



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
«ΠΡΟΗΓΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΚΗΣ – ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ & ΤΕΧΝΗΤΗΣ
ΝΟΗΜΟΣΥΝΗΣ»

ΑΠΑΛΛΑΚΤΙΚΗ ΕΡΓΑΣΙΑ
Μάθημα : Ειδικά Θέματα Τεχνολογίας Λογισμικού
ΔΙΔΑΣΚΟΥΣΑ : Καθηγήτρια Μ. Βίρβου

| | |
|------------------------|-----------|
| Ασημακόπουλος Γεώργιος | ΜΠΣΠ20004 |
| Κασώτης Δημήτριος | ΜΠΣΠ20017 |

ΠΕΙΡΑΙΑΣ
Ακαδημαϊκό Έτος 2020-2021

Περιεχόμενα

| | | |
|-------|--|----|
| 1 | Εισαγωγή..... | 3 |
| 1.1 | Στόχοι της εργασίας..... | 3 |
| 1.2 | Ορισμός του προβλήματος προς επίλυση..... | 3 |
| 2 | Σύντομη παρουσίαση της RUP..... | 4 |
| 3 | Φάση: Έναρξη (Inception) | 6 |
| 3.1 | Σύλληψη απαιτήσεων | 6 |
| 3.1.1 | Ανάλυση αναγκών (Needs Analysis) | 6 |
| 3.1.2 | Ανάλυση Χρηστών (User Analysis) | 6 |
| 3.1.3 | Ανάλυση Εργασιών (Task Analysis) | 7 |
| 3.2 | Ανάλυση Σχεδιασμός | 7 |
| 3.2.1 | Περιπτώσεις Χρήσης - Use Case..... | 7 |
| 3.2.2 | Διάγραμμα Τάξεων..... | 8 |
| 4 | Φάση: Εκπόνηση Μελέτης (Elaboration)..... | 10 |
| 4.1 | Ανάλυση - Σχεδιασμός | 10 |
| 4.1.1 | Διαγράμματα Περιπτώσεων Χρήσης (2η έκδοση)..... | 10 |
| 4.1.2 | Διαγράμματα Τάξεων (2η έκδοση) | 12 |
| 4.1.3 | Διαγράμματα Αντικειμένων (1η έκδοση) | 15 |
| 4.1.4 | Διαγράμματα Συνεργασίας (1η έκδοση) | 16 |
| 4.1.5 | Διαγράμματα Σειράς (1η έκδοση) | 18 |
| 4.1.6 | Διαγράμματα Δραστηριοτήτων (1η έκδοση)..... | 21 |
| 4.1.7 | Διαγράμματα Καταστάσεων (1η έκδοση) | 24 |
| 4.1.8 | Διαγράμματα Εξαρτημάτων (1η έκδοση) | 26 |
| 4.1.9 | Διαγράμματα Διανομής (1η έκδοση) | 27 |
| 4.2 | Υλοποίηση-Έλεγχος | 28 |
| 4.2.1 | Υλοποίηση: 1η εκτελέσιμη έκδοση | 28 |
| 4.2.2 | Αναφορά ελέγχου για την 1η εκτελέσιμη έκδοση | 29 |
| 5 | Φάση: Κατασκευή (Construction) | 30 |
| 5.1 | Ανάλυση-Σχεδιασμός | 30 |
| 5.1.1 | Διαγράμματα Περιπτώσεων Χρήσης (3η έκδοση)..... | 30 |
| 5.1.2 | Διαγράμματα Τάξεων (3η έκδοση) | 31 |
| 5.1.3 | Διαγράμματα Αντικειμένων (2η έκδοση) | 36 |
| 5.1.4 | Διαγράμματα Συνεργασίας (2η έκδοση) | 36 |
| 5.1.5 | Διαγράμματα Σειράς (2η έκδοση) | 38 |
| 5.1.6 | Διαγράμματα Δραστηριοτήτων (2η έκδοση)..... | 41 |
| 5.1.7 | Διαγράμματα Καταστάσεων (2η έκδοση) | 44 |

| | | |
|-------|--|----|
| 5.1.8 | Διαγράμματα Εξαρτημάτων (2η έκδοση) | 47 |
| 5.1.9 | Διαγράμματα Διανομής (2η έκδοση) | 47 |
| 5.2 | Υλοποίηση-Έλεγχος | 48 |
| 5.2.1 | Υλοποίηση: Τελική εκτελέσιμη έκδοση..... | 48 |
| 5.2.2 | Αναφορά ελέγχου για την τελική εκτελέσιμη έκδοση | 48 |
| 6 | Εγχειρίδιο Χρήστη | 49 |
| 6.1 | Σύντομη παρουσίαση του προγράμματος | 49 |
| 6.2 | Παρουσίαση σεναρίων λειτουργίας | 49 |

1 Εισαγωγή

1.1 Στόχοι της εργασίας

Στόχος της εργασίας αυτής, είναι ο σχεδιασμός και η λειτουργία τοπικής εφαρμογής που αφορά στην διαχείριση κρατήσεων ξενοδοχειακής μονάδας μέσω της μεθοδολογίας RUP και τη χρήση αντικειμενοστραφούς σχεδίασης τεκμηριωμένης με τη γλώσσα περιγραφής UML.

Στο πλαίσιο της θα τεκμηριωθεί με τη βοήθεια κειμένων και διαγραμμάτων η διαδικασία καθορισμού απαιτήσεων, σχεδιασμού και ανάπτυξης του αντίστοιχου λογισμικού.

Ως εκ τούτου το έργο διαιρέθηκε σε τρεις φάσεις της RUP , οι οποίες εξελίσσονται επαναληπτικά, σύμφωνα με τις αρχές της μεθοδολογίας.

Οι φάσεις αυτές είναι:

1. Η Έναρξη του έργου
2. Η Εκπόνηση Μελέτης
3. Η Κατασκευή του λογισμικού

Οι βασικές διαδικασίες οι οποίες εξελίσσονται στη διάρκεια των φάσεων αυτών είναι: η μοντελοποίηση των διαδικασιών, η καταγραφή των απαιτήσεων, η ανάλυση και ο σχεδιασμός. Επιπρόσθετα θα καταγραφεί και η διαδικασία της υλοποίησης, κατά την οποία αναπτύσσεται το λογισμικό καθαυτό και εγκαθίσταται σε περιβάλλον λειτουργίας. Η εργασία θα καλύψει και τις διαδικασίες ελέγχου του παραχθέντος λογισμικού.

Οι φάσεις που προαναφέρθηκαν αποτελούν τα κύρια ορόσημα του έργου. Με την ολοκλήρωσή τους παραδίδεται το αντίστοιχο προϊόν το οποίο αναπτύσσεται στη διάρκειά τους.

1.2 Ορισμός του προβλήματος προς επίλυση

Το πρόβλημα προς επίλυση αφορά την οργάνωση του συστήματος διαχείρισης ξενοδοχειακής μονάδας.

Οι κύριοι χρήστες του συστήματος διαχείρισης είναι ο υπάλληλος στην υποδοχή και ο manager του ξενοδοχείου.

Ο υπάλληλος στην υποδοχή μέσω του συστήματος διαχείρισης του ξενοδοχείου εξυπηρετεί τους πελάτες.

Ελέγχει τα δωμάτια του ξενοδοχείου, διαχειρίζεται τις κρατήσεις των πελατών και καταχωρεί τα στοιχεία των πελατών στη βάση δεδομένων του συστήματος.

Ο manager μέσω του συστήματος διαχείρισης του ξενοδοχείου μπορεί να κάνει ότι και ο υπάλληλος υποδοχής, να βλέπει τη λίστα των receptionist και επιπρόσθετα μπορεί να λάβει εκτυπώσεις που αφορούν στο ξενοδοχείο, όπως:

- Οικονομικά στοιχεία τρέχοντος ή προηγούμενου έτους
- Στατιστικά στοιχεία τρέχοντος ή προηγούμενου έτους

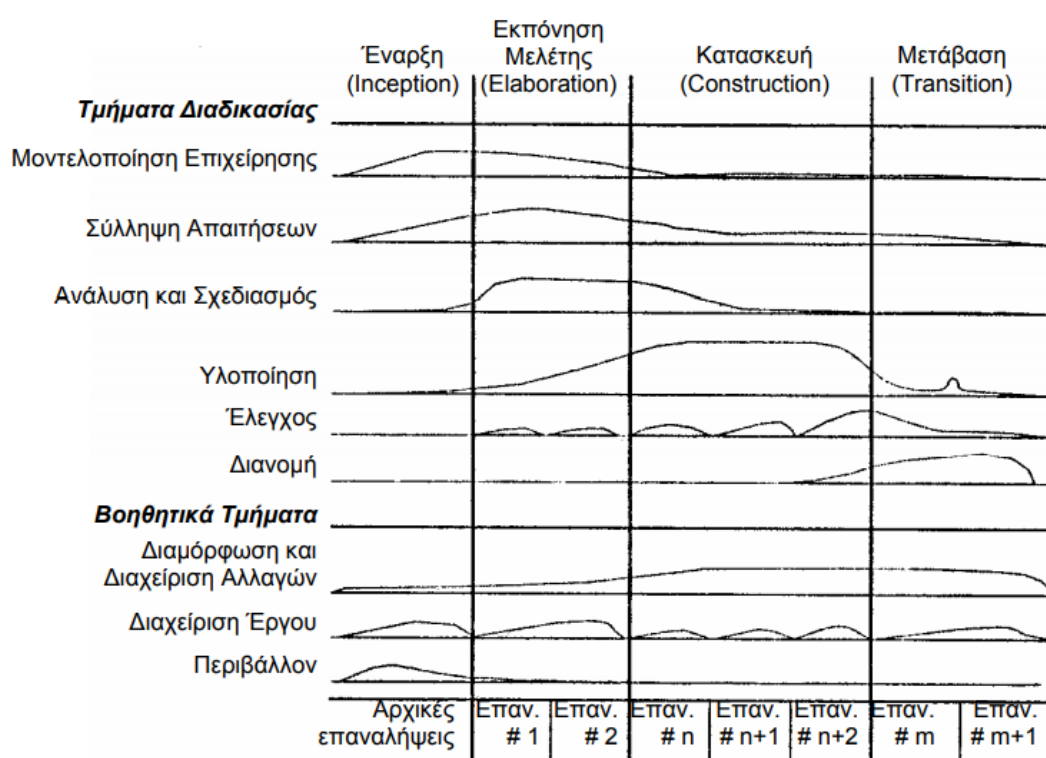
Όλες οι παραπάνω πληροφορίες καταγράφονται στη βάση δεδομένων του συστήματος.

2 Σύντομη παρουσίαση της RUP

Η Rational Unified Process (RUP) είναι μεθοδολογία για την ανάπτυξη αντικειμενοστραφών συστημάτων και έχει αναπτυχθεί από τους δημιουργούς της αντικειμενοστραφούς γλώσσας μοντελοποίησης UML, τους Booch, Rumbaugh και Jacobson.

Βασίζεται στο μοντέλο του καταρράκτη, αλλά θεωρεί ότι η ανάλυση απαιτήσεων, ο σχεδιασμός, η υλοποίηση και ο έλεγχος δεν συμπίπτουν με χρονικές φάσεις αλλά αντιπροσωπεύουν τμήματα διαδικασίας τα οποία λαμβάνουν χώρα σε διάφορες χρονικές φάσεις.

Το μοντέλο αυτό εισάγει μια δεύτερη διάσταση όπως φαίνεται στο σχήμα 1.1, η οποία αφορά στο χρόνο. Σύμφωνα με τη χρονική σειρά που διεξάγονται οι διαδικασίες, το μοντέλο ορίζει 4 χρονικές φάσεις, την έναρξη, εκπόνηση μελέτης, κατασκευή και μετάβαση



Σχήμα 1.1 Κύκλος Ζωής Ανάπτυξης Λογισμικού

Ο κύκλος ζωής λογισμικού προτείνεται να είναι επαναληπτικός. Η ανάπτυξη δηλαδή να προχωρεί σε μια σειρά επαναλήψεων μέχρι να εξελιχθεί το τελικό προϊόν. Η διαδικασία Unified της Rational αποτελείται από ένα σύνολο οδηγιών σχετικά με τις τεχνικές και οργανωτικές απόψεις της ανάπτυξης λογισμικού. Η διαδικασία αυτή αφορά κυρίως στην Ανάλυση Απαιτήσεων και στο Σχεδιασμό.

Η διαδικασία Unified είναι δομημένη σε δύο διαστάσεις :

- 1) Χρόνο Χωρισμός του κύκλου ζωής σε φάσεις και επαναλήψεις.
- 2) Τμήματα διαδικασίας Καλά ορισμένες ενέργειες.

Η δόμηση ενός έργου σε σχέση με το χρόνο ακολουθεί τις εξής φάσεις που έχουν σχέση με το χρόνο :

- 1) Έναρξη (Inception) : Καθορίζει την προοπτική του έργου
- 2) Εκπόνηση μελέτης (Elaboration) : Σχεδιασμός των απαιτούμενων δραστηριοτήτων και πόρων. Καθορισμός των χαρακτηριστικών και σχεδιασμός της αρχιτεκτονικής
- 3) Κατασκευή (Construction) : Ανάπτυξη του προϊόντος σε μια σειρά βηματικών επαναλήψεων
- 4) Μετάβαση (Transition) : Προμήθεια του προϊόντος στους χρήστες (παραγωγή, διανομή, εκπαίδευση)

Η δόμηση έργου σύμφωνα με τη διάσταση των τμημάτων διαδικασίας περιλαμβάνει τις ακόλουθες δραστηριότητες :

- 1) Σύλληψη απαιτήσεων (Requirements capture) : Μια αφήγηση του τι πρέπει να κάνει το σύστημα
- 2) Ανάλυση και σχεδιασμός (Analysis and design) : Μια περιγραφή του πώς θα υλοποιηθεί το σύστημα
- 3) Υλοποίηση (Implementation) : Η παραγωγή του κώδικα
- 4) Έλεγχος (Test) : Η επαλήθευση του συστήματος

3 Φάση: Έναρξη (Inception)

3.1 Σύλληψη απαιτήσεων

Στο πρώτο μέρος της Φάσης Έναρξη, θα πρέπει να ασχοληθούμε με την καταγραφή των απαιτήσεων του project και την ανάλυσή τους ώστε να δημιουργηθεί ένα αρχικό σχεδιάγραμμα το οποίο θα περιγράφει τις διάφορες λειτουργίες τις οποίες θα πρέπει το σύστημα που θα υλοποιήσουμε να εκτελεί.

Για να γίνει αυτό θα πρέπει να γίνουν οι παρακάτω αναλύσεις:

- Ανάλυση αναγκών (Needs Analysis)
- Ανάλυση Χρηστών (User Analysis)
- Ανάλυση Εργασιών (Task Analysis)

3.1.1 Ανάλυση αναγκών (Needs Analysis)

Το σύστημα που θέλουμε να υλοποιήσουμε, θα μπορεί να εκτελέσει κάποιες συγκεκριμένες εργασίες. Με σκοπό την ομαλή και ορθή εκτέλεση αυτών των διαδικασιών, πρέπει να προβούμε σε μία ανάλυση απαιτήσεων ως προς τον τρόπο υλοποίησης τους και των τεχνικών τις οποίες πρέπει να υλοποιήσουμε.

Αρχικά πρέπει όλα τα δεδομένα να αποθηκεύονται κεντρικά σε ένα σύστημα διαχείρισης το οποίο θα μας επιτρέπει με την χρήση ερωτημάτων να μπορούμε να εξάγουμε τις απαντήσεις μας. Αυτό προϋποθέτει την ύπαρξη μίας βάσης δεδομένων η οποία θα μας διευκολύνει τόσο στην καταχώρηση των δεδομένων μας όσο και στην εξαγωγή reports – αποτελεσμάτων.

Το σύστημα διαχείρισης ξενοδοχειακής μονάδας που υλοποιούμε θα εκτελείται σε τοπικό - desktop περιβάλλον με χρήση Java και θα καταχωρούνται /αντλούνται δεδομένα από μία βάση δεδομένων SQLite.

3.1.2 Ανάλυση Χρηστών (User Analysis)

Κατά την υλοποίηση της εφαρμογής θα έχουμε δύο ειδών χρήστες στους στόχους, όσο αφορά την υλοποίηση αλλά και τις διαφορετικές λειτουργίες τις οποίες θα μπορεί να διατελεί κάθε ένας από τους δύο διαφορετικούς χρήστες.

User (Receptionist): Ο χρήστης Receptionist έχει πρόσβαση σε ορισμένες λειτουργίες του συστήματος.

Επιγραμματικά, αυτές οι λειτουργίες είναι:

- Είσοδος – Έξοδος από το Σύστημα
- Εισαγωγή και προβολή στοιχείων πελατών
- Εισαγωγή και προβολή των κρατήσεων του Ξενοδοχείου
- Προβολή στοιχείων δωματίων
- Είσπραξη από πελάτη η οποία αφορά κράτηση ή κρατήσεις

User (Manager): Ο χρήστης Manager έχει πρόσβαση σε περισσότερες λειτουργίες του συστήματος. Επιγραμματικά, αυτές οι λειτουργίες είναι:

- Είσοδος – Έξοδος από το Σύστημα
- Εισαγωγή και προβολή στοιχείων πελατών
- Εισαγωγή και προβολή των κρατήσεων του Ξενοδοχείου
- Προβολή στοιχείων δωματίων
- Είσπραξη από πελάτη η οποία αφορά κράτηση ή κρατήσεις
- Οικονομικά στοιχεία τρέχοντος ή προηγούμενου έτους
- Στατιστικά στοιχεία τρέχοντος ή προηγούμενου έτους
- Προβολή λίστας Receptionist

3.1.3 Ανάλυση Εργασιών (Task Analysis)

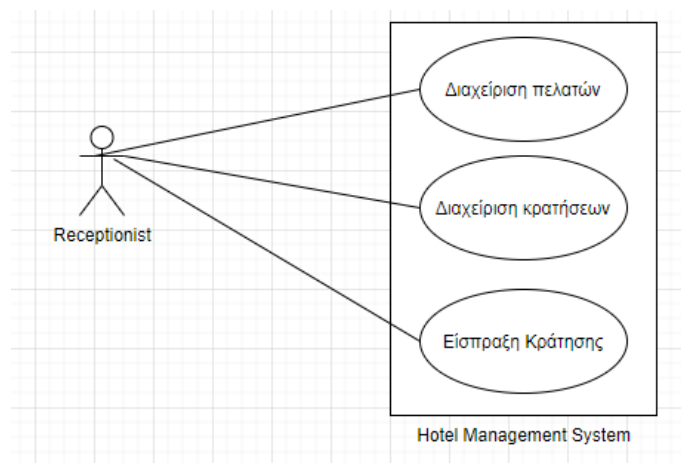
Σε αυτό το σημείο θα οργανώσουμε την υλοποίηση του συστήματος σε διάφορες φάσεις – στάδια τα οποία θα μας βοηθήσουν να διαχωρίσουμε σε τμήματα την όλη διαδικασία.

- Βάση Δεδομένων: Πρώτη θα πρέπει να αναλυθεί και να δημιουργηθεί η βάση δεδομένων στην οποία θα καταχωρούνται / αντλούνται δεδομένα. Η δομή της βάσης θα μας οδηγήσει εύκολα στην υλοποίηση των επιμέρους τμημάτων τα οποία τόσο θα εισάγουν, όσο και θα εξάγουν δεδομένα από αυτή με την βοήθεια διάφορων ερωτημάτων.
- Δημιουργία τάξεων: Μετά την δημιουργία της βάσης θα προχωρήσουμε στη δημιουργία τάξεων σε Java περιβάλλον όπου θα αποτελέσουν τον κορμό της εφαρμογής.
- Διασύνδεση βάσης δεδομένων με τις τάξεις: Στη συνέχεια θα υλοποιηθεί με κώδικα η διασύνδεση της βάσης δεδομένων με τις τάξεις.

3.2 Ανάλυση Σχεδιασμός

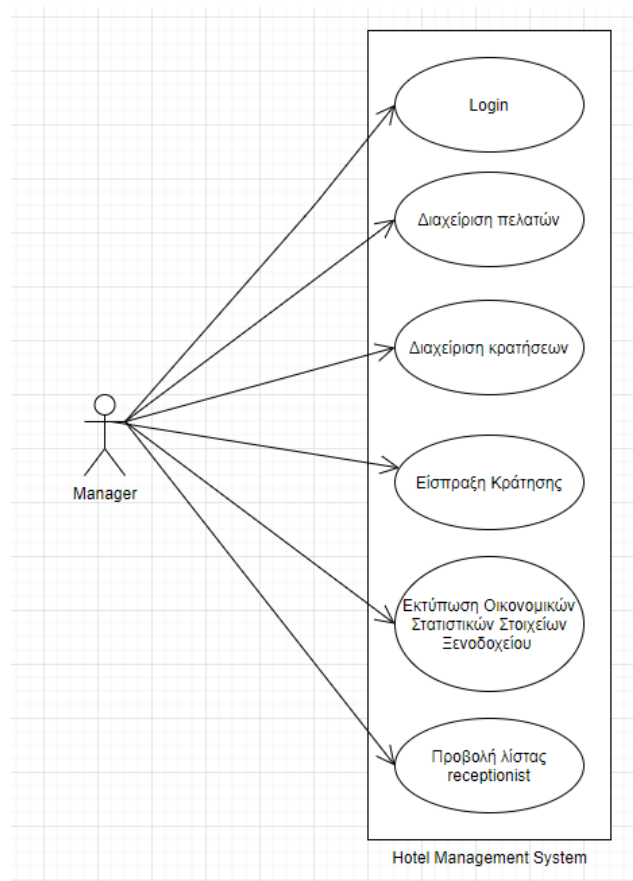
Στη δραστηριότητα αυτή καταγράφεται σε ένα πρώτο επίπεδο η λειτουργικότητα του συστήματος, μέσω των αντίστοιχων διαγραμμάτων Περιπτώσεων Χρήσης και Τάξεων.

3.2.1 Περιπτώσεις Χρήσης - Use Case



Εικόνα 1: Διάγραμμα Περίπτωση χρήσης Receptionist

Στην εικόνα 1 απεικονίζονται οι ενέργειες τις οποίες θέλουμε να μπορεί να πραγματοποιήσει ο χρήστης Receptionist.



Εικόνα 2: Διάγραμμα Περίπτωση χρήσης Manger

Στην εικόνα 2, απεικονίζονται οι ενέργειες τις οποίες θέλουμε να μπορεί να πραγματοποιήσει ο χρήστης Manager.

3.2.2 Διάγραμμα Τάξεων

Ένα διάγραμμα τάξης περιγράφει τους τύπους των αντικειμένων στο σύστημα και τα διάφορα είδη στατικών σχέσεων που υπάρχουν μεταξύ τους (αμφίδρομες και μονόδρομες συσχετίσεις, υπότυποι, πολλαπλότητα, κληρονομικότητα, κλπ.). Στα διαγράμματα αυτά αποτυπώνονται ακόμα οι ιδιότητες των τάξεων και οι λειτουργίες αυτών.

Οι τάξεις (classes) αποτελούν τη βάση της κατασκευής οποιουδήποτε αντικειμενοστραφούς συστήματος. Σκοπός του διαγράμματος τάξεων (class diagram) είναι η απεικόνιση της στατικής δομής του συστήματος.

Μια τάξη σχεδιάζεται ως ένα ορθογώνιο με τρία τμήματα. Το πρώτο τμήμα περιέχει το όνομα της τάξης, το δεύτερο περιέχει τα χαρακτηριστικά της τάξης και το τρίτο της λειτουργίες της. Με βάση τις απαιτήσεις που συλλέξαμε, έχουμε θεωρήσει τις ακόλουθες τάξεις για την εφαρμογή μας:

Customer: Είναι η τάξη που αναφέρεται στους πελάτες που καταχωρούνται στην εφαρμογή.

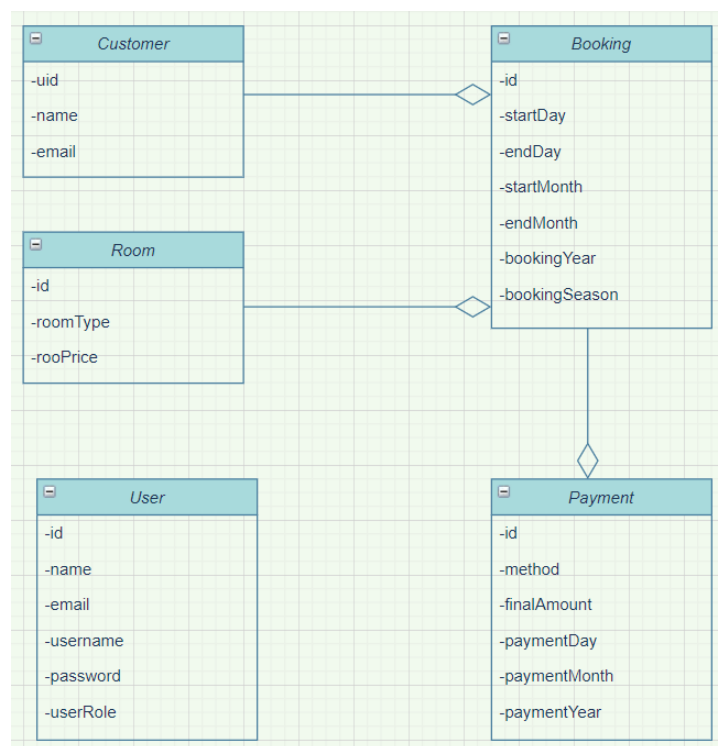
Booking: Είναι η τάξη που αναφέρεται στις κρατήσεις που καταχωρούνται στην εφαρμογή.

Payment: Είναι η τάξη που αναφέρεται στις πληρωμές των κρατήσεων που καταχωρούνται στην εφαρμογή.

Room: Είναι η τάξη που αναφέρεται στα δωμάτια που υπάρχουν στην εφαρμογή.

User: Είναι η τάξη που αναφέρεται στους χρήστες της εφαρμογής

Οι παραπάνω τάξεις δεν λειτουργούν ανεξάρτητα η μία από την άλλη αλλά αντίθετα αλληλεπιδρούν μέσω σχέσεων. Ακολουθως περιγράφουμε ποιες είναι οι συσχετίσεις μεταξύ των τάξεων.



Εικόνα 3: Διάγραμμα Τάξεων

Από το παραπάνω διάγραμμα καταλαβαίνουμε ότι οι τάξεις που συνδέονται με σχέση συναρμολόγησης μπορούν να υφίστανται ανεξάρτητα. Επίσης από το διάγραμμα διαπιστώνουμε πως η τάξη Booking περιέχει τις τάξεις customer και Room ενώ η τάξη Payment περιέχει την τάξη Booking. Τέλος υφίσταται η τάξη User όπου θα μπορεί να διαχειριστεί τις υπόλοιπες τάξεις.

4 Φάση: Εκπόνηση Μελέτης (Elaboration)

Ο αντικειμενικός σκοπός της δεύτερης φάσης της μεθοδολογίας RUP (Εκπόνηση Μελέτης) είναι ο προσδιορισμός της λεπτομερούς αρχιτεκτονικής δομής του συστήματος και η δημιουργία των βάσεων για την ανάπτυξη του αντίστοιχου λογισμικού στις επόμενες φάσεις. Σύμφωνα με τον οδηγό της RUP για υλοποίηση μικρών έργων, η φάση αυτή περιλαμβάνει τις ακόλουθες δραστηριότητες:

1. Προσδιορισμός και επικύρωση της αρχιτεκτονικής του συστήματος.
2. Αναπροσαρμογή του «Οράματος» του έργου, βάσει των πληροφοριών που αντλήθηκαν. Κατανόηση των κρίσιμων περιπτώσεων χρήσης, οι οποίες και επηρεάζουν τις αποφάσεις του αρχιτεκτονικού σχεδιασμού.
3. Δημιουργία του περιβάλλοντος ανάπτυξης του λογισμικού. Το περιβάλλον αυτό περιλαμβάνει τις διαδικασίες, τα εργαλεία λογισμικού και τους αυτοματισμούς, οι οποίοι θα υποστηρίξουν την ομάδα ανάπτυξης.
4. Καθορισμός των στοιχείων που απαιτούνται για τη σύνθεση του συστήματος. Αξιολογούνται διαφορετικές εκδοχές και λαμβάνονται αποφάσεις σχετικά με την αγορά ή ανάπτυξη κάποιου (ή κάποιων) απαραίτητων στοιχείων. Τα αρχιτεκτονικά αυτά στοιχεία ολοκληρώνονται στο περιβάλλον ανάπτυξης και αξιολογούνται σε σχέση με τα αρχικά σενάρια. Τα συμπεράσματα αυτής της διαδικασίας έχουν ως αποτέλεσμα τον πιθανό ανασχεδιασμό της αρχιτεκτονικής και εναλλακτικούς σχεδιασμούς ή και αναθεώρηση των απαιτήσεων από το σύστημα.

4.1 Ανάλυση - Σχεδιασμός

Οι δραστηριότητες της φάσης αυτής στοχεύουν στο σχεδιασμό των ενοτήτων του συστήματος βάσει των απαιτήσεων που αντλήθηκαν στη φάση «Έναρξη». Επίσης καταγράφονται οι προδιαγραφές του συστήματος, οι οποίες θα τροφοδοτήσουν τη φάση «Υλοποίηση». Η εφαρμογή θα αναπτυχθεί για να εξυπηρετήσει τις κρατήσεις ξενοδοχειακής μονάδας.

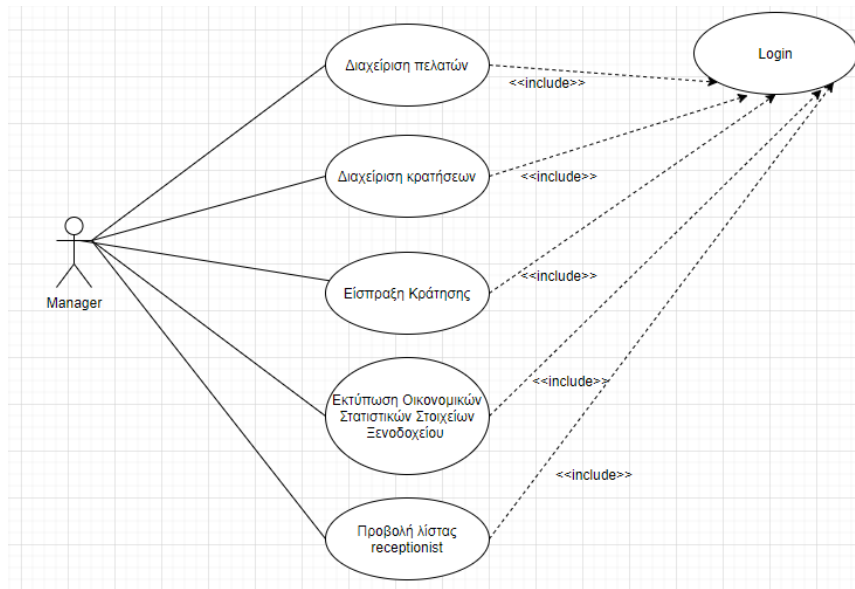
4.1.1 Διαγράμματα Περιπτώσεων Χρήσης (2η έκδοση)



Εικόνα 4: Διάγραμμα Περίπτωσης χρήσης Receptionist (2^η έκδοση)

Στην εικόνα 4, απεικονίζονται οι ενέργειες τις οποίες μπορεί να πραγματοποιήσει ο χρήστης Receptionist εφόσον συνδεθεί στην εφαρμογή.

Στη βάση δεδομένων μπορεί, εφόσον κάνει login, να δημιουργεί νέους πελάτες ή να ενημερώνει τα στοιχεία τους, να εισάγει νέες κρατήσεις και να πραγματοποιεί είσπραξη κράτησης.



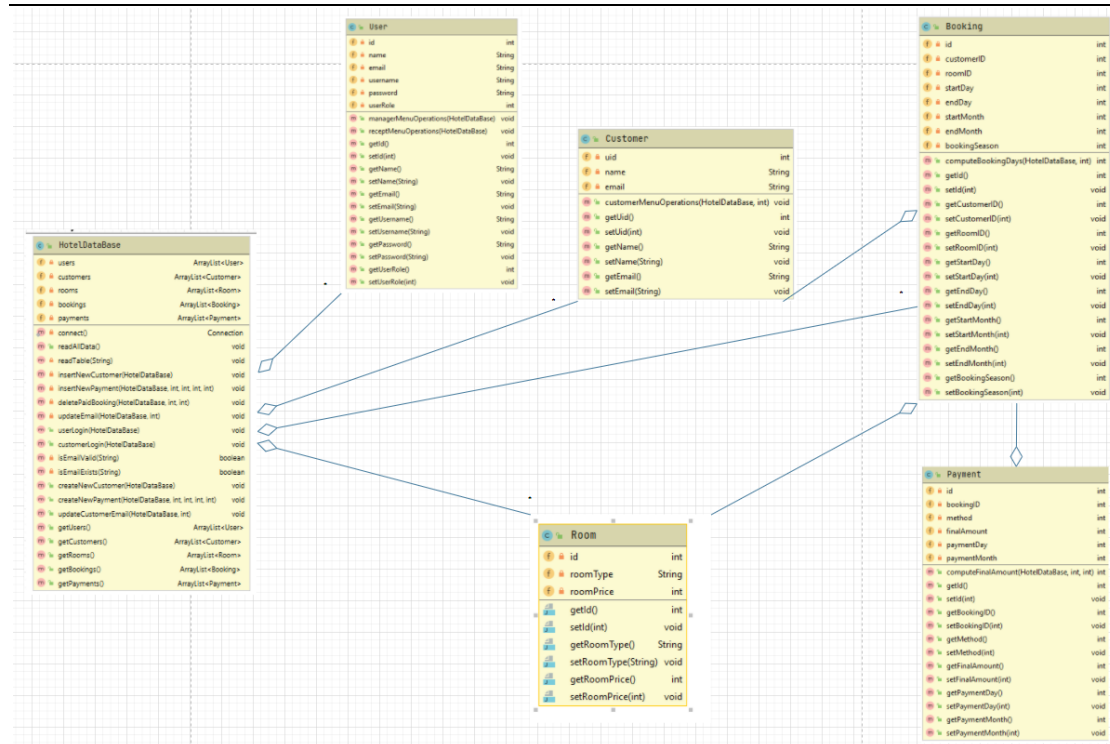
Εικόνα 5: Διάγραμμα Περίπτωση χρήσης Manager (2^η έκδοση)

Στην εικόνα 5 απεικονίζονται οι ενέργειες τις οποίες μπορεί να πραγματοποιήσει ο χρήστης Manager εφόσον συνδεθεί στην εφαρμογή.

Στη βάση δεδομένων μπορεί, εφόσον κάνει login, να δημιουργεί νέους πελάτες ή να ενημερώνει τα στοιχεία τους, να εισάγει νέες κρατήσεις και να πραγματοποιεί είσπραξη κράτησης. Επίσης μπορεί να προβάλει Report οικονομικών στατιστικών στοιχείων του ξενοδοχείου καθώς και να προβάλει τη λίστα των Receptionist.

4.1.2 Διαγράμματα Τάξεων (2η έκδοση)

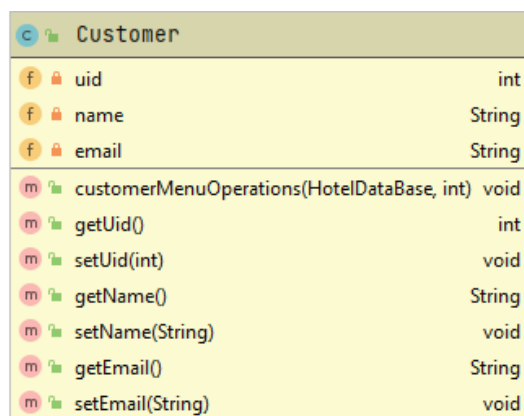
Το τροποποιημένο διάγραμμα τάξεων για το πληροφοριακό σύστημα της εφαρμογής μας φαίνεται παρακάτω. Παραθέτουμε τις έξι τάξεις με τις μεθόδους που έχει η κάθε μία. Η τάξη πελάτη (Customer), η τάξη κράτηση (Booking), η τάξη δωμάτιο (Room) η τάξη πληρωμή (Payment) η τάξη χρήστη (User) και η τάξη της βάσης δεδομένων (HotelDataBase).



Εικόνα 6: Διάγραμμα Τάξεων (2^η έκδοση)

Ακολουθως περιγράφονται αναλυτικά οι λειτουργίες κάθε τάξης:

Τάξη Customer



Στην τάξη Customer έχουμε τα εξής αντικείμενα:

- uid: int
- name : String
- email : String

Όπως και τις παρακάτω λειτουργίες:

- customerMenuOperations()
- getId()
- setId()
- getName()
- setName()
- getEmail()
- setEmail()

Τάξη Room

| Room | | |
|------|---------------------|--------|
| f | id | int |
| f | roomType | String |
| f | roomPrice | int |
| m | getId() | int |
| m | setId(int) | void |
| m | getRoomType() | String |
| m | setRoomType(String) | void |
| m | getRoomPrice() | int |
| m | setRoomPrice(int) | void |

Στην τάξη Room έχουμε τα εξής αντικείμενα:

- id: int
- roomType : String
- roomPrice : int

Όπως και τις παρακάτω λειτουργίες:

- getId()
- setId()
- getRoomType()
- setRoomType()
- getRoomPrice()
- setRoomPrice()

Τάξη Payment

| Payment | | |
|---------|---|------|
| f | id | int |
| f | bookingID | int |
| f | method | int |
| f | finalAmount | int |
| f | paymentDay | int |
| f | paymentMonth | int |
| m | computeFinalAmount(HotelDataBase, int, int) | int |
| m | getId() | int |
| m | setId(int) | void |
| m | getBookingID() | int |
| m | setBookingID(int) | void |
| m | getMethod() | int |
| m | setMethod(int) | void |
| m | getFinalAmount() | int |
| m | setFinalAmount(int) | void |
| m | getPaymentDay() | int |
| m | setPaymentDay(int) | void |
| m | getPaymentMonth() | int |
| m | setPaymentMonth(int) | void |

Στην τάξη Payment έχουμε τα εξής αντικείμενα:

- id: int
- bookingId : int
- method : int
- finalAmount : int
- paymentDay : int
- paymentMonth : int

Όπως και τις παρακάτω λειτουργίες:

- computeFinalAmount()
- getId()
- setId()
- getBookingId()
- setBookingId()
- getMethod()
- setMethod()
- getFinalAmount()
- setFinalAmount()
- getPaymentDay()
- setPaymentDay()
- getPaymentMonth()
- setPaymentMonth()

Τάξη Booking

| Booking | | |
|---------|--|------|
| f | id | int |
| f | customerID | int |
| f | roomID | int |
| f | startDay | int |
| f | endDay | int |
| f | startMonth | int |
| f | endMonth | int |
| f | bookingSeason | int |
| m | computeBookingDays(HotelDataBase, int) | int |
| m | getId() | int |
| m | setId(int) | void |
| m | getCustomerID() | int |
| m | setCustomerID(int) | void |
| m | getRoomID() | int |
| m | setRoomID(int) | void |
| m | getStartDay() | int |
| m | setStartDay(int) | void |
| m | getEndDay() | int |
| m | setEndDay(int) | void |
| m | getStartMonth() | int |
| m | setStartMonth(int) | void |
| m | getEndMonth() | int |
| m | setEndMonth(int) | void |
| m | getBookingSeason() | int |
| m | setBookingSeason(int) | void |

Στην τάξη Booking έχουμε τα εξής αντικείμενα:

- id: int
- CustomerId : int
- roomId : int
- startDay : int
- endDay : int
- startMonth : int
- endMonth : int
- bookingSeason : int

Όπως και τις παρακάτω λειτουργίες:

- computeBookingDays()
- getId()
- setId()
- getCustomerId()
- setCustomerId()
- getRoomId()
- setRoomId()
- getStartDay()
- setStartDay()
- getEndDay()
- setEndDay()
- getStartMonth()
- setStartMonth()
- getEndMonth()
- setEndMonth()
- getBookingSeason()
- setBookingSeason()

Τάξη HotelDataBase

| HotelDataBase | | |
|---------------|---|---------------------|
| f | users | ArrayList<User> |
| f | customers | ArrayList<Customer> |
| f | rooms | ArrayList<Room> |
| f | bookings | ArrayList<Booking> |
| f | payments | ArrayList<Payment> |
| m | connect() | Connection |
| m | readAllData() | void |
| m | readTable(String) | void |
| m | insertNewCustomer(HotelDataBase) | void |
| m | insertNewPayment(HotelDataBase, int, int, int, int) | void |
| m | deletePaidBooking(HotelDataBase, int, int) | void |
| m | updateEmail(HotelDataBase, int) | void |
| m | userLogin(HotelDataBase) | void |
| m | customerLogin(HotelDataBase) | void |
| m | isEmailValid(String) | boolean |
| m | isEmailExists(String) | boolean |
| m | createNewCustomer(HotelDataBase) | void |
| m | createNewPayment(HotelDataBase, int, int, int, int) | void |
| m | updateCustomerEmail(HotelDataBase, int) | void |
| m | getUsers() | ArrayList<User> |
| m | getCustomers() | ArrayList<Customer> |
| m | getRooms() | ArrayList<Room> |
| m | getBookings() | ArrayList<Booking> |
| m | getPayments() | ArrayList<Payment> |

Στην τάξη HotelDataBase έχουμε τα εξής αντικείμενα:

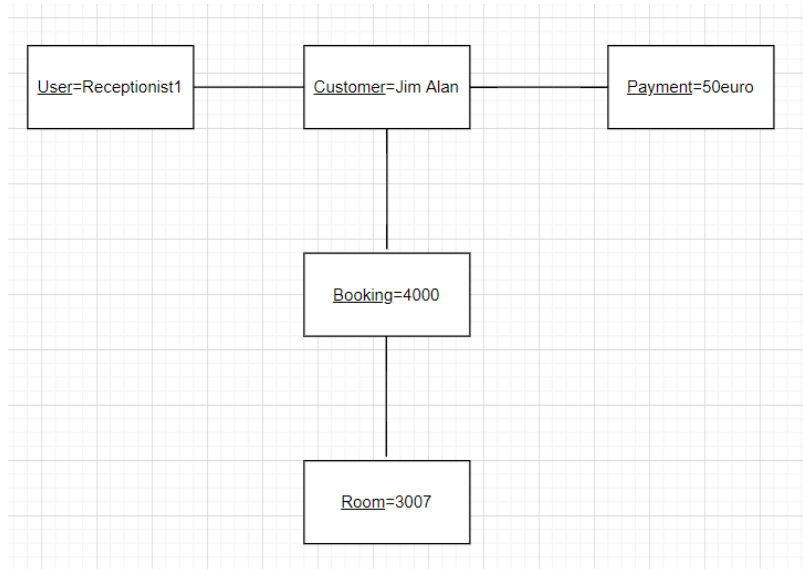
- users
- customers
- rooms
- bookings
- payments

και τις παρακάτω λειτουργίες:

- connect()
- readAllData()
- readTable()
- insertNewCustomer()
- insertNewPayment()
- deletePaidBooking()
- updateEmail()
- userLogin()
- customerLogin()
- isEmailValid()
- isEmailExists()

- createNewCustomer()
- createNewPayment()
- updateCustomerEmail()
- getUsers()
- getCustomers()
- getRooms()
- getBookings()
- getPayments()

4.1.3 Διαγράμματα Αντικειμένων (1η έκδοση)



Εικόνα 7: Διάγραμμα Αντικειμένων (1^η έκδοση)

Το διάγραμμα αντικειμένων φαίνεται παραπάνω. Αναπαριστούν αντικείμενα και τις σχέσεις τους και αντιστοιχούν σε απλοποιημένα διαγράμματα συνεργασίας που δεν αναπαριστούν μετάδοση μηνυμάτων.

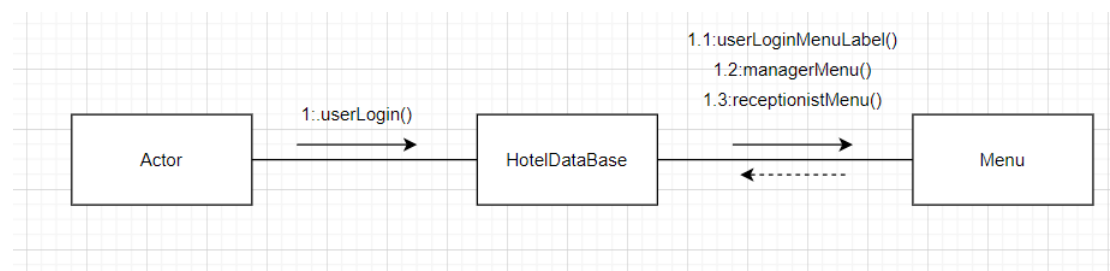
4.1.4 Διαγράμματα Συνεργασίας (1η έκδοση)

Τα διαγράμματα συνεργασίας είναι σημαντικά κατά την αντικειμενοστραφή ανάλυση και σχεδιασμό διότι απεικονίζουν τις αλληλεπιδράσεις ανάμεσα στα αντικείμενα. Εκφράζουν το περιβάλλον μεταξύ των αντικειμένων μέσω των συνδέσμων, αλλά και την αλληλεπίδραση μεταξύ των αντικειμένων μέσω μηνυμάτων που εμφανίζονται κατά μήκος των συνδέσμων.

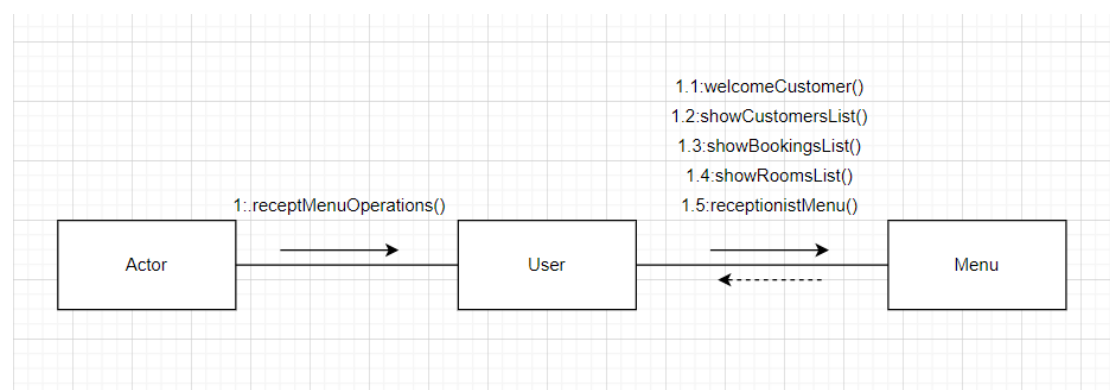
Στα διαγράμματα συνεργασίας ο χρόνος δεν αναπαρίσταται, με αποτέλεσμα τα διάφορα μηνύματα να αριθμούνται για να δηλώσουν τη σειρά αποστολής. Η αναπαράσταση των αλληλεπιδράσεων γίνεται ως εξής.

Το περιεχόμενο μιας αλληλεπίδρασης περιλαμβάνει τα ορίσματα, τις τοπικές μεταβλητές που δημιουργήθηκαν κατά την διάρκεια της εκτέλεσης και τους συνδέσμους ανάμεσα στα αντικείμενα που συμμετέχουν στην αλληλεπίδραση. Μια αλληλεπίδραση εκτελείται από ένα σύνολο αντικειμένων που συνεργάζονται ανταλλάσσοντας μηνύματα. Αυτά τα μηνύματα εμφανίζονται κατά μήκος των συνδέσμων που συνδέουν τα αντικείμενα χρησιμοποιώντας βέλη που δείχνουν προς τον παραλήπτη του μηνύματος.

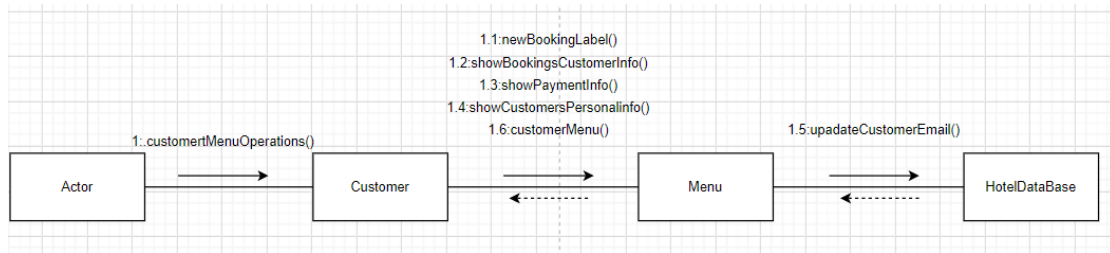
Τέλος, τα διαγράμματα αυτά δείχνουν τις αλληλεπιδράσεις ανάμεσα στα αντικείμενα και ταυτόχρονα τις δομικές σχέσεις που διευκολύνουν αυτές τις αλληλεπιδράσεις.



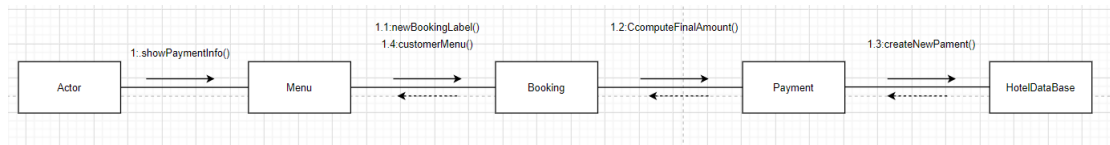
Εικόνα 8: Διάγραμμα Συνεργασίας 1 (1^η έκδοση)



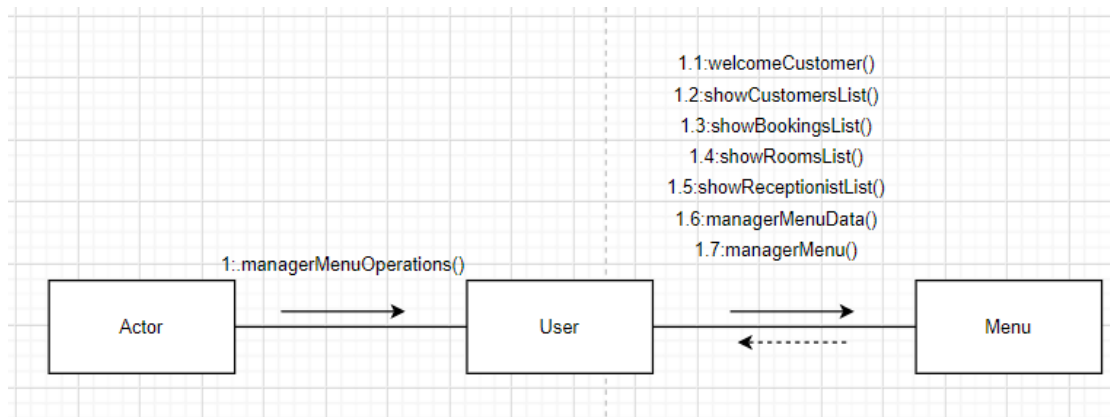
Εικόνα 9: Διάγραμμα Συνεργασίας 2 (1^η έκδοση)



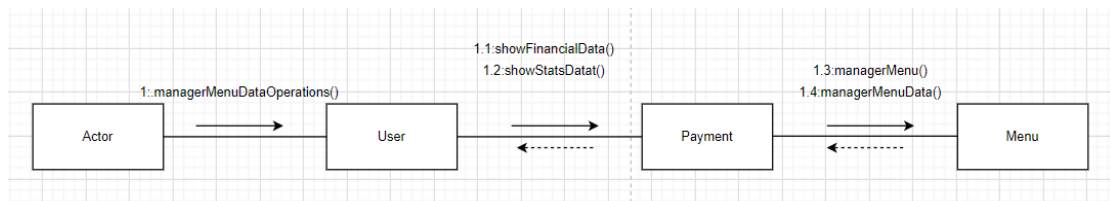
Εικόνα 10: Διάγραμμα Συνεργασίας 3 (1^η έκδοση)



Εικόνα 11: Διάγραμμα Συνεργασίας 4 (1^η έκδοση)



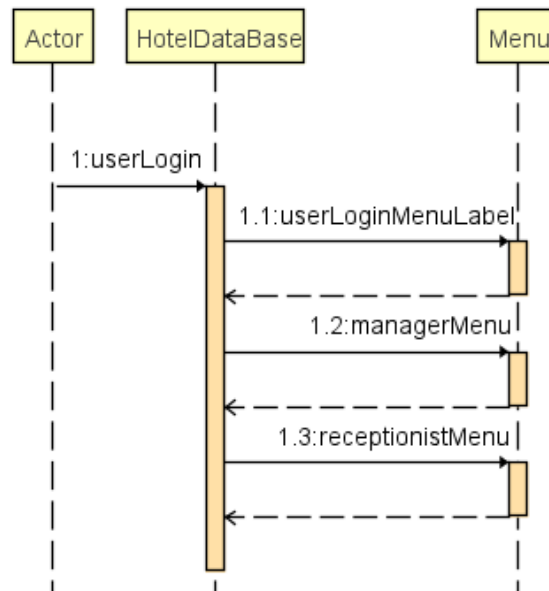
Εικόνα 12: Διάγραμμα Συνεργασίας 5 (1^η έκδοση)



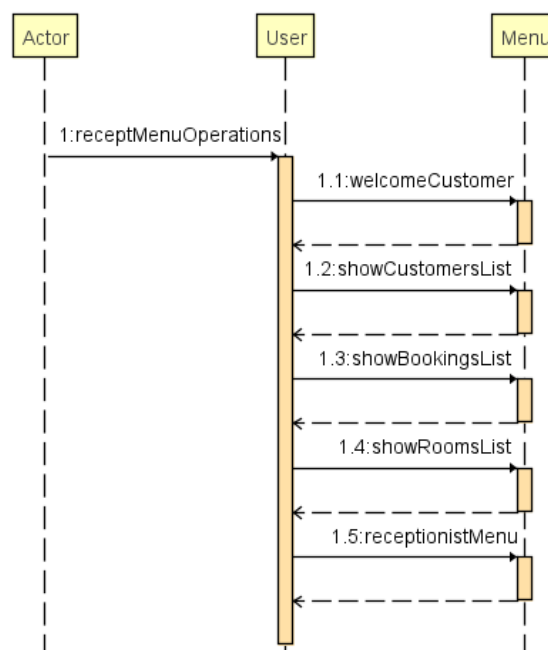
Εικόνα 13: Διάγραμμα Συνεργασίας 6 (1^η έκδοση)

4.1.5 Διαγράμματα Σειράς (1η έκδοση)

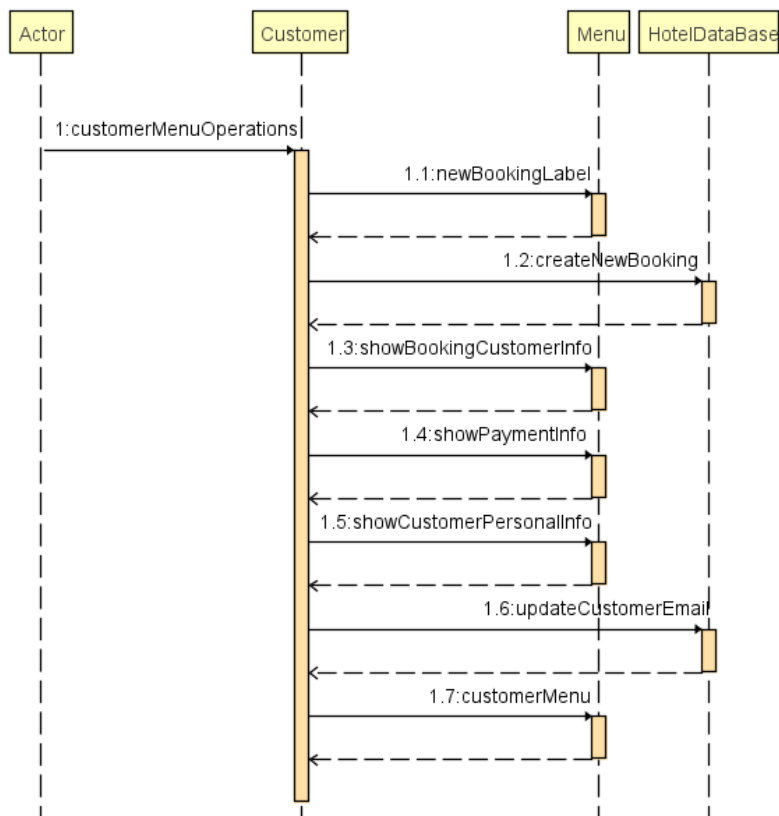
Τα διαγράμματα σειράς αναπαριστούν αλληλεπιδράσεις ανάμεσα στα αντικείμενα από μια χρονική άποψη. Σε αντίθεση με τα διαγράμματα συνεργασίας, το περιβάλλον των αντικειμένων δεν αναπαρίσταται σαφώς. Η αναπαράσταση επικεντρώνεται στην έκφραση των αλληλεπιδράσεων. Ένα διάγραμμα σειράς αναπαριστά μια αλληλεπίδραση ανάμεσα σε αντικείμενο αναπαρίσταται με ένα ορθογώνιο και μία κάθετη γραμμή, που καλείται γραμμή ζωής του αντικειμένου.



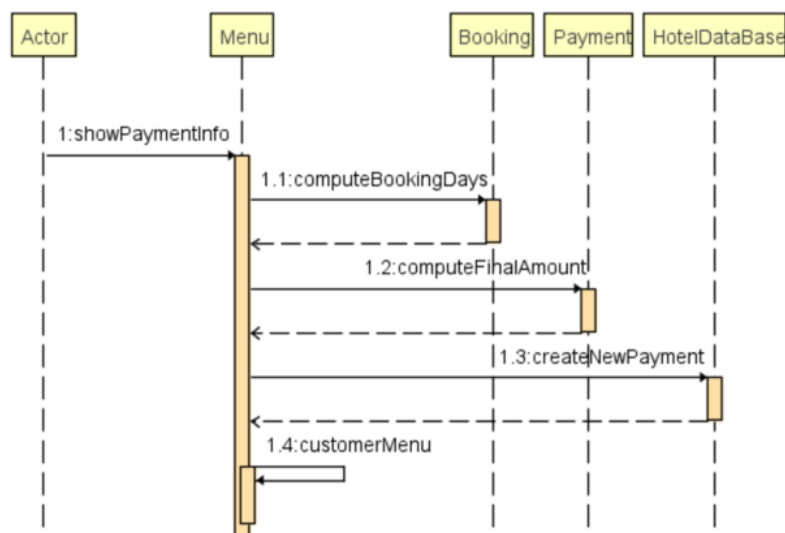
Εικόνα 14: Διάγραμμα Σειράς 1 (1^η έκδοση)



Εικόνα 15: Διάγραμμα Σειράς 2 (1^η έκδοση)



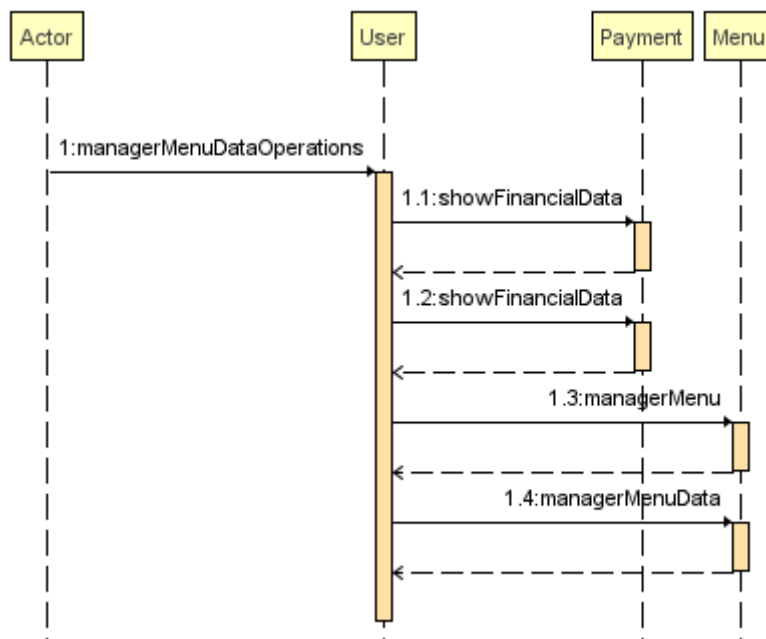
Εικόνα 16: Διάγραμμα Σειράς 3 (1^η έκδοση)



Εικόνα 17: Διάγραμμα Σειράς 4 (1^η έκδοση)



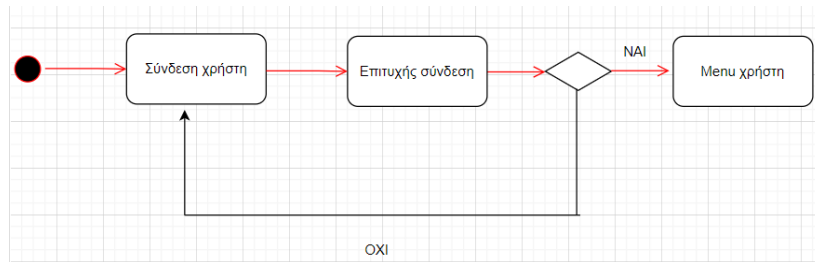
Εικόνα 18: Διάγραμμα Σειράς 5 (1^η έκδοση)



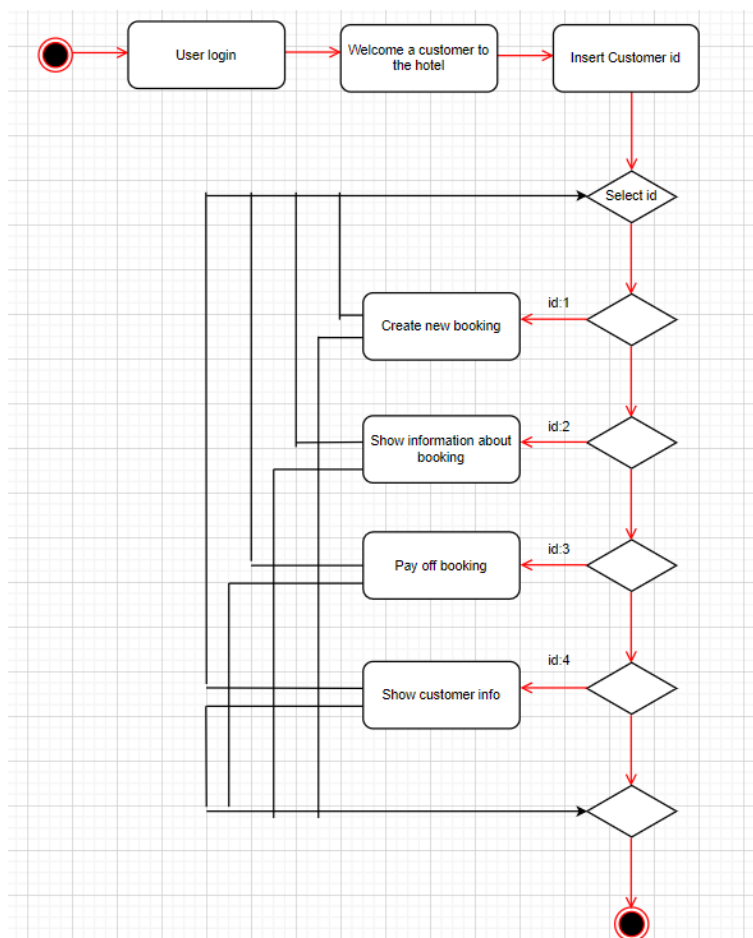
Εικόνα 19: Διάγραμμα Σειράς 6 (1^η έκδοση)

4.1.6 Διαγράμματα Δραστηριοτήτων (1η έκδοση)

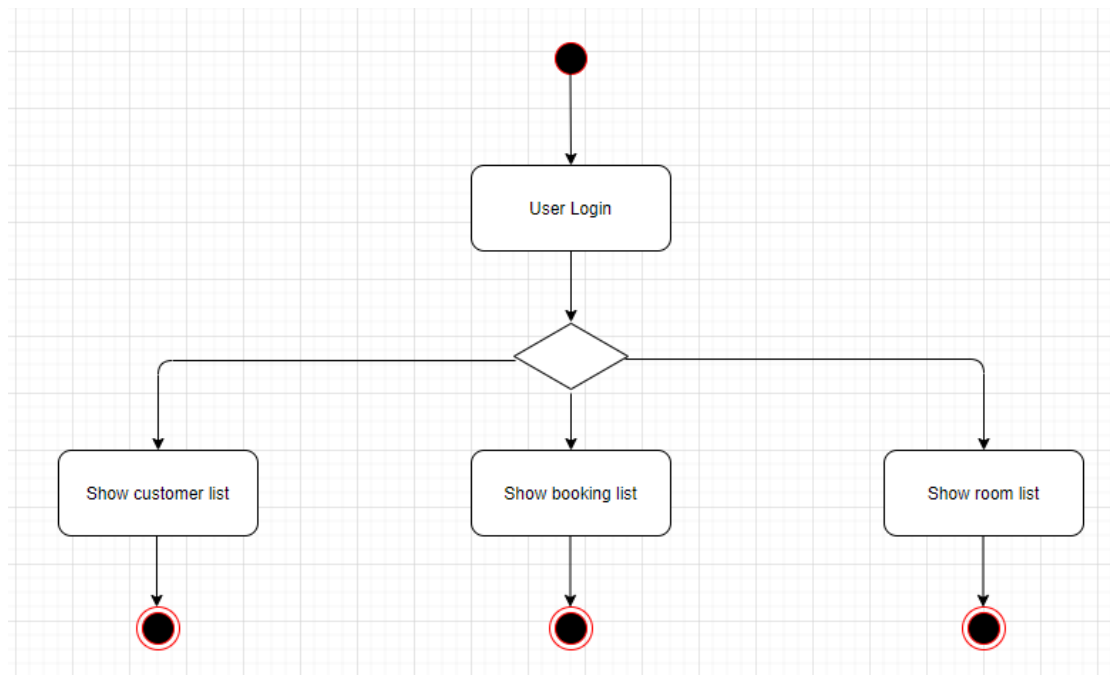
Ένα διάγραμμα δραστηριοτήτων είναι μια παραλλαγή των διαγραμμάτων καταστάσεων οργανωμένο σύμφωνα με ενέργειες και στοχεύει κυρίως στην αναπαράσταση της εσωτερικής συμπεριφοράς μιας μεθόδου ή μιας περίπτωσης χρήσης σαν μια ακολουθία βημάτων. Τα σύμβολα που χρησιμοποιούνται για τον σχεδιασμό αυτών των διαγραμμάτων είναι τα ίδια με τα διαγράμματα καταστάσεων απλά η σημασία τους αλλάζει. Πιο συγκεκριμένα η αρχική και τελική κατάσταση θεωρείται η αρχή και το τέλος της λειτουργίας ενώ οι λεγόμενες ενδιάμεσες καταστάσεις αποτελούν της δραστηριότητες της λειτουργίας. Τέλος οι δραστηριότητες συνδέονται με αυτόματες μεταβάσεις που αναπαρίστανται με βέλη.



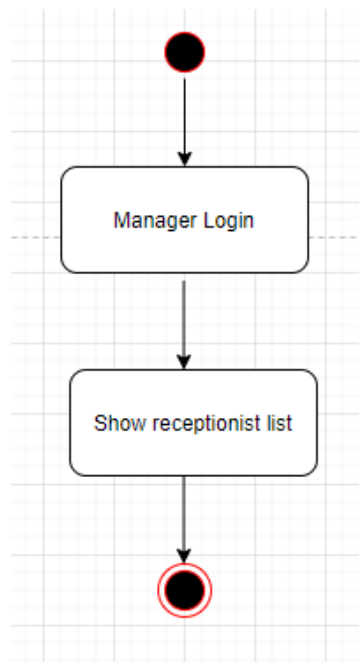
Εικόνα 20: Διάγραμμα Δραστηριοτήτων 1 (1^η έκδοση)



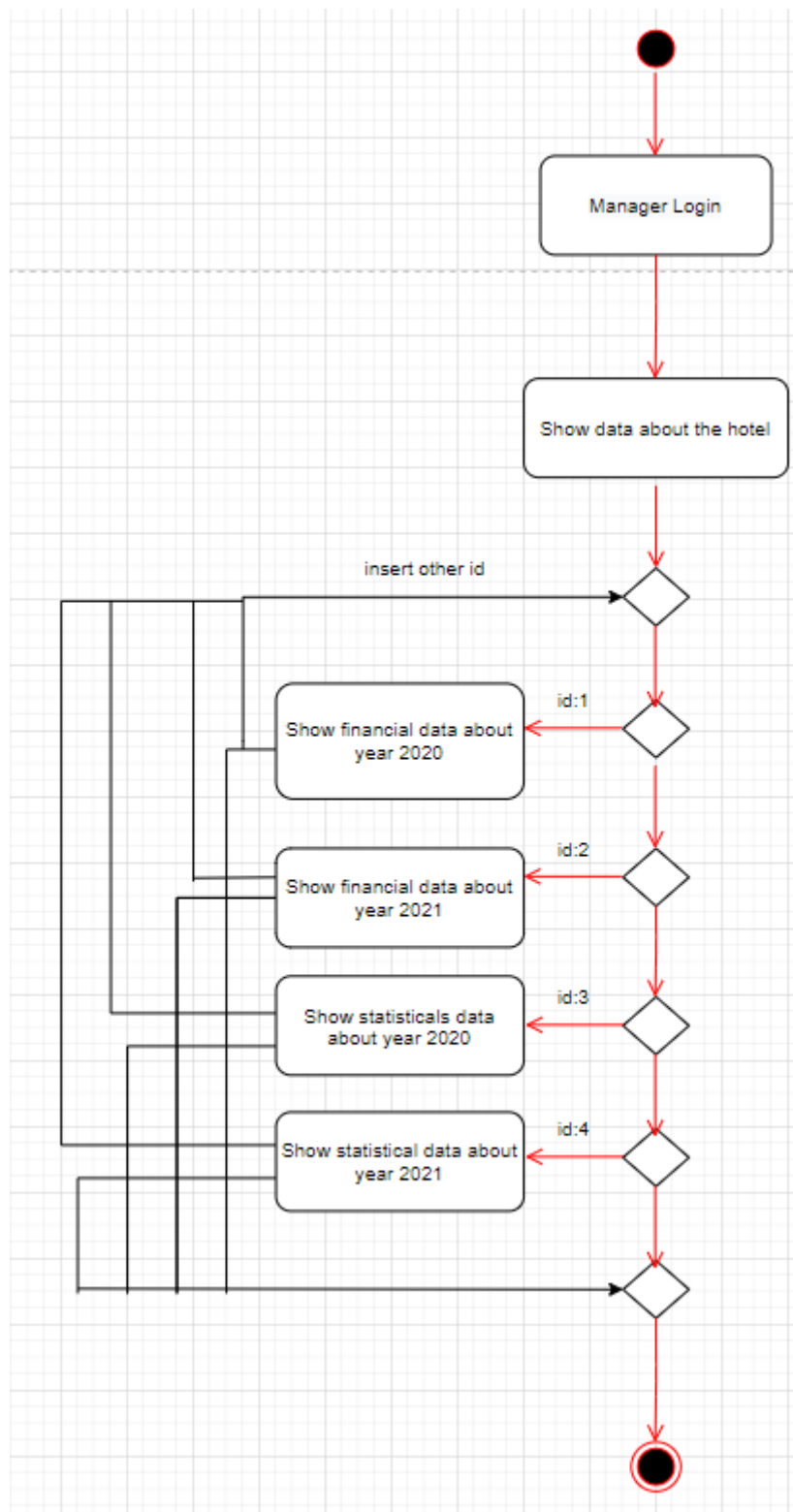
Εικόνα 21: Διάγραμμα Δραστηριοτήτων 2 (1^η έκδοση)



Εικόνα 22: Διάγραμμα Δραστηριοτήτων 3 (1^η έκδοση)

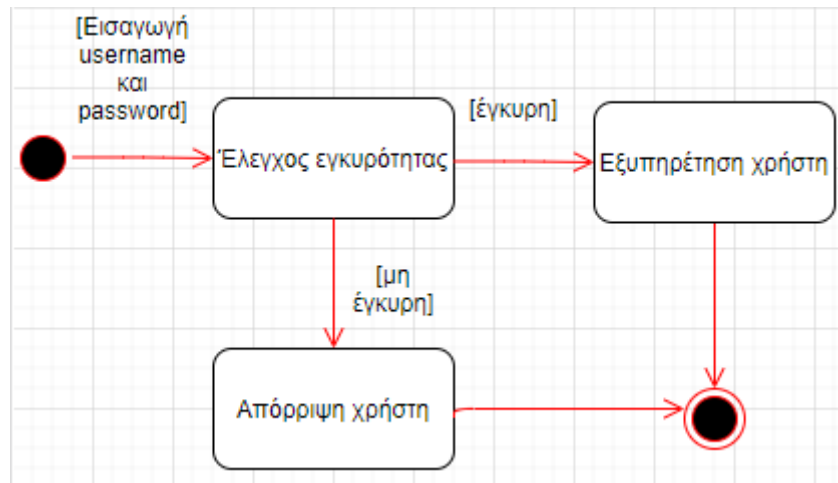


Εικόνα 23: Διάγραμμα Δραστηριοτήτων 4 (1^η έκδοση)

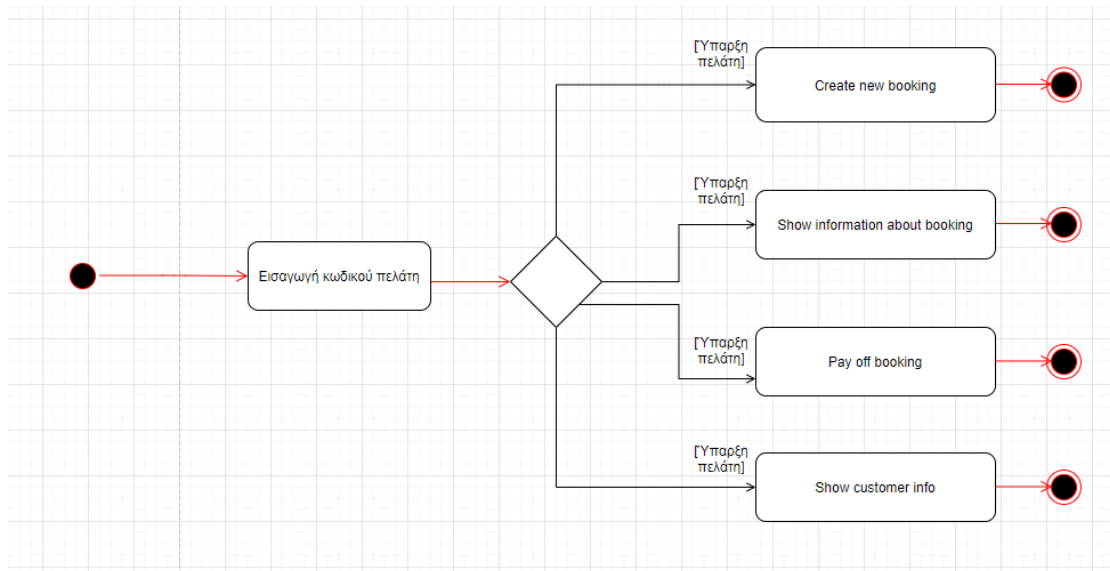


Εικόνα 24: Διάγραμμα Δραστηριοτήτων 5 (1^η έκδοση)

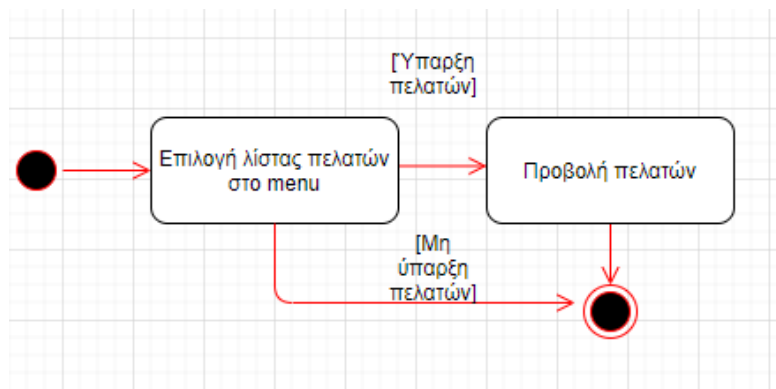
4.1.7 Διαγράμματα Καταστάσεων (1η έκδοση)



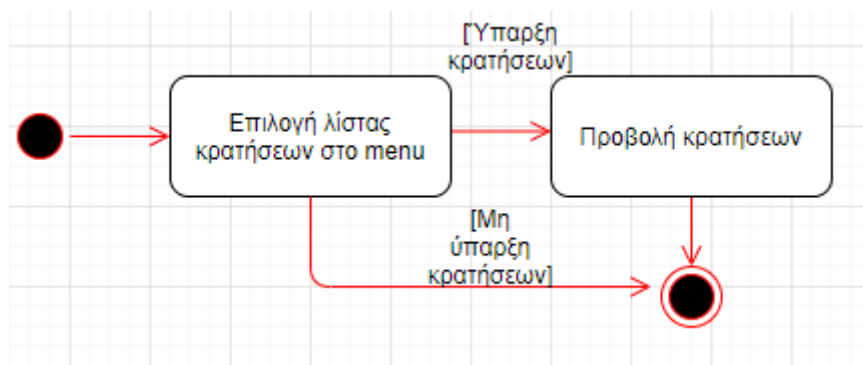
Εικόνα 25: Διάγραμμα Καταστάσεων 1 (1^η έκδοση)



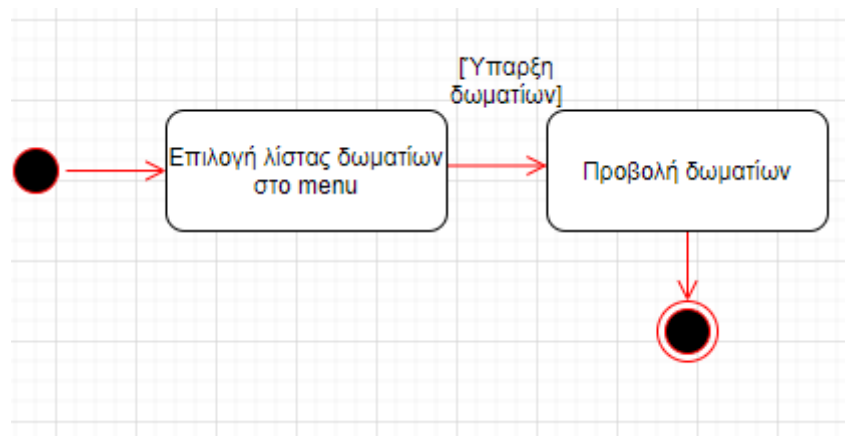
Εικόνα 26: Διάγραμμα Καταστάσεων 2 (1^η έκδοση)



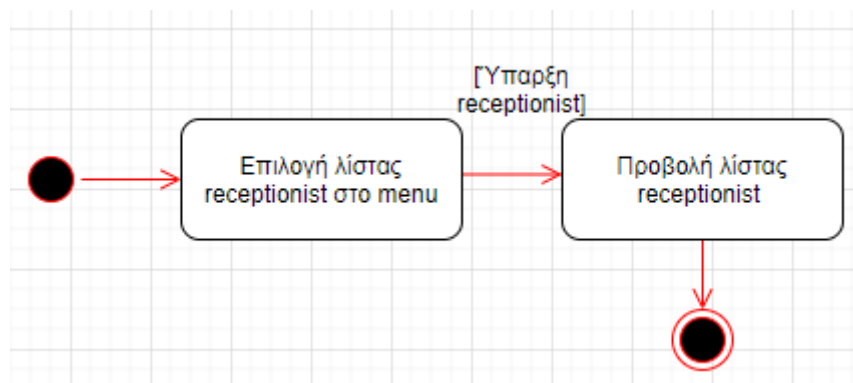
Εικόνα 27: Διάγραμμα Καταστάσεων 3 (1^η έκδοση)



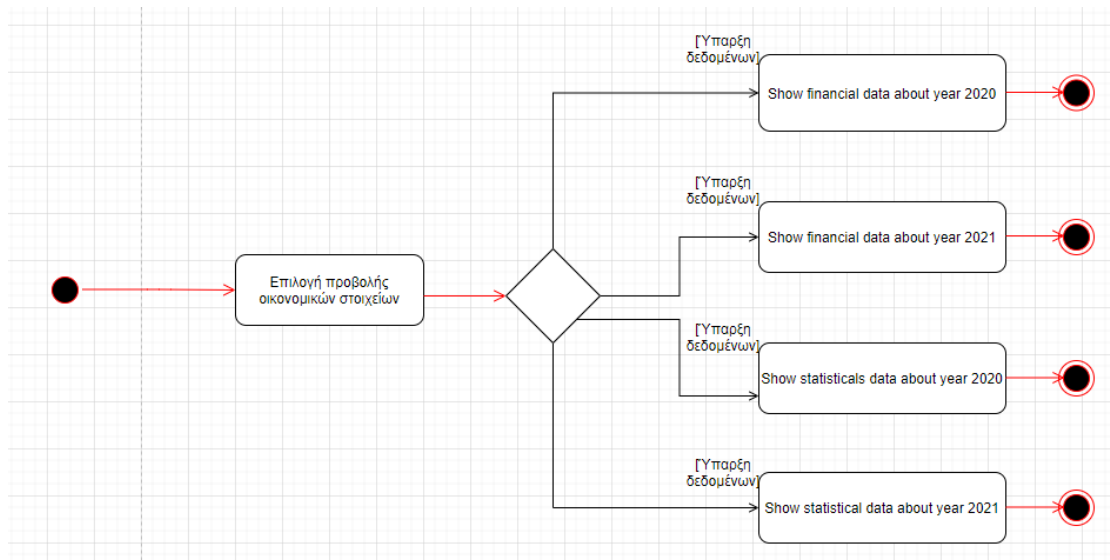
Εικόνα 28: Διάγραμμα Καταστάσεων 4 (1^η έκδοση)



Εικόνα 29: Διάγραμμα Καταστάσεων 5 (1^η έκδοση)



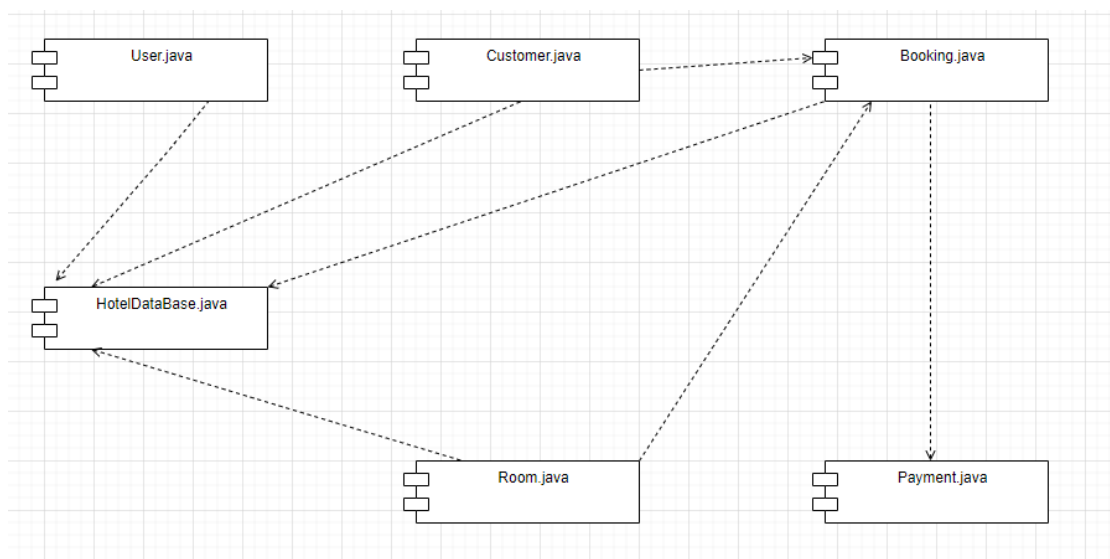
Εικόνα 30: Διάγραμμα Καταστάσεων 6 (1^η έκδοση)



Εικόνα 31: Διάγραμμα Καταστάσεων 7 (1^η έκδοση)

4.1.8 Διαγράμματα Εξαρτημάτων (1^η έκδοση)

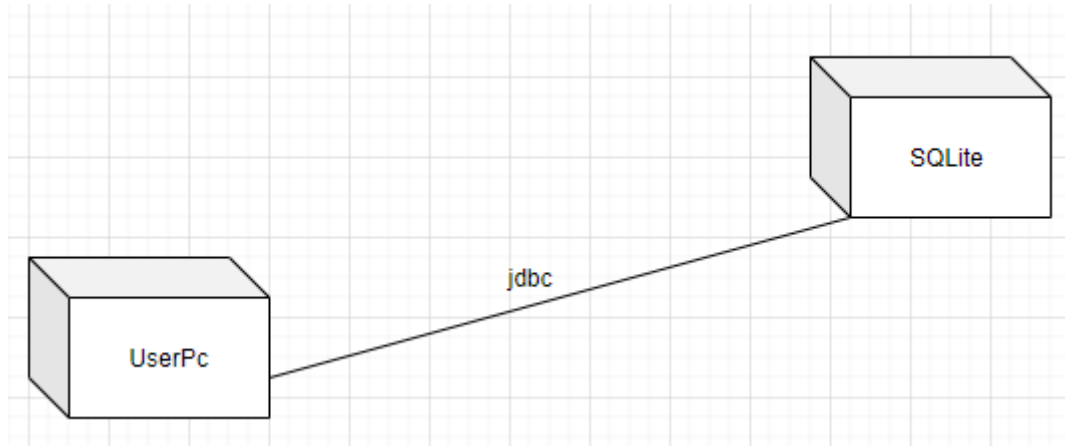
Τα διαγράμματα εξαρτημάτων περιγράφουν τμήματα λογισμικού και τη σχέση τους με το περιβάλλον υλοποίησης. Υποδηλώνουν τις επιλογές που γίνονται κατά τη χρονική στιγμή της υλοποίησης.



Εικόνα 32: Διάγραμμα Εξαρτημάτων (1^η έκδοση)

4.1.9 Διαγράμματα Διανομής (1η έκδοση)

Τα διαγράμματα διανομής δείχνουν τη φυσική διάταξη των διαφόρων (κόμβων) υλικού που αποτελούν ένα σύστημα, όπως και τη διανομή των εκτελέσιμων προγραμμάτων σε αυτό το υλικό. Οι διάφοροι κόμβοι συνδέονται μεταξύ τους με απλές γραμμές που αναπαριστούν μία υποδομή επικοινωνίας.



Εικόνα 33: Διάγραμμα Διανομής (1^η έκδοση)

4.2 Υλοποίηση-Έλεγχος

4.2.1 Υλοποίηση: 1η εκτελέσιμη έκδοση

Σύμφωνα με την παραπάνω ανάλυση απαιτήσεων, προκύπτει ότι η βάση δεδομένων θα πρέπει να περιλαμβάνει τους παρακάτω πίνακες.

Πίνακες στη βάση δεδομένων

User

Ο πίνακας user περιλαμβάνει πληροφορίες σχετικά με τους χρήστες και το ρόλο που μπορεί να έχει κάθε χρήστης και περιέχει τις εξής στήλες:

- ID: integer (ξεκινά από 1000 και ανεβαίνουμε) (primary key)
- name: String
- email: String
- username: String
- password: String
- userRole: integer (0 για Receptionist, 1 για Manager)

Customer

Ο πίνακας Customer περιλαμβάνει πληροφορίες σχετικά με τους πελάτες και περιέχει τις εξής στήλες:

- ID: integer (ξεκινά από 2000 και ανεβαίνουμε) (primary key)
- name: String
- email: String

Room

Ο πίνακας Room περιλαμβάνει πληροφορίες σχετικά με τα δωμάτια και περιέχει τις εξής στήλες:

- ID: integer. (ξεκινά από 3000 και ανεβαίνουμε) (primary key)
- roomType: String (single, double, suite)
- roomPrice: integer (single = 20 E/day, double = 30 E/day, suite = 50 E/day)

Booking

Ο πίνακας Booking περιλαμβάνει πληροφορίες σχετικά με τις κρατήσεις και περιέχει τις εξής στήλες:

- ID: integer (ξεκινά από 4000 και ανεβαίνουμε) (primary key)
- customerID: integer
- roomID: integer
- startDay: integer
- endDay: integer
- startMonth: integer

- endMonth: integer (0 για low season, στο διάστημα Οκτώβριος-Δεκέμβριος, 1 για medium season, στο διάστημα Ιανουάριος-Απρίλιος, 2 για high season, στο διάστημα Μάιος-Σεπτέμβριος). → (low: +1 Ε/ημέρα, medium: +5 Ε/ ημέρα, high: +10 Ε/ημέρα)

Payment

Ο πίνακας Payment περιλαμβάνει πληροφορίες σχετικά με τις πληρωμές και περιέχει τις εξής στήλες:

- ID: integer (ξεκινά από 5000 και ανεβαίνουμε) (primary key)
- bookingID: integer
- method: integer (0 για μετρητά, 1 για πιστωτική κάρτα)
- finalAmount: integer
- paymentDay: integer (θα ταυτίζεται με τη endDay από Booking)
- paymentMonth: integer (θα ταυτίζεται με τη endMonth από Booking)

Επίσης όπως έχει ήδη αναφερθεί η εφαρμογή διαχειρίζεται:

- Χρήστες
- Πελάτες
- Κρατήσεις
- Δωμάτια
- Πληρωμές

Οπότε οι κλάσεις που δημιουργήθηκαν είναι οι

- User
- Customer
- Booking
- Room
- Payment

Επιπρόσθετα δημιουργήθηκε η τάξη

- HotelDataBase

4.2.2 Αναφορά ελέγχου για την 1η εκτελέσιμη έκδοση

Κατά τη λειτουργία της εφαρμογής διαπιστώθηκε ότι:

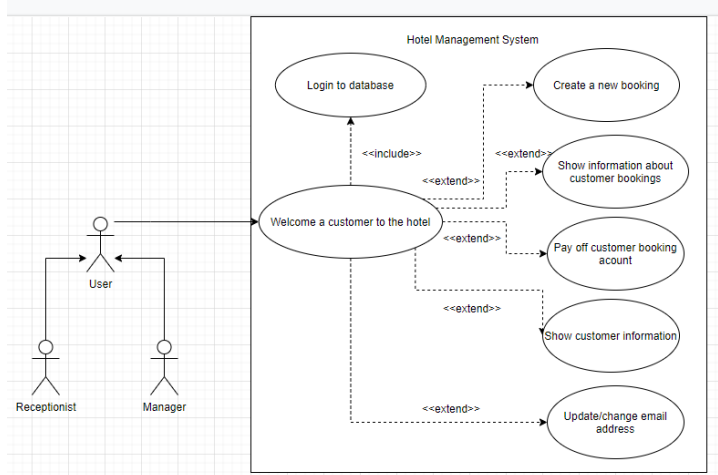
- 1) δε λειτουργούσε η επιλογή Create new booking, που αφορά τη δημιουργία νέας κράτησης για πελάτη.
- 2) Επιλέγοντας συνδυασμό εντολών στην τάξη customer, έκλεινε η εφαρμογή.
- 3) Τα reports Show statistical data about year 2020) και Show statistical data about year 2021 (so far) στο menu Hotel Data, δεν παράγουν αποτέλεσμα

5 Φάση: Κατασκευή (Construction)

5.1 Ανάλυση-Σχεδιασμός

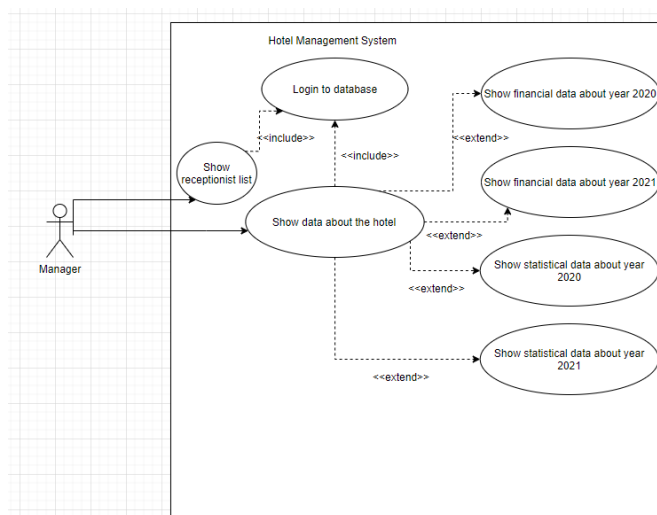
Στη φάση αυτή θα αναθεωρηθεί η ανάλυση και ο σχεδιασμός της προηγούμενης και θα αναπτυχθούν οι επόμενες εκδόσεις των αντίστοιχων διαγραμμάτων.

5.1.1 Διαγράμματα Περιπτώσεων Χρήσης (3η έκδοση)



Εικόνα 34: Διάγραμμα 1 Περίπτωση χρήσης Welcome customer to hotel (3^η έκδοση)

Στο παραπάνω διάγραμμα απεικονίζεται το Διάγραμμα Περίπτωσης Χρήσης του χρήστη για την περίπτωση που ο χρήστης επιλέγει να καλωσορίσει πελάτη στο ξενοδοχείο. Σε αυτή την περίπτωση ο χρήστης θα πρέπει υποχρεωτικά να έχει κάνει login στο σύστημα. Σε αυτή την περίπτωση μπορεί να δημιουργήσει νέα κράτηση για τον πελάτη, να δει πληροφορίες σχετικά με τις κρατήσεις του πελάτη, να πληρώσει την κράτηση του πελάτη, να δει πληροφορίες για τον πελάτη και να κάνει αλλαγή/ενημέρωση στο mail του πελάτη.



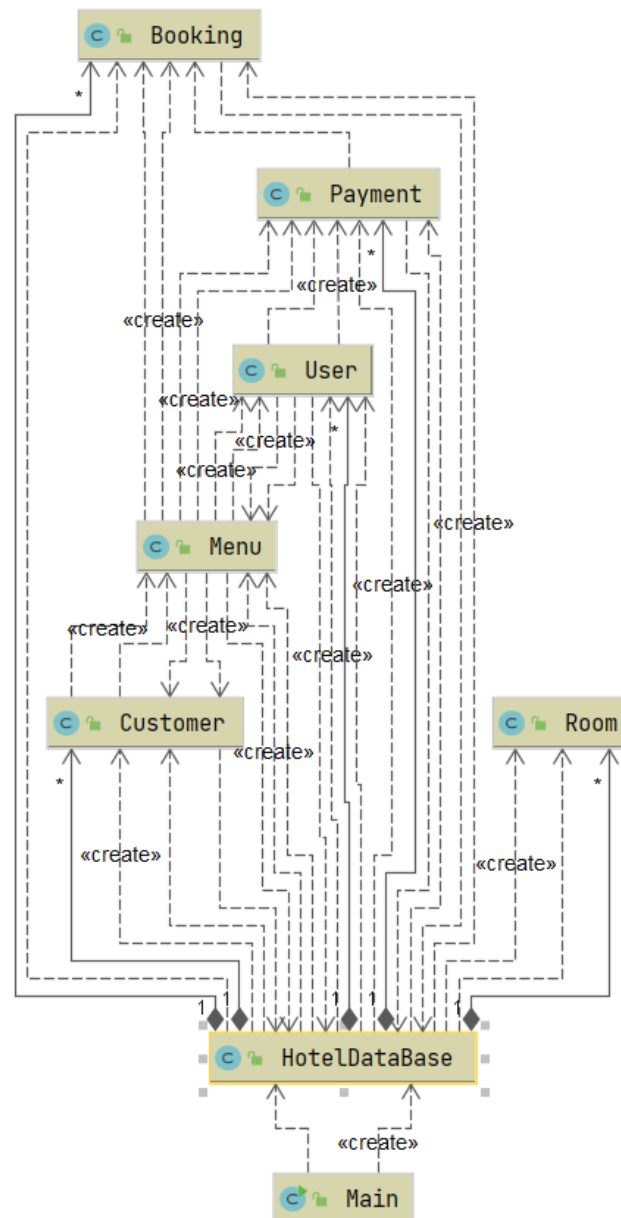
Εικόνα 35: Διάγραμμα 2 Περίπτωση χρήσης, Show data about the hotel, show receptionist list (3^{ης} Έκδοση)

Στο παραπάνω διάγραμμα απεικονίζεται το Διάγραμμα Περίπτωσης Χρήσης του χρήστη για την περίπτωση που ο χρήστης manager επιλέγει να δει οικονομικά και στατιστικά στοιχεία

του ξενοδοχείου έτους 2020 και 2021, και να προβάλει τη λίστα των receptionist. Ο χρήστης θα πρέπει υποχρεωτικά να έχει κάνει login στο σύστημα.

5.1.2 Διαγράμματα Τάξεων (3η έκδοση)

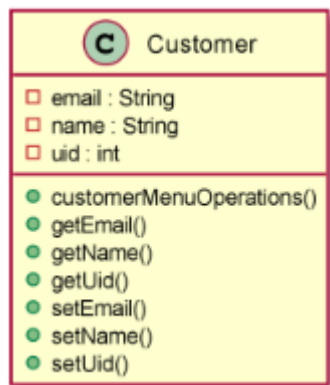
Το τροποποιημένο διάγραμμα τάξεων για το πληροφοριακό σύστημα της εφαρμογής μας φαίνεται παρακάτω. Υπάρχουν 8 τάξεις, η τάξη πελάτη (Customer), η τάξη κράτηση (Booking), η τάξη δωμάτιο (Room), η τάξη πληρωμή (Payment), που υπήρχαν στο διάγραμμα τάξεων 2^{ης} έκδοσης και επιπρόσθετα η τάξη χρήστη (User), η τάξη μενού (Menu), η τάξη της βάσης δεδομένων (HotelDataBase) και τέλος η κλάση της Main.



Εικόνα 36: Διάγραμμα Τάξεων (3^η έκδοση)

Ακολούθως περιγράφονται αναλυτικά οι λειτουργίες κάθε τάξης:

Τάξη Customer



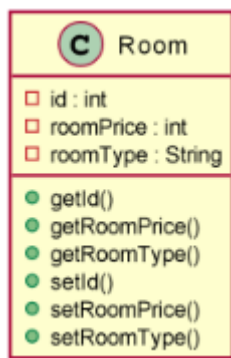
Στην τάξη Customer έχουμε τα εξής αντικείμενα:

- name : String
- email : String
- uid: int

Όπως και τις παρακάτω λειτουργίες:

- customerMenuOperations()
- getEmail()
- getName()
- getUid()
- setEmail()
- setName()
- setUid()

Τάξη Room



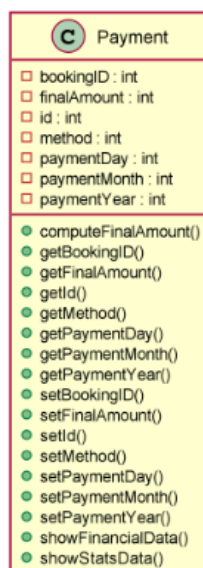
Στην τάξη Room έχουμε τα εξής αντικείμενα:

- id: int
- roomPrice : int
- roomType : String

Όπως και τις παρακάτω λειτουργίες:

- getId()
- getRoomPrice()
- getRoomType()
- setId()
- setRoomPrice()
- setRoomType()

Τάξη Payment



Στην τάξη Payment έχουμε τα εξής αντικείμενα:

- bookingId : int
- finalAmount : int
- id: int
- method : int
- paymentDay : int
- paymentMonth : int
- paymentYear : int

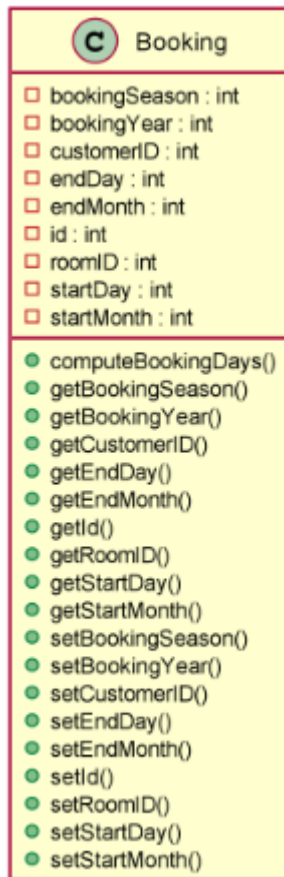
Όπως και τις παρακάτω λειτουργίες:

- computeFinalAmount()
- getBookingId()
- getFinalAmount()
- getId()
- getMethod()
- getPaymentDay()

- getPaymentMonth()
- getPaymentYear()
- setBookingId()
- setFinalAmount()
- setId()

- setPaymentDay()
- setPaymentMonth()
- setPaymentYear()
- showFinancialData()
- showStatsData()

Τάξη Booking



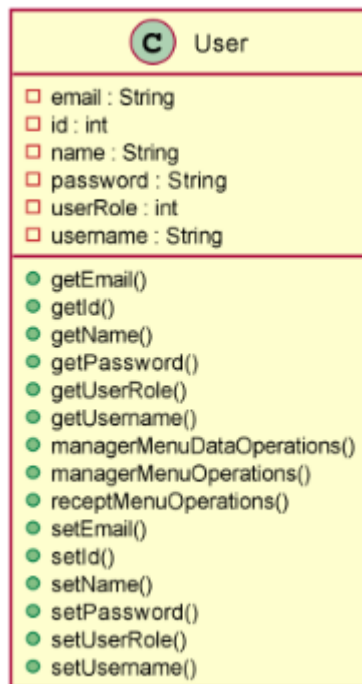
Στην τάξη Booking έχουμε τα εξής αντικείμενα:

- bookingSeason : int
- bookingYear : int
- CustomerId : int
- endDay : int
- endMonth : int
- id : int
- roomId : int
- startDay : int
- startMonth : int

Όπως και τις παρακάτω λειτουργίες:

- computeBookingDays()
- getBookingSeason()
- getBookingYear()
- getCustomerId()
- getEndDay()
- getEndMonth()
- getId()
- getRoomId()
- getStartDay()
- getStartMonth()
- setBookingSeason()
- setBookingYear()
- setCustomerId()
- setEndDay()
- setEndMonth()
- setId()
- setRoomId()
- setStartDay()
- setStartMonth()

Τάξη User



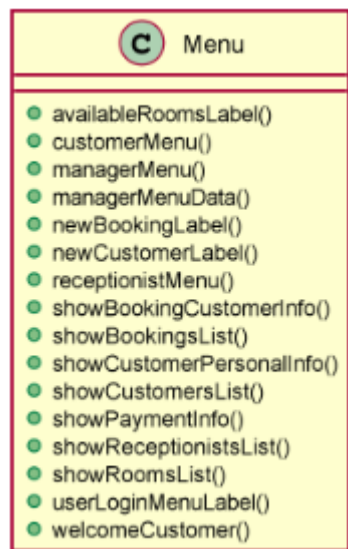
Στην τάξη User έχουμε τα εξής αντικείμενα:

- email : String
- id: int
- name : String
- password : String
- userRole : int
- username : String

Όπως και τις παρακάτω λειτουργίες:

- getEmail()
- getId()
- getName()
- getPassword()
- getUserRole()
- getUsername()
- managerMenuDataOperations()
- managerMenuOperations()
- receiptMenuOperations()
- setEmail()
- setId()
- setName()
- setPassword()
- setUserRole()
- setUsername()

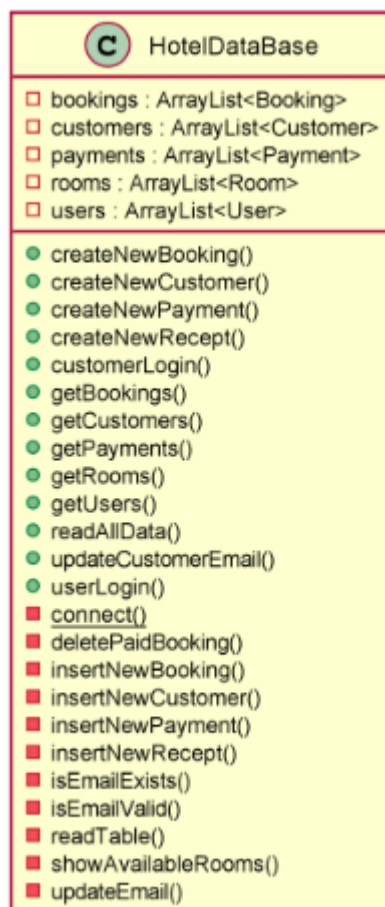
Τάξη Menu



Στην τάξη Menu έχουμε τις παρακάτω λειτουργίες:

- availableRoomsLabel()
- customerMenu()
- managerMenu()
- managerMenuData()
- newBookingLabel()
- newCustomerLabel()
- receptionistMenu()
- showBookingCustomerInfo()
- showBookingsList()
- showCustomerPersonalInfo()
- showCustomersList()
- showPaymentInfo()
- showReceptionistsList()
- showRoomsList()
- userLoginMenuLabel()
- welcomeCusotmer()

Τάξη HotelDataBase



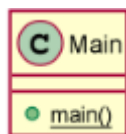
Στην τάξη HotelDataBase έχουμε τα εξής αντικείμενα:

- bookings
- customers
- payments
- rooms
- users

και τις παρακάτω λειτουργίες:

- createNewBooking()
- createNewCustomer()
- createNewPayment()
- customerLogin()
- getBookings()
- getCustomers()
- getPayments()
- getRooms()
- getUsers()
- readAllData()
- updateCustomerEmail()
- userLogin()
- connect()
- readTable()
- deletePaidBooking()
- insertNewBooking()
- insertNewCustomer()
- insertNewPayment()
- insertNewReceipt()
- isEmailExists()
- isEmailValid()
- readTable()
- showAvailableRooms()
- updateEmail()

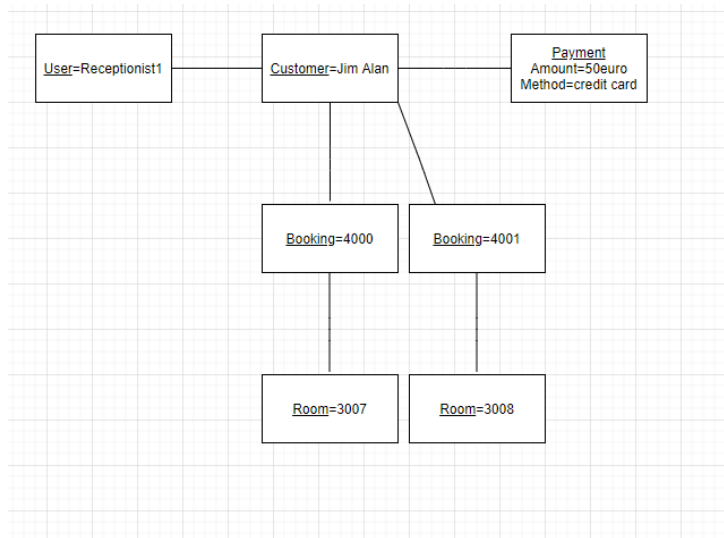
Τάξη Main



Η τάξη Main έχει την παρακάτω λειτουργία:

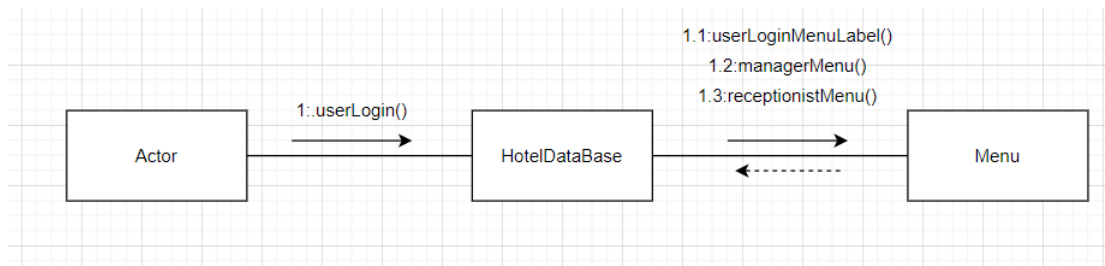
- main()

5.1.3 Διαγράμματα Αντικειμένων (2η έκδοση)

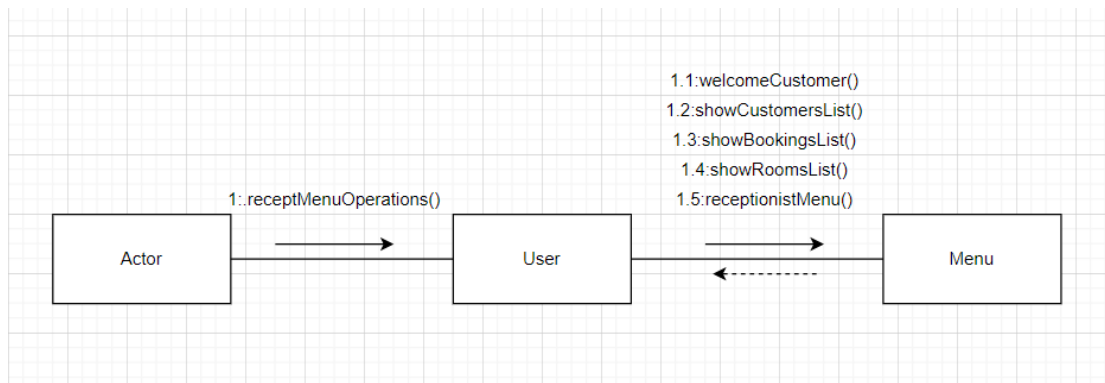


Εικόνα 37: Διάγραμμα Αντικειμένων (2^η έκδοση)

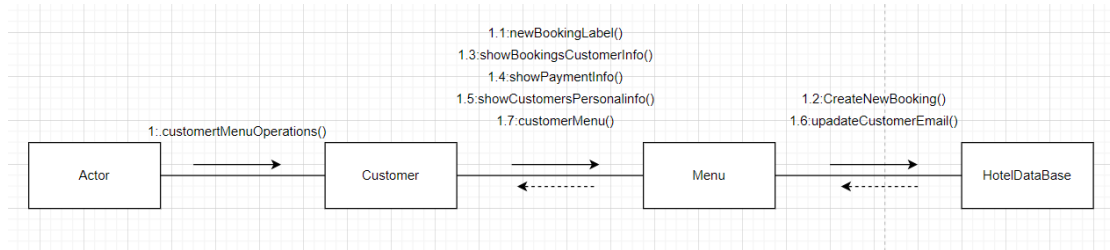
5.1.4 Διαγράμματα Συνεργασίας (2η έκδοση)



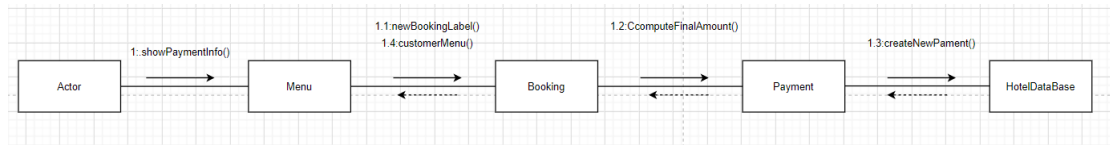
Εικόνα 38: Διάγραμμα Συνεργασίας 1 (2^η έκδοση)



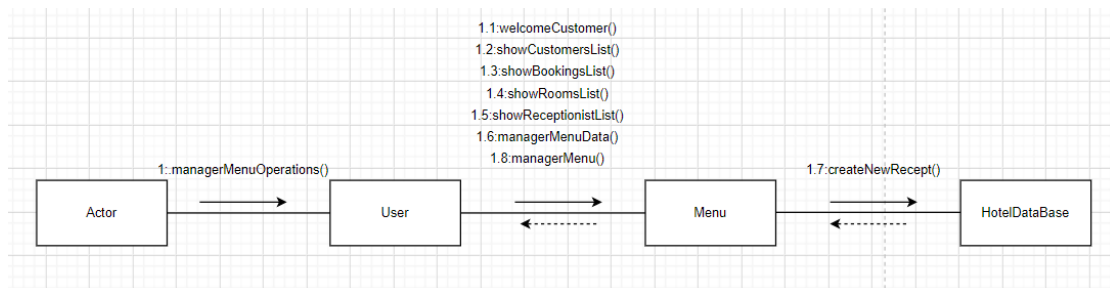
Εικόνα 39: Διάγραμμα Συνεργασίας 2 (2^η έκδοση)



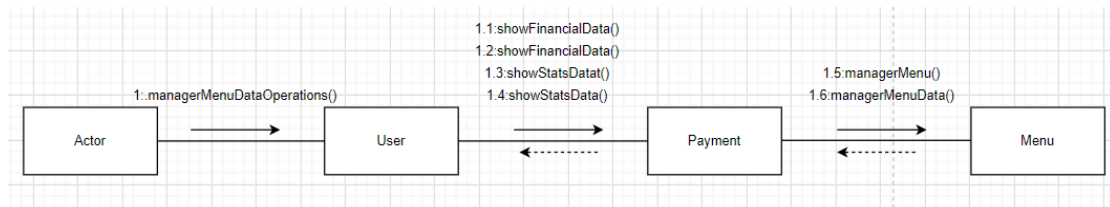
Εικόνα 40: Διάγραμμα Συνεργασίας 3 (2^η έκδοση)



Εικόνα 41: Διάγραμμα Συνεργασίας 4 (2^η έκδοση)

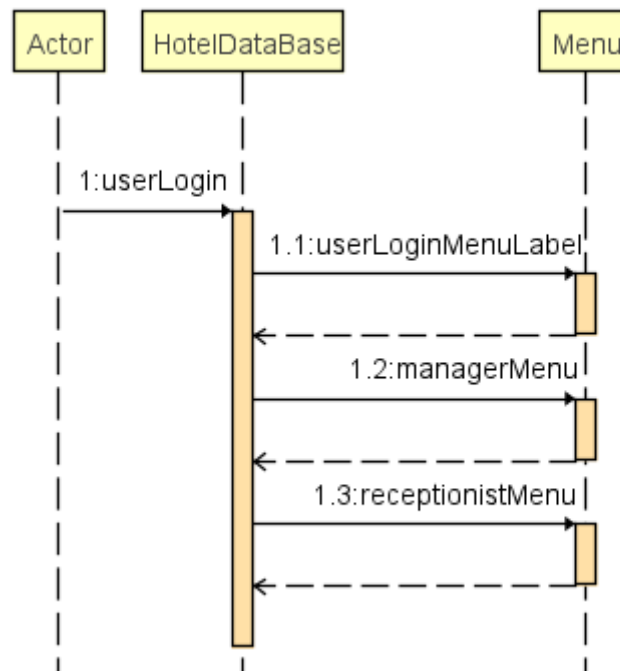


Εικόνα 42: Διάγραμμα Συνεργασίας 5 (2^η έκδοση)

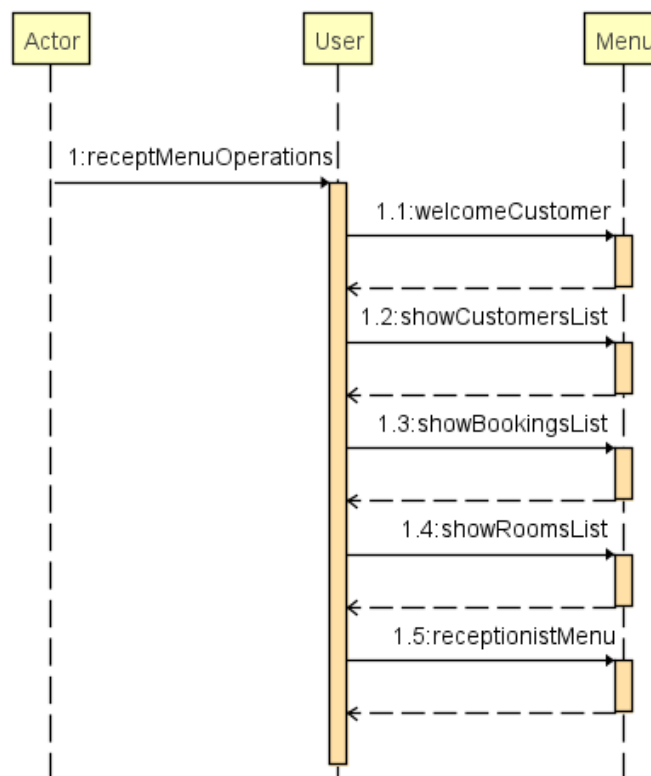


Εικόνα 43: Διάγραμμα Συνεργασίας 6 (2^η έκδοση)

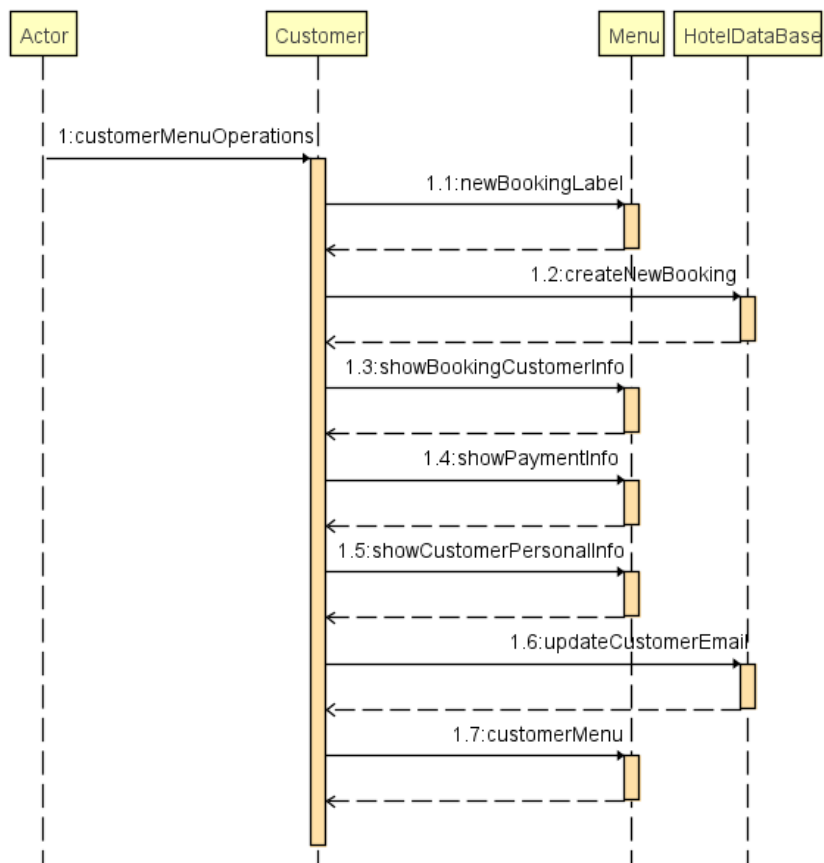
5.1.5 Διαγράμματα Σειράς (2η έκδοση)



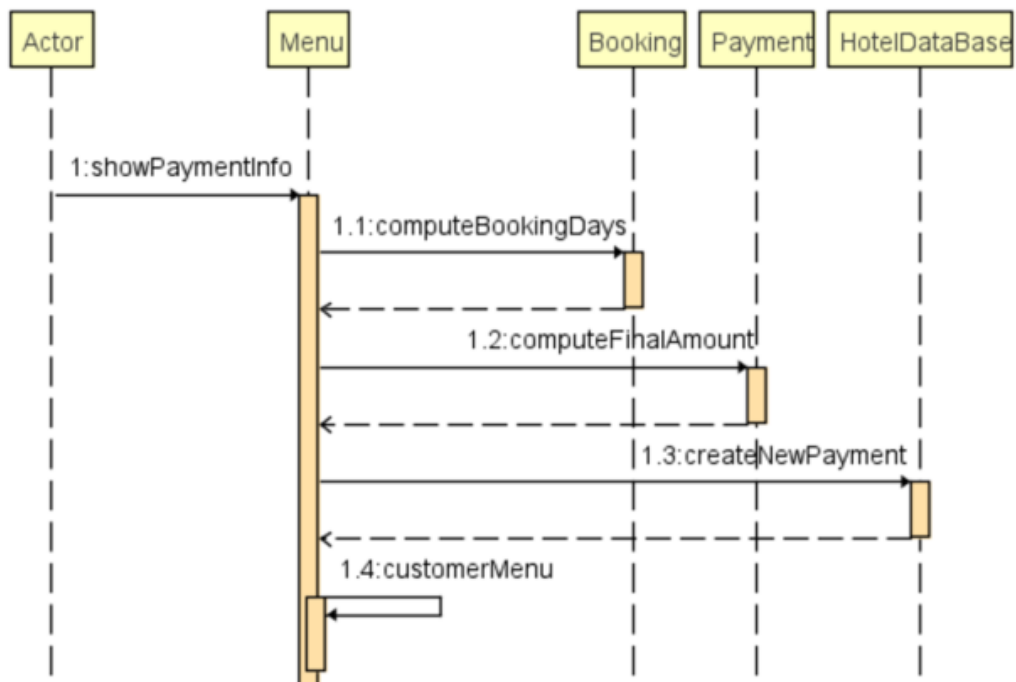
Εικόνα 44: Διάγραμμα Σειράς 1 (2^η έκδοση)



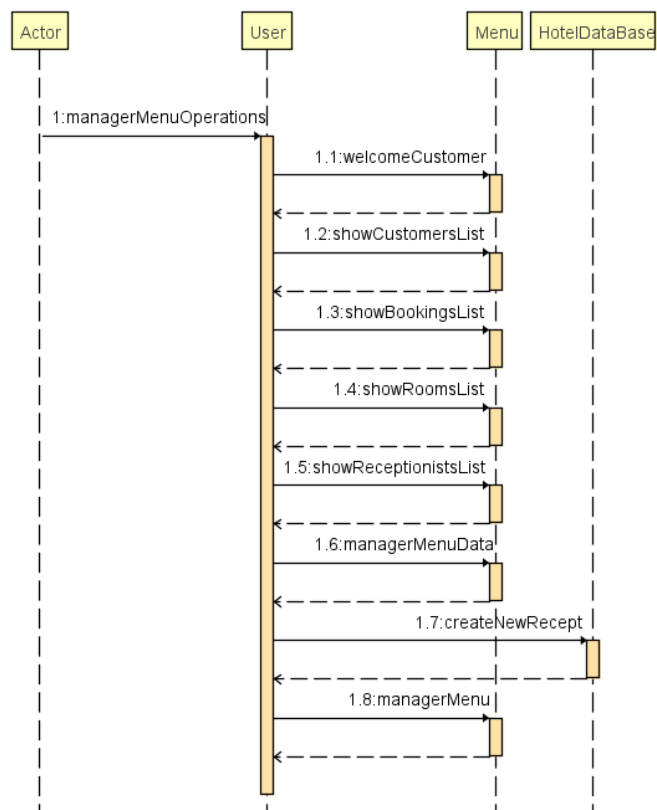
Εικόνα 45: Διάγραμμα Σειράς 2 (2^η έκδοση)



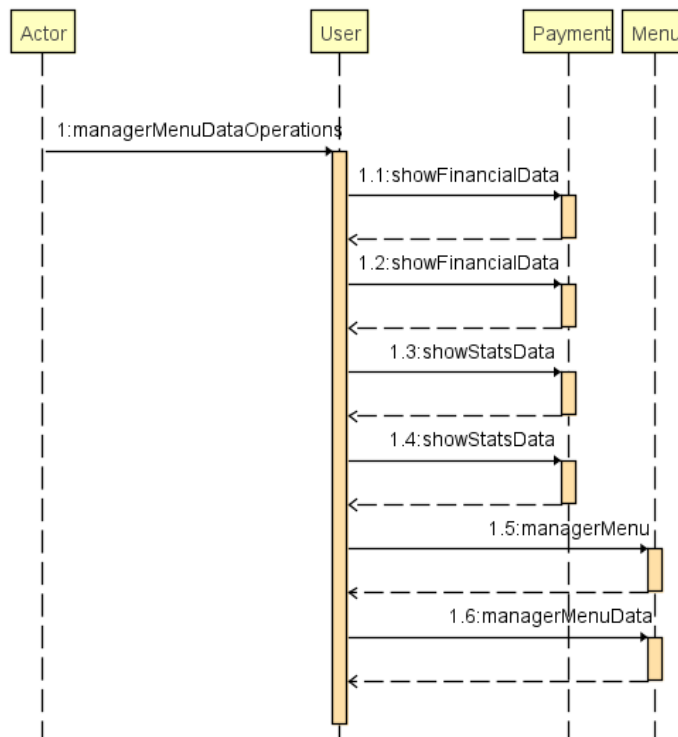
Εικόνα 46: Διάγραμμα Σειράς 3 (2^η έκδοση)



Εικόνα 47: Διάγραμμα Σειράς 4 (2^η έκδοση)

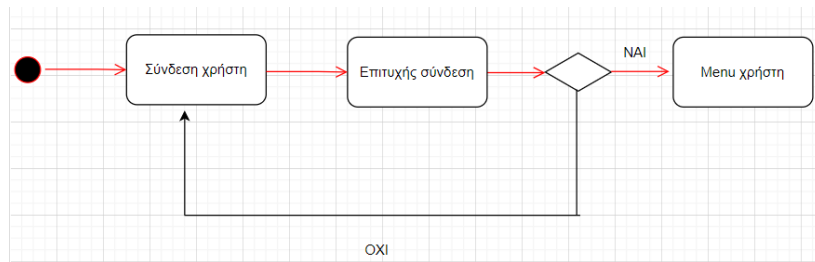


Εικόνα 48: Διάγραμμα Σειράς 5 (2^η έκδοση)

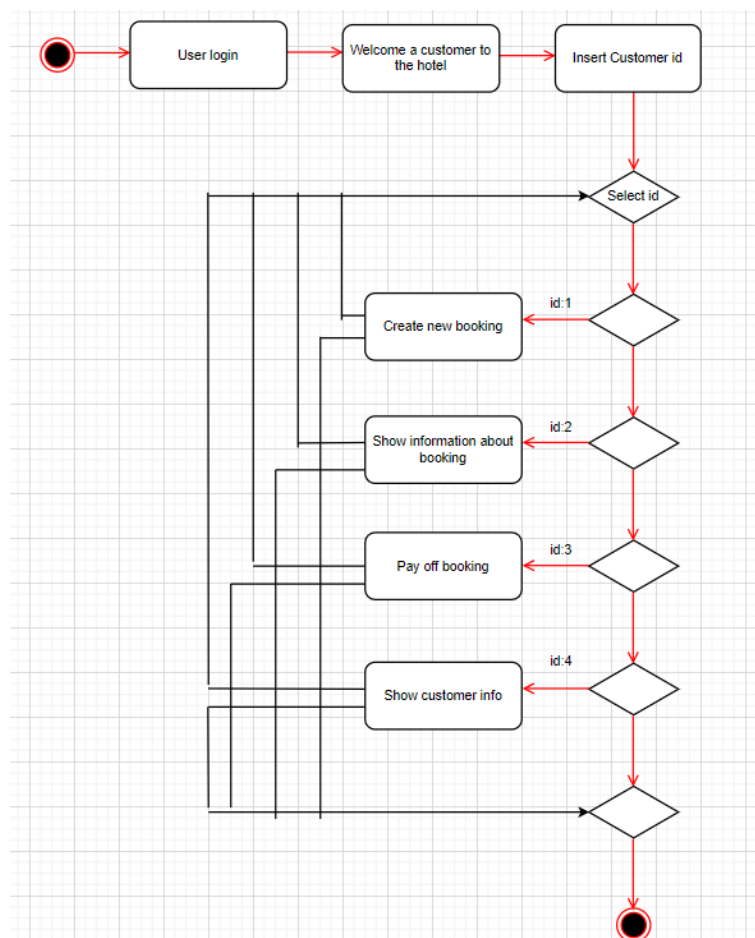


Εικόνα 49: Διάγραμμα Σειράς 6 (2^η έκδοση)

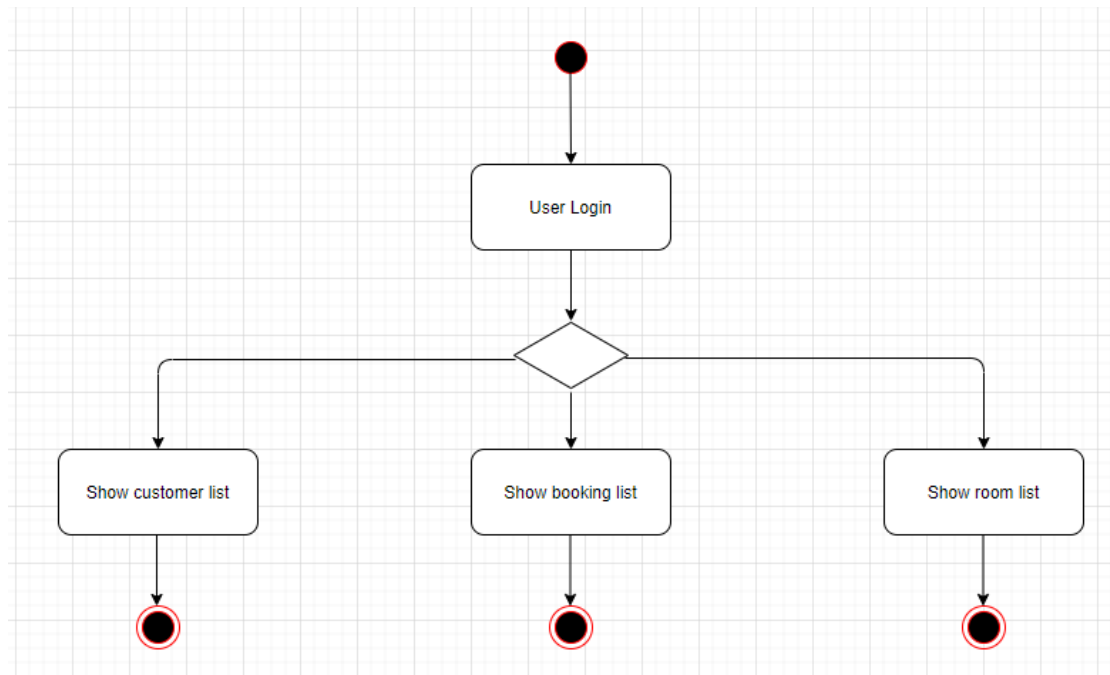
5.1.6 Διαγράμματα Δραστηριοτήτων (2η έκδοση)



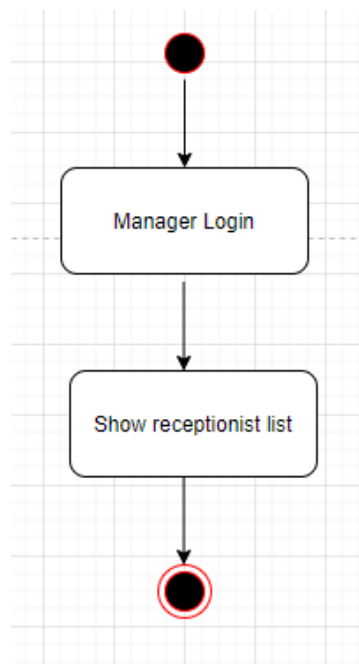
Εικόνα 50: Διάγραμμα Δραστηριοτήτων 1 (2^η έκδοση)



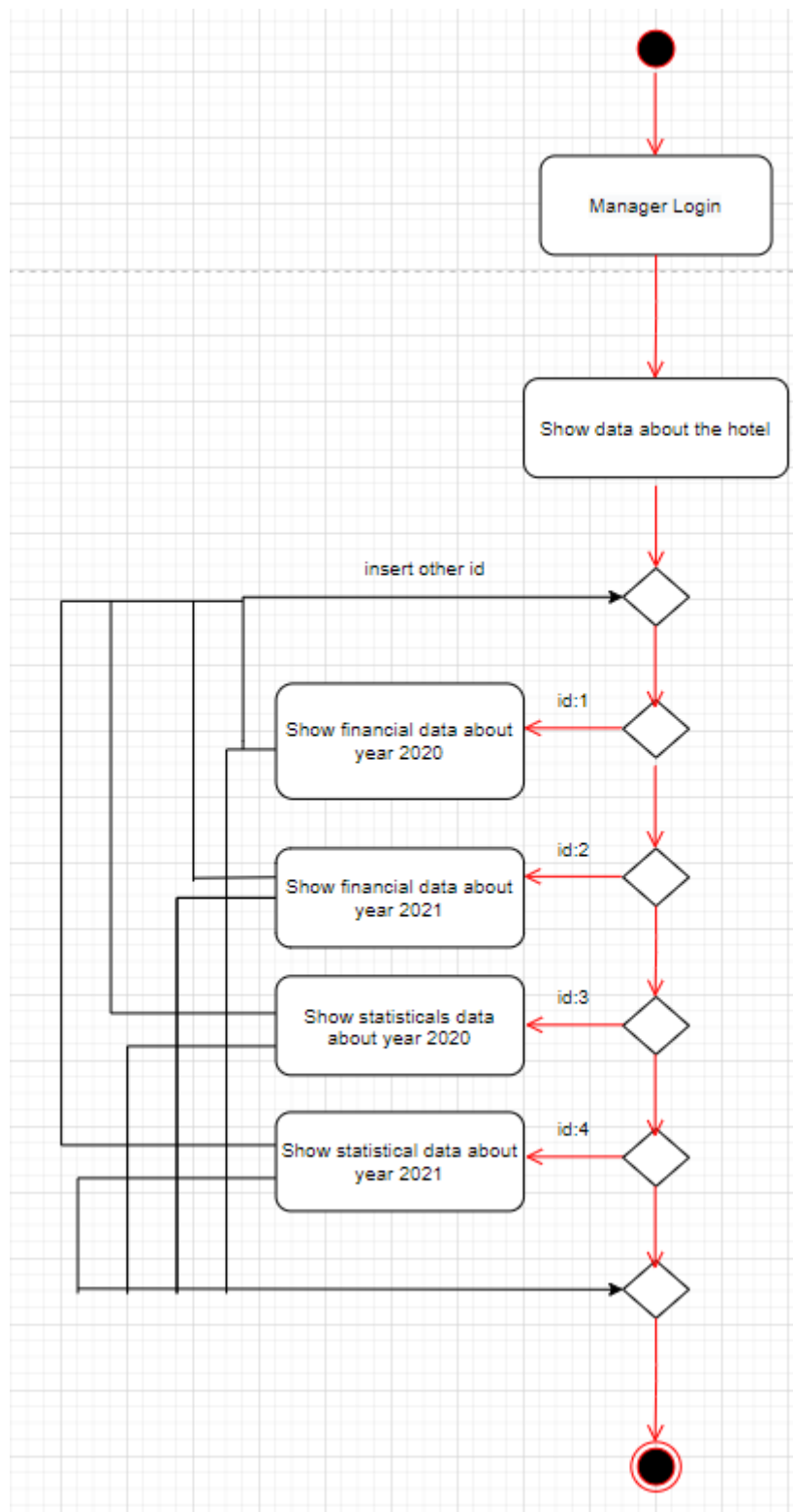
Εικόνα 51: Διάγραμμα Δραστηριοτήτων 2 (2^η έκδοση)



Εικόνα 52: Διάγραμμα Δραστηριοτήτων 3 (2^η έκδοση)

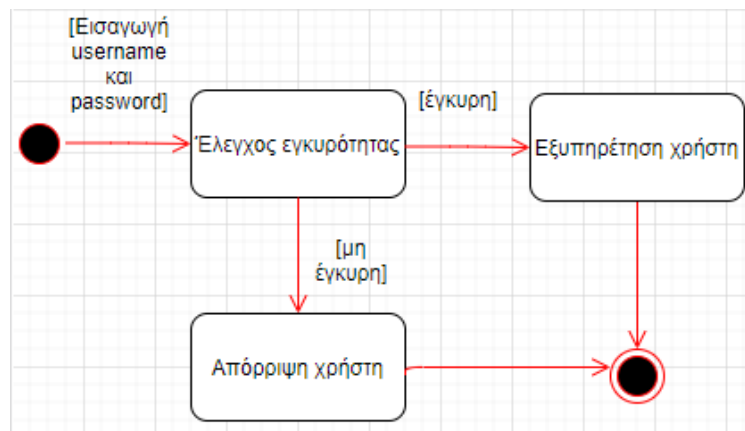


Εικόνα 53: Διάγραμμα Δραστηριοτήτων 4 (2^η έκδοση)

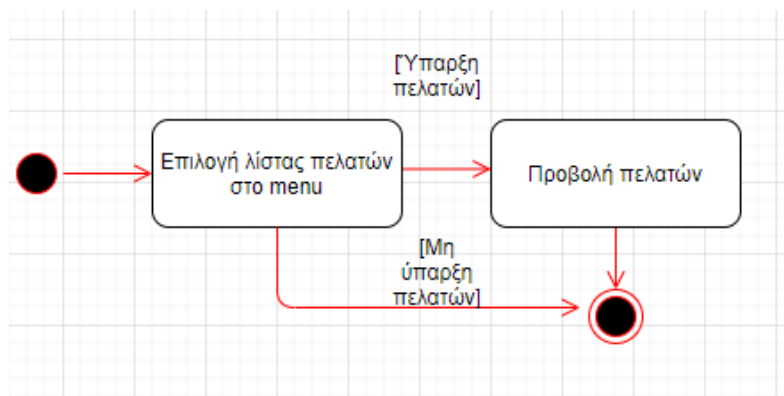


Εικόνα 54: Διάγραμμα Δραστηριοτήτων 5 (2^η έκδοση)

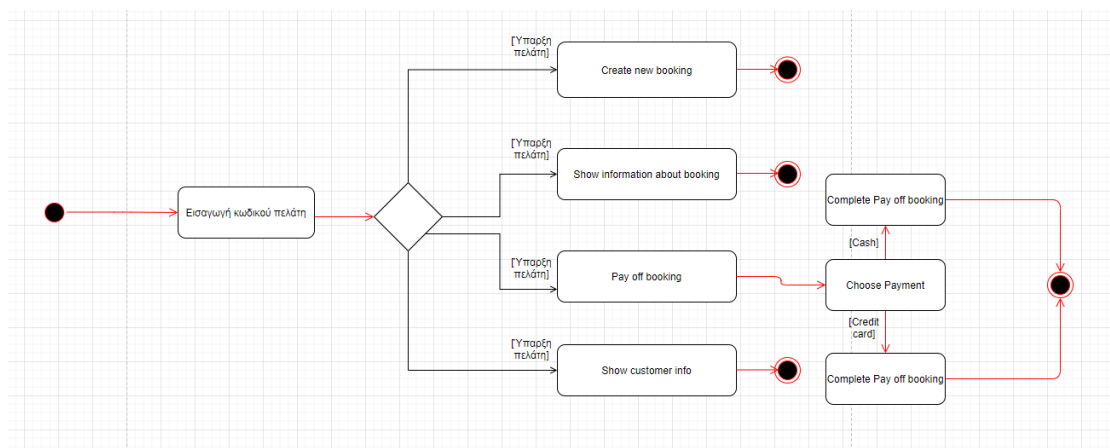
5.1.7 Διαγράμματα Καταστάσεων (2η έκδοση)



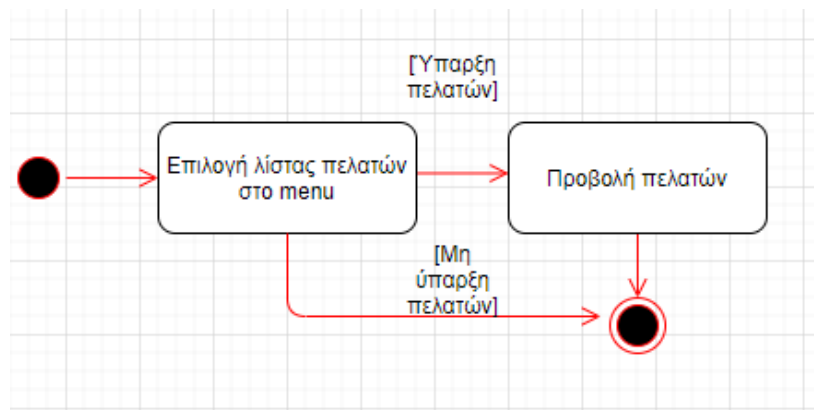
Εικόνα 55: Διάγραμμα Καταστάσεων 1 (2^η έκδοση)



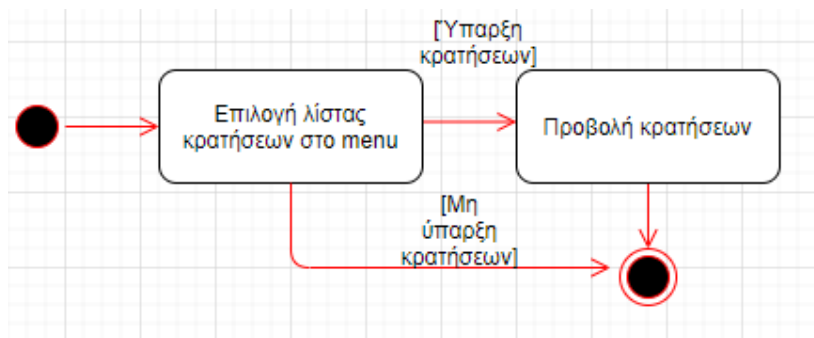
Εικόνα 56: Διάγραμμα Καταστάσεων 2 (2^η έκδοση)



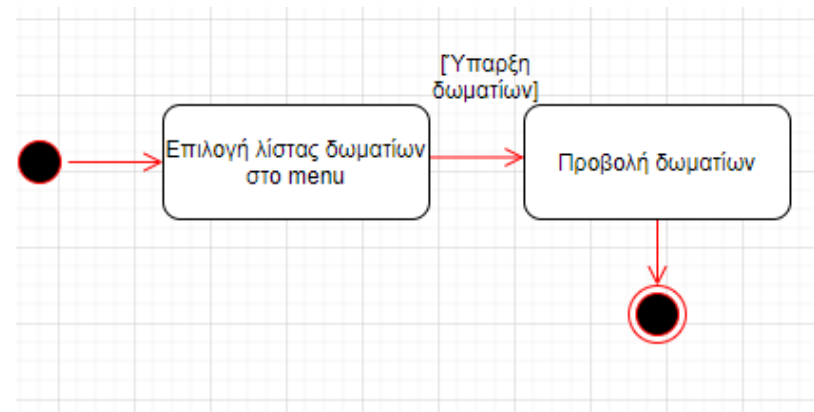
Εικόνα 57: Διάγραμμα Καταστάσεων 3 (2^η έκδοση)



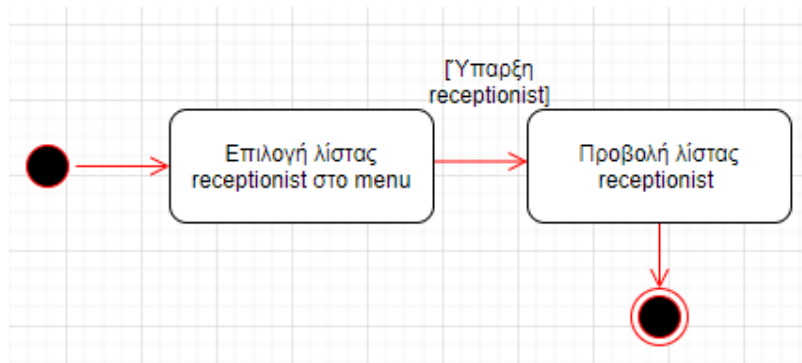
Εικόνα 58: Διάγραμμα Καταστάσεων 4 (2^η έκδοση)



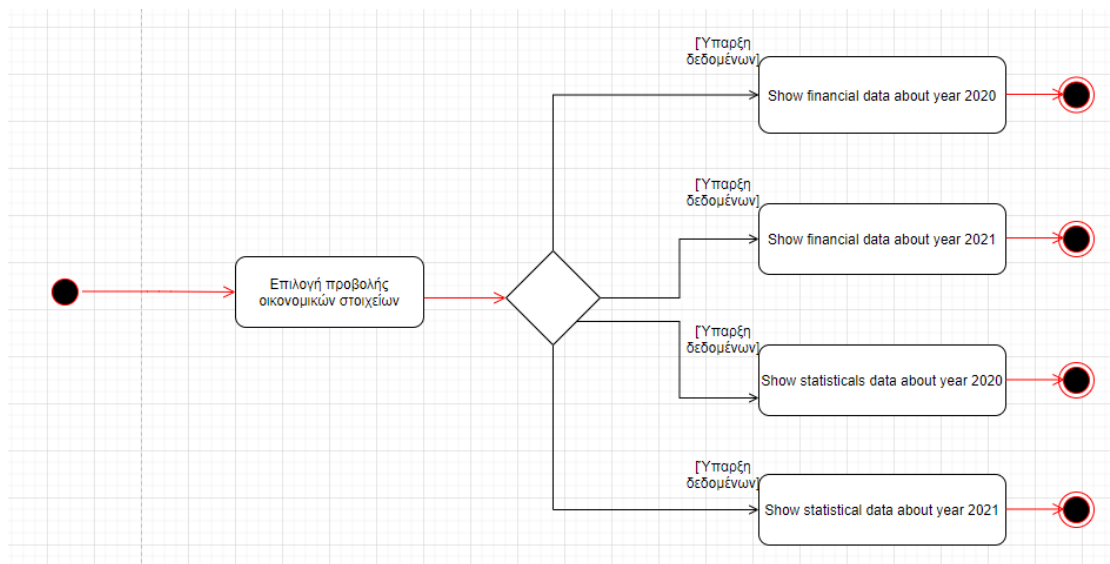
Εικόνα 59: Διάγραμμα Καταστάσεων 5 (2^η έκδοση)



Εικόνα 60: Διάγραμμα Καταστάσεων 6 (2^η έκδοση)

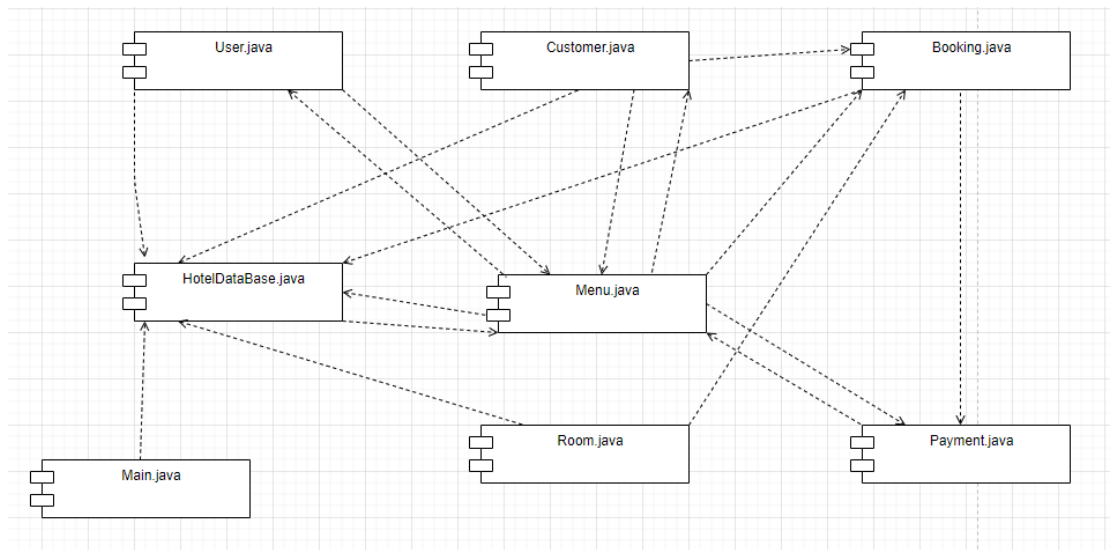


Εικόνα 61: Διάγραμμα Καταστάσεων 7 (2^η έκδοση)



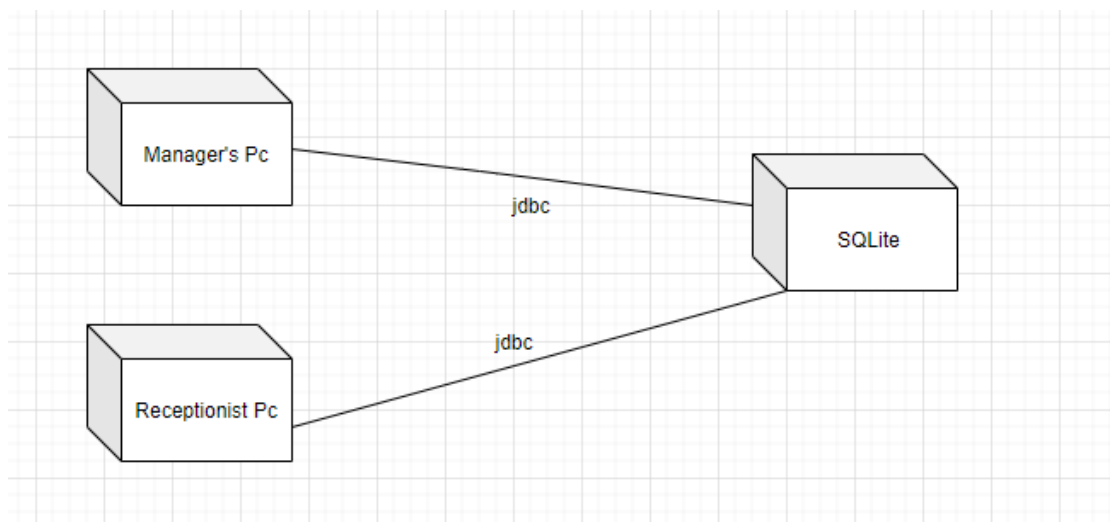
Εικόνα 62: Διάγραμμα Καταστάσεων 8 (2^η έκδοση)

5.1.8 Διαγράμματα Εξαρτημάτων (2η έκδοση)



Εικόνα 63: Διάγραμμα Εξαρτημάτων (2^η έκδοση)

5.1.9 Διαγράμματα Διανομής (2η έκδοση)



Εικόνα 64: Διάγραμμα Διανομής (2^η έκδοση)

5.2 Υλοποίηση-Έλεγχος

5.2.1 Υλοποίηση: Τελική εκτελέσιμη έκδοση

Η τελική έκδοση του Hotel Management System ενσωμάτωσε όλες τις παρατηρήσεις της αναφοράς ελέγχου της 1^{ης} εκτελέσιμης έκδοσης, ώστε να μην παρατηρούνται οι δυσλειτουργίες και οι αστοχίες που καταγράφηκαν.

Πιο συγκεκριμένα

- 1) Λειτουργεί πλέον η επιλογή Create new booking, που αφορά τη δημιουργία νέας κράτησης για πελάτη.
- 2) Επιλέγοντας συνδυασμό εντολών στην τάξη customer, δεν κλείνει πλέον η εφαρμογή.
- 3) Τα reports Show statistical data about year 2020) και Show statistical data about year 2021 (so far) στο menu Hotel Data, παράγουν πλέον αποτέλεσμα

5.2.2 Αναφορά ελέγχου για την τελική εκτελέσιμη έκδοση

Η αναφορά ελέγχου της τελικής έκδοσης ανέφερε τα εξής:

1. Μόλις ολοκληρωθεί η κράτηση τερματίζει η εφαρμογή.

6 Εγχειρίδιο Χρήστη

6.1 Σύντομη παρουσίαση του προγράμματος

Πρόκειται για μία desktop εφαρμογή κονσόλας όπου οι χρήστες του συστήματος διαχείρισης είναι ο υπάλληλος στην υποδοχή και ο manager του ξενοδοχείου.

Ο υπάλληλος στην υποδοχή, μέσω του συστήματος διαχείρισης του ξενοδοχείου, εξυπηρετεί τους πελάτες, ελέγχει τα δωμάτια του ξενοδοχείου, διαχειρίζεται τις κρατήσεις των πελατών και καταχωρεί τα στοιχεία των πελατών στη βάση δεδομένων του συστήματος.

Ο manager, μέσω του συστήματος διαχείρισης του ξενοδοχείου, μπορεί να κάνει ότι και ο υπάλληλος υποδοχής και επιπρόσθετα μπορεί να λάβει εκτυπώσεις που αφορούν στο ξενοδοχείο, όπως:

- Οικονομικά στοιχεία τρέχοντος ή προηγούμενου έτους
- Στατιστικά στοιχεία τρέχοντος ή προηγούμενου έτους

Όλες οι παραπάνω πληροφορίες καταγράφονται στη βάση δεδομένων του συστήματος.

6.2 Παρουσίαση σεναρίων λειτουργίας

Σύνδεση χρήστη Receptionist

Ο χρήστης συνδέεται στο σύστημα εισάγοντας το Username και Password.

```
-----  
|          USER LOGIN TO BASE          |  
-----
```

```
Username: recept1  
Password: recept1
```

Εάν ο χρήστης καταχωρήσει λάθος στοιχεία εμφανίζεται σχετικό μήνυμα λάθους.

```
Error! Please try again!
```

Εφόσον εισάγει τα στοιχεία του επιτυχώς τότε εμφανίζεται το menu χρήστη.

```
-----  
|          RECEPTIONIST MENU          |  
-----
```

```
Select one operation (type the ID number).  
ID: 1 ---> Welcome a customer to the hotel.  
ID: 2 ---> Show customers list.  
ID: 3 ---> Show bookings list.  
ID: 4 ---> Show rooms list.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU):
```

Επιλέγοντας το ID: 1 (Welcome a customer to the hotel) στο menu τότε εμφανίζεται ερώτηση αν έχει ο πελάτης καταχωρημένο λογαριασμό στο σύστημα. Με επιλογή "no" ο χρήστης μπορεί να εισάγει τα στοιχεία πελάτη ώστε να δημιουργηθεί ο λογαριασμός του.

Enter the menu ID number (ID: 0 ---> MENU): 1

```
-----  
|           WELCOME TO OUR HOTEL           |  
-----
```

Do you have an account? (Yes or No): no

```
-----  
|           NEW CUSTOMER                     |  
-----
```

Please give me your personal information.
First name: Andrew
Last name: Kotzafiliotis
Email: adny@gmail.com
New customer created successfully!

Εφόσον υπάρχει ο λογαριασμός, τότε εισάγοντας το μοναδικό αριθμό πελάτη, εμφανίζονται οι διαθέσιμες επιλογές για τον πελάτη.

```
-----  
|           WELCOME TO OUR HOTEL           |  
-----
```

Do you have an account? (Yes or No): yes
Please give me your unique ID: 20006
Success customer log in!

```
-----  
|           CUSTOMER MENU                   |  
-----
```

Select one operation (type the ID number).
ID: 1 ---> Create a new booking.
ID: 2 ---> Show information about your booking(s).
ID: 3 ---> Pay off your booking account.
ID: 4 ---> Show your personal information.
ID: 5 ---> Update/change your email address.
ID: -1 ---> Exit program.
Enter the menu ID number (ID: 0 ---> MENU):

Επιλέγοντας το ID: 1 (Create a new booking) στο Customer menu, εμφανίζονται τα πεδία προς εισαγωγή για να προχωρήσει η νέα κράτηση.

Enter the menu ID number (ID: 0 ---> MENU): 1

```
-----  
|           NEW BOOKING                     |  
-----
```

Start day: 1
End day: 5
Month: 4
Room Type (Single, Double or Suite): suite

Έπειτα εμφανίζονται τα διαθέσιμα δωμάτια και επιλέγει ο χειριστής το room id για να ολοκληρωθεί η καταχώρηση.

```
-----  
Room # 14  
-----  
ID: 3013  
Price: 50 €/day
```

Give me the room ID you want to book: 14
New booking record created successfully!

Επιλέγοντας το ID: 2 (Show information about your booking(s)) στο Customer menu,
εμφανίζεται/ονται η/οι κράτηση/εις του πελάτη.

Enter the menu ID number (ID: 0 ---> MENU): 2

```
-----  
|          BOOKING INFO          |  
-----
```

Your booking # 1

Booking's ID: 4000
Room's ID: 3000
Booking's period: From 11.2.2021 to 15.2.2021
Season: Medium

Your booking # 2

Booking's ID: 4007
Room's ID: 3015
Booking's period: From 1.5.2021 to 10.5.2021
Season: High

Your booking # 3

Booking's ID: 4042
Room's ID: 3022
Booking's period: From 1.10.2021 to 5.10.2021
Season: Low

Enter the menu ID number (ID: 0 ---> MENU): |

Επιλέγοντας το ID: 3 (Pay off your booking account) στο Customer menu, εμφανίζεται/ονται η/οι κράτηση/εις του πελάτη με το συνολικό ποσό πληρωμής.

Enter the menu ID number (ID: 0 ----> MENU): 3

```
-----
|                PAYMENT INFO                |
|-----|
```

Your booking # 1

Booking's ID: 4000
Booking's period: From 11.2.2021 to 15.2.2021
Room's ID: 3000
Total amount: 88 €

Your booking # 2

Booking's ID: 4007
Booking's period: From 1.5.2021 to 10.5.2021
Room's ID: 3015
Total amount: 207 €

Your booking # 3

Booking's ID: 4042
Booking's period: From 1.10.2021 to 5.10.2021
Room's ID: 3022
Total amount: 124 €

You have more than one booking active in our hotel.

Are you sure you want to pay one of your bookings? (Yes or No): |

Με επιλογή "yes", εισάγει ο χρήστης τον αριθμό κράτησης και το ποσό και εμφανίζεται η επιλογή μεθόδου πληρωμής όπου ο πελάτης μπορεί να πληρώσει με μετρητά ή κάρτα.

You have more than one booking active in our hotel.

Are you sure you want to pay one of your bookings? (Yes or No): yes

Please choose the booking ID you want to pay now: 4000

Fill the amount of current booking: 88

Payment method

1. Credit card (-20 € discount).

2. Cash.

Customer's choice: 1

Thank you so much! See you soon!

Με επιλογή "no", επιστρέφει ο χρήστης στο customer menu.

Επιλέγοντας το ID: 4 (Show your personal information) στο Customer menu, εμφανίζονται πληροφορίες σχετικά με τον πελάτη.

```
-----  
|          CUSTOMER INFO          |  
-----  
  
UID: 20006  
Name: Jim Alan  
Email: jim_alan__22@t-online.de
```

Επιλέγοντας το ID: 5 (Update/change your mail address) στο Customer menu, ο χρήστης μπορεί να αλλάξει το email του πελάτη.

```
-----  
|          CUSTOMER MENU          |  
-----  
  
Select one operation (type the ID number).  
ID: 1 ---> Create a new booking.  
ID: 2 ---> Show information about your booking(s).  
ID: 3 ---> Pay off your booking account.  
ID: 4 ---> Show your personal information.  
ID: 5 ---> Update/change your email address.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU): 5  
  
Give me your new email: test@gamil.com  
Customer's email updated successfully!
```

Επιλέγοντας το ID:-1 (Exit Program) επιστρέφει στο menu χρήστη.

```
-----  
|          RECEPTIONIST MENU      |  
-----  
  
Select one operation (type the ID number).  
ID: 1 ---> Welcome a customer to the hotel.  
ID: 2 ---> Show customers list.  
ID: 3 ---> Show bookings list.  
ID: 4 ---> Show rooms list.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU):
```

Επιλέγοντας το ID: 2 (Show customer list) στο menu χρήστη, εμφανίζεται η λίστα πελατών που είναι καταχωρημένοι στο σύστημα.

```
-----  
|          CUSTOMERS LIST          |  
-----  
  
-----  
Customer # 1  
-----  
UID: 20000  
Name: Santiago Perez  
Email: santiagop@yahoo.de  
  
-----  
Customer # 2  
-----  
UID: 20001  
Name: Jessica Brosman  
Email: jess_bros@hotmail.it
```

Επιλέγοντας το ID: 3 (Show bookings list) στο menu χρήστη, εμφανίζεται η λίστα των κρατήσεων που είναι καταχωρημένες στο σύστημα.

```
-----  
Booking # 35  
-----  
ID: 4056  
Customer's UID: 20053  
Room's ID: 3007  
Booking's period: From 28.9.2021 to 30.9.2021  
Season: High
```

```
-----  
Booking # 36  
-----  
ID: 4057  
Customer's UID: 20008  
Room's ID: 3014  
Booking's period: From 21.4.2021 to 23.4.2021  
Season: Medium
```

Επιλέγοντας το ID: 4 (Show rooms list) στο menu χρήστη, εμφανίζεται η λίστα των δωματίων που είναι καταχωρημένα στο σύστημα.

```
-----  
Room # 50  
-----  
ID: 3049  
Type: Single  
Price: 20 €/day
```

```
-----  
Room # 51  
-----  
ID: 3050  
Type: Double  
Price: 30 €/day
```

Επιλέγοντας το ID: -1 (Exit Program) στο menu χρήστη, τερματίζεται η λειτουργία του συστήματος.

Σύνδεση χρήστη Manger

Ο χρήστης συνδέεται στο σύστημα εισάγοντας το Username και Password.

```
-----  
|                USER LOGIN TO BASE                |  
-----
```

```
Username: manager1  
Password: manager1
```

Εάν ο χρήστης καταχωρήσει λάθος στοιχεία εμφανίζεται σχετικό μήνυμα λάθους.

Error! Please try again!

Εφόσον εισάγει τα στοιχεία του επιτυχώς τότε εμφανίζεται το menu χρήστη. Το menu έχει τις 4 λειτουργίες που αναφέρθηκαν παραπάνω. Επιπρόσθετα έχει και άλλες δύο επιλογές που θα εξηγήσουμε παρακάτω:

```
-----  
|                MANAGER MENU                |  
-----
```

```
Select one operation (type the ID number).  
ID: 1 ---> Welcome a customer to the hotel.  
ID: 2 ---> Show customers list.  
ID: 3 ---> Show bookings list.  
ID: 4 ---> Show rooms list.  
ID: 5 ---> Show receptionists list.  
ID: 6 ---> Show data about hotel.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU):
```

Επιλέγοντας το ID: 1 (Welcome a customer to the hotel) στο menu τότε εμφανίζεται ερώτηση αν έχει ο πελάτης καταχωρημένο λογαριασμό στο σύστημα. Με επιλογή "no" ο χρήστης μπορεί να εισάγει τα στοιχεία πελάτη ώστε να δημιουργηθεί ο λογαριασμός του.

```
Enter the menu ID number (ID: 0 ---> MENU): 1
```

```
-----  
|                WELCOME TO OUR HOTEL                |  
-----
```

```
Do you have an account? (Yes or No): no
```

```
-----  
|                NEW CUSTOMER                |  
-----
```

```
Please give me your personal information.  
First name: Andrew  
Last name: Kotzafilios  
Email: adny@gmail.com  
New customer created successfully!
```

Εφόσον υπάρχει ο λογαριασμός, τότε εισάγοντας το μοναδικό αριθμό πελάτη, εμφανίζονται οι διαθέσιμες επιλογές για τον πελάτη.


```
-----  
|           WELCOME TO OUR HOTEL           |  
-----
```

```
Do you have an account? (Yes or No): yes  
Please give me your unique ID: 20006  
Success customer log in!
```

```
-----  
|           CUSTOMER MENU                   |  
-----
```

```
Select one operation (type the ID number).  
ID: 1 ----> Create a new booking.  
ID: 2 ----> Show information about your booking(s).  
ID: 3 ----> Pay off your booking account.  
ID: 4 ----> Show your personal information.  
ID: 5 ----> Update/change your email address.  
ID: -1 ----> Exit program.  
Enter the menu ID number (ID: 0 ----> MENU):
```

Επιλέγοντας το ID: 1 (Create a new booking) στο Customer menu, εμφανίζονται τα πεδία προς εισαγωγή για να προχωρήσει η νέα κράτηση.

```
Enter the menu ID number (ID: 0 ----> MENU): 1
```

```
-----  
|           NEW BOOKING                     |  
-----
```

```
Start day: 1  
End day: 5  
Month: 4  
Room Type (Single, Double or Suite): suite
```

Έπειτα εμφανίζονται τα διαθέσιμα δωμάτια και επιλέγει ο χειριστής το room id για να ολοκληρωθεί η καταχώρηση.

```
-----  
Room # 14  
-----
```

```
ID: 3013  
Price: 50 €/day
```

```
Give me the room ID you want to book: 14  
New booking record created successfully!
```

Επιλέγοντας το ID: 2 (Show information about your booking(s)) στο Customer menu, εμφανίζεται/ονται η/οι κράτηση/εις του πελάτη.

Enter the menu ID number (ID: 0 ---> MENU): 2

```
-----  
|          BOOKING INFO          |  
-----
```

Your booking # 1

Booking's ID: 4000
Room's ID: 3000
Booking's period: From 11.2.2021 to 15.2.2021
Season: Medium

Your booking # 2

Booking's ID: 4007
Room's ID: 3015
Booking's period: From 1.5.2021 to 10.5.2021
Season: High

Your booking # 3

Booking's ID: 4042
Room's ID: 3022
Booking's period: From 1.10.2021 to 5.10.2021
Season: Low
Enter the menu ID number (ID: 0 ---> MENU): |

Επιλέγοντας το ID: 3 (Pay off your booking account) στο Customer menu, εμφανίζεται/ονται η/οι κράτηση/εις του πελάτη με το συνολικό ποσό πληρωμής.

Enter the menu ID number (ID: 0 ---> MENU): 3

```
-----  
|          PAYMENT INFO          |  
-----
```

Your booking # 1

Booking's ID: 4000
Booking's period: From 11.2.2021 to 15.2.2021
Room's ID: 3000
Total amount: 88 €

Your booking # 2

Booking's ID: 4007
Booking's period: From 1.5.2021 to 10.5.2021
Room's ID: 3015
Total amount: 207 €

Your booking # 3

Booking's ID: 4042
Booking's period: From 1.10.2021 to 5.10.2021
Room's ID: 3022
Total amount: 124 €

You have more than one booking active in our hotel.

Are you sure you want to pay one of your bookings? (Yes or No): |

Με επιλογή "yes", εισάγει ο χρήστης τον αριθμό κράτησης και το ποσό και εμφανίζεται η επιλογή μεθόδου πληρωμής όπου ο πελάτης μπορεί να πληρώσει με μετρητά ή κάρτα.

```
You have more than one booking active in our hotel.  
Are you sure you want to pay one of your bookings? (Yes or No): yes
```

```
Please choose the booking ID you want to pay now: 4000  
Fill the amount of current booking: 88
```

```
-----  
Payment method  
-----  
1. Credit card (-20 € discount).  
2. Cash.  
Customer's choice: 1
```

```
-----  
Thank you so much! See you soon!
```

Με επιλογή "no", επιστρέφει ο χρήστης στο customer menu.

Επιλέγοντας το ID: 4 (Show your personal information) στο Customer menu, εμφανίζονται πληροφορίες σχετικά με τον πελάτη.

```
-----  
|          CUSTOMER INFO          |  
-----
```

```
UID: 20006  
Name: Jim Alan  
Email: jim_alan__22@t-online.de
```

Επιλέγοντας το ID: 5 (Update/change your mail address) στο Customer menu, ο χρήστης μπορεί να αλλάξει το email του πελάτη.

```
-----  
|          CUSTOMER MENU          |  
-----
```

```
Select one operation (type the ID number).  
ID: 1 ---> Create a new booking.  
ID: 2 ---> Show information about your booking(s).  
ID: 3 ---> Pay off your booking account.  
ID: 4 ---> Show your personal information.  
ID: 5 ---> Update/change your email address.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU): 5  
  
Give me your new email: test@gamil.com  
Customer's email updated successfully!
```

Επιλέγοντας το ID:-1 (Exit Program) επιστρέφει στο menu χρήστη.

```
-----  
|          RECEPTIONIST MENU      |  
-----
```

```
Select one operation (type the ID number).  
ID: 1 ---> Welcome a customer to the hotel.  
ID: 2 ---> Show customers list.  
ID: 3 ---> Show bookings list.  
ID: 4 ---> Show rooms list.  
ID: -1 ---> Exit program.  
Enter the menu ID number (ID: 0 ---> MENU):
```

Επιλέγοντας το ID: 2 (Show customer list) στο menu χρήστη, εμφανίζεται η λίστα πελατών που είναι καταχωρημένοι στο σύστημα.

```
-----  
|                CUSTOMERS LIST                |  
-----
```

```
-----  
Customer # 1
```

```
-----  
UID: 20000
```

```
Name: Santiago Perez
```

```
Email: santiagop@yahoo.de
```

```
-----  
Customer # 2
```

```
-----  
UID: 20001
```

```
Name: Jessica Brosman
```

```
Email: jess_bros@hotmail.it
```

Επιλέγοντας το ID: 3 (Show bookings list) στο menu χρήστη, εμφανίζεται η λίστα των κρατήσεων που είναι καταχωρημένες στο σύστημα.

```
-----  
Booking # 35
```

```
-----  
ID: 4056
```

```
Customer's UID: 20053
```

```
Room's ID: 3007
```

```
Booking's period: From 28.9.2021 to 30.9.2021
```

```
Season: High
```

```
-----  
Booking # 36
```

```
-----  
ID: 4057
```

```
Customer's UID: 20008
```

```
Room's ID: 3014
```

```
Booking's period: From 21.4.2021 to 23.4.2021
```

```
Season: Medium
```

Επιλέγοντας το ID: 4 (Show rooms list) στο menu χρήστη, εμφανίζεται η λίστα των δωματίων που είναι καταχωρημένα στο σύστημα.

```
-----  
Room # 50
```

```
-----  
ID: 3049
```

```
Type: Single
```

```
Price: 20 €/day
```

```
-----  
Room # 51
```

```
-----  
ID: 3050
```

```
Type: Double
```

```
Price: 30 €/day
```

Επιλέγοντας το ID: -1 (Exit Program) στο menu χρήστη, τερματίζεται η λειτουργία του συστήματος.

Επιλέγοντας το ID: 5 (Show receptionist list) στο menu χρήστη, εμφανίζεται η λίστα των receptionist που είναι καταχωρημένοι στο σύστημα.

```
-----
|          RECEPTIONISTS LIST          |
-----

-----
Receptionist # 1
-----
ID: 1002
Name: Nick Pepperman
Email: recept1@hotel.com
Username: recept1
Password: recept1

-----
Receptionist # 2
-----
ID: 1003
Name: George Lopez
Email: recept2@hotel.com
Username: recept2
Password: recept2
```

Επιλέγοντας το ID: 6 (Show data about hotel) στο menu χρήστη, εμφανίζεται τα διαθέσιμα reports που μπορεί να δει ο manager.

Enter the menu ID number (ID: 0 ---> MENU): 6

```
-----
|          HOTEL DATA          |
-----

Select an option (type the ID number).
ID: 1 ---> Show financial data about year 2020.
ID: 2 ---> Show financial data about year 2021 (so far).
ID: 3 ---> Show statistical data about year 2020.
ID: 4 ---> Show statistical data about year 2021 (so far).
ID: 5 ---> Back to manager's basic menu.
Enter the data menu ID number (ID: 0 --> DATA MENU): |
```

Επιλέγοντας το ID: 1 (Show financial data about year 2020) στο menu Hotel Data, εμφανίζονται οι παρακάτω πληροφορίες:

```
Total amount for 2020: 4433 €
October's 2020 total amount: 632 €
November's 2020 total amount: 2280 €
December's 2020 total amount: 1521 €
Enter the data menu ID number (ID: 0 --> DATA MENU):
```

Επιλέγοντας το ID: 2 (Show financial data about year 2021 (so far)) στο menu Hotel Data, εμφανίζονται οι παρακάτω πληροφορίες:

```
Total amount for 2021: 68 €
January's 2021 total amount: 0 €
Enter the data menu ID number (ID: 0 --> DATA MENU):
```

Επιλέγοντας το ID: 3 (Show statistical data about year 2020) στο menu Hotel Data, εμφανίζονται οι παρακάτω πληροφορίες:

```
-----  
|          YEAR 2020 STATS          |  
-----
```

```
Total paid bookings in 2020: 22  
Total active bookings in 2020: 0  
Enter the data menu ID number (ID: 0 --> DATA MENU):
```

Επιλέγοντας το ID: 4 (Show statistical data about year 2021 (so far)) στο menu Hotel Data, εμφανίζονται οι παρακάτω πληροφορίες

```
-----  
|          YEAR 2021 STATS          |  
-----
```

```
Total paid bookings in 2021: 18  
Total active bookings in 2021: 22  
Enter the data menu ID number (ID: 0 --> DATA MENU):
```

Εισάγοντας ID: 5 ο χρήστης επιστρέφει στο menu χρήστη.