεται αριστερα το technical support chat από την μεριά του χρηστη ενώ δεξια από την μεριά του διαχειριστη.



Εικόνα 6: Mock-Up screen αρχικής σελίδας διαχειριστή και η τελική υλοποίηση.

Στην Εικόνα 6 φαίνεται αρχική σελίδα του διαχειριστή και οι λειτουργίες που προσφέρονται. Αρχικά και ο διαχειριστής έχει την δυνατότητα να επεξεργαστεί και αυτός το προφίλ του. Έπειτα φαίνονται τα τρία κουμπιά «Offers», «Show Income» και «Edit Parking Hours» όπου είναι και τα κουμπιά για τις βασικές λειτουργίες του διαχειριστή. Τέλος υπάρχει και το κουμπί «Technical Support» για την επικοινωνία χρήστη-διαχειριστή.



Εικόνα 7: Mock-Up screen σελίδας «Show Income» διαχειριστή.

Τέλος μεγάλες αλλαγές φαίνονται και στο αρχικό screen της προβολής εσόδων. Πια υπάρχουν κουμπιά για την επιλογή παρκινγκ και το διάστημα ημερομηνιών που θέλει ο διαχειριστής να παρακολουθήσει την εισροή εσόδων.



**Μετά από ομόφωνη συμφωνία της ομάδας η κατανομή της εργασίας του κάθε συμμετέχοντα συμφωνήθηκε ως εξής:**

**Ραφαήλ Νικόλαος Μίχος: 1.10/5**

**Δημήτρης Αλβανός: 1.10/5**

**Ηλίας Μάριος Παπαγγελής: 1.10/5**

**Γιώργος Σωτηρόπουλος: 0.85/5**

**Πασχάλης Δρουκόπουλος: 0.85/5**

### **Αναφορές:**

Το παρόν τεχνικό κείμενο δημιουργήθηκε στα πλαίσια του μαθήματος Τεχνολογία Λογισμικού κατά τη διάρκεια του ακαδημαϊκού έτους 2025.

Το Mock-up screen της Εικόνας 1 δημιουργήθηκε μέσω του figma (<https://www.figma.com/>).

Το ημερολόγιο που χρησιμοποιήθηκε στο Mock-up screen της Εικόνας 6 χρησιμοποιήθηκε ως παράδειγμα από το ημερολόγιο των Microsoft Windows.

Τα Mock-up screens των υπολοίπων Εικόνων δημιουργήθηκαν μέσω του Microsoft Paint.

Το logo της εφαρμογής δημιουργήθηκε μέσω του looka ([https://looka.com/?gad\\_source=1&gclid=Cj0KCQjw5cOwBhCiARIsAJ5njubXKJC7wiQMK8uGg7O5\\_8iz8jyGDgVekHWnyyo14Cos0YtAnS4GIbcaAusdEALw\\_wcB](https://looka.com/?gad_source=1&gclid=Cj0KCQjw5cOwBhCiARIsAJ5njubXKJC7wiQMK8uGg7O5_8iz8jyGDgVekHWnyyo14Cos0YtAnS4GIbcaAusdEALw_wcB)).

**Ονομασία Έργου: Smart Park**

**Use-Cases**

**Έκδοση: v1.0**



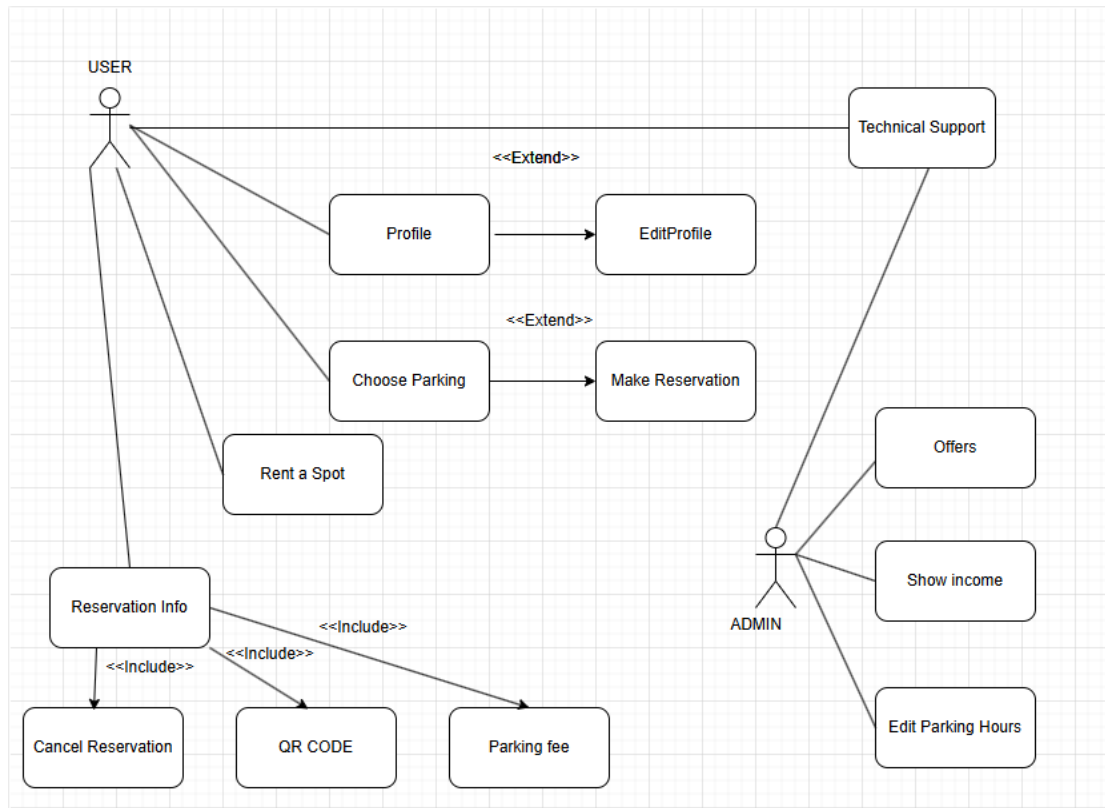
Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

Editor: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Contributor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ, ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ,  
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ

Reviewer: ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ

## USE CASE DIAGRAM



### Παραδοχές:

1. Παρόλο που οι χρήστες έχουν την δυνατότητα να επιστρέψουν στην αρχική σελίδα από όλες τις σελίδες της εφαρμογής για λόγους απλούστευσης συχνά δεν αναφέρεται.
2. Η εφαρμογή δουλεύει κατά βάση με ειδικά διαμορφωμένους χώρους παρκινγκ που διαθέτουν σκάνερ QRCode ώστε να επιτρέπουν την είσοδο/έξοδο των οχημάτων.
3. Τα πρόστιμα λειτουργούν ως χρηματική επιβάρυνση για τους πελάτες σε περιπτώσεις αργοπορίας πέραν της λήξης της κράτησης. Αυτό θα υπολογίζεται με βάση την ώρα που βγήκε ο χρήστης από το παρκινγκ όταν θα ξανασκανάρει το QR Code. Προφανώς αυτό δεν μπορεί να υλοποιηθεί σε κώδικα καθώς δεν διαθέτουμε τα δεδομένα από την μπάρα. Παρόλα αυτά θεωρούμε ότι πρέπει να παραμείνει σαν ιδέα και use case σε περίπτωση που στο μέλλον υπάρξει αυτή η δυνατότητα.
4. Όλοι οι διαχειριστές χρησιμοποιούν το ίδιο chat (λογαριασμό chat) για την τεχνική υποστήριξη.
5. Όπου αναφέρουμε αρχική οθόνη της εφαρμογής εννοούμε το 'screen' αφετηρίας που ανοίγει μετά την σύνδεση του κάθε χειριστή στην εφαρμογή. Ο κάθε χειριστής έχει διαφορετική αρχική σελίδα εφαρμογής.
6. Όλες οι εναλλακτικές ροές ακολουθούν την αρίθμηση του manual και των διαφανειών του φροντιστηρίου.

Με κόκκινο χρώμα έχουν επισημανθεί οι αλλαγές από το v0.1 στο v0.2

Το v0.3 είναι ίδιο με το v0.2 με αλλαγές μόνο σε γραμματοσειρές.

Με μπλε χρώμα έχουν επισημανθεί οι αλλαγές από το v0.3 στο v1.0

Τα κύρια 10 **use cases** που θα αναλυθούν είναι τα:

1. Προβολή και Επεξεργασία Προφίλ
2. Προβολή Εσόδων
3. Technical support (user, admin)
4. Διαχείριση Προσφορών και Εκπτώσεων
5. Κράτηση Θέσεως παρκινγκ
6. Είσοδος/ Έξοδος από το παρκινγκ
7. Πληρωμή Πρόστιμου
8. Τροποποίηση ωρών λειτουργίας παρκινγκ
9. Ενοικίαση θέσης παρκινγκ
10. Ακύρωση Κράτησης

## USE CASES

**Τίτλος:** Προβολή και Επεξεργασία Προφίλ (View and Edit Profile)

**Χειριστές:** Χρήστης (user)

**Περιγραφή:** Ο χρήστης να έχει την δυνατότητα να προβάλλει και να επεξεργάζεται τα προσωπικά του δεδομένα.

**Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** και διαλέγει την επιλογή «Profile».
2. Το σύστημα φορτώνει και εμφανίζει τα δεδομένα του προφίλ του χρήστη **στην οθόνη Προφίλ Χρήστη**.
3. Στην οθόνη **Προφίλ Χρήστη** εμφανίζονται τα προσωπικά δεδομένα του χρήστη, δηλαδή το όνομα, email κλπ.
4. Ο χρήστης επιλέγει ~~το κουμπί~~ «Edit».
5. Το σύστημα **εμφανίζει την οθόνη Επεξεργασίας Στοιχείων**.
6. Ο χρήστης αλλάζει τα δεδομένα στα αντίστοιχα πεδία (π.χ. αλλαγή Ονόματος) **και επιλέγει «Save»**.
- ~~7. Ο χρήστης επιλέγει «Save» κατά την ολοκλήρωση των αλλαγών.~~
7. Το σύστημα ελέγχει την εγκυρότητα των δεδομένων και ανανεώνει την βάση δεδομένων.

8. Το σύστημα εμφανίζει θετικό μήνυμα ‘Saved Changes’ στην οθόνη του χρήστη.

#### ***Εναλλακτική ροή 1:***

**6.1.1** Ο χρήστης επιλέγει να επεξεργαστεί το προφίλ αλλά αποφασίζει να μην προχωρήσει με τις αλλαγές.

**6.1.2** Επιλέγει το κουμπί «[Πίσω](#)».

**6.1.3** Το σύστημα τον οδηγεί στην αρχική σελίδα της εφαρμογής.

#### ***Εναλλακτική ροή 2:***

**7.1.1** Το σύστημα διαπιστώνει ότι ορισμένα δεδομένα δεν είναι έγκυρα (π.χ. λανθασμένη μορφή email, password κλπ.).

**7.1.2** Το σύστημα εμφανίζει μήνυμα σφάλματος ‘False Change’.

**7.1.3** Το σύστημα γυρίζει τον χρήστη στο Βήμα 5 της Βασικής Ροής.

**Τίτλος:** Προβολή Εσόδων (Show incomes)

**Χειριστές:** Διαχειριστής (admin)

**Περιγραφή:** Αυτή η περίπτωση χρήσης αρχίζει όταν ο διαχειριστής επιλέξει «Show Incomes» και του επιτρέπει αφού επιλέξει ένα παρκινγκ και τις ημερομηνίες που τον ενδιαφέρουν να δει τα συνολικά έσοδα όπου και τελειώνει η περίπτωση χρήσης.

#### **Βασική ροή:**

1. Ο διαχειριστής πλοηγείται στην **Αρχική Οθόνη διαχειριστή** και επιλέγει «Show incomes».
2. Το σύστημα **εμφανίζει την οθόνη Show Incomes.**
3. Ο διαχειριστής επιλέγει ένα από τα παρκινγκ και το διάστημα ημερομηνιών που θέλει να δει.
4. Το σύστημα αντλεί δεδομένα με βάση τις ημερομηνίες εισαγωγής του χρήστη και την επιλογή εσόδων.
5. Το σύστημα υπολογίζει τα έσοδα με βάση δεδομένα εισαγωγής.
6. Το σύστημα εμφανίζει στον διαχειριστή το σύνολο των εσόδων σύμφωνα με τις προηγούμενες επιλογές του.

#### ***Εναλλακτική ροή 1:***

**4.1.1** Το σύστημα αντλεί δεδομένα με βάση τις επιλογές του χρήστη αλλά δεν υπάρχουν έσοδα.

**4.1.2** Το σύστημα εμφανίζει μήνυμα ‘No incomes between the selected dates’.

**4.1.3** Το σύστημα επιστρέφει τον διαχειριστή στο βήμα 3 της Βασικής Ροής.

**Τίτλος: Τεχνική υποστήριξη χρήστη (Technical support user).**

**Χειριστές:** Χρήστης (user)

**Περιγραφή:** Αυτή η περίπτωση χρήσης αρχίζει όταν οι χρήστες επιλέξουν ‘Technical Support’. Τους επιτρέπει να επικοινωνήσουν με έναν διαχειριστή του συστήματος για οποιοδήποτε πρόβλημα έχει προκύψει. Η περίπτωση χρήσης τελειώνει όταν οι χρήστης επιλέξουν να γυρίσουν στην αρχική σελίδα.

**Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** της εφαρμογής και επιλέγει ‘Technical Support’.
2. Το σύστημα αντλεί πληροφορίες για τα μηνύματα όπου έχουν σταλθεί μεταξύ του συγκεκριμένου χρήστη και των διαχειριστών.
3. Το σύστημα εμφανίζει το ιστορικό ανταλλαγής μηνυμάτων και χώρο για εισαγωγή νέων μηνυμάτων **οθόνη Technical Support**.
4. Ο χρήστης γράφει το μήνυμα που επιθυμεί και **επιλέγει το «Αποστολή»**.
5. Το σύστημα αποθηκεύει το μήνυμα και **ενημερώνει την Βάση Δεδομένων**.
6. Το σύστημα εμφανίζει στο ιστορικό ανταλλαγής μηνυμάτων το καινούριο μήνυμα του χρήστη.
7. Ο χρήστης τελειώνει τη συζήτηση και επιλέγει **«Πίσω»**.
8. Το σύστημα τον οδηγεί στην **Αρχική Οθόνη**.

**Τίτλος: Τεχνική Υποστήριξη διαχειριστή (Technical Support Admin)**

**Χειριστές:** Διαχειριστής (admin)

**Περιγραφή:** Αυτή η περίπτωση χρήσης αρχίζει όταν ο διαχειριστής επιλέξει ‘Technical Support’ και του επιτρέπει να δει τα μηνύματα του από χρήστες ή ιδιοκτήτες, να απαντήσει ή να τα διαγράψει. Η περίπτωση χρήσης τελειώνει όταν ο διαχειριστής κλείσει το παράθυρο του μηνύματος, απαντήσει ή διαγράψει το μήνυμα.

**Βασική ροή:**

1. Ο διαχειριστής πλοηγείται στην **Αρχική Οθόνη διαχειριστή** της εφαρμογής και επιλέγει ‘Technical Support’.
2. Το σύστημα αντλεί πληροφορίες για τα μηνύματα όπου έχουν ανταλλαχθεί μεταξύ των χρηστών και του διαχειριστή.
3. Το σύστημα εμφανίζει τα ιστορικά ανταλλαγής μηνυμάτων μεταξύ του διαχειριστή και των χρηστών καθώς και δίνει την δυνατότητα στον διαχειριστή να επιλέξει ποιο από αυτά θα ανοίξει.
4. Ο διαχειριστής επιλέγει μία συνομιλία και αφού την διαβάσει στέλνει την απάντηση του.
5. Το σύστημα αποθηκεύει την απάντηση και ενημερώνει **την Βάση Δεδομένων**.
- ~~6. Το σύστημα εμφανίζει την απάντηση του διαχειριστή στον αρχικό αποστολέα του μηνύματος.~~

**Εναλλακτική ροή 1:**

- 4.1.1 Ο διαχειριστής επιλέγει στην **οθόνη Technical Support** μία συνομιλία και αφού την διαβάσει επιλέγει να την διαγράψει.
- 4.1.2 Το σύστημα αποθηκεύει τις αλλαγές και ενημερώνει **την Βάση Δεδομένων**.

**4.1.3** Το σύστημα εμφανίζει μήνυμα “This **conversation** has been deleted!” και επιστρέφει τον χρήστη στο βήμα 4 της βασικής ροής της περίπτωσης χρήσης.

#### **Τίτλος: Διαχείριση Προσφορών και Εκπτώσεων (Offers)**

**Χειριστές:** Διαχειριστής (admin)

**Περιγραφή:** Ο διαχειριστής θα μπορεί να προσθέτει, να επεξεργάζεται και να διαγράφει προσφορές και εκπτώσεις που αφορούν την εφαρμογή.

#### **Βασική Ροή:**

1. Ο διαχειριστής πλοηγείται στην **Αρχική Οθόνη διαχειριστή** της εφαρμογής και επιλέγει ~~το κουμπί~~ «Offers».
2. Το σύστημα εμφανίζει τα παρκινγκ **στην οθόνη «Offers»**.
3. Ο διαχειριστής επιλέγει ένα από τα παρκινγκ.
4. Το σύστημα εμφανίζει την λίστα με τις διαθέσιμες προσφορές και εκπτώσεις **στην οθόνη «Parkings»**.
5. Ο διαχειριστής επιλέγει να προσθέσει μια νέα προσφορά ή έκπτωση.
6. Ο διαχειριστής αποθηκεύει τις αλλαγές.
7. Το σύστημα **ενημερώνει** την βάση δεδομένων με τις καινούργιες αλλαγές.
8. Το σύστημα εμφανίζει θετικό μήνυμα ‘Saved Changes’.
- ~~9. Ο διαχειριστής επιλέγει να γυρίσει στην αρχική σελίδα.~~
- ~~10. Το σύστημα τον οδηγεί στην αρχική σελίδα της εφαρμογής.~~

#### **Εναλλακτική ροή 1**

- 5.1.1 Ο διαχειριστής επιλέγει μια προσφορά για διαγραφή εάν δεν είναι πλέον επιθυμητή.
- 5.1.2 Ο διαχειριστής πατάει ~~το κουμπί~~ «Delete».
- 5.1.3 Το σύστημα ανανεώνει την βάση δεδομένων και διαγράφει την προσφορά.
- 5.1.4 Το σύστημα εμφανίζει θετικό μήνυμα ‘Offer Deleted’.
- ~~5.1.5. Ο διαχειριστής επιλέγει να γυρίσει στην αρχική σελίδα.~~
- ~~5.1.6. Το σύστημα τον οδηγεί στην αρχική σελίδα της εφαρμογής.~~

#### **Τίτλος: Κράτηση Θέσεως παρκινγκ (Make Reservation)**

**Χειριστές:** Χρήστης (user)

**Περιγραφή:** Η περίπτωση χρήσης αρχίζει όταν ο χρήστης ανοίγει την εφαρμογή στο κινητό του. Ο χρήστης μπορεί να επιλέξει ~~το είδος του οχήματος του και στη συνέχεια επιλέγει~~ το πάρκινγκ που θέλει, την θέση και την ώρα της προτίμησης του.

#### **Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** και επιλέγει ~~το κουμπί~~ «Make Reservation».
- ~~2. Το σύστημα αντλεί τα δεδομένα που επέλεξε ο χρήστης.~~
2. Το σύστημα εμφανίζει την οθόνη επιλογής παρκινγκ.
3. Ο χρήστης επιλέγει το παρκινγκ της αρεσκείας του.



4. Το σύστημα αντλεί δεδομένα από την Βάση Δεδομένων και εμφανίζει στο χρήστη την οθόνη επιλογής θέσης **παρκινγκ**.
5. Ο χρήστης επιλέγει όποια θέση προτιμάει από αυτές που είναι διαθέσιμες.
6. Το σύστημα εμφανίζει όλες τις διαθέσιμες ώρες που μπορεί να επιλέξει για την συγκεκριμένη θέση στην οθόνη επιλογής ώρας.
7. Ο χρήστης επιλέγει τις ώρες που θέλει να κλείσει την θέση.
8. Το σύστημα ελέγχει την διαθεσιμότητα της θέσης στην Βάση Δεδομένων.
9. Το σύστημα εμφανίζει την **οθόνη πληρωμής**.
10. Ο χρήστης επιλέγει την πληρωμή κράτησης.
- ~~12. Το σύστημα υπολογίζει το ποσό πληρωμής της κράτησης.~~
- ~~13. Το σύστημα ελέγχει και αντλεί τα δεδομένα της κάρτας του χρήστη.~~
- ~~14. Το σύστημα εμφανίζει το ποσό και ένα κουμπί «Έγκριση πληρωμής»~~
- ~~15. Ο χρήστης πατάει το κουμπί έγκρισης για να ολοκληρώσει την πληρωμή.~~
- ~~16. Το σύστημα συνδέεται με την τράπεζα και αντλεί το κατάλληλο χρηματικό ποσό.~~
11. Το σύστημα επεξεργάζεται τη συναλλαγή μέσω της επιλεγμένης πλατφόρμας.
12. Το σύστημα εμφανίζει μήνυμα **‘Η κράτηση ολοκληρώθηκε με επιτυχία’** και στέλνει το QR Code της κράτησης στο «Reservation Info».

### **Εναλλακτική ροή 1**

- 8.1.1 Το σύστημα κατά τον έλεγχο διαθεσιμότητας βρίσκει κάποια άλλη κράτηση με παρόμοιες ώρες στην θέση που επέλεξε ο χρήστης.
- 8.1.2 Το σύστημα εμφανίζει **‘Η θέση είναι ήδη δεσμευμένη’** και επιστρέφει τον χρήστη στο **Βήμα 6** της Βασικής Ροής.

### **Εναλλακτική ροή 2**

- ~~13.1.1 Το σύστημα αντλεί τα δεδομένα της κάρτας του χρήστη αλλά δεν είναι έγκυρη.~~
- ~~13.1.2 Το σύστημα εμφανίζει στην οθόνη του χρήστη αρνητικό μήνυμα ‘Wrong Credit Card’.~~
- ~~13.1.3 Το σύστημα επιστρέφει τον χρήστη στο Βήμα 12 της Βασικής Ροής.~~

### **Εναλλακτική ροή 3**

- ~~16.1.1 Το σύστημα συνδέεται με την τράπεζα αλλά ο χρήστης δεν κατέχει το κατάλληλο χρηματικό ποσό.~~
- ~~16.1.2 Το σύστημα εμφανίζει στην οθόνη του χρήστη αρνητικό μήνυμα ‘Insufficient Balance’.~~
- ~~16.1.3 Το σύστημα επιστρέφει τον χρήστη στο Βήμα 12 της Βασικής Ροής.~~

**Τίτλος:** Είσοδος/ Έξοδος από το παρκινγκ (Reservation QR Code)

**Χείριστες:** Χρήστης (user)

**Περιγραφή:** Η περίπτωση χρήσης ξεκινά όταν ο χρήστης επιλέξει «Reservation Info» και το σύστημα εμφανίζει το QR Code της κράτησης ώστε ο χρήστης να εισέρθει ή εξέλθει από το παρκινγκ.

### **Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** της εφαρμογής και επιλέγει το «Reservation Info».

2. Το σύστημα αντλεί τα απαραίτητα στοιχεία για τον χρήστη από την Βάση Δεδομένων.
3. Το σύστημα εμφανίζει **νέα οθόνη Reservation Info** και το QR Code της κράτησης.
4. Ο χρήστης σκανάρει το QR Code ~~στον αισθητήρα εισόδου του παρκινγκ.~~
5. **Το σύστημα ελέγχει** την εγκυρότητα του QR Code.
6. Το QR Code εγκρίνεται και το σύστημα στέλνει θετικό μήνυμα στη μπάρα εισόδου.
7. **Το σύστημα ανοίγει** την μπάρα εισόδου .
8. Ο χρήστης εισέρχεται στο παρκινγκ.

#### ***Εναλλακτική ροή 1:***

- 2.1.1 Το σύστημα δεν βρήκε κάποια κράτηση άρα δεν υπάρχει QR Code.
- 2.1.2 Το σύστημα εμφανίζει μήνυμα «No Reservation».

#### ***Εναλλακτική ροή 2:***

- 6.1.1 Το QR Code δεν εγκρίνεται.
- 6.1.2 Το σύστημα εμφανίζει στην οθόνη του χρηστή αρνητικό μήνυμα ‘Wrong QR Code’ και επιστρέφει τον χρήστη στο Βήμα 4 της Βασικής Ροής.

### **Τίτλος: Πληρωμή Πρόστιμου (Reservation Fine)**

**Χείριστες:** Χρήστης (user)

**Περιγραφή:** Η περίπτωση χρήσης ξεκινά όταν ο χρήστης επιλέξει «Reservation Info» και το σύστημα εμφανίζει την πληρωμή προστίμου.

#### **Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** της εφαρμογής και επιλέγει το «Reservation Info».
2. Το σύστημα αντλεί τα απαραίτητα στοιχεία για τον χρήστη **από την Βάση Δεδομένων.**
3. Το σύστημα εμφανίζει **την Οθόνη Reservation Info .**
4. Ο χρήστης επιλέγει την «πληρωμή πρόστιμου».
5. Το σύστημα αντλεί δεδομένα από την ΒΔ και υπολογίζει το ποσό πληρωμής.
6. Το σύστημα **εμφανίζει σε νέα οθόνη Πληρωμή Προστίμου το ποσό, και τα στοιχεία της κράτησης και το «Έγκριση πληρωμής».**
7. Ο χρήστης επιλέγει «Έγκριση Πληρωμής».
8. Το σύστημα επεξεργάζεται τη συναλλαγή μέσω της επιλεγμένης πλατφόρμας.
9. Το σύστημα εμφανίζει θετικό μήνυμα πληρωμής “Fine Paid” στην οθόνη ~~του~~ **χρήστη** Πληρωμής Προστίμου.

#### ***Εναλλακτική ροή 1***

- 6.1.1 Το σύστημα αντλεί τα δεδομένα της κάρτας του χρήστη αλλά δεν είναι έγκυρη.
- 6.1.2 Το σύστημα εμφανίζει στην οθόνη του χρηστή αρνητικό μήνυμα ‘Μη Έγκυρη Κάρτα’ και επιστρέφει τον χρήστη στο Βήμα 4 της Βασικής Ροής.

#### ***Εναλλακτική ροή 2***

~~9.1.1 Το σύστημα συνδέεται με την τράπεζα αλλά ο χρήστης δεν κατέχει το κατάλληλο χρηματικό ποσό.~~

~~9.1.2 Το σύστημα εμφανίζει στην οθόνη του χρηστή αρνητικό μήνυμα ‘Ανεπαρκές Υπόλοιπο’ και επιστρέφει τον χρήστη στο Βήμα 4 της Βασικής Ροής.~~

**Τίτλος:** Τροποποίηση ωρών λειτουργίας παρκινγκ (Edit Parking Hours)

**Χείριστες:** Διαχειριστής (admin)

**Περιγραφή:** Η περίπτωση χρήσης ξεκινά όταν ο διαχειριστής επιλέξει «Edit Parking Hours». Το σύστημα δίνει την επιλογή στο διαχειριστή να ελέγξει τις ώρες λειτουργίας του παρκινγκ.

**Βασική Ροή:**

1. Ο διαχειριστής πλοηγείτε στην **Αρχική Οθόνη Διαχειριστή** της εφαρμογής και επιλέγει το «Edit Parking Hours» .
2. Το σύστημα αντλεί δεδομένα από την Βάση Δεδομένων και εμφανίζει νέα **σελίδα οθόνη Edit Parking Hours** με το πίνακα με τις ώρες λειτουργίας του παρκινγκ.
3. Ο διαχειριστής επιλεγεί το Parking το οποίο θέλει να τροποποιήσει.
4. Ο διαχειριστής επιλεγεί τις ώρες που θα είναι **κλειστό ανοιχτό** το παρκινγκ.
5. Το σύστημα ελέγχει τις ήδη υπάρχουσες ώρες λειτουργίας.
6. Ο διαχειριστής **πατάει κουμπί** επιλέγει «**Αποθήκευση** νέου ωραρίου λειτουργίας».
7. Το σύστημα λαμβάνει την υποβολή και ανανεώνει την βάση δεδομένων.
8. Το σύστημα εμφανίζει τα καινούργια ωράρια στις οθόνες των χρηστών.

**Εναλλακτική ροή 1:**

**4.2.1** Το σύστημα βρήκε παρόμοιες ώρες λειτουργίας.

**4.2.2** Το σύστημα εμφανίζει μήνυμα «Wrong Hours» και επανέρχεται στο βήμα 2 της Βασικής Ροής.

**Τίτλος: Ενοικίαση Θέσης Parking (Rent a Spot)**

**Χειριστές:** Χρήστης (user)

**Περιγραφή:** Ο χρήστης έχει την δυνατότητα να ενοικιάσει οποιαδήποτε θέση παρκινγκ θέλει, για χρονικό διάστημα ενός μήνα.

**Βασική Ροή:**

1. Ο χρήστης πλοηγείται στην **Αρχική Οθόνη** της εφαρμογής και επιλέγει «Rent a Spot».
2. Το σύστημα εμφανίζει την οθόνη επιλογής παρκινγκ.
3. Ο χρήστης επιλέγει το παρκινγκ της αρεσκείας του.
4. Το σύστημα αντλεί δεδομένα από την Βάση Δεδομένων και εμφανίζει στο χρήστη την οθόνη επιλογής θέσης.
5. Ο χρήστης επιλέγει όποια θέση προτιμάει από αυτές που είναι διαθέσιμες.
6. Το σύστημα εμφανίζει όλες τις διαθέσιμες ημερομηνίες που μπορεί να επιλέξει για την συγκεκριμένη θέση.
7. Ο χρήστης επιλέγει τις ημερομηνίες που θέλει να κλείσει την θέση.
8. Το σύστημα ελέγχει την διαθεσιμότητα της θέσης στην Βάση Δεδομένων.
9. Το σύστημα εμφανίζει την οθόνη πληρωμής.
10. Ο χρήστης επιλέγει την πληρωμή ενοικίασης.
- ~~11. Το σύστημα υπολογίζει το ποσό πληρωμής της ενοικίασης.~~
- ~~12. Το σύστημα ελέγχει και αντλεί τα δεδομένα της κάρτας του χρήστη.~~
- ~~13. Το σύστημα εμφανίζει το ποσό και ένα κουμπί «Έγκριση πληρωμής»~~
- ~~14. Ο χρήστης πατάει το κουμπί έγκρισης για να ολοκληρώσει την πληρωμή.~~
- ~~15. Το σύστημα συνδέεται με την τράπεζα και αντλεί το κατάλληλο χρηματικό ποσό.~~
9. Το σύστημα επεξεργάζεται τη συναλλαγή μέσω της επιλεγμένης πλατφόρμας.
10. Το σύστημα εμφανίζει μήνυμα ‘Successful Rental’ και στέλνει το QR Code της κράτησης στο «Reservation Info».

**Εναλλακτική ροή 1**

- 8.1.1 Το σύστημα κατά τον έλεγχο διαθεσιμότητας βρίσκει κάποια άλλη ενοικίαση με παρόμοιες ημερομηνίες στην θέση που επέλεξε ο χρήστης.
- 8.1.2 Το σύστημα εμφανίζει ‘Δεν είναι διαθέσιμη σε αυτές τις ημερομηνίες’
- 8.1.3 Το σύστημα επιστρέφει το χρηστή στο Βήμα 4 της Βασικής Ροής.

**Εναλλακτική ροή 2**

- 15.1.1 Το σύστημα αντλεί τα δεδομένα της κάρτας του χρήστη αλλά δεν είναι έγκυρη.
- ~~15.1.2 Το σύστημα εμφανίζει στην οθόνη του χρηστή αρνητικό μήνυμα ‘Wrong Credit Card’.~~

### ***Εναλλακτική Ροή 3***

**18.1.1** Το σύστημα συνδέεται με την τράπεζα αλλά ο χρήστης δεν κατέχει το κατάλληλο χρηματικό ποσό.

**18.1.2** Το σύστημα εμφανίζει στην οθόνη του χρηστή αρνητικό μήνυμα 'Insufficient Balance'.

### **Τίτλος: Ακύρωση Κράτησης (Cancel Reservation)**

**Χειριστές:** Χρήστης (user)

**Περιγραφή:** Ο χρήστης μπορεί να ακυρώσει μια κράτηση 1 ώρα πριν από την έναρξη της. Αν η ακύρωση γίνει 1 ώρα πριν την κράτηση, γίνεται πλήρης επιστροφή χρημάτων.

### **Βασική Ροή:**

- 1.Ο χρήστης πλοηγείται στην Αρχική Οθόνη και επιλέγει το «Reservation Info».
- 2.Το σύστημα αντλεί δεδομένα από την Βάση Δεδομένων και εμφανίζει την οθόνη **Reservation Info** με τις ενεργές κρατήσεις του χρήστη.
- 3.Ο χρήστης επιλέγει «Ακύρωση Κράτησης» την κράτηση που θέλει να ακυρώσει.
- 4.Το σύστημα εμφανίζει νέα οθόνη Ακύρωσης Κράτησης με τις λεπτομέρειες της κράτησης και επιλογή ακύρωσης.
- 5.Ο χρήστης επιλέγει «Cancel Reservation».
- 6.Αν η ακύρωση είναι έγκαιρη, το σύστημα επιστρέφει πλήρως τα χρήματα στην κάρτα με την οποία έγινε η πληρωμή.
- 7.Το σύστημα ακυρώνει την κράτηση και εμφανίζει μήνυμα επιβεβαίωσης «Reservation Cancelled».

### ***Εναλλακτική Ροή 1 (Εκπρόθεσμη Ακύρωση)***

- 6.1.1 Το σύστημα βρίσκει ότι η κράτηση είναι κοντά στην ώρα έναρξης.
- 6.1.2 Το σύστημα ενημερώνει τον χρήστη ότι δεν μπορεί να ολοκληρωθεί η επιστροφή χρημάτων και εμφανίζει μήνυμα «Reservation Cancelled».
- 6.1.4 Το σύστημα επιστρέφει τον χρηστή στο Βήμα 2 της Βασικής Ροής.

**Ονομασία Έργου: Smart Park**

**Robustness-Diagrams**

**Έκδοση: v1.0**



Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

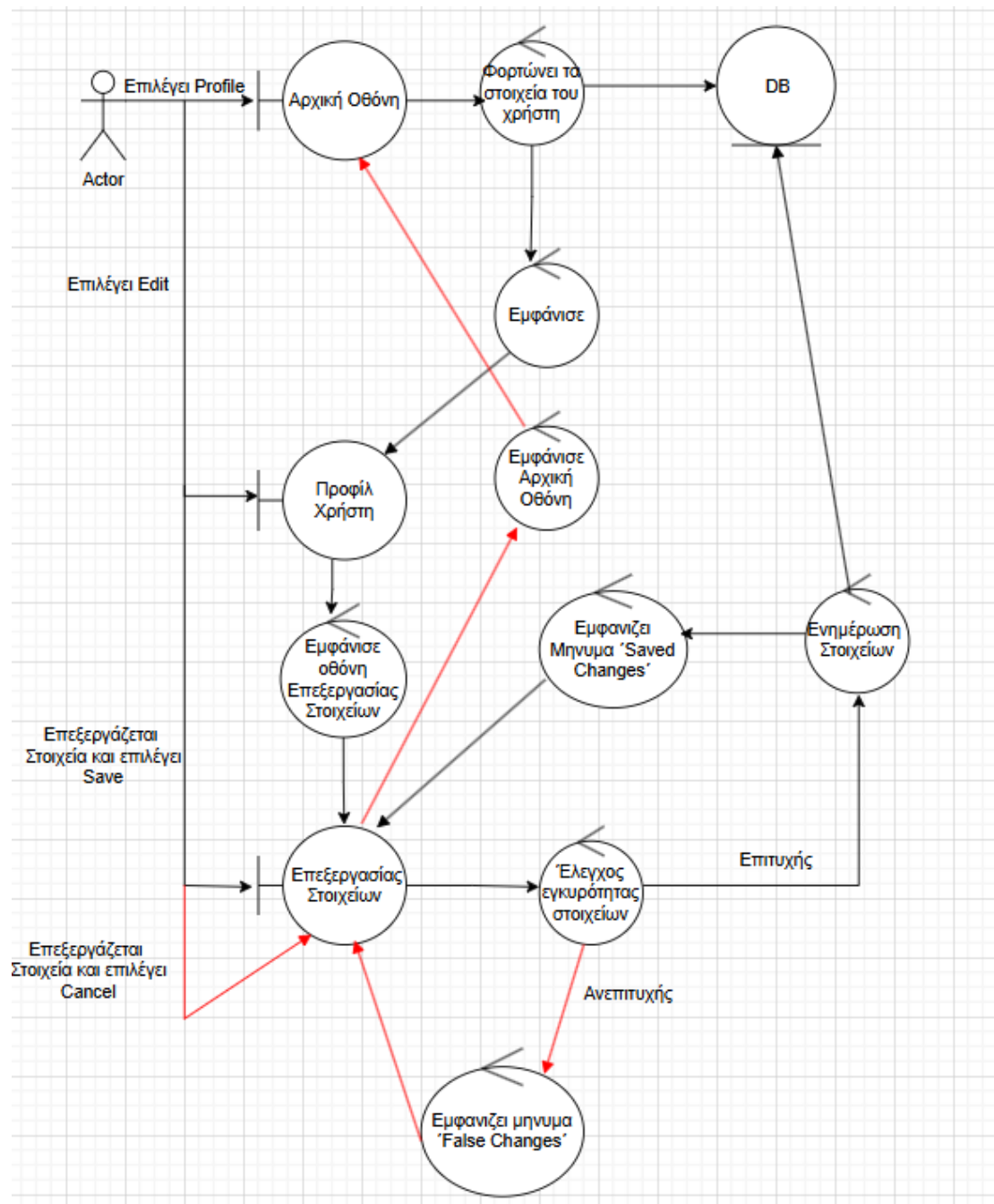
Editor: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Contributor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ, ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ,  
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ

Reviewer: ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ

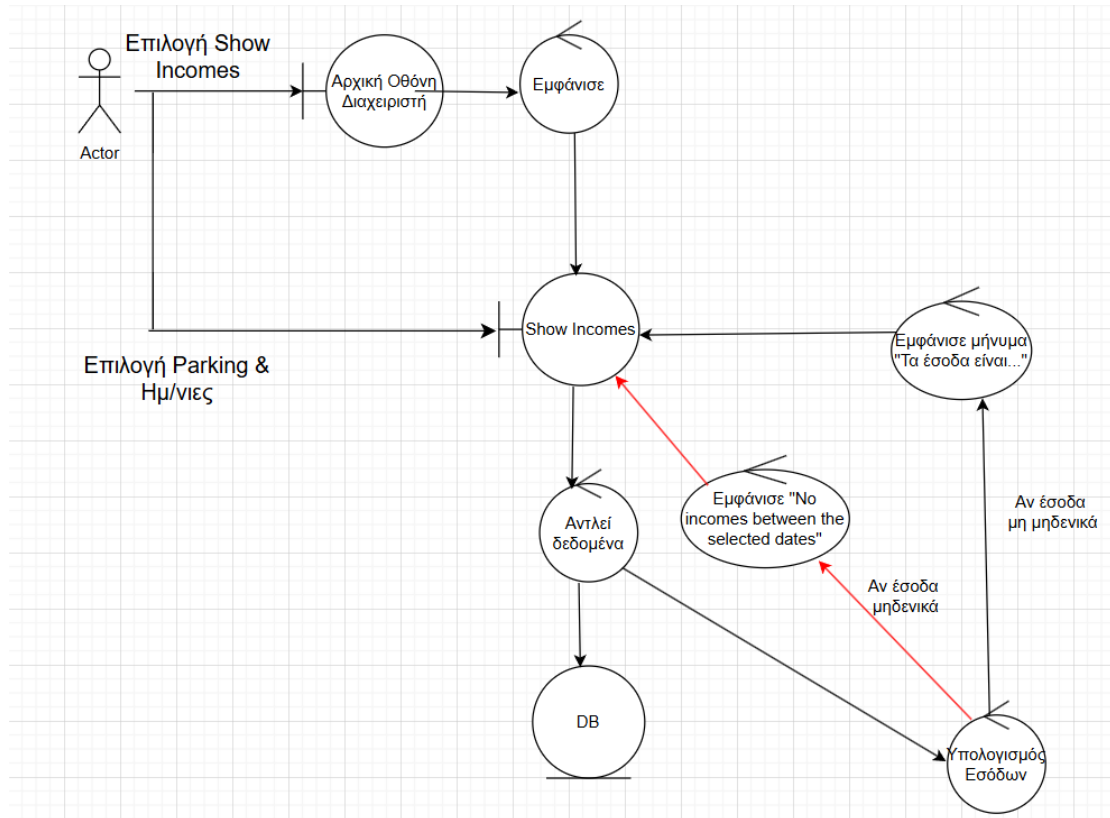
Το v1.0 είναι το ίδιο με το v0.2

### Τίτλος: Προβολή και Επεξεργασία Προφίλ (View and Edit Profile)

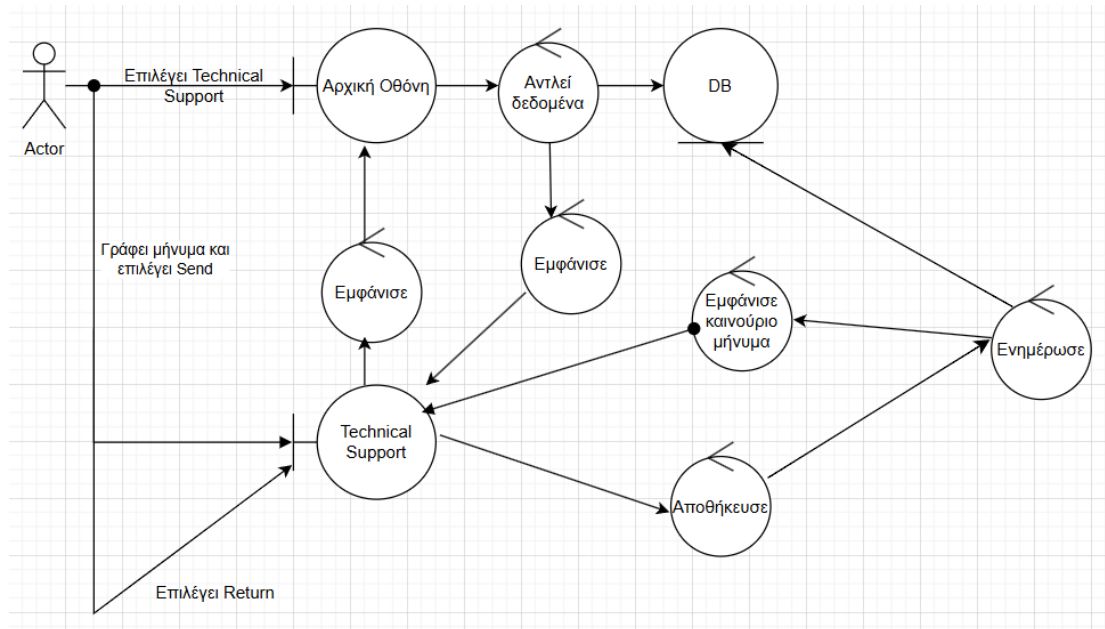




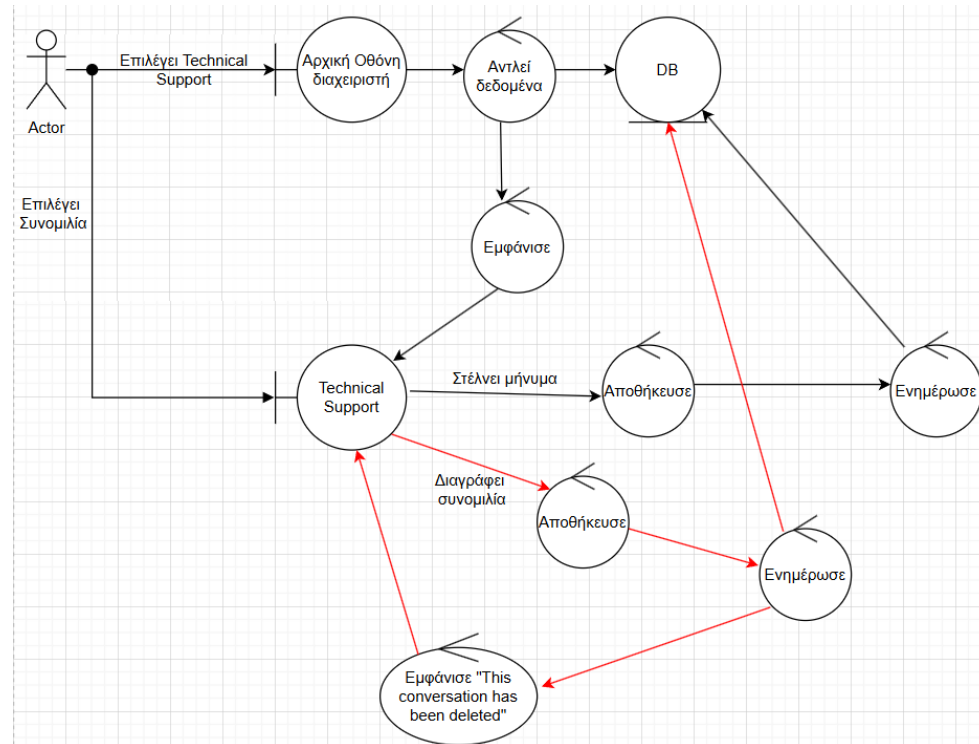
## Τίτλος: Προβολή Εσόδων (Show incomes)



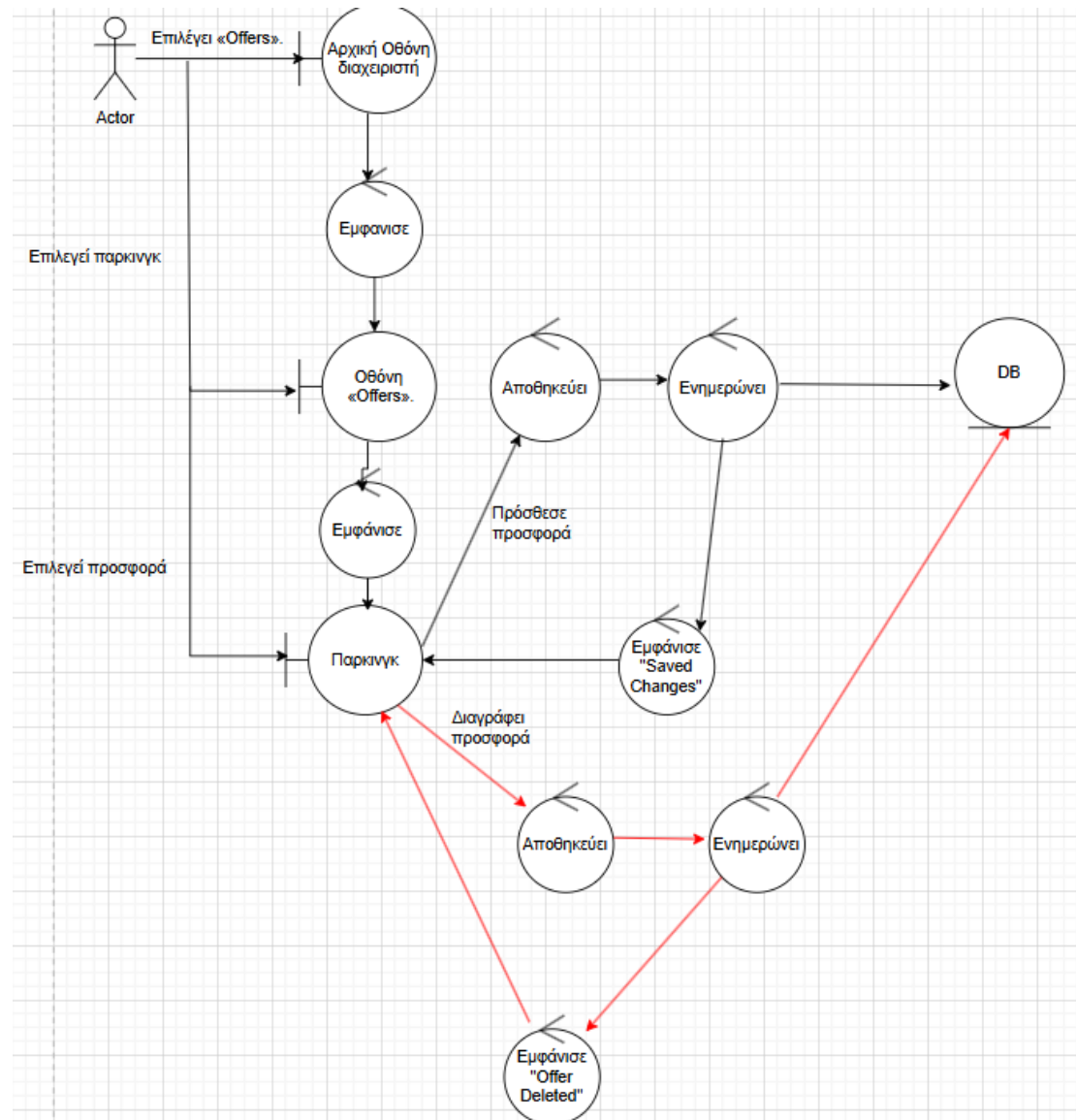
**Τίτλος: Τεχνική υποστήριξη χρήστη (Technical support user).**



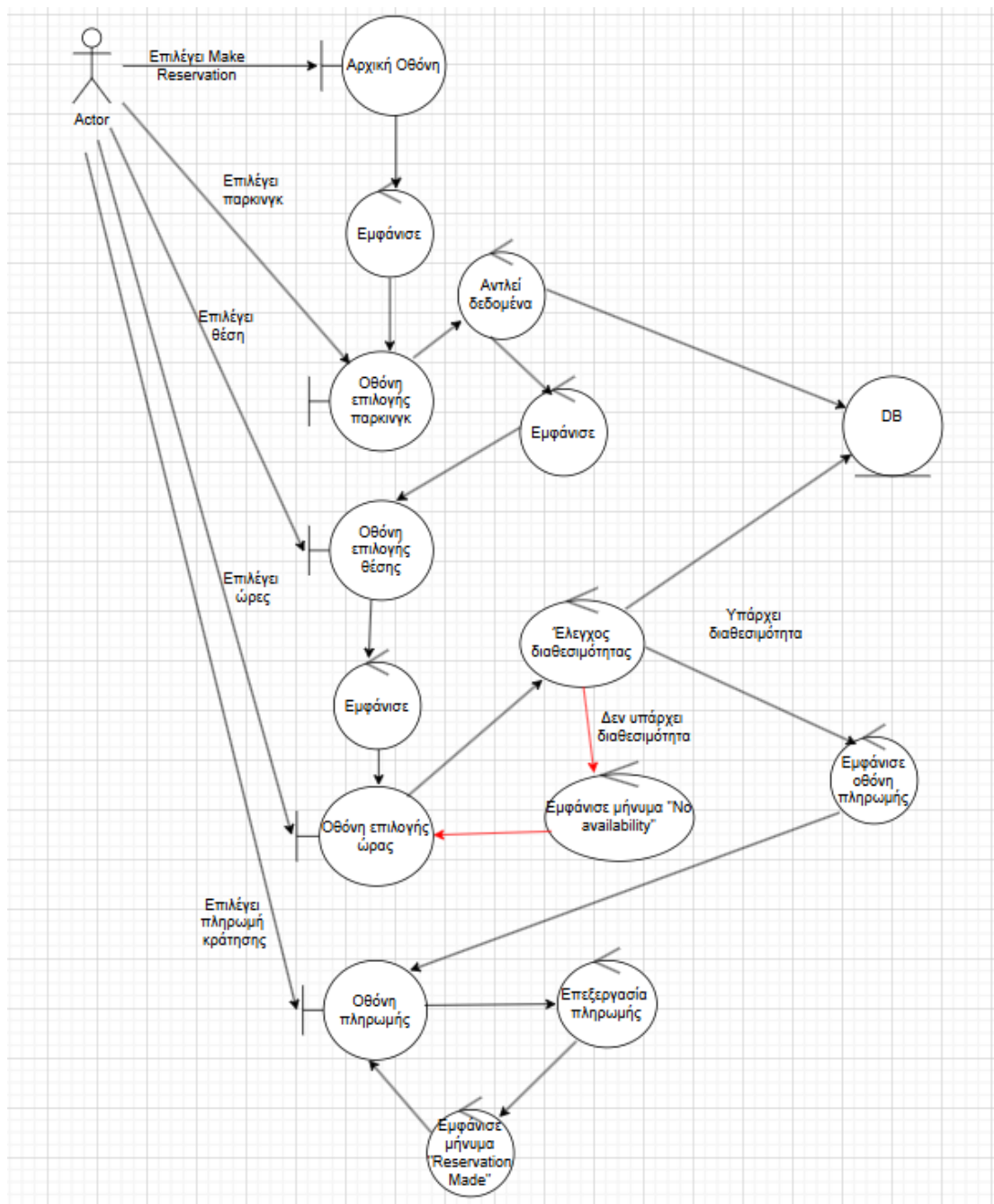
## Τίτλος: Τεχνική Υποστήριξη διαχειριστή (Technical Support Admin)



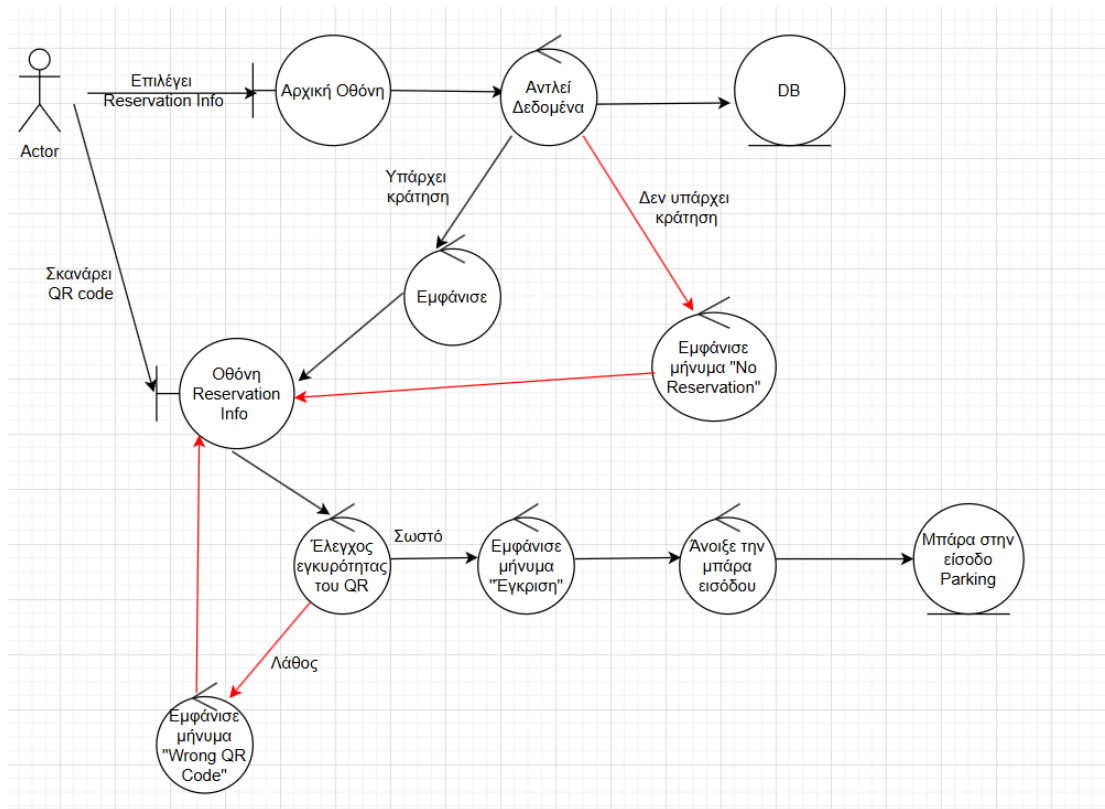
## Τίτλος: Διαχείριση Προσφορών και Εκπτώσεων (Offers)



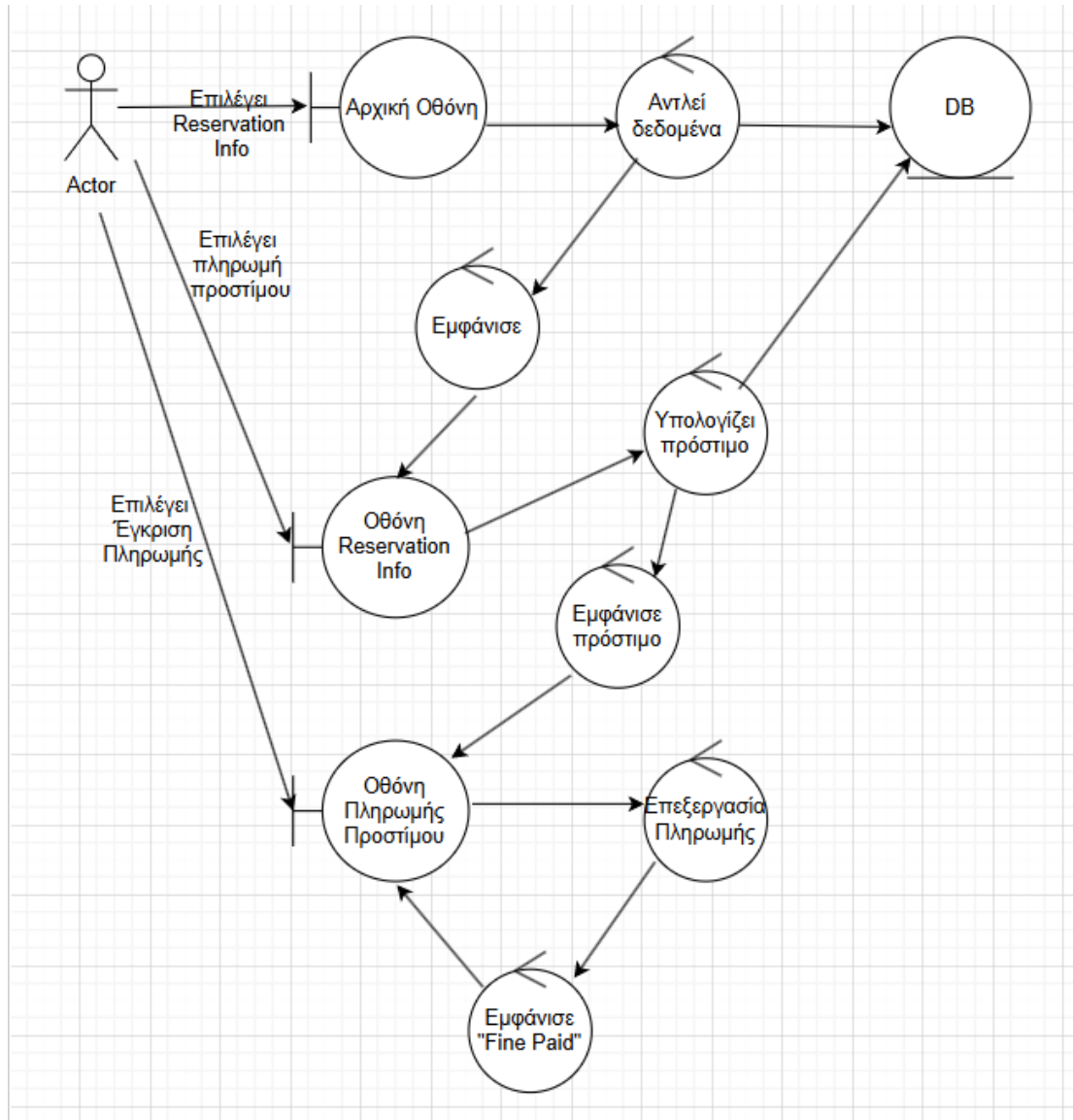
## Τίτλος: Κράτηση Θέσεως παρκινγκ (Make Reservation)



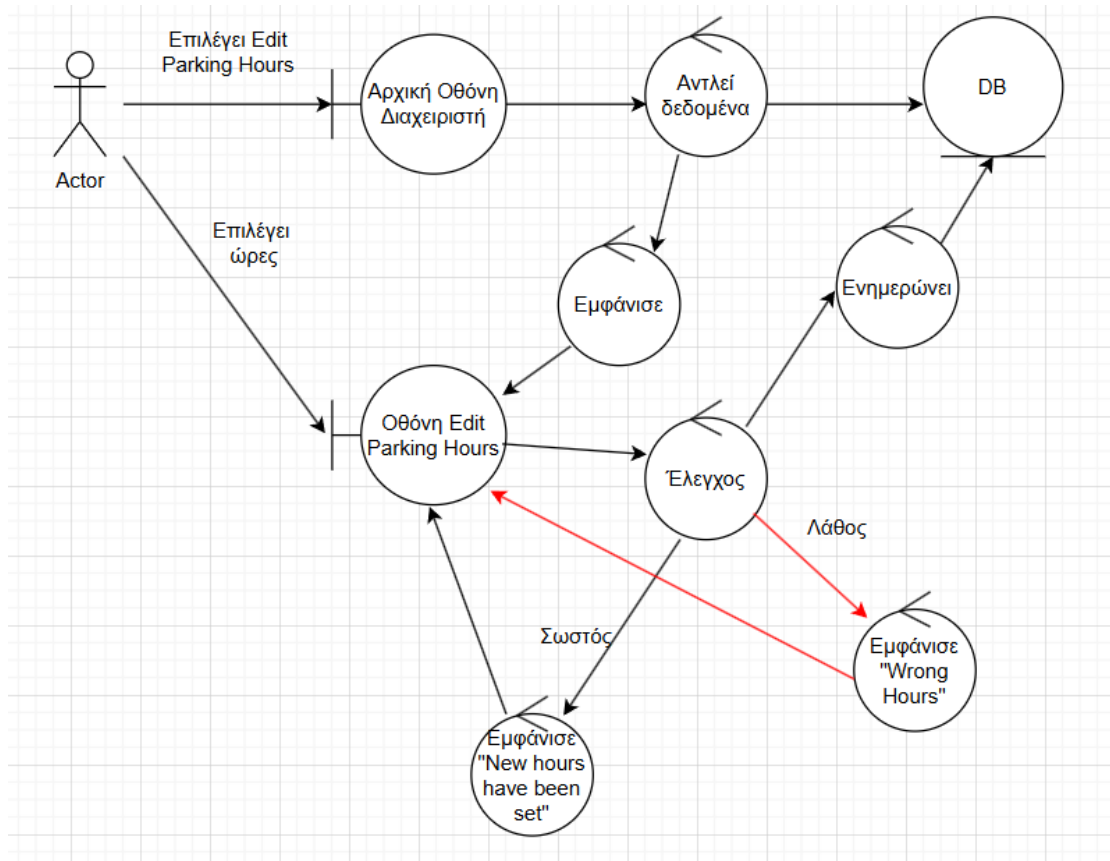
## Τίτλος: Είσοδος/ Έξοδος από το παρκινγκ (Reservation QR Code)



## Τίτλος: Πληρωμή Πρόστιμου (Reservation Fine)



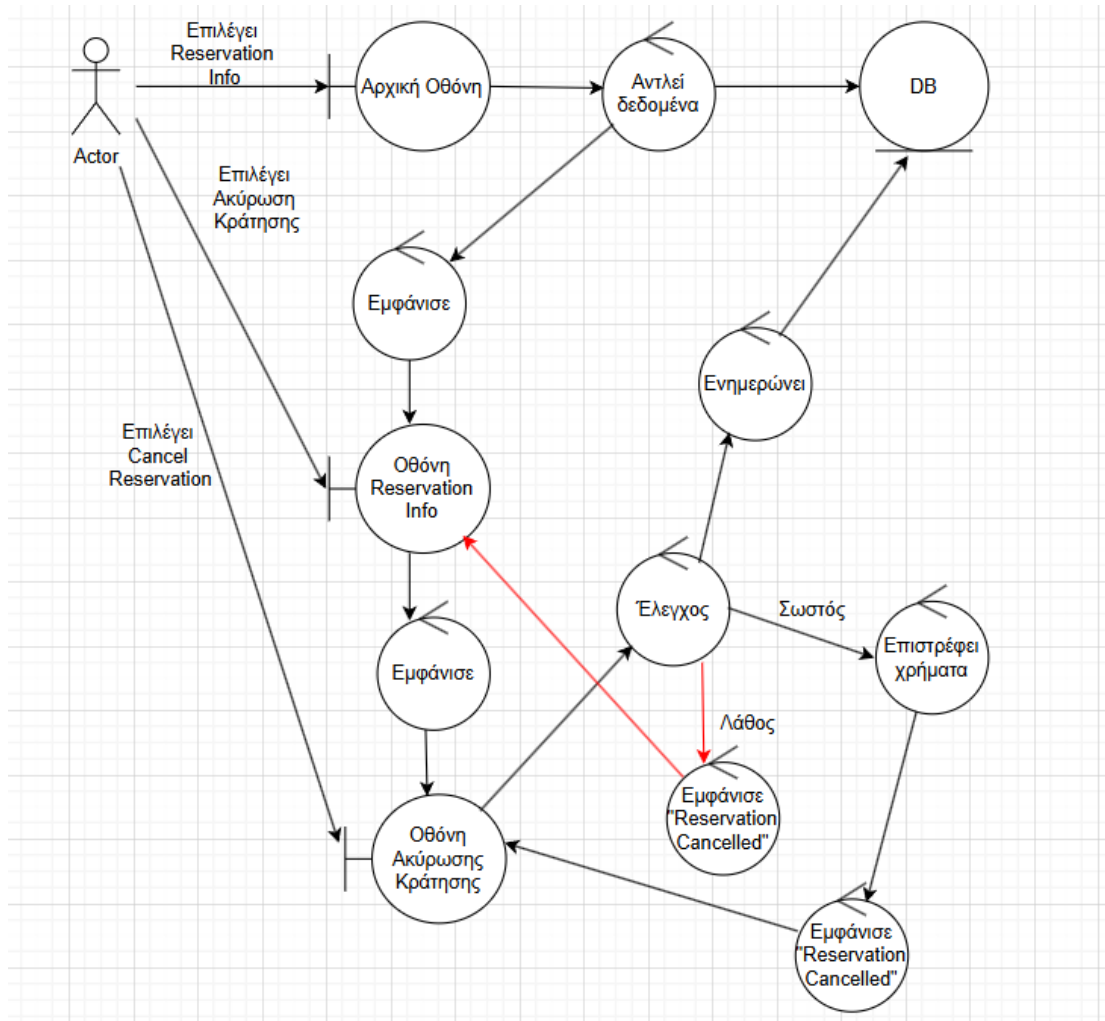
## Τίτλος: Τροποποίηση ωρών λειτουργίας παρκινγκ (Edit Parking Hours)







## Τίτλος: Ακύρωση Κράτησης (Cancel Reservation)



Για τα διαγράμματα χρησιμοποιήσαμε το draw.io.

**Ονομασία Έργου: Smart Park**

**Sequence-Diagrams**

**Έκδοση: v1.0**



Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

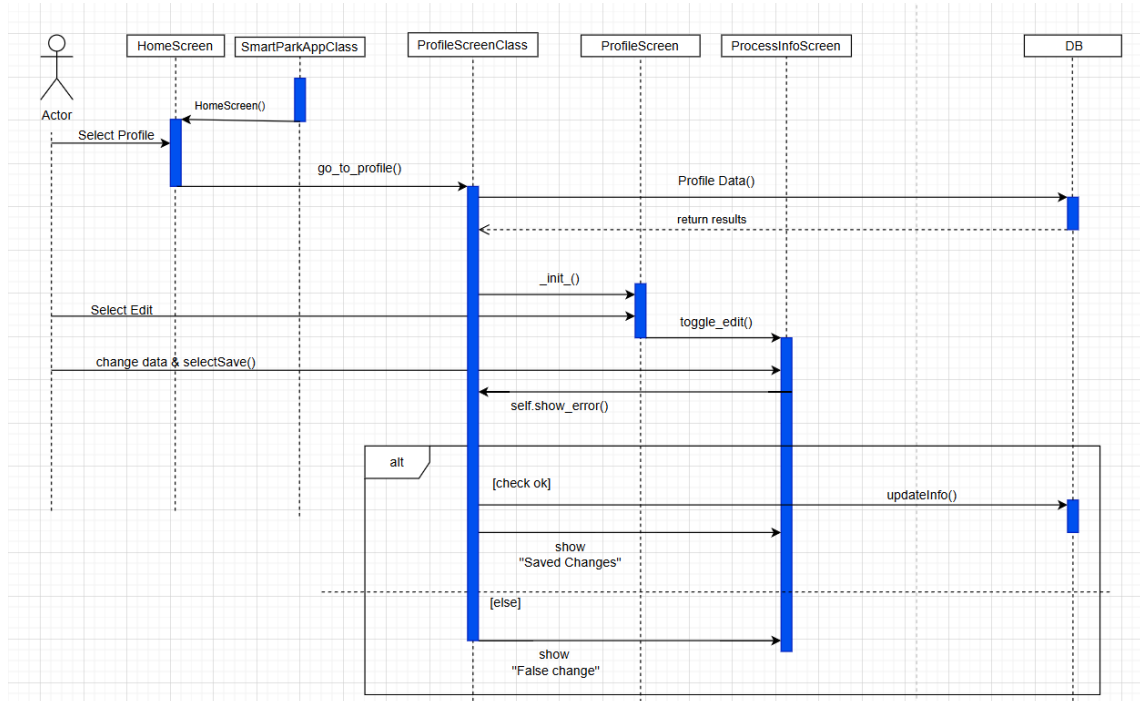
Editor: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Contributor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ, ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ,  
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ

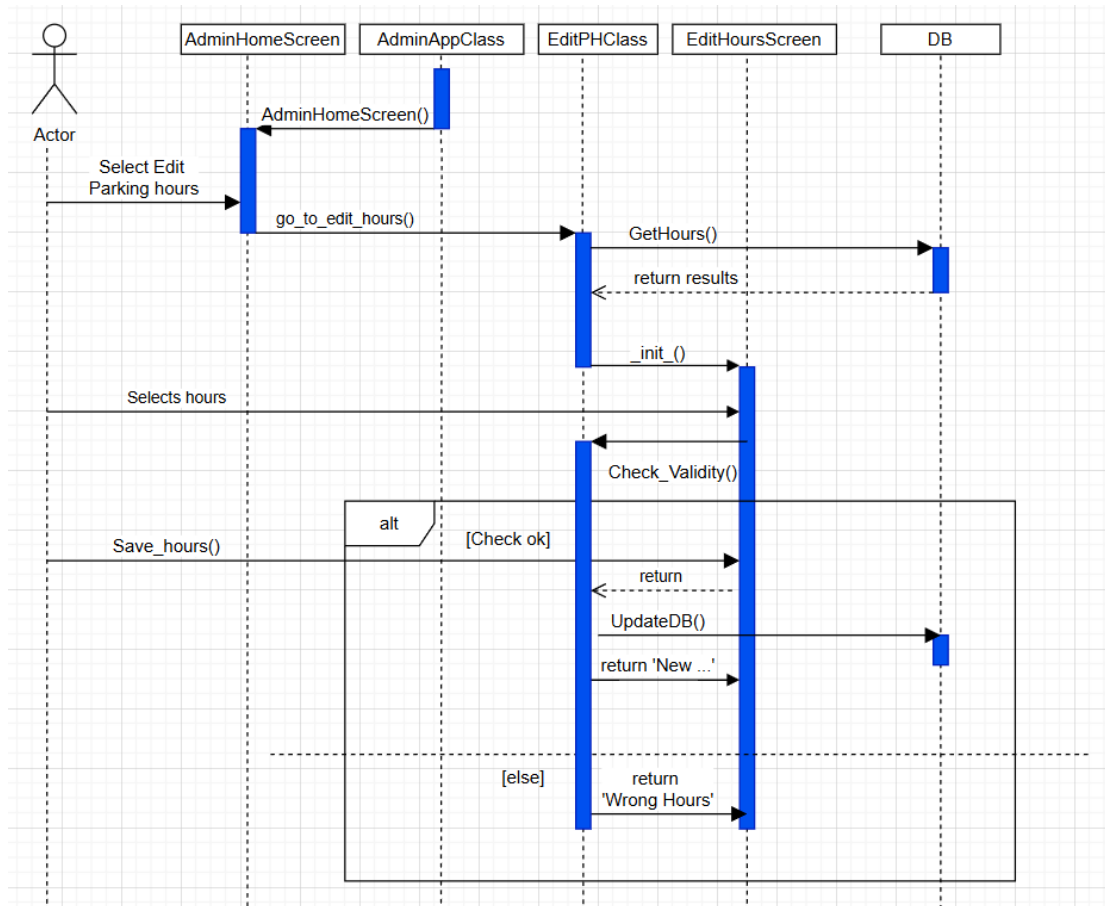
Reviewer: ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ

**Από την έκδοση v0.1 στην έκδοση v1.0 έγιναν αλλαγές σε όλα τα διαγράμματα για συνάδουν καλύτερα τον κώδικα και τις ονομασίες των μεθόδων αλλά και μερικές αλλαγές στην σύνδεση των κλάσεων.**

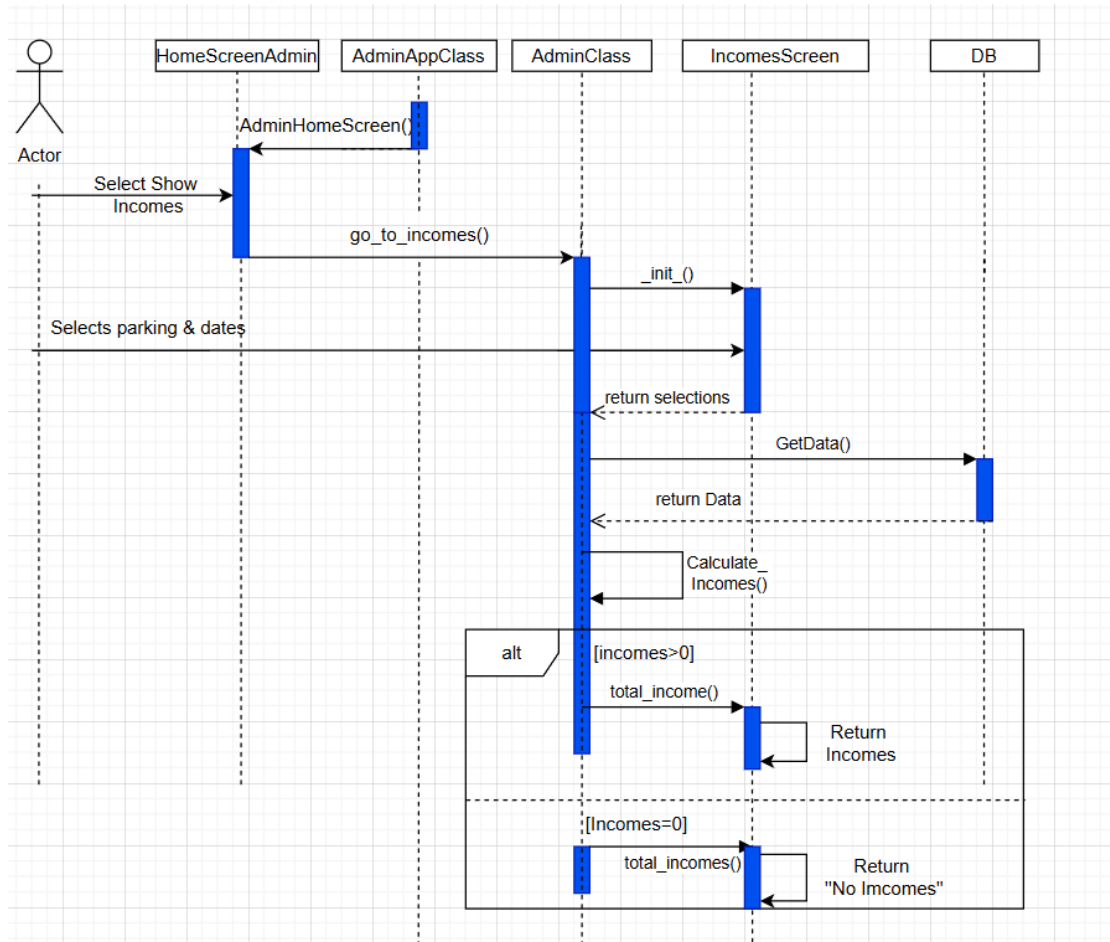
# 1. Προβολή και Επεξεργασία Προφίλ



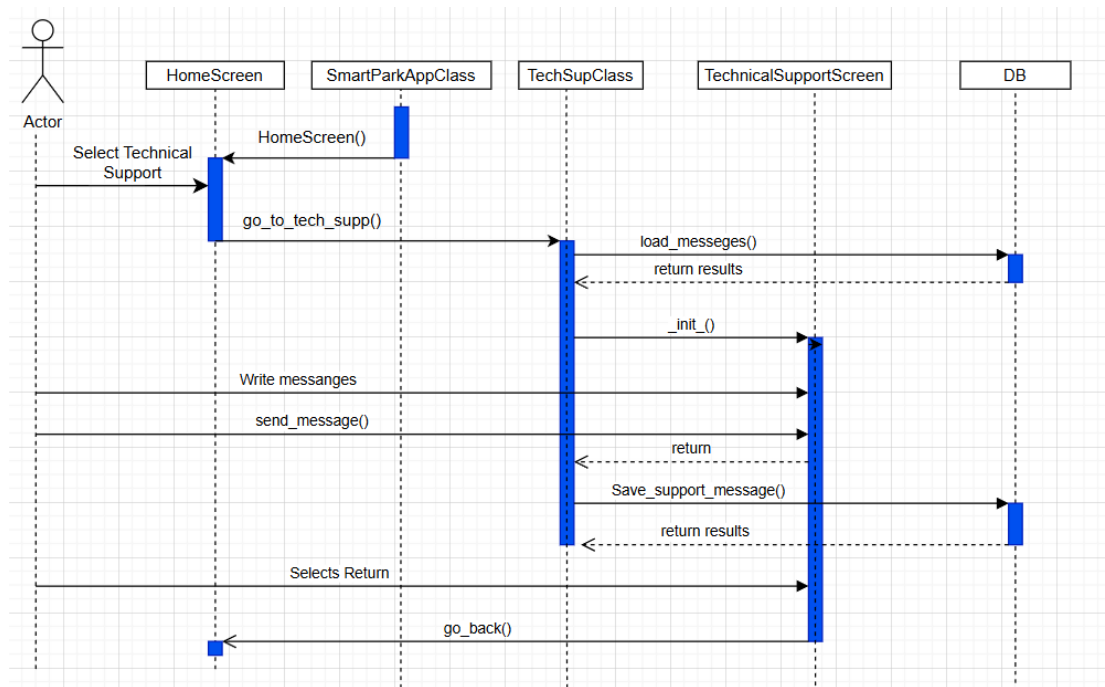
## 2. Τροποποίηση ωρών λειτουργίας παρκινγκ



### 3. Προβολή Εσόδων

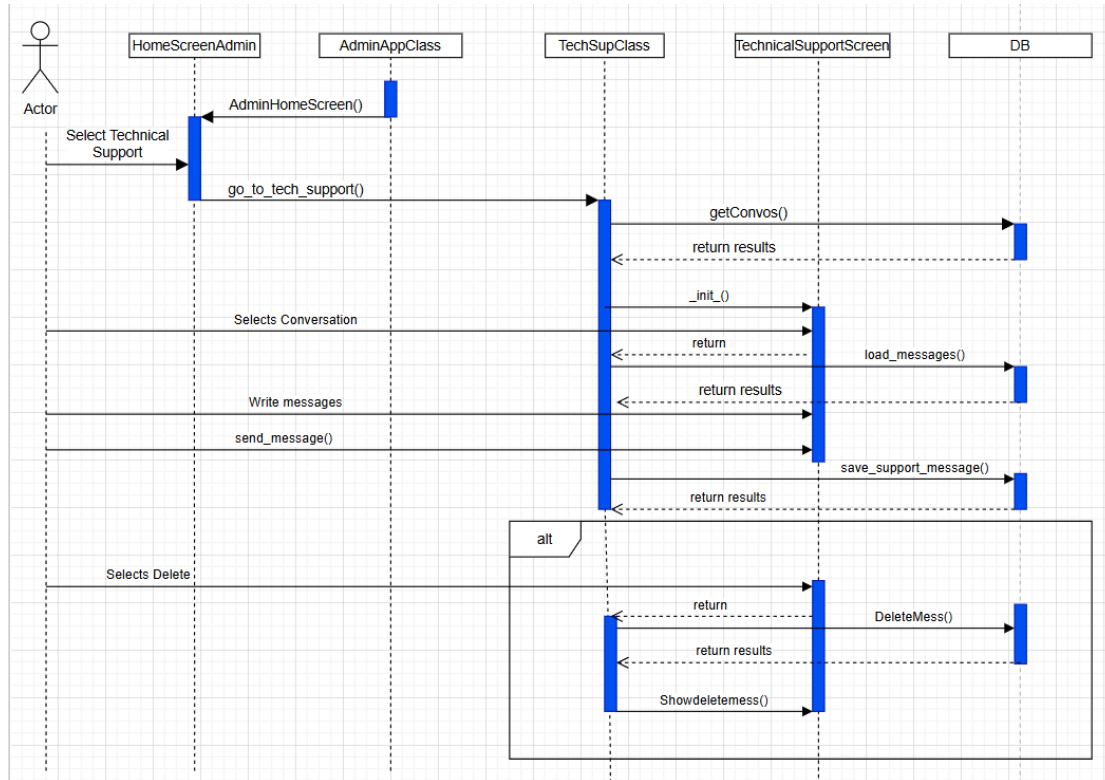


## 4. Technical support (User)

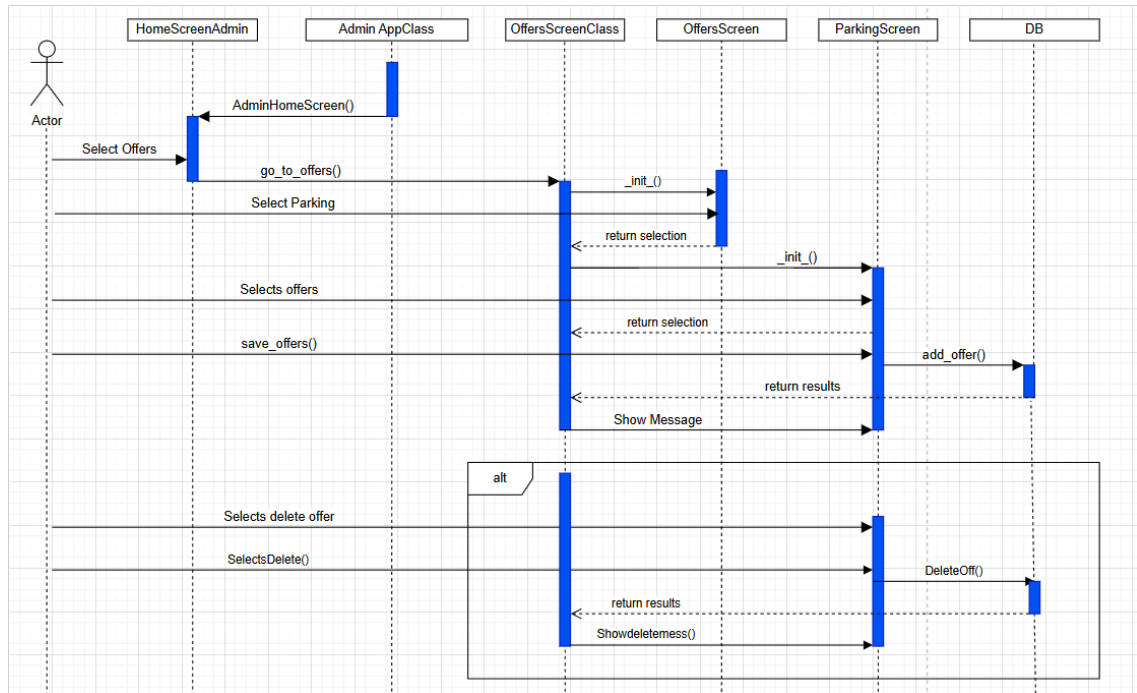




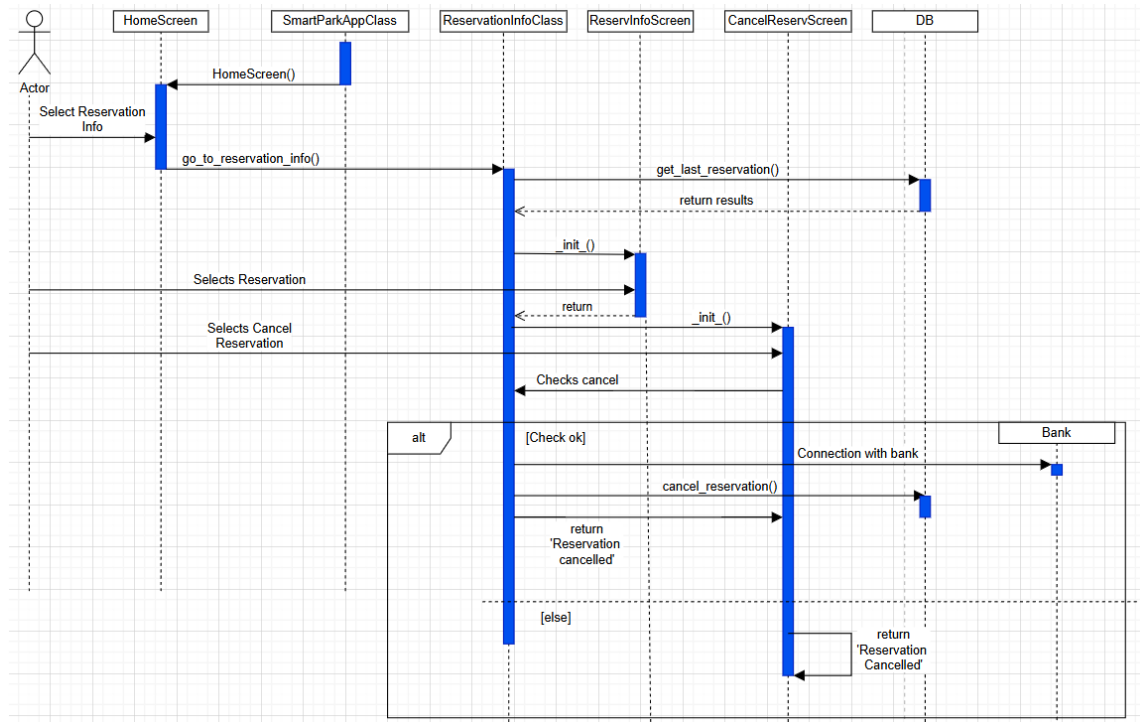
## 4. Technical support (Admin)



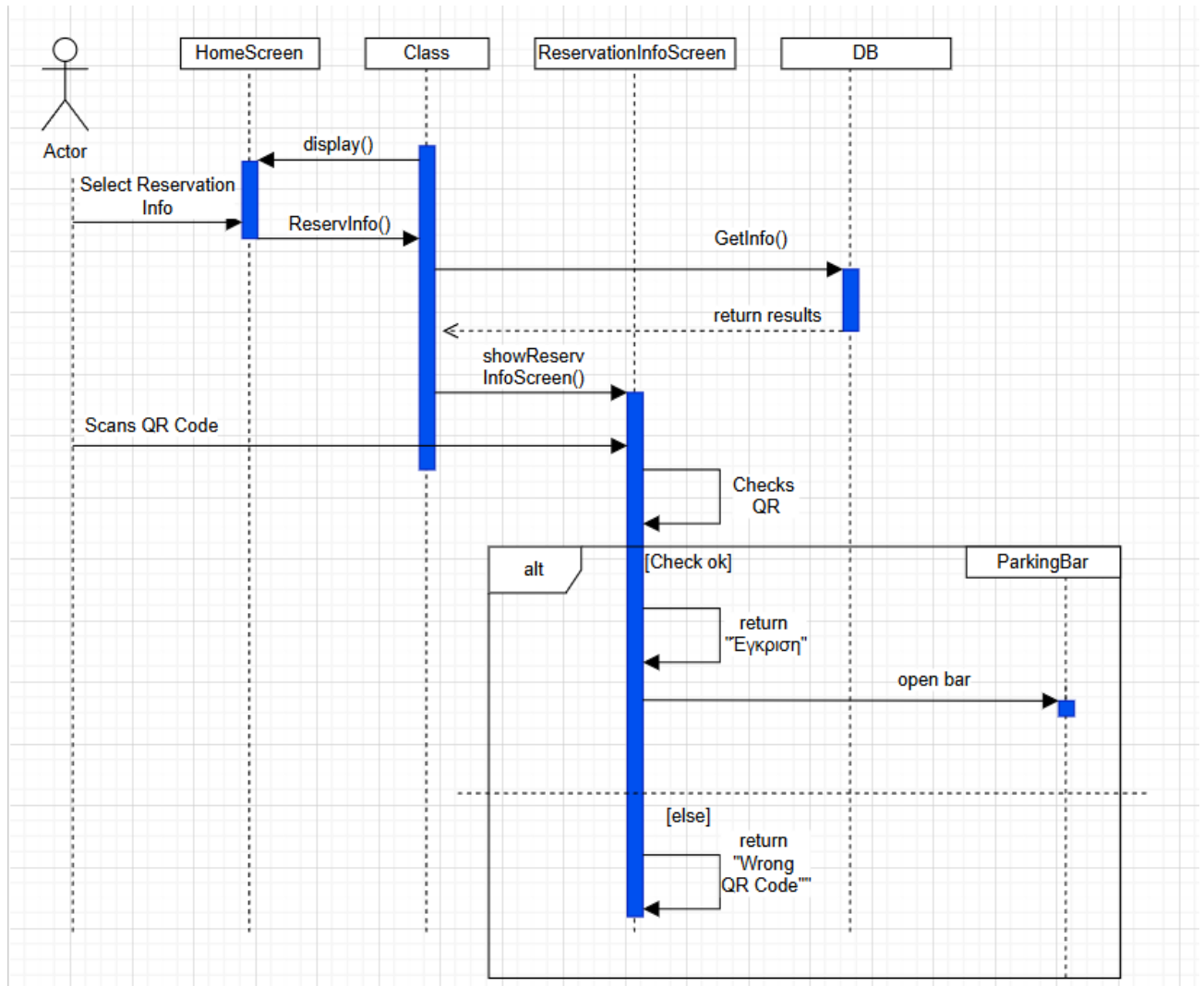
## 5. Διαχείριση Προσφορών και Εκπτώσεων



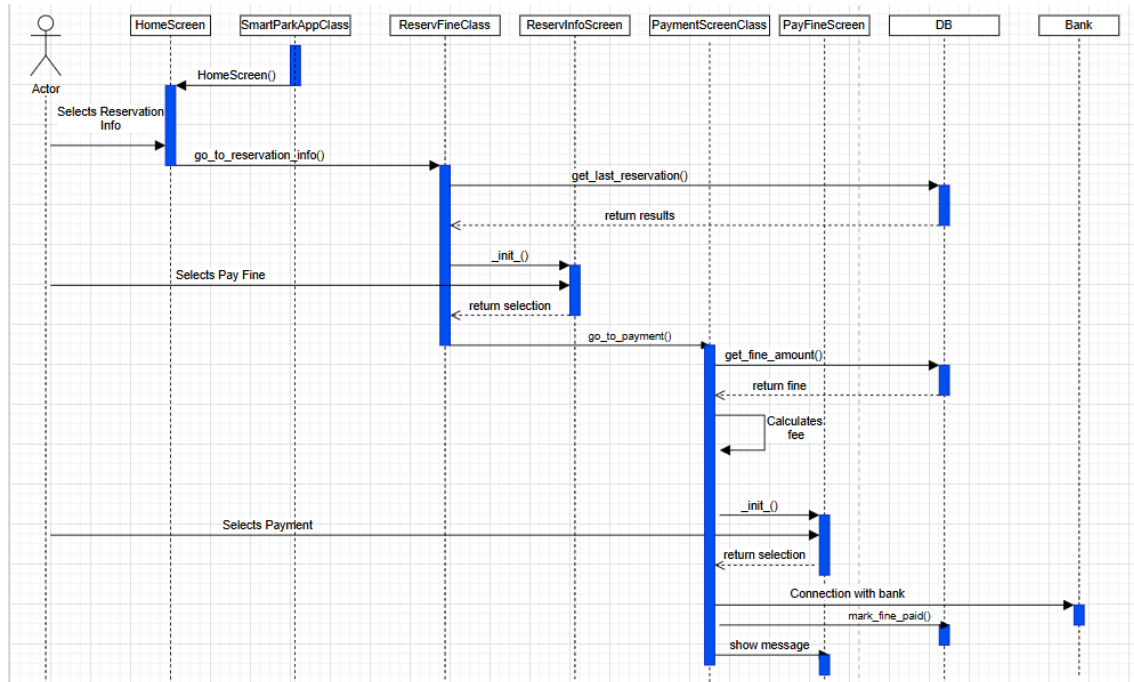
## 6. Ακύρωση Κράτησης



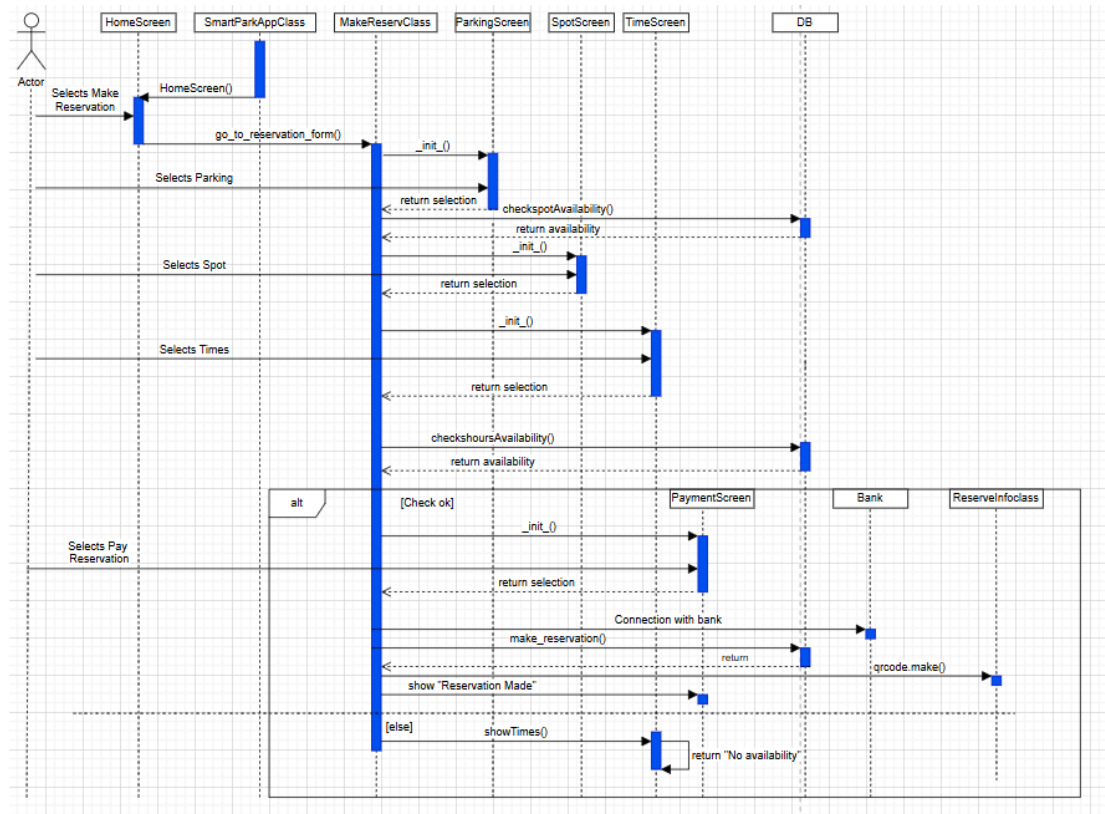
## 7. Είσοδος/Έξοδος από το παρκινγκ



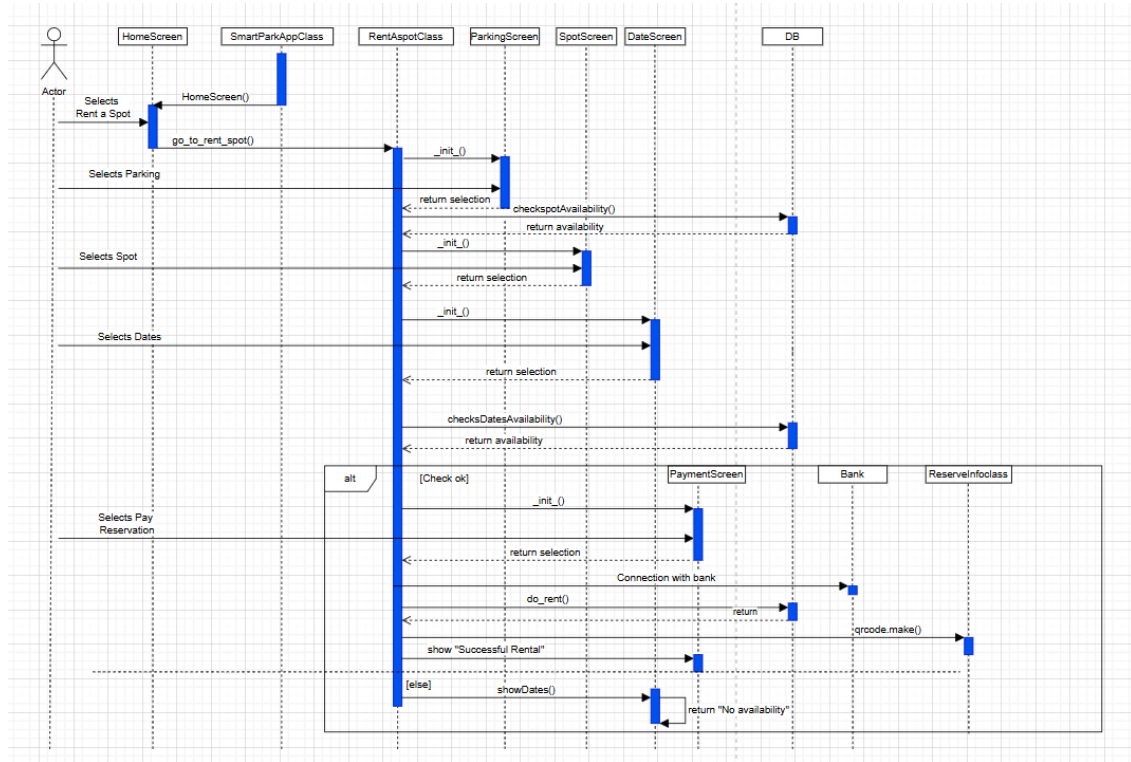
## 8. Πληρωμή Πρόστιμου



## 9. Κράτηση Θέσεως παρκινγκ



## 10. Ενοικίαση θέσης παρκινγκ



Για τα διαγράμματα χρησιμοποιήσαμε το draw.io.

**Ονομασία Έργου: Smart Park**

**Domain-Model**

**Έκδοση: v1.0**





Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

Editor: ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ

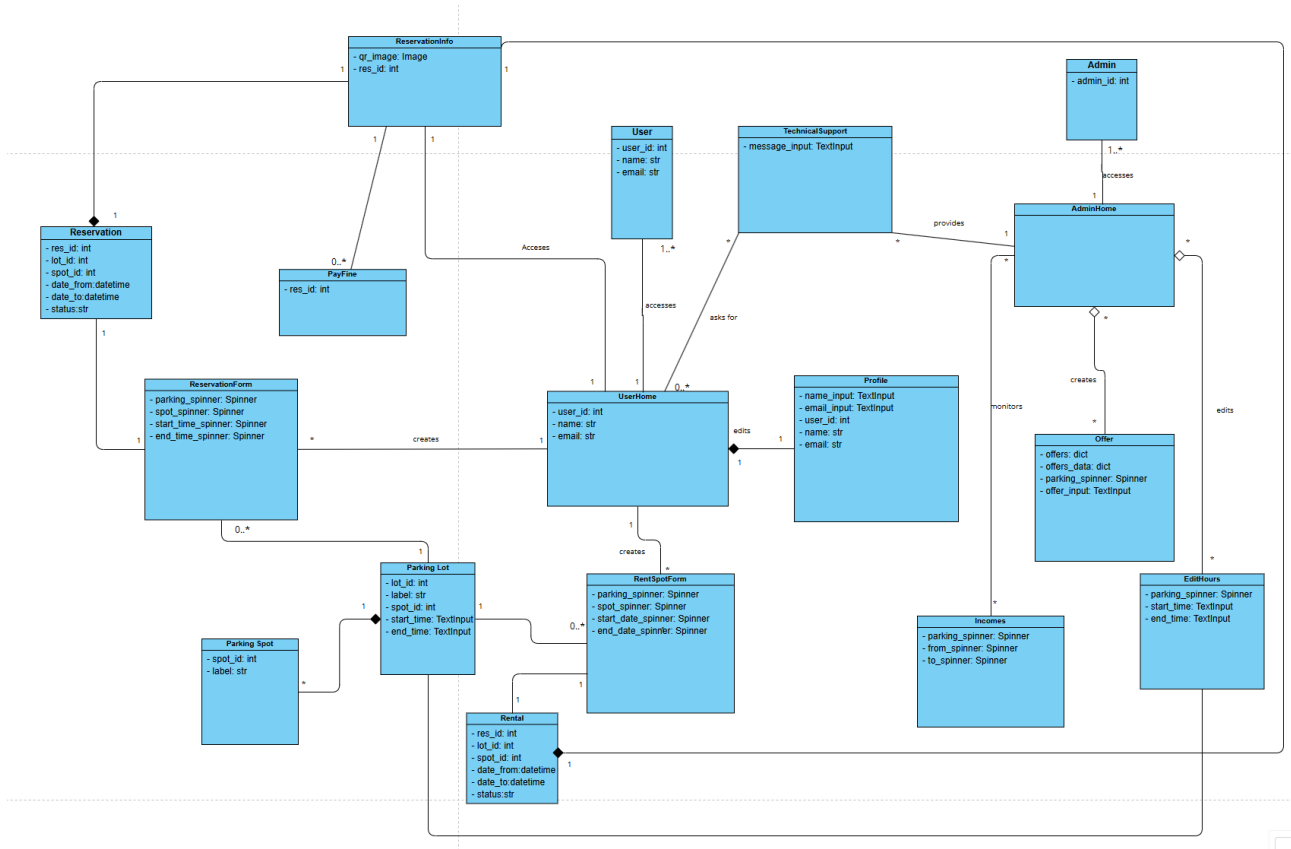
Contributor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ

Reviewer: ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ

Οι κλάσεις που φαίνονται είναι οι εξής:

- User: Γενική κλάση η οποία αναφέρεται στο άτομο που συνδέεται και έχει πρόσβαση σε όλες τις υπηρεσίες (π.χ ReservationInfo, Technical Support κ.α).
- Admin: Κλάση η οποία μπορεί να αλλάξει διάφορα πράγματα στις κλάσεις όπως στο EditHours και να παρέχει υπηρεσίες όπως στο Technical Support.
- ReservationInfo: Κλάση που περιέχει πληροφορίες για μια κράτηση όπως το QR code και περιέχει κουμπιά για ακύρωση κράτησης και πληρωμής προστίμου.
- PayFine: Κλάση που υπολογίζει το ποσό προστίμου σε περίπτωση που υπάρχει.
- Reservation: Κλάση που είναι αντικείμενο της κλάσης ReservationForm και αποθηκεύεται στην Βάση Δεδομένων. Περιέχει id, την θέση που έχει γίνει κράτηση καθώς και την χρονική διάρκεια της.
- ReservationForm: Κλάση που αποθηκεύει μια κράτηση με βάση το παρκινγκ, την θέση και την ώρα.
- Parking Lot: Κλάση που είναι αντικείμενο της κλάσης RentSpotForm και αντιπροσωπεύει ένα parking. Έχει id και όνομα.
- Parking Spot: Κλάση που αντιπροσωπεύει μια θέση παρκινγκ. Ανήκει στο Parking Lot. Έχει id και όνομα.
- Rental: Κλάση που αντιπροσωπεύει μια ενοικίαση που σχετίζεται με RentSpotForm. Περιέχει θέση παρκινγκ με χρονική διάρκεια.
- RentSpotForm: Κλάση που αποθηκεύει μια ενοικίαση με βάση το παρκινγκ, την θέση και την ώρα.
- Offer: Κλάση που επιτρέπει στον διαχειριστή να προσθέσει, να εμφανίσει και να αποθηκεύει προσφορές για όποιο παρκινγκ επιθυμεί.
- Profile: Κλάση εμφανίζει τα στοιχεία του user, ο οποίος μπορεί και να τα επεξεργαστεί.
- Incomes: Κλάση που υπολογίζει τα έσοδα από ένα παρκινγκ με βάση τις επιλογές που έκανε ο admin.
- EditHours: Κλάση για την αλλαγή ωραρίου παρκινγκ. Ο διαχειριστής επιλέγει παρκινγκ και το ωράριο λειτουργίας του.
- Rental: Κλάση που είναι αντικείμενο της κλάσης RentSpotForm και αποθηκεύεται στην Βάση Δεδομένων. Περιέχει id, την θέση που έχει γίνει κράτηση καθώς και την χρονική διάρκεια της.
- Technical Support: Κλάση που σχετίζεται με την υποστήριξη του user από τον admin μέσω συνομιλίας τους και τα μηνύματα αποθηκεύονται στην Βάση Δεδομένων.

## DOMAIN MODEL DIAGRAM



Έγιναν πολλές αλλαγές από το v0.2 στο v1.0 καθώς στα πρώτα στάδια το domain model ήταν πιο ενδεικτικό και λογικό μοντέλο του πως λειτουργεί η εφαρμογή μας. Σε αυτή τη φάση περιγράφουμε λίγο περισσότερα entities και τις σχέσεις τους καθώς σιγά σιγά θα καταλήξουμε στο class diagram. Για το διάγραμμα χρησιμοποιήσαμε το Visual Paradigm.

**Ονομασία Έργου: Smart Park**

**Class-Diagram**

**Έκδοση: v1.0**



Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

Editor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ

Contributor: ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ

Reviewer: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Οι κλάσεις που φαίνονται είναι οι εξής:

**Σε κάθε κλάση υπάρχει method() : \_\_init\_\_() μέσα στην κλάση η οποία δημιουργεί την οθόνη που βρισκόμαστε καθώς δεν έχουμε μια καθολική “base screen κλάση”**

- User: Γενική κλάση η οποία αναφέρεται στο άτομο που συνδέεται και έχει πρόσβαση σε όλες τις υπηρεσίες (π.χ ReservationInfo, Technical Support κ.α).
- Admin: Κλάση η οποία μπορεί να αλλάξει διάφορα πράγματα στις κλάσεις όπως στο EditHours και να παρέχει υπηρεσίες όπως στο Technical Support.

- ReservationInfo: Κλάση που περιέχει πληροφορίες για μια κράτηση όπως το QR code και περιέχει κουμπιά για ακύρωση κράτησης και πληρωμής προστίμου.

Attributes:

-qr\_image: Εικόνα QR

-Buttons: πληρωμή προστίμου, ακύρωση

Methods():

- cancel\_reservation() : Ακυρώνει την κράτηση

- go\_to\_payment() : Μεταφέρει το χρήστη στην οθόνη πληρωμής Προστίμου

- get\_last\_reservation() : Φέρνει απο τη βάση την τελευταία κράτηση

- PayFine: Κλάση που υπολογίζει το ποσό προστίμου σε περίπτωση που υπάρχει.

Attributes:

- id του reservation και το status του

- Button για μετάβαση στην πληρωμή

Methods():

-get\_fine\_amount(): Φορτώνει το ποσό απο τη βάση

-do\_payment(): Αποθηκεύει την πληρωμή και το status του reservation η rental ως paid

- Reservation: Κλάση που είναι αντικείμενο της κλάσης ReservationForm και αποθηκεύεται στην Βάση Δεδομένων. Περιέχει id, την θέση που έχει γίνει κράτηση καθώς και την χρονική διάρκεια της.

- ReservationForm: Κλάση που αποθηκεύει μια κράτηση με βάση το παρκινγκ, την θέση και την ώρα.

Attributes:

- Spinners για parking, θέση και ώρα
- Buttons για έλεγχο διαθεσιμότητας και κράτηση
- result\_label για αποτελέσματα

Methods():

- check\_availability(): Ελέγχει διαθεσιμότητα θέσης
- make\_reservation(): Καταχωρεί την κράτηση»

- **Parking Lot:** Κλάση που είναι αντικείμενο της κλάσης RentSpotForm και αντιπροσωπεύει ένα parking. Έχει id και όνομα.
- **Parking Spot:** Κλάση που αντιπροσωπεύει μια θέση παρκινγκ. Ανήκει στο Parking Lot. Έχει id και όνομα.  
Methods():  
is\_spot\_available(): Ελέγχει απο την Βάση αν η θέση είναι ελεύθερη
- **RentSpotForm:** Κλάση που αποθηκεύει μια ενοικίαση με βάση το παρκινγκ, την θέση και την ώρα.  
Attributes:
  - Spinners για parking, θέση και ώρα
  - Buttons για έλεγχο διαθεσιμότητας και ενοικίαση
  - result\_label για αποτελέσματα
 Methods():
  - check\_availability(): Ελέγχει διαθεσιμότητα θέσης
  - do\_rent(): Καταχωρεί την ενοικίαση
- **Offer:** Κλάση που επιτρέπει στον διαχειριστή να προσθέσει, να εμφανίσει και να αποθηκεύει προσφορές για όποιο παρκινγκ επιθυμεί.  
Attributes:
  - Spinner για parking
  - Buttons για προσθήκη και αποθήκευση
  - offer\_input για προσθήκη προσφοράς
  - offers: dict για αποθήκευση των offers τοπικά της κλάσης
 Methods():
  - add\_offers(): Προσθέτει καινούριο offer
  - save\_offers(): Αποθηκεύει το νέο offer
  - load\_offers(): Κάνει load τα offers απο το dict
- **Profile:** Κλάση που εμφανίζει τα στοιχεία του user, ο οποίος μπορεί και να τα επεξεργαστεί.  
Attributes:
  - name\_input / email\_input
  - name\_label / email\_label
  - edit\_btn: κουμπί εναλλαγής προβολής/επεξεργασίας
 Methods():
  - toggle\_edit(): Εναλλαγή κατάστασης επεξεργασίας
  - show\_error(): Προβολή σφαλμάτων εισόδου
  - build\_static(): Επαναφορά σε προβολή

- **Incomes:** Κλάση που υπολογίζει τα έσοδα από ένα παρκινγκ με βάση τις επιλογές που έκανε ο admin.  
**Attributes:**
  - Spinners για επιλογή Parking, θέσης και ώρας
  - Button για υπολογισμό και Label για αποτέλεσμα**Methods():**
  - GetData ( ) : Φέρνει απο την βάση τα δεδομένα
  - Calculate\_incomes ( ) : Υπολογίζει τα συνολικά έσοδα
- **EditHours:** Κλάση για την αλλαγή ωραρίου παρκινγκ. Ο διαχειριστής επιλέγει πάρκινγκ και το ωράριο λειτουργίας του.  
**Attributes:**
  - Spinners για επιλογή Parking και ώρας
  - Button για αποθήκευση και Label για αποτέλεσμα**Methods():**
  - GetHours ( ) : Φέρνει απο την βάση τα δεδομένα
  - SaveHours ( ) : Αποθηκεύει στη βάση το νέο ωράριο
- **Rental:** Κλάση που είναι αντικείμενο της κλάσης RentSpotForm και αποθηκεύεται στην Βάση Δεδομένων. Περιέχει id, την θέση που έχει γίνει κράτηση καθώς και την χρονική διάρκεια της.
- **Technical Support:** Κλάση που σχετίζεται με την υποστήριξη του user από τον admin μέσω συνομιλίας τους και τα μηνύματα αποθηκεύονται στην Βάση Δεδομένων.  
**Attributes:**
  - chat\_area, chat\_box, message\_input για δημιουργία του chat και εγγραφή μηνυμάτων
  - Button για αποστολή**Methods():**
  - Load\_messages ( ) : Φέρνει απο την βάση τα δεδομένα
  - Save\_support\_message ( ) : Αποθηκεύει στη βάση το μήνυμα
  - Send\_message(): Στέλνει το μήνυμα
- **UserHome:** Η αρχική οθόνη χρήστη για εναλλαγή οθονών με τα buttons της.
- **AdminHome:** Η αρχική οθόνη Διαχειριστή για εναλλαγή οθονών με τα buttons της.



## **CLASS DIAGRAM**



**Ονομασία έργου: Smart Park**

**Test-cases**

**Έκδοση: v1.0**



Μέλος	Αριθμός Μητρώου	Έτος Φοίτησης
ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ	1067439	7 <sup>ο</sup>
ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ	1080700	6 <sup>ο</sup>
ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ	1071109	7 <sup>ο</sup>
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ	1067474	7 <sup>ο</sup>
ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ	1072541	6 <sup>ο</sup>

Editor: ΔΗΜΗΤΡΗΣ ΑΛΒΑΝΟΣ

Contributor: ΗΛΙΑΣ-ΜΑΡΙΟΣ ΠΑΠΑΓΓΕΛΗΣ, ΓΙΩΡΓΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ ,  
ΠΑΣΧΑΛΗΣ ΔΡΟΥΚΟΠΟΥΛΟΣ

Reviewer: ΡΑΦΑΗΛ-ΝΙΚΟΛΑΟΣ ΜΙΧΟΣ

Από την έκδοση v0.1 στην έκδοση v1.0 έγιναν αλλαγές που επισημαίνονται με  
χρώμα μπλε.

## BLACK BOX TESTING

### user test cases

#### Test Case ID 1

Περιγραφή: Άνοιγμα αρχικής σελίδας Χρήστη

Βήματα για εκτέλεση: Εκκίνηση της εφαρμογής

Επιθυμητό αποτέλεσμα: Εμφανίζεται η αρχική οθόνη με 4 κουμπιά

Κατάσταση: pass

- Test Case ID 2

Περιγραφή: Πλοήγηση από Home σε Make Reservation

Βήματα για εκτέλεση: Ο χρήστης πατάει το κουμπί "Make Reservation" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη κράτησης.

Κατάσταση: pass

#### Test Case ID 3

Περιγραφή: Κράτηση Θέσης (rent a spot, make reservation)

Βήματα για εκτέλεση: Ο χρήστης επιλέγει πάρκινγκ, θέση, ημερομηνίες και πατάει "Έλεγχος Διαθεσιμότητας" και εφόσον η θέση είναι διαθέσιμη ο χρήστης πατάει Κράτηση

Input:

Parking A

Θέση 1

Από: 2025-05-20

Έως: 2025-05-22

Επιθυμητό αποτέλεσμα: Εμφανίζεται μήνυμα "Η θέση είναι διαθέσιμη!" και εφόσον ο χρήστης επιλέγει Κράτηση εμφανίζεται μήνυμα επιτυχούς κράτησης

Κατάσταση: pass

Test Case ID 4

Περιγραφή: Επεξεργασία Προφίλ

Βήματα για εκτέλεση: Ο χρήστης αλλάζει το όνομά ή/και το email του από "XXX" σε "YYY".

Επιθυμητό αποτέλεσμα: Το νέο όνομα ή/και email εμφανίζεται στην οθόνη προφίλ.

Input: Πατάει Edit > αλλάζει κείμενο > Save.

Κατάσταση: pass

- Test Case ID 5

Περιγραφή: Πλοήγηση από Home σε Reservation Info

Βήματα για εκτέλεση: Ο χρήστης πατάει το κουμπί "Reservation Info" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη πληροφοριών.

Κατάσταση: pass

- 

- Test Case ID 6

Περιγραφή: Πλοήγηση από Home σε Rent a Spot

Βήματα για εκτέλεση: Ο χρήστης πατάει το κουμπί "Rent a spot" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη κράτησης (rent\_spot).

Κατάσταση: pass

- Test Case ID 7

Περιγραφή: Πλοήγηση από Home σε Technical Support

Βήματα για εκτέλεση: Ο χρήστης πατάει το κουμπί "Techincal Support" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη Τεχνικής υποστήριξης.

Κατάσταση: pass

- Test Case ID 8

Περιγραφή: Πλοήγηση από Home σε Profile

Βήματα για εκτέλεση: Ο χρήστης πατάει το κουμπί "Profile" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη profile.

Κατάσταση: pass

- Test Case ID 9

Περιγραφή: Επιτυχής αποστολή αιτήματος υποστήριξης

Βήματα για εκτέλεση: Ο χρήστης συμπληρώνει μήνυμα και πατάει “Αποστολή”

Επιθυμητό αποτέλεσμα: Εμφανίζεται το μήνυμα που στείλατε και απο κάτω ενα ακόμα μήνυμα: “Ευχαριστούμε για το μήνυμά σας! Θα σας βοηθήσουμε σύντομα.”

Input: Πληκτρολόγηση μηνύματος > Αποστολή

Κατάσταση: pass

- Test Case ID 10

Περιγραφή: Ακύρωση κράτησης

Βήματα για εκτέλεση: Ο χρήστης επιλέγει το κουμπί ακύρωσης κράτησης.

Επιθυμητό αποτέλεσμα: Η κράτηση ακυρώνεται, εμφανίζεται μήνυμα Reservation Cancelled

Κατάσταση: pass

- Test Case ID 11

Περιγραφή: Πληρωμή Προστίμου

Βήματα για εκτέλεση: Ο χρήστης επιλέγει το κουμπί πληρωμής προστίμου και εφόσον οφείλει κάποιο ποσό τον μεταφέρει σε νέα σελίδα και επιλέγει το κουμπί έγκριση πληρωμής.

Επιθυμητό αποτέλεσμα: Το ποσό πληρώνεται , εμφανίζεται μήνυμα ‘Η πληρωμή ολοκληρώθηκε με επιτυχία!’

Κατάσταση: pass

Πίνακας 1. user test cases

ID	1	2	3	4	5	6	7	8	9	10	11
RESULT	pass	pass	pass	pass	pass	pass	pass	pass	pass	pass	pass

### admin test cases

- Test Case ID 1

Περιγραφή: Πλοήγηση από Home σε Technical Support

Βήματα για εκτέλεση: Ο admin πατάει το κουμπί "Techincal Support" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη Τεχνικής υποστήριξης.

Κατάσταση: pass



- Test Case ID 2

Περιγραφή: Επιτυχής αποστολή μηνύματος στους χρήστες.

Βήματα για εκτέλεση: Ο admin συμπληρώνει μήνυμα και πατάει “Αποστολή”

Επιθυμητό αποτέλεσμα: Εμφανίζεται το μήνυμα που στείλατε καθώς και όλα τα μηνύματα της συνομιλίας με τον χρήστη

Input: Πληκτρολόγηση μηνύματος > Αποστολή

Κατάσταση: pass

- Test Case ID 3

Περιγραφή: Διαγραφή μηνυμάτων.

Βήματα για εκτέλεση: Ο admin επιλέγει το μήνυμα που θέλει να διαγράψει και πατάει ‘διαγραφή’.

Επιθυμητό αποτέλεσμα: Το μήνυμα της επιλογής εξαφανίζεται απο το chat.

Input: επιλογή μηνύματος > Διαγραφή

Κατάσταση: fail

- Test Case ID 4

Περιγραφή: Πλοήγηση από Home σε show incomes

Βήματα για εκτέλεση: Ο admin πατάει το κουμπί "show incomes" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη show incomes.

Κατάσταση: pass

- Test Case ID 5

Περιγραφή: Πλοήγηση από Home σε Edit Parking Hours

Βήματα για εκτέλεση: Ο admin πατάει το κουμπί "Edit Parking Hours" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη Parking Hours.

Κατάσταση: pass

- Test Case ID 6

Περιγραφή: Πλοήγηση από Home σε Offers

Βήματα για εκτέλεση: Ο admin πατάει το κουμπί "Offers" από την αρχική σελίδα.

Επιθυμητό αποτέλεσμα: Η εφαρμογή αλλάζει στην οθόνη Offers.

Κατάσταση: pass

- Test Case ID 7

Περιγραφή: Δημιουργία Προσφορών

Βήματα για εκτέλεση: Ο admin επιλέγει το parking στο οποίο θέλει να προσθέσει την προσφορά, δημιουργεί μια νέα προσφορά και την αποθηκεύει.

Επιθυμητό αποτέλεσμα: Αποθηκεύονται οι νέες προσφορές.

Κατάσταση: pass

- Test Case ID 8

Περιγραφή: Προβολή εσόδων

Βήματα για εκτέλεση: Ο admin επιλέγει το parking στο οποίο θέλει να παρακολουθήσει τα έσοδά του και το διάστημα το οποίο τον ενδιαφέρει και επιλέγει ‘Υπολογισμός Εσόδων’

Επιθυμητό αποτέλεσμα: Εμφανίζεται το Σύνολο Εσόδων του.

Κατάσταση: pass

- 
- 
- 
- Test Case ID 9

Περιγραφή: Διαχείριση ωρών λειτουργίας

Βήματα για εκτέλεση: Ο admin επιλέγει το parking στο οποίο θέλει να ρυθμίσει το ωράριο , τις ώρες τις οποίες θέλει να λειτουργεί και τα αποθηκεύει.

Επιθυμητό αποτέλεσμα: Εμφανίζεται το νέο ωράριο στο parking επιλογής.

Κατάσταση: pass

Πίνακας 2. admin test cases

ID	1	2	3	4	5	6	7	8	9
RES ULT	pass	pass	pass	pass	pass	pass	pass	pass	pass

## **UNIT TESTING**

Πιο συγκεκριμένα χρησιμοποιήσαμε:

- τη βιβλιοθήκη unittest της Python
- μεμονωμένα test cases για κάποιες σημαντικές λειτουργίες του συστήματος

## USER

### Προφιλ χρήστη

- Μπαίνουμε σε edit mode (toggle\_edit)
- Αλλάζουμε το όνομα σε "Νέο Όνομα"
- Ξαναπατάμε toggle\_edit για να κάνουμε save
- Ελέγχουμε αν το Label εμφανίζει τη νέα τιμή

```
def test_profile_edit_toggle(self):  
    screen = ProfileScreen(self.sm)  
    screen.toggle_edit(None) # ενεργοποίηση edit  
    screen.name_input.text = "Νέο Όνομα"  
    screen.toggle_edit(None) # αποθήκευση αλλαγών  
    self.assertIn("Νέο Όνομα", screen.name_label.text)
```

### Ακύρωση κράτησης

-Δημιουργούμε την ReservationInfoScreen

-Πατάμε cancel\_reservation()

-Ψάχνουμε στα widgets αν υπάρχει Label με την ένδειξη **"Cancelled"**

```
def test_reservation_cancel(self):  
    screen = ReservationInfoScreen(self.sm)  
    screen.cancel_reservation(None)  
    self.assertTrue(any("Cancelled" in w.text for w in screen.children if isinstance(w, Label)))
```

### Κράτηση Θέσης (make reservation)

-Δημιουργούμε Κράτηση με dummy δεδομένα

-Ελέγχουμε αν υπάρχει στη βάση

```
def test_create_reservation(self):
```

```
    # Δημιουργία κράτησης
```

```
    parking = "Parking A"
```

```
    spot = "Θέση 1"
```

```
    date_from = "8:00"
```

```
date_to = "14:00"

save_reservation(parking, spot, date_from, date_to)

conn = sqlite3.connect(DB_PATH)

cursor = conn.cursor()


# Έλεγχος αν υπάρχει στη βάση


cursor.execute("SELECT * FROM reservations")

rows = cursor.fetchall()

conn.close()

self.assertEqual(len(rows), 1)

self.assertEqual(rows[0][1], parking)

self.assertEqual(rows[0][2], spot)

self.assertEqual(rows[0][3], date_from)

self.assertEqual(rows[0][4], date_to)
```

- Δημιουργούμε Κράτηση με dummy δεδομένα
- Πληρωμή προστίμου
- Ελέγχουμε αν έγινε marked στην βάση σαν paid

```
def test_pay_fine_updates_status(self):  
    # Δημιουργία κράτησης  
    save_reservation("Parking A", "Θέση 1", "08:00", "10:00")  
  
    # Πάρε το ID της τελευταίας κράτησης  
    self.cursor.execute("SELECT id FROM reservations ORDER BY id DESC  
LIMIT 1")  
    reservation_id = self.cursor.fetchone()[0]  
  
    # Πλήρωσε το πρόστιμο  
    mark_fine_paid(reservation_id)  
  
    # Έλεγχος αν ενημερώθηκε το status στη βάση δεδομένων  
    self.cursor.execute("SELECT status FROM reservations WHERE id = ?",  
(reservation_id,))  
    status = self.cursor.fetchone()[0]  
    self.assertEqual(status, "paid")
```

ADMIN

### Μήνυμα τεχνικής υποστήριξης

-Στέλνουμε μήνυμα απο τον admin.

-Ελέγχουμε αν υπάρχει το μήνυμα του admin στην βάση.

```
def test_admin_sends_support_message(self):  
    screen = AdminTechSupportScreen(self.sm)  
    screen.message_input.text = "Μήνυμα από admin"  
    screen.send_message(None)  
  
    # Έλεγχος αν αποθηκεύτηκε το μήνυμα του admin  
    messages = load_support_messages()  
  
    self.assertTrue(any(m[0] == 'admin' and m[1] == "Μήνυμα από admin" for  
m in messages))
```

### Προσθήκη προσφοράς

-Δημιουργούμε μια προσφορά και την αποθηκεύουμε.

-Ελέγχουμε αν υπάρχει η προσφορά στο dictionary.



```
def test_add_offer_text(self):  
    screen = OffersScreen(self.sm)  
    screen.parking_spinner.values = ("Parking A",)  
    screen.parking_spinner.text = "Parking A"  
    screen.offer_input.text = "30% έκπτωση"  
    screen.add_offer(None)  
    self.assertEqual(screen.offers_data["Parking A"], ["30% έκπτωση"])
```

### Διαχείριση ωρών λειτουργίας

-Ελέγχουμε αν το αντικείμενο screen έχει *attributes* με τα ονόματα "start\_time" και "end\_time".

```
def test_edit_hours_fields_exist(self):  
    screen = EditHoursScreen(self.sm)  
    self.assertTrue(hasattr(screen, "start_time"))  
    self.assertTrue(hasattr(screen, "end_time"))
```