



UNIVERSITY OF  
PATRAS  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

# Class Diagram v1.0

Τεχνολογία Λογισμικού



Στοιχεία Φοιτητών			
Επώνυμο	Όνομα	Αριθμός Μητρώου	Έτος
Θώδη	Μαριάνθη	1084576	4ο
Καγιούλη	Αλεξάνδρα	1084546	4ο
Μισαηλίδου	Βασιλική	1072505	5ο
Νασιέλη	Βασιλική	1090034	4ο

Απρίλιος 2024

# Contents

<b>1</b>	<b>Εισαγωγή</b>	<b>3</b>
<b>2</b>	<b>Σημειώσεις για το Class Diagram</b>	<b>3</b>
<b>3</b>	<b>Class Diagram v1.0</b>	<b>4</b>
<b>4</b>	<b>Ανάλυση</b>	<b>4</b>
4.1	Κλάση User . . . . .	4
4.2	Κλάση BusinessPartner . . . . .	5
4.3	Κλάση Admin . . . . .	5
4.4	Κλάση Beneficiary . . . . .	6
4.5	Κλάση SimpleUser . . . . .	6
4.6	Κλάση ServiceProvider . . . . .	7
4.7	Κλάση Membership . . . . .	7
4.8	Κλάση Card . . . . .	8
4.9	Κλάση Points . . . . .	8
4.10	Κλάση PointsHistory . . . . .	9
4.11	Κλάση Country . . . . .	9
4.12	Κλάση City . . . . .	10
4.13	Κλάση Business . . . . .	10
4.14	Κλάση Market . . . . .	10
4.15	Κλάση FoodAndBeverage . . . . .	11
4.16	Κλάση Hotels . . . . .	11
4.17	Κλάση Bars . . . . .	12
4.18	Κλάση Service . . . . .	12
4.19	Κλάση Accommodation . . . . .	13
4.20	Κλάση Car . . . . .	13
4.21	Κλάση Activity . . . . .	13
4.22	Κλάση Booking . . . . .	14
4.23	Κλάση ReviewApp . . . . .	14
4.24	Κλάση DestinationGui . . . . .	15
4.25	Κλάση FriendRequestGUI . . . . .	15
4.26	Κλάση FriendsApp . . . . .	16
4.27	Κλάση SocialBondingGUI . . . . .	16
4.28	Κλάση ChattingGUI . . . . .	17
<b>5</b>	<b>ER Diagram</b>	<b>18</b>

# 1 Εισαγωγή

Τα διαγράμματα κλάσεων είναι ένα εργαλείο στον αντικειμενοστραφή προγραμματισμό. Χρησιμοποιούνται για να δείξουν τη δομή ενός συστήματος, παρουσιάζοντας τις κλάσεις, τα χαρακτηριστικά και τις μεθόδους τους, καθώς και τις σχέσεις μεταξύ τους. Κάθε κλάση εμφανίζεται ως ένα κουτί που χωρίζεται σε τρία μέρη: το όνομα της κλάσης, τα χαρακτηριστικά της (όπως οι ιδιότητες) και τις μεθόδους της (όπως οι λειτουργίες). Οι σχέσεις μεταξύ των κλάσεων δείχνονται με γραμμές που τις συνδέουν.

Σε αυτό το έγγραφο, παρουσιάζουμε το αρχικό διάγραμμα κλάσεων για το σύστημά μας, καθώς και την περιγραφή των βασικών οντοτήτων και των σχέσεών τους. Αυτό το διάγραμμα θα μας βοηθήσει να προχωρήσουμε στον σχεδιασμό και την υλοποίηση του συστήματος.

## 2 Σημειώσεις για το Class Diagram

Στην συγκεκριμένη φάση ανάπτυξης δεν έχουν υλοποιηθεί όλες οι οντότητες ως κλάσεις υπό μορφή κώδικα.

Η ύπαρξη ορισμένων συναρτήσεων μετάβασης, θεωρείται αμελητέα (απλή πλοήγηση/εναλλαγή ανάμεσα στα παράθυρα) και θα αφαιρεθούν στην τελική έκδοση, καθώς πρόκειται για ενέργειες οι οποίες υλοποιούνται με αμεσότερο τρόπο, μέσω του συστήματος και του τρόπου γραφής του κώδικα.

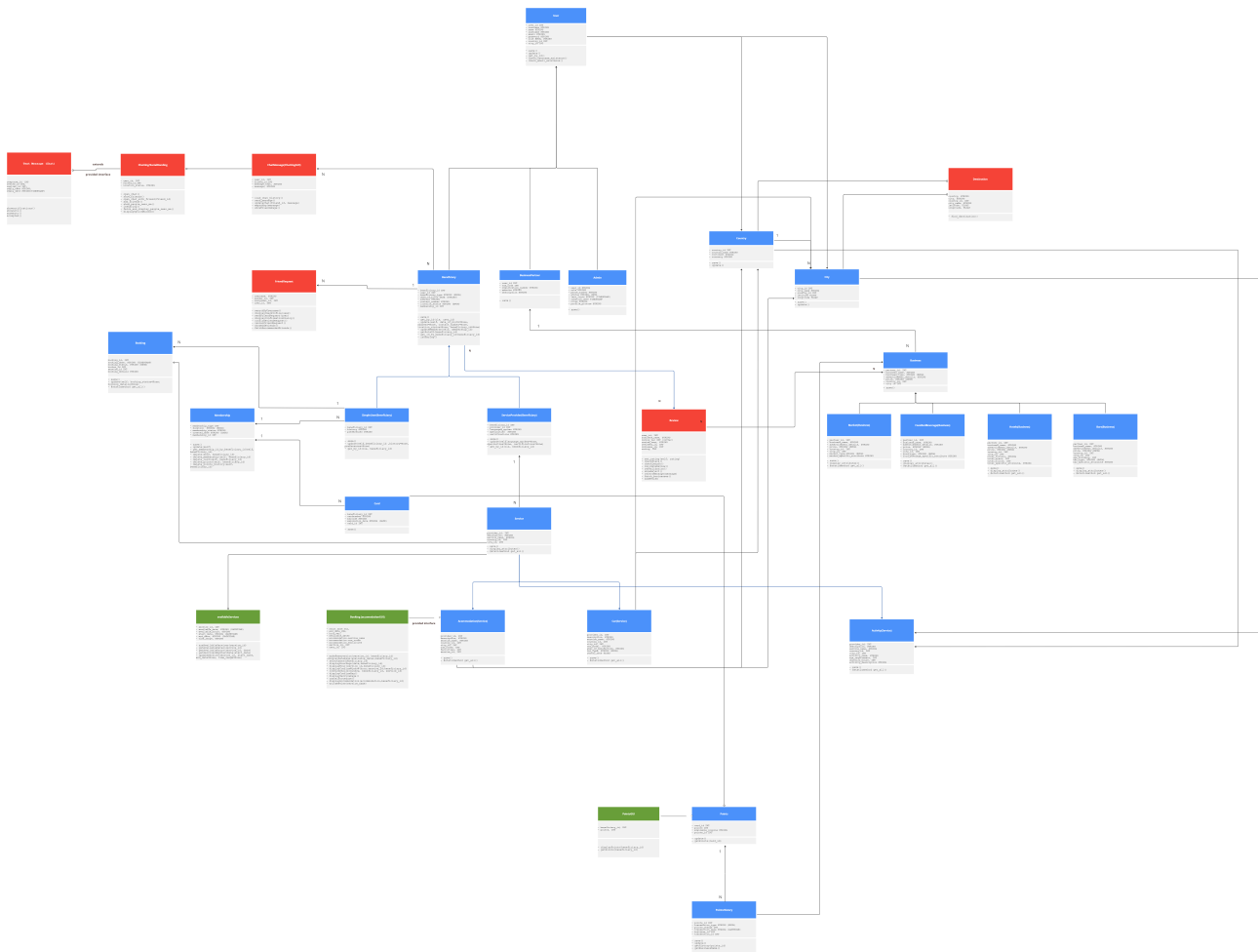
Δεν λαμβάνονται υπόψη τα στοιχεία του tkinter τα οποία χρησιμοποιούνται για την αναπαράσταση των GUI, κατά την υλοποίηση των κλάσεων.

### **Κλάση FriendRequest:**

Οι παρακάτω συναρτήσεις ενώ έχουν ενσωματωθεί στο αντίστοιχο sequence diagram, δεν έχουν υλοποιηθεί σε λειτουργική μορφή στον κώδικα. Πρόκειται για συναρτήσεις οι οποίες αναπαριστώνται στον ήδη υλοποιημένο κώδικα, έμμεσα, ως παράμετροι.

```
clickAddFriends()  
TypeUsername()  
rejectSearch()  
rejectSearchMes()  
ClickRecFriend()  
updateFrReq()
```

### 3 Class Diagram v1.0



## 4 Ανάλυση

### 4.1 Κλάση User

Η κλάση User αναπαριστά έναν χρήστη του συστήματος με διάφορες ιδιότητες. Περιέχει μεθόδους για την αποθήκευση, ενημέρωση και ανάκτηση των δεδομένων ενός χρήστη.

- **\_\_init\_\_(self, user\_id=None, username=None, name=None, lastname=None, email=None, password=None, role=None, country\_id=None, city\_id=None):**
  - Αρχικοποιεί τα χαρακτηριστικά του χρήστη.
- **save(self):**
  - Αποθηκεύει τον χρήστη στη βάση δεδομένων. Αν ο user\_id είναι None, προσθέτει νέο χρήστη, αλλιώς εκτυπώνει μήνυμα για χρήση της μεθόδου update.
- **update(self, username=None, password=None):**

- Ενημερώνει το `username` και το `password` του χρήστη στη βάση δεδομένων.
- `get_by_id(cls, user_id)`:
  - Επιστρέφει έναν χρήστη με βάση το `user_id`.
- `check_username_existence(username)`:
  - Ελέγχει αν υπάρχει το `username` στη βάση δεδομένων.
- `check_email_existence(email)`:
  - Ελέγχει αν υπάρχει το `email` στη βάση δεδομένων.

## 4.2 Κλάση `BusinessPartner`

Η κλάση `BusinessPartner` επεκτείνει την κλάση `User` και αναπαριστά έναν συνεργάτη με επιπλέον χαρακτηριστικά.

- `__init__(self, user_id=None, tax_code=None, registration_number=None, website=None, description=None)`:
  - Αρχικοποιεί τα χαρακτηριστικά του επιχειρηματικού συνεργάτη.
- `save(self)`:
  - Αποθηκεύει τον συνεργάτη στη βάση δεδομένων. Πρώτα καλεί τη μέθοδο `save` της κλάσης `User` και μετά αποθηκεύει τα επιπλέον δεδομένα του `BusinessPartner`.

## 4.3 Κλάση `Admin`

Η κλάση `Admin` επεκτείνει την κλάση `User` και αναπαριστά έναν διαχειριστή με επιπλέον χαρακτηριστικά.

- `__init__(self, user_id=None, role=None, phone_number=None, status='active', last_login=None, creation_date=None, notes=None, profile_picture=None)`:
  - Αρχικοποιεί τα χαρακτηριστικά του διαχειριστή.
- `save(self)`:
  - Αποθηκεύει τον διαχειριστή στη βάση δεδομένων. Πρώτα καλεί τη μέθοδο `save` της κλάσης `User` και μετά αποθηκεύει τα επιπλέον δεδομένα του `Admin`.

## 4.4 Κλάση Beneficiary

Η κλάση Beneficiary αναπαριστά έναν ωφελούμενο με διάφορες ιδιότητες και περιέχει μεθόδους για την αποθήκευση, ενημέρωση και ανάκτηση των δεδομένων ενός ωφελούμενου.

- `__init__(self, beneficiary_id=None, user_id=None, beneficiary_type=None, date_of_birth=None, address=None, contact_number=None, location_status='Inactive', membership_id=None):`
  - Αρχικοποιεί τα χαρακτηριστικά του ωφελούμενου.
- `save(self):`
  - Αποθηκεύει τον ωφελούμενο στη βάση δεδομένων.
- `get_by_id(cls, user_id):`
  - Επιστρέφει έναν ωφελούμενο με βάση το `user_id`.
- `update(self, date_of_birth=None, address=None, contact_number=None, location_status=None, beneficiary_id=None):`
  - Ενημερώνει τα χαρακτηριστικά του ωφελούμενου στη βάση δεδομένων.
- `updateMemberID(self, membership_id):`
  - Ενημερώνει το `membership_id` του ωφελούμενου στη βάση δεδομένων.
- `getRoleID(beneficiary_id):`
  - Επιστρέφει τον τύπο του ωφελούμενου (`beneficiary_type`) με βάση το `beneficiary_id`.
- `get_id_by_beneficiary_id(beneficiary_id):`
  - Επιστρέφει το `user_id` με βάση το `beneficiary_id`.
- `isSharing(self):`
  - Ελέγχει αν ο ωφελούμενος μοιράζεται την τοποθεσία του (το `location_status` είναι 'Active').

## 4.5 Κλάση SimpleUser

Η κλάση SimpleUser επεκτείνει την κλάση Beneficiary και αναπαριστά έναν απλό χρήστη με επιπλέον χαρακτηριστικά.

- `__init__(self, beneficiary_id=None, bistory=None, preferences=None):`
  - Αρχικοποιεί τα χαρακτηριστικά του απλού χρήστη.
- `save(self):`
  - Αποθηκεύει τον απλό χρήστη στη βάση δεδομένων.

- **update(self, beneficiary\_id, bistory=None, preferences=None):**
  - Ενημερώνει τα χαρακτηριστικά του απλού χρήστη στη βάση δεδομένων.
- **get\_by\_id(cls, beneficiary\_id):**
  - Επιστρέφει έναν απλό χρήστη με βάση το beneficiary\_id.

## 4.6 Κλάση ServiceProvider

Η κλάση ServiceProvider επεκτείνει την κλάση Beneficiary και αναπαριστά έναν πάροχο υπηρεσιών με επιπλέον χαρακτηριστικά.

- **\_\_init\_\_(self, beneficiary\_id=None, provider\_id=None, languages\_spoken=None, specialties=None, certifications=None):**
  - Αρχικοποιεί τα χαρακτηριστικά του παρόχου υπηρεσιών.
- **save(self):**
  - Αποθηκεύει τον πάροχο υπηρεσιών στη βάση δεδομένων.
- **update(self, language\_spoken=None, specialties=None, certifications=None):**
  - Ενημερώνει τα χαρακτηριστικά του παρόχου υπηρεσιών στη βάση δεδομένων.
- **get\_by\_id(cls, beneficiary\_id):**
  - Επιστρέφει έναν πάροχο υπηρεσιών με βάση το beneficiary\_id.

## 4.7 Κλάση Membership

Η κλάση Membership αναπαριστά μια ιδιότητα μέλους με διάφορες ιδιότητες και περιέχει μεθόδους για την αποθήκευση, ενημέρωση και διαγραφή των δεδομένων μιας ιδιότητας μέλους.

- **\_\_init\_\_(self, membership\_type=None, duration=None, membership\_status=None, created\_date=None, membership\_id=None):**
  - Αρχικοποιεί τα χαρακτηριστικά της ιδιότητας μέλους.
- **save(self):**
  - Αποθηκεύει την ιδιότητα μέλους στη βάση δεδομένων.
- **update(self):**
  - Ενημερώνει τα χαρακτηριστικά της ιδιότητας μέλους στη βάση δεδομένων.
- **get\_membership\_id\_by\_beneficiary\_id(self, beneficiary\_id):**
  - Επιστρέφει το membership\_id με βάση το beneficiary\_id.
- **delete(self, beneficiary\_id):**

- Διαγράφει την ιδιότητα μέλους και τα συνδεδεμένα δεδομένα από τη βάση δεδομένων.
- **delete\_membership(self, beneficiary\_id):**
  - Διαγράφει την ιδιότητα μέλους από τη βάση δεδομένων.
- **delete\_card(self, beneficiary\_id):**
  - Διαγράφει την κάρτα από τη βάση δεδομένων.
- **delete\_points(self, beneficiary\_id):**
  - Διαγράφει τους πόντους από τη βάση δεδομένων.
- **delete\_points\_history(self, beneficiary\_id):**
  - Διαγράφει το ιστορικό πόντων από τη βάση δεδομένων.

## 4.8 Κλάση Card

Η κλάση Card αντιπροσωπεύει μια κάρτα μέλους με τις πληροφορίες της και περιλαμβάνει μια μέθοδο για την αποθήκευσή της στη βάση δεδομένων.

- **\_\_init\_\_(self, beneficiary\_id=None, cardnumber=None, barcode=None, expiration\_date=None, card\_id=None):**
  - Αρχικοποιεί τα χαρακτηριστικά της κάρτας.
- **save(self):**
  - Αποθηκεύει την κάρτα στη βάση δεδομένων.

## 4.9 Κλάση Points

Η κλάση Points αντιπροσωπεύει τους πόντους που συνδέονται με μια κάρτα μέλους και περιλαμβάνει μεθόδους για την αποθήκευση, ενημέρωση και λήψη πόντων από τη βάση δεδομένων.

- **\_\_init\_\_(self, card\_id=None, points=None, available\_coupons=None, points\_id=None):**
  - Αρχικοποιεί τα χαρακτηριστικά των πόντων.
- **save(self):**
  - Αποθηκεύει τους πόντους στη βάση δεδομένων.
- **update(self):**
  - Ενημερώνει τους πόντους στη βάση δεδομένων.
- **getPoints(card\_id):**
  - Επιστρέφει τους πόντους για μια συγκεκριμένη κάρτα.



## 4.10 Κλάση PointsHistory

Η κλάση PointsHistory καταγράφει το ιστορικό των πόντων για μια κάρτα μέλους. Περιλαμβάνει μεθόδους για την αποθήκευση και την ενημέρωση του ιστορικού πόντων.

- `__init__(self, points_id=None, transaction_type=None, points_change=None, transaction_date=None, business_id=None, transaction_id=None)`:
  - Αρχικοποιεί τα χαρακτηριστικά του ιστορικού των πόντων.
- `save(self)`:
  - Αποθηκεύει το ιστορικό των πόντων στη βάση δεδομένων.
- `update(self)`:
  - Ενημερώνει το ιστορικό των πόντων στη βάση δεδομένων.
- `getHistory(points_id)`:
  - Επιστρέφει το ιστορικό των πόντων για μια συγκεκριμένη κάρτα, ταξινομημένο κατά ημερομηνία.
- `getBusinessName(self)`:
  - Επιστρέφει το όνομα της επιχείρησης συνδεδεμένο με την εγγραφή ιστορικού πόντων.

## 4.11 Κλάση Country

Η κλάση Country αντιπροσωπεύει μια χώρα και περιλαμβάνει πληροφορίες όπως το όνομα της χώρας, την ηπειρωτική της θέση και το νόμισμά της.

- `__init__(self, country_id=None, country_name=None, continent=None, currency=None)`:
  - Αρχικοποιεί τα χαρακτηριστικά της χώρας.
- `save(self)`:
  - Αποθηκεύει την χώρα στη βάση δεδομένων.
- `update(self)`:
  - Ενημερώνει τις πληροφορίες της χώρας στη βάση δεδομένων.

## 4.12 Κλάση City

Η κλάση City αντιπροσωπεύει μια πόλη και περιλαμβάνει πληροφορίες όπως το όνομά της, το αναγνωριστικό της χώρας όπου βρίσκεται, καθώς και τις γεωγραφικές της συντεταγμένες.

- `__init__(self, city_id=None, city_name=None, country_id=None, latitude=None, longitude=None)`:
  - Αρχικοποιεί τα χαρακτηριστικά της πόλης.
- `save(self)`:
  - Αποθηκεύει την πόλη στη βάση δεδομένων.
- `update(self)`:
  - Ενημερώνει τις πληροφορίες της πόλης στη βάση δεδομένων.

## 4.13 Κλάση Business

Η κλάση Business αντιπροσωπεύει μια επιχείρηση και περιλαμβάνει πληροφορίες όπως το όνομα της επιχείρησης, τον τύπο της επιχείρησης, λεπτομέρειες διαφήμισης, την τιμή της υπηρεσίας, καθώς και τα αναγνωριστικά της χώρας και της πόλης όπου βρίσκεται.

- `__init__(self, partner_id, business_name, business_type, advertisement_details, price, country_id, city_id)`:
  - Αρχικοποιεί τα χαρακτηριστικά της επιχείρησης.
- `save(self)`:
  - Αποθηκεύει την επιχείρηση στη βάση δεδομένων.

## 4.14 Κλάση Market

Η κλάση Market είναι μια υποκλάση της κλάσης Business και αντιπροσωπεύει μία επιχείρηση-αγορά. Περιλαμβάνει επιπλέον πληροφορίες όπως τον τύπο της επιχείρησης και τυχόν συγκεκριμένα χαρακτηριστικά της.

- `__init__(self, partner_id, business_name, advertisement_details, price, country_id, city_id, market_type, market_specific_attribute)`:
  - Καλεί τον constructor της κλάσης Business με τα κατάλληλα ορίσματα και προσθέτει τον τύπο της επιχείρησης-αγοράς και τυχόν συγκεκριμένα χαρακτηριστικά.
- `save(self)`:
  - Αποθηκεύει την επιχείρηση-αγορά στη βάση δεδομένων.
- `display_attributes(self)`:

- Επιστρέφει μια αλφαριθμητική αναπαράσταση των επιπρόσθετων χαρακτηριστικών της επιχείρησης-αγοράς
- **get\_all():**
  - Επιστρέφει μια λίστα με όλες τις επιχειρήσεις-αγορές που βρίσκονται στη βάση δεδομένων.

## 4.15 Κλάση FoodAndBeverage

Η κλάση FoodAndBeverage είναι μια υποκλάση της κλάσης Business και αντιπροσωπεύει μια επιχείρηση εστίασης. Περιλαμβάνει επιπλέον πληροφορίες όπως τον τύπο φαγητού που προσφέρεται και τυχόν συγκεκριμένα χαρακτηριστικά της επιχείρησης εστίασης.

- **\_\_init\_\_(self, partner\_id, business\_name, advertisement\_details, price, country\_id, city\_id, food\_type, food\_beverage\_specific\_attribute):**
  - Καλεί τον constructor της κλάσης Business με τα κατάλληλα ορίσματα και προσθέτει τον τύπο φαγητού και τυχόν συγκεκριμένα χαρακτηριστικά της επιχείρησης εστίασης.
- **save(self):**
  - Αποθηκεύει την επιχείρηση εστίασης στη βάση δεδομένων.
- **display\_attributes(self):**
  - Επιστρέφει μια αλφαριθμητική αναπαράσταση των επιπρόσθετων χαρακτηριστικών της επιχείρησης εστίασης.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλες τις επιχειρήσεις εστίασης που βρίσκονται στη βάση δεδομένων.

## 4.16 Κλάση Hotels

Η κλάση Hotels είναι μια υποκλάση της κλάσης Business και αντιπροσωπεύει ένα ξενοδοχείο. Περιλαμβάνει επιπλέον πληροφορίες όπως τα φίλτρα ξενοδοχείου, τα αστέρια του ξενοδοχείου, τους ορόφους του ξενοδοχείου και τυχόν συγκεκριμένα χαρακτηριστικά του ξενοδοχείου.

- **\_\_init\_\_(self, partner\_id, business\_name, advertisement\_details, price, country\_id, city\_id, hotel\_filters, hotel\_stars, hotel\_floors, hotel\_specific\_attribute):**
  - Καλεί τον constructor της κλάσης Business με τα κατάλληλα ορίσματα και προσθέτει τα φίλτρα του ξενοδοχείου, τα αστέρια του ξενοδοχείου, τους ορόφους του ξενοδοχείου και τυχόν συγκεκριμένα χαρακτηριστικά του ξενοδοχείου.
- **save(self):**
  - Αποθηκεύει το ξενοδοχείο στη βάση δεδομένων.
- **display\_attributes(self):**

- Επιστρέφει μια αλφαριθμητική αναπαράσταση των επιπρόσθετων χαρακτηριστικών του ξενοδοχείου.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλα τα ξενοδοχεία που βρίσκονται στη βάση δεδομένων.

## 4.17 Κλάση Bars

Η κλάση Bars είναι μια υποκλάση της κλάσης Business και αντιπροσωπεύει ένα μπαρ. Περιλαμβάνει επιπλέον πληροφορίες όπως το όριο ηλικίας, τον τύπο του μπαρ και τυχόν συγκεκριμένα χαρακτηριστικά του μπαρ.

- **\_\_init\_\_(self, partner\_id, business\_name, advertisement\_details, price, country\_id, city\_id, age\_boundary, bar\_type, bar\_specific\_attribute):**
  - Καλεί τον constructor της κλάσης Business με τα κατάλληλα ορίσματα και προσθέτει το όριο ηλικίας, τον τύπο του μπαρ και τυχόν συγκεκριμένα χαρακτηριστικά του μπαρ.
- **save(self):**
  - Αποθηκεύει το μπαρ στη βάση δεδομένων.
- **display\_attributes(self):**
  - Επιστρέφει μια αλφαριθμητική αναπαράσταση των επιπρόσθετων χαρακτηριστικών του μπαρ.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλα τα μπαρ που βρίσκονται στη βάση δεδομένων.

## 4.18 Κλάση Service

Η κλάση Service αντιπροσωπεύει ένα υπηρεσιών που προσφέρεται από έναν πάροχο. Περιλαμβάνει πληροφορίες όπως τον αναγνωριστικό του παρόχου, την περιγραφή της υπηρεσίας, το όνομα της υπηρεσίας, καθώς και το αναγνωριστικό της χώρας και της πόλης.

- **\_\_init\_\_(self, provider\_id, description, service\_name, country\_id, city\_id):**
  - Δημιουργεί ένα αντικείμενο υπηρεσίας με τις κατάλληλες πληροφορίες.
- **save(self):**
  - Αποθηκεύει την υπηρεσία στη βάση δεδομένων.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλες τις υπηρεσίες που υπάρχουν στη βάση δεδομένων.

## 4.19 Κλάση Accommodation

Η κλάση Accommodation αντιπροσωπεύει μια διαθέσιμη διαμονή. Περιλαμβάνει πληροφορίες όπως τον αριθμό των δωματίων, τις εγκαταστάσεις και το αναγνωριστικό της υπηρεσίας.

- **\_\_init\_\_(self, provider\_id, description, service\_name, country\_id, city\_id, num\_rooms, facilities, service\_id):**
  - Δημιουργεί ένα αντικείμενο διαμονής με τις κατάλληλες πληροφορίες.
- **save(self):**
  - Αποθηκεύει τη διαμονή στη βάση δεδομένων.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλες τις διαθέσιμες διαμονές από τη βάση δεδομένων.

## 4.20 Κλάση Car

Η κλάση Car αντιπροσωπεύει ένα διαθέσιμο αυτοκίνητο προς ενοικίαση. Περιλαμβάνει πληροφορίες όπως το μοντέλο, το έτος κατασκευής, τον τύπο και την τιμή ενοικίασης.

- **\_\_init\_\_(self, provider\_id, description, service\_name, country\_id, city\_id, car\_model, year\_of\_manufacture, car\_type, rental\_rate):**
  - Δημιουργεί ένα αντικείμενο αυτοκινήτου με τις κατάλληλες πληροφορίες.
- **save(self):**
  - Αποθηκεύει το αυτοκίνητο στη βάση δεδομένων.
- **get\_all():**
  - Επιστρέφει μια λίστα με όλα τα διαθέσιμα αυτοκίνητα προς ενοικίαση από τη βάση δεδομένων.

## 4.21 Κλάση Activity

The ‘Activity’ class represents an available activity. It includes information such as the activity name, age requirement, duration in hours, and activity description.

- **\_\_init\_\_(self, provider\_id, description, service\_name, country\_id, city\_id, activity\_name, age\_requirement, duration\_hours, activity\_description):**
  - Creates an activity object with the appropriate information.
- **save(self):**
  - Saves the activity to the database.
- **get\_all():**
  - Returns a list of all available activities from the database.

## 4.22 Κλάση Booking

The ‘Booking’ class represents a booking made by a user for a particular service. It includes information such as the booking date, booking status, booker ID, service ID, and booking details.

- **\_\_init\_\_(self, booking\_id=None, booking\_date=None, booking\_status=None, booker\_id=None, service\_id=None, booking\_details=None):**
  - Creates a booking object with the specified information. If no booking status is provided, it defaults to ‘pending’.
- **save(self):**
  - Saves the booking information to the database.
- **update(self, booking\_status=None, booking\_details=None):**
  - Updates the booking status and/or details in the database.
- **get\_all():**
  - Returns a list of all bookings from the database.

## 4.23 Κλάση ReviewApp

Η κλάση ‘ReviewApp’ είναι υπεύθυνη για τη διαχείριση του συστήματος δημοσίευσης αξιολογήσεων. Επιτρέπει στους χρήστες να υποβάλλουν αξιολογήσεις για επιχειρήσεις.

- **\_\_init\_\_(self, root, user\_id, business\_name):**
  - Αρχικοποιεί το αντικείμενο ‘ReviewApp’ με τις παρεχόμενες παραμέτρους.
  - Διαμορφώνει τον τίτλο του παραθύρου, το μέγεθος και διάφορα στοιχεία GUI για την καταγραφή πληροφοριών αξιολόγησης.
- **set\_rating(self, rating):**
  - Ορίζει τη βαθμολογία για την αξιολόγηση με βάση την είσοδο του χρήστη.
- **checkStars(self):**
  - Ελέγχει αν ο χρήστης έχει παρέχει βαθμολογία για την υπηρεσία.
- **checkLength(self):**
  - Ελέγχει αν το μήκος του κειμένου της αξιολόγησης υπερβαίνει τους 100 χαρακτήρες.
- **validateReview(self):**
  - Επικυρώνει την αξιολόγηση ελέγχοντας τόσο τη βαθμολογία όσο και το μήκος του κειμένου της αξιολόγησης.
- **refValidation(self):**

- Εμφανίζει ένα μήνυμα επιτυχίας μετά τη δημιουργία μιας αξιολόγησης με επιτυχία.
- **showReject(self):**
  - Εμφανίζει ένα μήνυμα σφάλματος αν υπάρξει αποτυχία στη δημιουργία μιας αξιολόγησης.
- **rejectMessage(self, message):**
  - Εμφανίζει ένα προσαρμοσμένο μήνυμα σφάλματος που λαμβάνεται από το σύστημα.
- **fetch\_\_businesses(self):**
  - Ανακτά τις επιχειρήσεις από τη βάση δεδομένων.
- **addReview(self):**
  - Προσθέτει μια νέα αξιολόγηση για μια επιχείρηση στη βάση δεδομένων.

## 4.24 Κλάση DestinationGui

Η κλάση 'DestinationGui' είναι υπεύθυνη για τη δημιουργία του παραθύρου εύρεσης προορισμού. Επιτρέπει στους χρήστες να εισάγουν χώρα και πόλη.

- **\_\_init\_\_(self, root):**
  - Αρχικοποιεί το αντικείμενο 'DestinationGui' με το παράθυρο του γραφικού περιβάλλοντος.
  - Διαμορφώνει τον τίτλο του παραθύρου, το μέγεθος και διάφορα στοιχεία GUI για την εισαγωγή πληροφοριών χώρας και πόλης.
- **find\_\_destination(self):**
  - Εντοπίζει τον προορισμό βάσει των εισαγμένων χώρας και πόλης, εμφανίζοντας τις συντεταγμένες του προορισμού.

## 4.25 Κλάση FriendRequestGUI

Η κλάση 'FriendRequestGUI' είναι υπεύθυνη για τη δημιουργία του παραθύρου συστήματος αιτημάτων φιλίας. Παρέχει διάφορες λειτουργίες όπως αναζήτηση χρηστών, αποστολή αιτημάτων φιλίας και φόρτωση προτεινόμενων φίλων.

- **\_\_init\_\_(self, root):**
- Αρχικοποιεί το αντικείμενο 'FriendRequestGUI' με το παράθυρο του γραφικού περιβάλλοντος.
- Δημιουργεί τα απαραίτητα στοιχεία GUI για την αναζήτηση, την προβολή και την αποστολή αιτημάτων φιλίας.

**searchByUsername(self):**

- Εκτελεί αναζήτηση χρήστη με βάση το όνομα χρήστη που δόθηκε από τον χρήστη.

**displayUserProfile(self, user):**

- Προβάλλει το προφίλ ενός χρήστη, συμπεριλαμβανομένης της επιλογής για αποστολή αιτήματος φιλίας.

**sendFriendRequest(self, user):**

- Αποστέλλει αίτημα φιλίας στον επιλεγμένο χρήστη.

**displayConfirmationDialog(self):**

- Προβάλλει ένα παράθυρο επιβεβαίωσης για την αποστολή αιτήματος φιλίας.

**confirmFriendRequest(self):**

- Επιβεβαιώνει την αποστολή αιτήματος φιλίας.

**cancelFriendRequest(self):**

- Ακυρώνει την αποστολή αιτήματος φιλίας.

**showAddFriends(self):**

- Προβάλλει τη λίστα των φίλων που έχουν προστεθεί ήδη.

**fetchRecommendedFriends(self):**

- Αναχτά προτεινόμενους φίλους που δεν είναι ήδη φίλοι με τον τρέχοντα χρήστη.

## 4.26 Κλάση FriendsApp

**profile\_show\_friends(self):**

- Εμφανίζει τους φίλους του χρήστη στο προφίλ του.

## 4.27 Κλάση SocialBondingGUI

Η κλάση SocialBondingGUI αναλαμβάνει τη δημιουργία του γραφικού περιβάλλοντος χρήστη (GUI).

- **\_\_init\_\_(self, root):**

- Αρχικοποιεί ένα αντικείμενο SocialBondingGUI με το παράθυρο του γραφικού περιβάλλοντος.
- Δημιουργεί τα απαραίτητα στοιχεία GUI, όπως τα κουμπιά για την προβολή φίλων, την προσθήκη φίλων και την προβολή ατόμων κοντά στον χρήστη.

- **open\_chat(self):**

- Ανοίγει ένα παράθυρο συνομιλίας όταν ο χρήστης κάνει κλικ στο κουμπί "Chat".

- **show\_friends(self):**

- Αναχτά και εμφανίζει τους φίλους του τρέχοντος χρήστη.
- Ανοίγει ένα παράθυρο με τους φίλους και τα ονόματά τους σε κουμπιά.



- **open\_chat\_with\_friend(self, friend\_id):**
  - Ανοίγει ένα παράθυρο συνομιλίας με έναν συγκεκριμένο φίλο.
- **add\_friends(self):**
  - Εμφανίζει ένα μήνυμα που επιβεβαιώνει την προσθήκη φίλων.
- **show\_people\_near\_me(self):**
  - Εμφανίζει ανθρώπους που βρίσκονται κοντά στον χρήστη, εάν ο χρήστης είναι διαθέσιμος για κοινοποίηση τοποθεσίας.
- **isSharing(self):**
  - Ελέγχει αν ο χρήστης κοινοποιεί την τοποθεσία του.
- **fetch\_and\_display\_people\_near\_me(self):**
  - Ανακτά και εμφανίζει τα άτομα που βρίσκονται κοντά στον χρήστη, εάν ο χρήστης κοινοποιεί την τοποθεσία του.

## 4.28 Κλάση ChattingGUI

Η κλάση ChattingGUI είναι υπεύθυνη για τη δημιουργία και τη διαχείριση ενός παραθύρου συνομιλίας. Περιλαμβάνει λειτουργίες όπως η φόρτωση του ιστορικού συνομιλίας, η αποστολή μηνυμάτων και η ενημέρωση του πεδίου εισαγωγής μηνυμάτων.

- **\_\_init\_\_(self, root, friend\_id=None):**
  - Αρχικοποιεί ένα αντικείμενο ChattingGUI με το παράθυρο της συνομιλίας.
  - Ορίζει τον τίτλο και τις διαστάσεις του παραθύρου.
  - Ορίζει το ID του φίλου προς τον οποίο θα πραγματοποιηθεί η συνομιλία.
  - Δημιουργεί τα απαραίτητα στοιχεία GUI για την προβολή του ιστορικού συνομιλίας, την εισαγωγή μηνυμάτων και το κουμπί αποστολής.
  - Αν δοθεί ένας φίλος (friend\_id), φορτώνει το ιστορικό συνομιλίας μεταξύ του τρέχοντος χρήστη και του φίλου αυτού.
- **load\_chat\_history(self):**
  - Φορτώνει το ιστορικό συνομιλίας μεταξύ του τρέχοντος χρήστη και ενός συγκεκριμένου φίλου.
  - Εκτελεί μια ερώτηση στη βάση δεδομένων για την ανάκτηση των μηνυμάτων μεταξύ των δύο χρηστών και τα ταξινομεί κατά χρονολογική σειρά.
- **send\_message(self):**
  - Αποστέλλει ένα μήνυμα που πληκτρολογεί ο χρήστης.
  - Ελέγχει αν το μήνυμα είναι κενό ή υπερβαίνει το μέγιστο μήκος.
  - Αποθηκεύει το μήνυμα στη βάση δεδομένων και ενημερώνει το ιστορικό συνομιλίας για να εμφανίσει το νέο μήνυμα.

## 5 ER Diagram

Κώδικα για την βάση δεδομένων μας μπορείτε να βρείτε στην σελίδα *github* του προτζεκτ:  
<https://github.com/palpatinedude/Travel-System/tree/main>

