



# Τμήμα Μηχανικών Η/Υ & Πληροφορικής

Πανεπιστήμιο Πατρών

Τεχνολογία Λογισμικού

## Class-Diagram-v1.0

ΣΥΝΘΕΣΗ ΟΜΑΔΑΣ

ΟΝΟΜΑ	ΑΜ	ΕΤΟΣ	ΡΟΛΟΣ
Αραβαντινός-Ζαφείρης Νικολαος	1059598	7ο	Manager
Θεοδοσίου Ανδρέας	1069889	6ο	Editor
Χαράλαμπος Πάρχας	1070935	6ο	Contributor
Βέρρας Φιλοποίμην	1067547	6ο	Contributor



Bill Guru

Η πλήρης λύση για την Διαχείριση Πολυκατοικιών!

### Σύντομη Περιγραφή Κλάσεων

Η κλάση `AnnouncementsDaoImpl` υλοποιεί το interface `AnnouncementsDao` και παρέχει λειτουργίες για τη διαχείριση των ανακοινώσεων στη βάση δεδομένων. Χρησιμοποιεί το `EntityManager` για την αλληλεπίδραση με την βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:

1. Ανάκτηση Ανακοίνωσης με ID: Μέθοδος `getAnnouncementById(long id)` που επιστρέφει μία ανακοίνωση βάσει του ID της.
2. Ανάκτηση Όλων των Ανακοινώσεων: Μέθοδος `getAllAnnouncements()` που επιστρέφει όλες τις ανακοινώσεις.
3. Ανάκτηση Ανακοινώσεων με Ημερομηνία: Μέθοδος `getAnnouncementByDate(Date date)` που επιστρέφει ανακοινώσεις βάσει της ημερομηνίας.
4. Ανάκτηση Ανακοινώσεων με Τίτλο: Μέθοδος `getAnnouncementsByTitle(String title)` που επιστρέφει ανακοινώσεις βάσει του τίτλου τους.
5. Ανάκτηση Ανακοινώσεων με ID Διαχειριστή: Μέθοδος `getAnnouncementsByAdminId(Admins admin)` που επιστρέφει ανακοινώσεις βάσει του διαχειριστή.
6. Αποθήκευση Ανακοίνωσης: Μέθοδος `saveOrder(Announcements announcement)` που αποθηκεύει μία νέα ανακοίνωση.
7. Ενημέρωση Ανακοίνωσης: Μέθοδος `updateOrder(Announcements announcement)` που ενημερώνει μία υπάρχουσα ανακοίνωση.
8. Διαγραφή Ανακοίνωσης: Μέθοδος `deleteOrder(Announcements announcement)` που διαγράφει μία ανακοίνωση.

**Η κλάση BaseDaoImpl υλοποιεί το interface BaseDao και παρέχει γενικές λειτουργίες για τη διαχείριση αντικειμένων στη βάση δεδομένων. Χρησιμοποιεί το EntityManager για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Αποθήκευση Αντικειμένου: Μέθοδος saveObject(Object object) που αποθηκεύει ή ενημερώνει ένα αντικείμενο στη βάση δεδομένων.
2. Διαγραφή Αντικειμένου: Μέθοδος deleteObject(Object object) που διαγράφει ένα αντικείμενο από τη βάση δεδομένων.
3. Αποθήκευση Όλων των Αντικειμένων: Μέθοδος saveAll(Collection collection) που αποθηκεύει ή ενημερώνει όλα τα αντικείμενα μίας συλλογής στη βάση δεδομένων.
4. Ανάκτηση Αντικειμένου με ID: Μέθοδος getObject(Class clazz, Serializable id) που επιστρέφει ένα αντικείμενο βάσει της κλάσης και του ID του.
5. Διαγραφή Αντικειμένου με ID: Μέθοδος removeObject(Class clazz, Serializable id) που διαγράφει ένα αντικείμενο από τη βάση δεδομένων βάσει της κλάσης και του ID του.

**Η κλάση BillsDaoImpl υλοποιεί το interface BillsDao και παρέχει λειτουργίες για τη διαχείριση των λογαριασμών στη βάση δεδομένων. Χρησιμοποιεί το EntityManager για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Ανάκτηση Λογαριασμού με ID: Μέθοδος getBillsById(long id) που επιστρέφει έναν λογαριασμό βάσει του ID του.
2. Ανάκτηση Όλων των Λογαριασμών: Μέθοδος getAllBills() που επιστρέφει όλους τους λογαριασμούς.
3. Ανάκτηση Λογαριασμών με Κατάσταση: Μέθοδος getBillsByStatus(String status) που επιστρέφει λογαριασμούς βάσει της κατάστασής τους.
4. Ανάκτηση Λογαριασμών με Ημερομηνία Λήξης: Μέθοδος getBillsByDueDate(Date dueDate) που επιστρέφει λογαριασμούς βάσει

**Η κλάση OffersDaoImpl υλοποιεί το interface OffersDao και παρέχει λειτουργίες για τη διαχείριση των προσφορών στη βάση δεδομένων. Χρησιμοποιεί το EntityManager για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Ανάκτηση Προσφοράς με ID: Μέθοδος `getOfferById(long id)` που επιστρέφει μία προσφορά βάσει του ID της.
2. Ανάκτηση Όλων των Προσφορών: Μέθοδος `getAllOffers()` που επιστρέφει όλες τις προσφορές.
3. Ανάκτηση Προσφορών με ID Διαχειριστή: Μέθοδος `getOfferByAdminId(Admins adminId)` που επιστρέφει προσφορές βάσει του ID του διαχειριστή.
4. Ανάκτηση Προσφορών με Έκπτωση: Μέθοδος `getOffersByDiscount(Double discount)` που επιστρέφει προσφορές βάσει του ποσοστού έκπτωσης.
5. Ανάκτηση Προσφορών με ID Χρήστη Premium: Μέθοδος `getOffersByPremUserId(PremiumUsers premUserId)` που επιστρέφει προσφορές βάσει του ID του χρήστη Premium.
6. Αποθήκευση Προσφοράς: Μέθοδος `saveOrder(Offers offer)` που αποθηκεύει μία νέα προσφορά.
7. Ενημέρωση Προσφοράς: Μέθοδος `updateOrder(Offers offer)` που ενημερώνει μία υπάρχουσα προσφορά.
8. Διαγραφή Προσφοράς: Μέθοδος `deleteOrder(Offers offer)` που διαγράφει μία προσφορά.

**Η κλάση ReportsDaoImpl υλοποιεί το interface ReportsDao και παρέχει λειτουργίες για τη διαχείριση των αναφορών στη βάση δεδομένων. Χρησιμοποιεί το EntityManager για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Ανάκτηση Αναφοράς με ID: Μέθοδος `getReportById(long id)` που επιστρέφει μία αναφορά βάσει του ID της.
2. Ανάκτηση Όλων των Αναφορών: Μέθοδος `getAllReports()` που επιστρέφει όλες τις αναφορές.
3. Ανάκτηση Αναφορών με Τίτλο: Μέθοδος `getReportByTitle(String title)` που επιστρέφει αναφορές βάσει του τίτλου τους.
4. Ανάκτηση Αναφορών με Κατάσταση: Μέθοδος `getReportsByStatus(String status)` που επιστρέφει αναφορές βάσει της κατάστασής τους.

5. Ανάκτηση Αναφορών με ID Χρήστη: Μέθοδος `getReportsByUserId(Users userId)` που επιστρέφει αναφορές βάσει του ID του χρήστη.
6. Αποθήκευση Αναφοράς: Μέθοδος `saveReport(Reports report)` που αποθηκεύει μία νέα αναφορά.
7. Ενημέρωση Αναφοράς: Μέθοδος `updateReport(Reports report)` που ενημερώνει μία υπάρχουσα αναφορά.
8. Διαγραφή Αναφοράς: Μέθοδος `'deleteReport(Reports report)'` που διαγράφει μία αναφορά.

**Η κλάση `'ReportsDaoImpl'` διαχειρίζεται αναφορές μέσω της χρήσης του `'EntityManager'` για πρόσβαση και διαχείριση δεδομένων στη βάση. Παρέχει μεθόδους για την αποθήκευση, ενημέρωση, ανάκτηση και διαγραφή αναφορών με βάση το ID, τον τίτλο, την κατάσταση και τον χρήστη στον οποίο ανήκουν. Αυτές οι λειτουργίες βοηθούν στην οργάνωση και διαχείριση των αναφορών στο σύστημα.**

**Η κλάση `VotesDaoImpl` υλοποιεί το `interface VotesDao` και παρέχει λειτουργίες για τη διαχείριση των ψήφων στη βάση δεδομένων. Χρησιμοποιεί το `EntityManager` για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Ανάκτηση Ψήφου με ID: Μέθοδος `getVoteById(long id)` που επιστρέφει μία ψήφο βάσει του ID της.
2. Ανάκτηση Όλων των Ψήφων: Μέθοδος `getAllVotes()` που επιστρέφει όλες τις ψήφους.
3. Ανάκτηση Ψήφων με Όνομα: Μέθοδος `getVoteByName(String name)` που επιστρέφει ψήφους βάσει του ονόματός τους.
4. Ανάκτηση Ψήφων με ID Διαχειριστή: Μέθοδος `getVotesByAdminId(Admins adminId)` που επιστρέφει ψήφους βάσει του ID του διαχειριστή.
5. Ανάκτηση Ψήφων με ID Χρήστη: Μέθοδος `getVotesByUserId(Users userId)` που επιστρέφει ψήφους βάσει του ID του χρήστη.
6. Αποθήκευση Ψήφου: Μέθοδος `saveVotes(Votes vote)` που αποθηκεύει μία νέα ψήφο.
7. Ενημέρωση Ψήφου: Μέθοδος `updateVotes(Votes vote)` που ενημερώνει μία υπάρχουσα ψήφο.

8. Διαγραφή Ψήφου: Μέθοδος deleteVotes(Votes vote) που διαγράφει μία ψήφο.

**Η κλάση UsersDaoImpl υλοποιεί το interface UsersDao και παρέχει λειτουργίες για τη διαχείριση των χρηστών στη βάση δεδομένων. Χρησιμοποιεί το EntityManager για την αλληλεπίδραση με τη βάση δεδομένων και υποστηρίζει τις εξής λειτουργίες:**

1. Ανάκτηση Χρήστη με ID: Μέθοδος getUserById(long id) που επιστρέφει έναν χρήστη βάσει του ID του.
2. Ανάκτηση Όλων των Χρηστών: Μέθοδος getAllUsers() που επιστρέφει όλους τους χρήστες.
3. Ανάκτηση Χρηστών με Όνομα: Μέθοδος getUserByName(String name) που επιστρέφει χρήστες βάσει του ονόματός τους.
4. Ανάκτηση Χρηστών με Επώνυμο: Μέθοδος getUserBySurname(String surname) που επιστρέφει χρήστες βάσει του επωνύμου τους.
5. Ανάκτηση Χρηστών με Διαμέρισμα: Μέθοδος getUserByAppartment(String appartment) που επιστρέφει χρήστες βάσει του διαμερίσματός τους.
6. Ανάκτηση Premium Χρηστών με Βαθμούς: Μέθοδος getPremiumUsersByPoints(int points) που επιστρέφει premium χρήστες βάσει των βαθμών τους.
7. Αποθήκευση Χρήστη: Μέθοδος saveUser(Users user) που αποθηκεύει έναν νέο χρήστη.
8. Ενημέρωση Χρήστη: Μέθοδος updateUser(Users user) που ενημερώνει έναν υπάρχοντα χρήστη.
9. Διαγραφή Χρήστη: Μέθοδος `deleteUser(Users user)` που διαγράφει έναν χρήστη. Η κλάση `UsersDaoImpl` διαχειρίζεται χρήστες μέσω της χρήσης του `EntityManager` για πρόσβαση και διαχείριση δεδομένων στη βάση. Παρέχει μεθόδους για την αποθήκευση, ενημέρωση, ανάκτηση και διαγραφή χρηστών με βάση το ID, το όνομα, το επώνυμο, το διαμέρισμα και τους βαθμούς. Αυτές οι λειτουργίες βοηθούν στην οργάνωση και διαχείριση των χρηστών στο σύστημα.

## Class-Diagram:

