



**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΦΥΣΙΚΗΣ**

**ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΣΤΟΝ  
ΗΛΕΚΤΡΟΝΙΚΟ ΑΥΤΟΜΑΤΙΣΜΟ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Σχεδιασμός και Ανάπτυξη Εφαρμογής Υποστήριξης  
Συρμών του Μετρό της Αθήνας  
(Απεικόνιση Θέσης και Ημερολογίου Συμβάντων)**

**Ανθή – Αναστασία Ε. Σακελλάρη**

**Επιβλέπων: Ιωάννης Κοτρώνης, Καθηγητής**

**ΑΘΗΝΑ  
ΜΑΙΟΣ 2020**

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Σχεδιασμός και Ανάπτυξη Εφαρμογής Υποστήριξης Συρμών του Μετρό της Αθήνας  
(Απεικόνιση Θέσης και Ημερολογίου Συμβάντων)

**Ανθή – Αναστασία Ε. Σακελλάρη**  
**A.M.: 2009527**

**ΕΠΙΒΛΕΠΩΝ: Ιωάννης Κοτρώνης, Καθηγητής**

**ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ: Ιωάννης Κοτρώνης, Καθηγητής**  
**Διονύσιος Ρεΐσης, Καθηγητής**  
**Αντώνιος Πασχάλης, Καθηγητής**

Μάιος 2020

## ΠΕΡΙΛΗΨΗ

Ο αστικός σιδηρόδρομος είναι από τις πλέον προτιμώμενες δημόσιες συγκοινωνίες στις μητροπόλεις της σύγχρονης εποχής. Ο στόχος του αστικού σιδηρόδρομου είναι να επιτύχει ένα καθορισμένο επίπεδο σιδηροδρομικής κυκλοφορίας σε ένα δεδομένο χρόνο με ασφάλεια. Η ανάπτυξη της τεχνολογίας και των πληροφοριακών συστημάτων βοηθούν στην επίτευξη αυτών των στόχων. Με γνώμονα τα παραπάνω δημιουργήθηκε το Ημερολόγιο Συμβάντων των Συρμών του Μετρό της Αθήνας για την παροχή αξιόπιστου και διαθέσιμου τροχαίου υλικού. Μέσω αυτής της εφαρμογής δίνεται η δυνατότητα στο προσωπικό που συντηρεί τους συρμούς να έχει μια πλήρη απεικόνιση της κατάστασης του τροχαίου υλικού σε πραγματικό χρόνο και συγκεκριμένα να ενημερώνεται άμεσα για τυχόν βλάβες, για τις συντηρήσεις και όλο το ιστορικό των συρμών .

Στην παρούσα διπλωματική εργασία, έχοντας ως κεντρική ιδέα σχεδιασμού το υπάρχον Ημερολόγιο Συμβάντων του Μετρό της Αθήνας, δημιουργείται με την βοήθεια της γλώσσας προγραμματισμού Java, μια νέα εφαρμογή η οποία μπορεί να «τρέξει» σε οποιοδήποτε λειτουργικό περιβάλλον, έχει καλύτερα γραφικά και πιο πλούσιες βιβλιοθήκες.

Αναπτύχθηκε μια εφαρμογή όπου μπορεί να καταγράφονται και να εμφανίζονται σε πραγματικό χρόνο, ή/και σε ιστορικό: η θέση, η κατάσταση, οι συντηρήσεις, οι καθαρισμοί και τα διανυθέντα χιλιόμετρα των συρμών.

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε η γλώσσα προγραμματισμού Java και συγκεκριμένα οι βιβλιοθήκες Swing, AWT. Επίσης έγινε χρήση της JavaFX για την δημιουργία πιο πλούσιων UIs. Επιπλέον, το σχεδιαστικό εργαλείο Scene Builder χρησιμοποιήθηκε για την δημιουργία του FXML κώδικα και της CSS για καλύτερη συνοχή στη σχεδίαση, περισσότερες επιλογές μορφοποίησης και ευκολία διατήρησης και ανανέωσης του κώδικα. Παράλληλα, χρησιμοποιήθηκε η βάση δεδομένων MySQL, όπου αποθηκεύονται και ανακτώνται όλα τα δεδομένα των συρμών. Η αλληλεπίδραση της εφαρμογής με τη Βάση γίνεται μέσω του Hibernate Framework και του JDBC driver. Επιπροσθέτως, γίνεται χρήση από το Apache POI πρότζεκ ενός Java API, του HSSF και XSSF για Excel Αρχεία που δίνει την δυνατότητα εξαγωγής των πινάκων όπως δηλαδή των ενεργών βλαβών και του ιστορικού, των χιλιομέτρων και των συντηρήσεων.

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ:** Διαχείριση Συρμών

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:** Διαχείριση Συρμών, Java, JavaFX, MySQL, JPA - JDBC driver

## **ABSTRACT**

The Metro is one of the most preferable public transportation in main cities in today's era. The goal of the Urban Railway is to achieve safely a defined level of rail traffic in a given time. The evolution of technology and IT helped to create an application that offers reliable and available rolling stock. So because of the need to monitor the state of the trains' fleet of the Athens Metro, the fleet log diary has been created. Through that application the maintenance personnel can monitor the state of trains in real time and more specifically to inform for failures, for maintenance schedule and the whole history.

In this thesis, the current fleet log diary of Athens Metro is used as a main design concept and a new Java application has been created which is platform-independent, has more sophisticated GUI features and a lot of libraries with a lot of great features.

In this application the status and the position of each unit of the fleet is recorded and displayed in real time. The preventive maintenance can be scheduled, modified and recalled. Moreover, the unit's current mileage can be entered but also a history of all the fleet mileages can be displayed as a table or extracted as Excel file. A new type of cleaning can be scheduled and a complete history of all cleanings can be extracted from the database in the form of a table or Excel document.

The programming language for this application was Java and particularly the libraries Swing, AWT. JavaFX was used for better graphics. Specifically, the Scene Builder was used to produce some JavaFX FXML UIs and JavaFX CSS for better consistency in design, more formatting options and easier to maintain and update the code. Additionally, the data of the fleet are stored and recalled from a MySQL database. The Java Framework Hibernate and the JDBC driver are used for the application to interact with the database. Also, the Java API HSSF and XSSF for Excel files from Apache POI project are used for extraction of data from the tables like active or history of failures, mileages and maintenances.

**SUBJECT AREA:** Train Management

**KEYWORDS:** Train Management, Java, JavaFX, MySQL, JPA - JDBC driver

στην οικογένειά μου.

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστήσω τον καθηγητή μου κύριο Ιωάννη Κοτρώνη, καθηγητή Ε.Κ.Π.Α., για την πολύτιμη βοήθεια του, την σωστή καθοδήγησή του και την άπλετη υπομονή του. Τον κύριο Ιωάννη Μυλωνά, Τομεάρχη Συστημάτων Πληροφορικής ΣΤΑ.ΣΥ., για την συμβολή τους στην διεκπεραίωση αυτής της εργασίας. Τον κύριο Χρίστο Μαμαλουκάκη, Διευθυντή Τροχαίου Υλικού ΣΤΑ.ΣΥ., για την συμπαράσταση του. Επίσης, θα ήθελα να ευχαριστήσω τον κ. Κυριάκο Καραγιαννίδη, Γενικό Διευθυντή Συντήρησης ΣΤΑ.ΣΥ. που είναι ο πρώτος που υλοποίησε το Ημερολόγιο Συμβάντων του Τροχαίου Υλικού του Μετρό της Αθήνας.

## ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΠΡΟΛΟΓΟΣ.....</b>	<b>12</b>
<b>1. ΕΙΣΑΓΩΓΗ .....</b>	<b>13</b>
<b>2. ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ .....</b>	<b>15</b>
<b>3. ΣΧΕΔΙΑΣΜΟΣ ΣΥΣΤΗΜΑΤΟΣ.....</b>	<b>17</b>
<b>4. ΕΡΓΑΛΕΙΑ .....</b>	<b>22</b>
4.1. NetBeans IDE .....	22
4.1.1. easyUML .....	22
4.2. MySQL Workbench .....	22
4.3. Αντικειμενοστραφής Γλώσσα Προγραμματισμού JAVA .....	23
4.4. JavaFX.....	23
4.5. CSS .....	25
4.6. Βάση Δεδομένων.....	25
4.7. MySQL .....	25
4.8. JDBC .....	26
4.9. MySQL Connector/J .....	26
4.10. Hibernate Framework.....	27
4.11. Java Persistence API (JPA) .....	27
4.12. APACHE POI .....	29
4.13. UML (Unified Modeling Language, Ενοποιημένη Γλώσσα Μοντελοποίησης) .....	29
<b>5. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ.....</b>	<b>30</b>
5.1. Αρχική Οθόνη.....	30
5.2. Κύριο Μενού.....	32
5.3. Εργασίες.....	33
5.3.1. Βλάβες.....	34
5.3.2. Συντήρηση.....	39
5.3.3. Κατάσταση.....	43
5.3.4. Διαχείριση Χρηστών .....	49
5.4. Αμαξοστάσια .....	53
5.5. Λογισμικά Συρμών.....	55

<b>6.</b>	<b>ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....</b>	<b>56</b>
<b>6.1.</b>	<b>ΣΥΜΠΕΡΑΣΜΑΤΑ.....</b>	<b>56</b>
<b>6.2.</b>	<b>ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....</b>	<b>56</b>
	<b>ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΑ.....</b>	<b>58</b>
	<b>ΠΑΡΑΡΤΗΜΑ II .....</b>	<b>59</b>
	<b>ΑΝΑΦΟΡΕΣ .....</b>	<b>65</b>



## **ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ**

<b>Σχήμα 1:</b> Αλληλεπίδραση εφαρμογής με Βάση δεδομένων.....	17
<b>Σχήμα 2:</b> UML διάγραμμα εφαρμογής.....	19
<b>Σχήμα 3:</b> ER διάγραμμα της Βάσης Δεδομένων «metroathensdb».....	20

## ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

<b>Εικόνα 1:</b> Αρχική οθόνη .....	30
<b>Εικόνα 2:</b> Μήνυμα λάθους εισαγωγής.....	30
<b>Εικόνα 3:</b> Μήνυμα λάθους εισαγωγής.....	31
<b>Εικόνα 4:</b> Αλλαγή κωδικού χρήστη .....	31
<b>Εικόνα 5:</b> Κύριο μενού .....	32
<b>Εικόνα 6:</b> Εργασίες .....	33
<b>Εικόνα 7:</b> Εισαγωγή Νέας Βλάβης .....	34
<b>Εικόνα 8:</b> Ενεργές Βλάβες .....	35
<b>Εικόνα 9:</b> Τροποποίηση Βλάβης.....	36
<b>Εικόνα 10:</b> Αποκατάσταση Βλάβης.....	37
<b>Εικόνα 11:</b> Ιστορικό Βλαβών .....	38
<b>Εικόνα 12:</b> Δημιουργία Προληπτικής Συντήρησης .....	39
<b>Εικόνα 13:</b> Προληπτική συντήρηση μετά από επιλογή ημισυρμού .....	40
<b>Εικόνα 14:</b> Ιστορικό Συντηρήσεων .....	41
<b>Εικόνα 15:</b> Διόρθωση Προληπτικής Συντήρησης .....	42
<b>Εικόνα 16:</b> Μήνυμα λάθους.....	42
<b>Εικόνα 17:</b> Εισαγωγή Διανυθέντων Χιλιομέτρων .....	43
<b>Εικόνα 18:</b> Ιστορικό χιλιομέτρων.....	44
<b>Εικόνα 19:</b> Μήνυμα λάθους.....	44
<b>Εικόνα 20:</b> Διόρθωση Χιλιομέτρων .....	45
<b>Εικόνα 21:</b> Καταχώριση νέου καθαρισμού .....	46
<b>Εικόνα 22:</b> Ιστορικό καθαρισμών .....	47
<b>Εικόνα 23:</b> Διόρθωση καθαρισμού.....	48
<b>Εικόνα 24:</b> Δημιουργία νέου χρήστη και επιλογή δικαιωμάτων .....	50
<b>Εικόνα 25:</b> Εισαγωγή Ονομ/νυμου και εύρεση χρήστη .....	51
<b>Εικόνα 26:</b> Τροποποίηση δικαιωμάτων χρήστη .....	51
<b>Εικόνα 27:</b> Πίνακας δικαιωμάτων χρηστών.....	52
<b>Εικόνα 28:</b> Επιλογή Αμαξοστασίου .....	53
<b>Εικόνα 29:</b> Μήνυμα « Υπό κατασκευή» .....	53
<b>Εικόνα 30:</b> Απεικόνιση Αμαξοστασίου Πειραιά .....	54
<b>Εικόνα 31:</b> Αλλαγή κατάσταση ημισυρμού.....	54
<b>Εικόνα 32:</b> Λογισμικά Συρμών .....	55

## ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Πιθανά δικαιώματα των χρηστών .....	49
--	----

## **ΠΡΟΛΟΓΟΣ**

Η παρούσα πτυχιακή εργασία εκπονήθηκε στα πλαίσια περάτωσης των μεταπτυχιακών μου σπουδών στο Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών.

## 1. ΕΙΣΑΓΩΓΗ

Η αυξανόμενη ζήτηση για ταχύτερες και ασφαλέστερες σταθερές συγκοινωνίες έχει δημιουργήσει την ανάγκη για συνεχή παρακολούθηση της κατάστασης του τροχαίου υλικού, την έγκαιρη συντήρηση, τη γρήγορη ανίχνευση των σφαλμάτων καθώς και την άμεση αποκατάσταση αυτών. Η αστοχία του τροχαίου υλικού μπορεί να προκαλέσει καθυστερήσεις και διαταραχή του συγκοινωνιακού έργου ή ακόμα να αποτελέσει αιτία δυστυχήματος.

Η αλματώδης ανάπτυξη της τεχνολογίας έχει δημιουργήσει νέα εργαλεία για την σωστή καταγραφή των αστοχιών των συστημάτων του τροχαίου υλικού, των ενεργειών αποκατάστασης αυτών και όλων γενικά των πληροφοριών που βοηθούν στη σωστή συντήρηση του τροχαίου υλικού. Με γνώμονα τα παραπάνω, στην παρούσα εργασία, αναπτύχθηκε μια εφαρμογή διαχείρισης συρμών σε περιβάλλον Java η οποία συνδέεται με μια MySQL βάση δεδομένων. Η εφαρμογή μπορεί να εκτελείται αυτόνομα σε διαφορετικές συσκευές / αρχιτεκτονικές, έχει εξελιγμένα γραφικά, μπορεί να υποστηρίξει τη χρήση πολυμέσων, να εξαγάγει τα δεδομένα σε άλλες μορφές, ώστε να ανοίγονται και από άλλες εφαρμογές μορφοποίησης δεδομένων.

Η εφαρμογή έχει σαν σκοπό τη γρήγορη εύρεση της κατάστασης των συρμών αξιοποιώντας πιο σύγχρονα γραφικά για την καλύτερη οπτικοποίησή τους. Η ανάπτυξή της βασίστηκε στην ιδέα της εξέλιξης του υπάρχοντος συστήματος διαχείρισης συρμών του μετρό της Αθήνας.

Για την υλοποίηση χρησιμοποιήθηκαν εκτός από την Java και η JavaFX για την απεικόνιση των παραθύρων και των πινάκων με πιο πλούσια υποστήριξη μέσων/γραφικών.

Παράλληλα, για την σχεδίαση και την υλοποίηση της βάσης δεδομένων χρησιμοποιήθηκε η MySQL, όπου αποθηκεύονται όλα τα δεδομένα του τροχαίου υλικού.

### **Εργαλεία που Χρησιμοποιήθηκαν**

Η εφαρμογή Java αναπτύχθηκε στο ενσωματωμένο περιβάλλον ανοιχτού κώδικα (IDE) NetBeans. Επιπλέον, η σχεσιακή βάση δεδομένων MySQL δημιουργήθηκε και διαχειρίστηκε στο γραφικό περιβάλλον MySQL WorkBench.

### **Τι είναι το Ημερολόγιο Συμβάντων και Απεικόνισης Θέσης του Συνεργείου Επισκευών Τροχαίου Υλικού**

Το Ημερολόγιο Συμβάντων είναι μια εφαρμογή η οποία χρησιμεύει για την καταγραφή της τρέχουσας κατάστασης του τροχαίου υλικού. Δηλαδή την καταγραφή των αστοχιών των συστημάτων που απαρτίζουν τον κάθε συρμό, την συντήρηση που πρέπει να ακολουθηθεί και την τρέχουσα τοποθεσία του.

Πιο συγκεκριμένα, μέσω κατάλληλης διεπαφής ο χρήστης μπορεί να καταχωρίσει αλλά και να ενημερωθεί για οποιαδήποτε βλάβη, το πρόγραμμα συντήρησης, τον καθαρισμό, τα διανυθέντα χιλιόμετρα και την τρέχουσα τοποθεσία του κάθε συρμού. Τα στοιχεία αυτά αποθηκεύονται στη βάση δεδομένων, στην οποία έχουν έμμεσα πρόσβαση όλοι οι χρήστες ανάλογα με τα δικαιώματα που τους έχουν εκχωρηθεί από την εφαρμογή.

Επιπρόσθετα, ο χρήστης μπορεί να επιλέξει και τις εφαρμογές που χρειάζονται για την ανάλυση των δεδομένων των συρμών.

### **Οργάνωση των Κεφαλαίων**

Στην παρούσα πτυχιακή εργασία περιγράφεται ένα πρόγραμμα διαχείρισης συρμών της Γραμμής 1 του Μετρό της Αθήνας.

Στο **1<sup>ο</sup> κεφάλαιο**, περιγράφεται το αντικείμενο της παρούσας πτυχιακής εργασίας, αναφέρονται εν περιλήψει τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίησή της και ο σκοπός χρήσης του Ημερολογίου.

Στο **2<sup>ο</sup> κεφάλαιο**, αναφέρονται η λειτουργικότητα, δηλαδή περιγράφονται οι δυνατότητες της εφαρμογής.

Στο **3<sup>ο</sup> κεφάλαιο**, αναλύεται ο σχεδιασμός του συστήματος.

Στο **4<sup>ο</sup> κεφάλαιο** αναλύονται τα εργαλεία που χρησιμοποιήθηκαν.

Στο **5<sup>ο</sup> κεφάλαιο** υπάρχει το εγχειρίδιο χρήστη όπου περιγράφεται αναλυτικά η εφαρμογή, παρουσιάζονται όλα τα μενού και οι λειτουργίες τους για έναν χρήστη με πλήρη δικαιώματα.

Τέλος, στο **τελευταίο κεφάλαιο** αναφέρονται τα συμπεράσματα και οι μελλοντικές επεκτάσεις που μπορεί να γίνουν σε αυτήν την εφαρμογή.

## 2. ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ

Το “ **Ημερολόγιο Συμβάντων και Απεικόνιση Θέσης** ” είναι ένας εύχρηστος τρόπος καταγραφής και παρακολούθησης της κατάστασης των συρμών. Στο παρελθόν η τήρηση του αρχείου γινόταν σε χαρτί που το καθιστούσε μια δύσκολη διαδικασία τήρησης και προσπέλασης των καταγραφών. Ως εκ τούτου, δημιουργήθηκε μια ηλεκτρονική εφαρμογή για να αντικαταστήσει την έντυπη καταγραφή: των συμβάντων που δημιουργήθηκαν κατά την λειτουργία των συρμών και των διορθωτικών ενεργειών, των συντηρήσεων, των χιλιομέτρων, των πλυσιμάτων και των μετακινήσεων. Επιπλέον, με την ηλεκτρονική καταγραφή μπορούν να διεξαχθούν χρήσιμα συμπεράσματα για την διατήρηση του Τροχαίου Υλικού σε λειτουργία.

Η εφαρμογή του Ημερολογίου Συρμών είναι ένα σημαντικό εργαλείο διαχείρισης του τροχαίου υλικού γιατί σε ελάχιστο χρόνο ο τεχνικός μπορεί να μάθει όποια πληροφορία χρειάζεται ή να ομαδοποιήσει τις πληροφορίες που χρειάζεται για την εκτέλεση της εργασίας του. Παρέχει δηλαδή τη δυνατότητα ελέγχου όλων των εργασιών του στόλου, όπως της καταγραφής και του προγραμματισμού, της απεικόνισης και της εμφάνισης των προγραμμάτων που διευκολύνουν τη διαχείριση του.

Η εφαρμογή που αναλύεται στην παρούσα διπλωματική εργασία έχει τη δυνατότητα να καταχωρεί τις βλάβες των συρμών, να τις εμφανίζει (σε μορφή πίνακα) είτε αυτές εξακολουθούν να υφίστανται, είτε ως ιστορικό. Στις καταγραφές των Βλαβών αναφέρονται ο συρμός, το υποσύστημα που δυσλειτουργεί, η αιτία που το προκάλεσε, ο χρήστης που έκανε την αναφορά, η ημερομηνία δημιουργίας και όλες οι διορθωτικές ενέργειες που απαιτούνται για είναι ο στόλος πάντα λειτουργικός. Ακόμα, όταν αποκατασταθεί η βλάβη, τότε καταγράφεται και η ημερομηνία αποκατάστασής της.

Αντίστοιχα, γίνεται καταγραφή μέσω της εφαρμογής των μελλοντικών συντηρήσεων που θα υποστούν οι συρμοί. Ακόμα, παρέχεται η δυνατότητα να εμφανιστούν και να διορθωθούν όλες οι εισαγωγές που έχουν γίνει.

Επιπροσθέτως, κρατείται ένα αρχείο χιλιομέτρων που χρησιμοποιείται για τον υπολογισμό του χρονοδιαγράμματος των εργασιών στους συρμούς ή ως δείκτης φθοράς.

Οι καθαρισμοί των τρένων επίσης προγραμματίζονται μέσω της εφαρμογής ή ανακτώνται ως ιστορικό.

Η εφαρμογή ακόμα δίνει την δυνατότητα της απεικόνισης της θέσης και της κατάστασης των συρμών. Αυτό συντελεί στο να ξέρουν οι τεχνικοί και οι ρυθμιστές κυκλοφορίας πού βρίσκεται ο κάθε συρμός και σε ποια κατάσταση είναι. Επίσης, τους παρέχεται η δυνατότητα να του αλλάξουν κατάσταση, όπως, για παράδειγμα, να τον δεσμεύσουν για να γνωστοποιήσουν ότι εκτελείται κάποια εργασία στον συρμό. Ο ρυθμιστής έχει και μία ακόμα

δυνατότητα: της μετακίνησης του συρμού ώστε να υπάρχει αντιστοίχιση με την φυσική θέση του.

Επιπροσθέτως, μέσω της εφαρμογής μπορεί να περιοριστούν τα δικαιώματα πρόσβασης των χρηστών σε κάποια μενού, ανάλογα με τη φύση της εργασίας του κάθε χρήστη.

Τέλος, εμφανίζονται συγκεντρωμένα κάποια λογισμικά τρίτων για την διευκόλυνση των χρηστών στις εργασίες που θέλουν να εκτελέσουν.



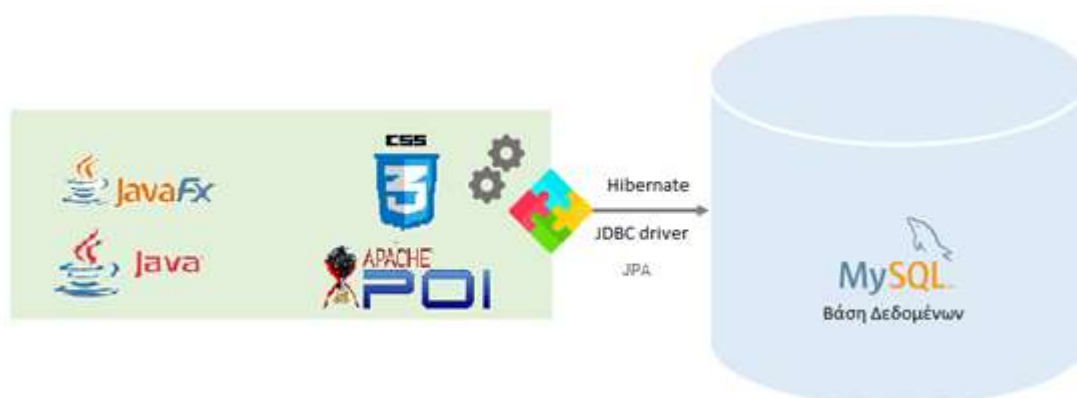
### 3. ΣΧΕΔΙΑΣΜΟΣ ΣΥΣΤΗΜΑΤΟΣ

Για την δημιουργία αυτής της εφαρμογής χρησιμοποιήθηκε η γλώσσα προγραμματισμού **Java** στο περιβάλλον του **Netbeans**. Οι πληροφορίες αποθηκεύονται και αντλούνται από την βάση δεδομένων **MySQL**. Η δημιουργία των πινάκων έγινε με το πρόγραμμα **MySQL Workbench**.

Επίσης, η διασύνδεση της εφαρμογής με την βάση δεδομένων γίνεται μέσω του **JDBC DriverManager Interface**, του **Hibernate Framework** ή του **JPA**, όπου ο Java Client επικοινωνεί με τον MySQL Server.

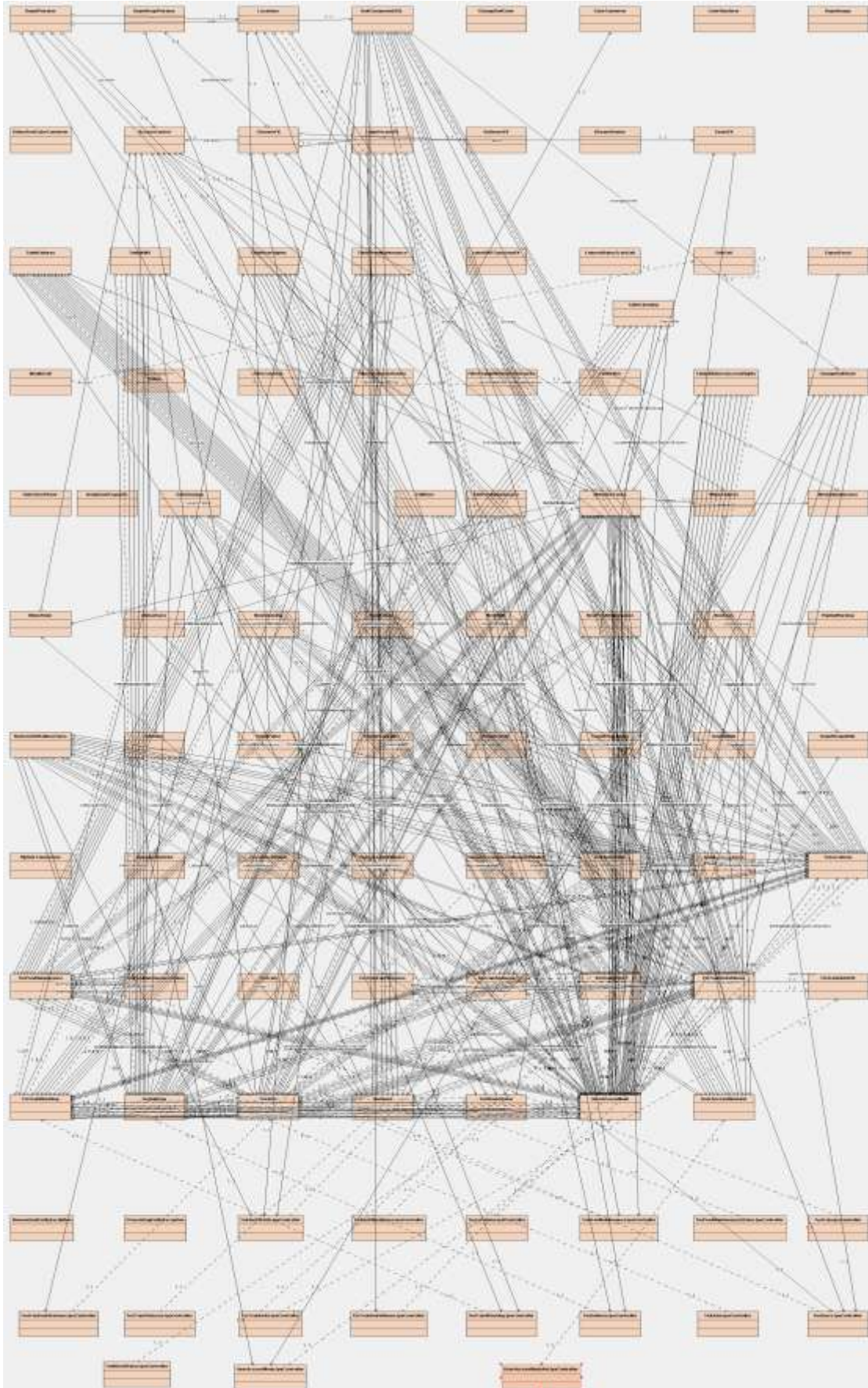
Το μοντέλο που χρησιμοποιήθηκε είναι 2- επιπέδων: η εφαρμογή επικοινωνεί άμεσα με τη βάση δεδομένων (**Σχήμα 1**).

Η εφαρμογή αναπτύχθηκε σε γλώσσα προγραμματισμού Java. Επίσης, για την δημιουργία των περισσότερων GUIs έγινε χρήση της JavaFX. Συγκεκριμένα, για την δημιουργία του FXML κώδικα χρησιμοποιήθηκε το πρόγραμμα Scene Builder. Επιπλέον, έγινε χρήση της CSS για καλύτερη συνοχή στη σχεδίαση, περισσότερες επιλογές μορφοποίησης και ευκολία διατήρησης και ανανέωσης του κώδικα. Ακόμα, χρησιμοποιήθηκε η Βάση Δεδομένων MySQL, όπου αποθηκεύονται και ανακτώνται τα δεδομένα των συρμών. Η αλληλεπίδραση της εφαρμογής με τη Βάση Δεδομένων γίνεται μέσω του **Hibernate Framework**, του **JDBC driver** και του **JPA**. Με την βοήθεια του wizard του NetBeans δημιουργήθηκαν οι Entity κλάσεις και οι Jpa controller Classes. Επίσης, γίνεται χρήση από το Apache POI πρότζεκτ ενός Java API, του HSSF and XSSF για Excel αρχεία για την εξαγωγή των πινάκων όπως των ενεργών / ιστορικού βλαβών, χιλιομέτρων, συντηρήσεων.



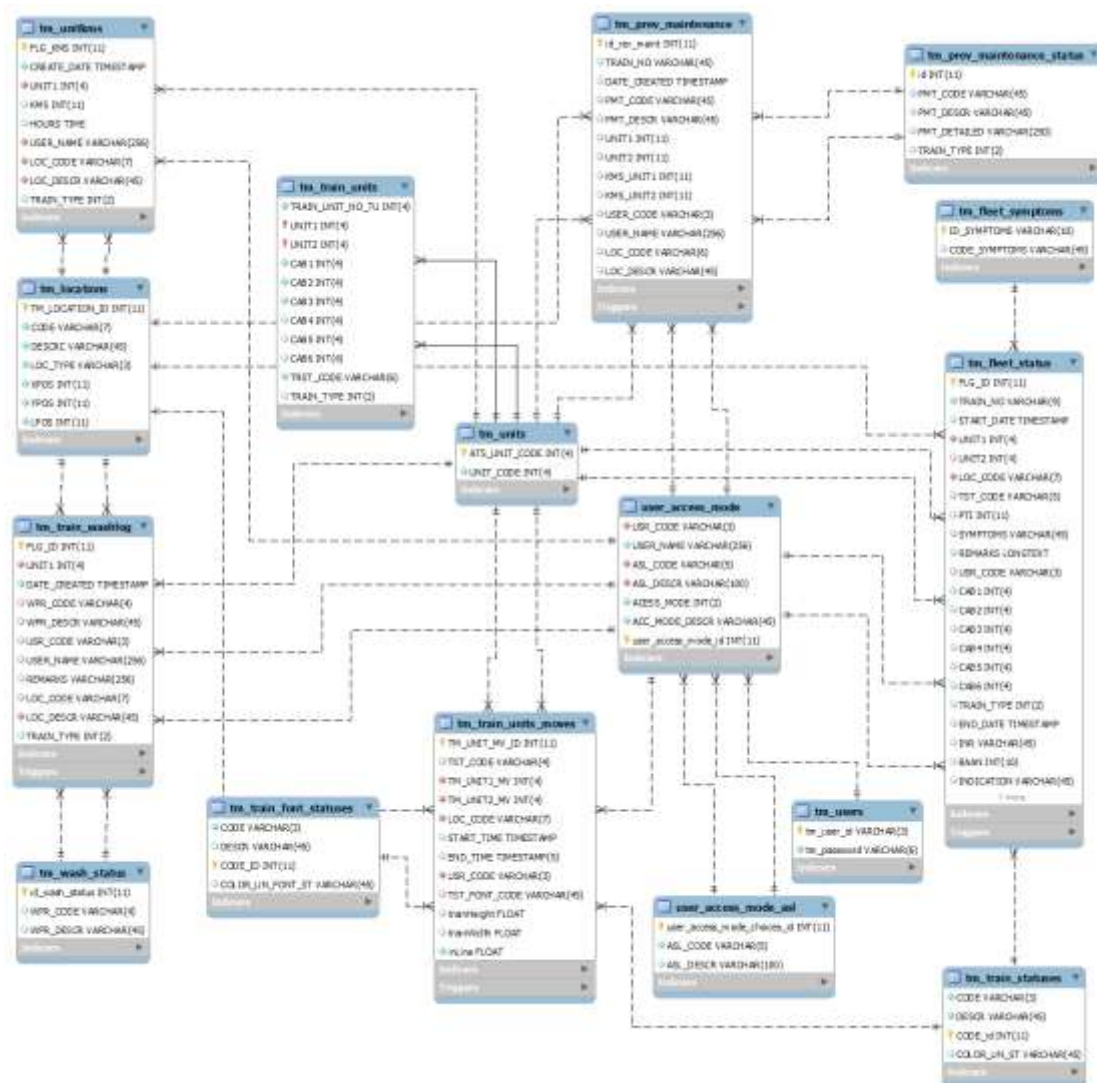
**Σχήμα 1:** Αλληλεπίδραση εφαρμογής με Βάση δεδομένων

Στην συνέχεια, απεικονίζεται ένα διάγραμμα κλάσεων της εφαρμογής, όπου φαίνονται οι εξαρτήσεις/ συσχετίσεις (**Σχήμα 2**) χωρίς να εμφανίζονται οι μέθοδοι και οι μεταβλητές που χρησιμοποιήθηκαν. Το διάγραμμα δημιουργήθηκε με την γλώσσα UML ( Unified Modeling Language) στο περιβάλλον του easyUML Designer.



**Σχήμα 2:** UML διάγραμμα εφαρμογής

Παρακάτω απεικονίζεται το ER διάγραμμα της παρούσας εργασίας (**Σχήμα 3**). Οι ονομασίες των Πινάκων και των Πεδίων αυτής της εργασίας έχουν αρκετές ομοιότητες με αυτές της Βάσης Δεδομένων του Μετρό της Αθήνας.



**Σχήμα 3:** ER διάγραμμα της Βάσης Δεδομένων «metroathensdb»

Περιγράφοντας το ER διάγραμμα βλέπουμε ότι κάθε ημισυρμός (“tm\_train\_units”) αποτελείται από οχήματα. Κάθε όχημα (“tm\_units”) έχει έναν μοναδικό αριθμό ATS. Κάθε ημισυρμός έχει καμπίνες οδήγησης, οχήματα και τύπο οχημάτων.

Η κατάσταση του ημισυρμού (“tm\_train\_statuses” και “tm\_train\_font\_statuses”) έχει έναν μοναδικό κωδικό, μια περιγραφή και ένα χρώμα.

Σχετικά με την μεταβολή της κατάστασης των ημισυρμών (“tm\_fleet\_status”) καταγράφεται ο αριθμός ATS των οχημάτων, το σύμπτωμα, η ημερομηνία και η ώρα δημιουργίας και αποκατάστασης της βλάβης, ο αριθμός PTI, η κατάσταση, οι παρατηρήσεις και ο τύπος των συρμών.



Το σύμπτωμα (“tm\_fleet\_symptoms”) έχει έναν μοναδικό κωδικό και μια περιγραφή.

Ένας ημισυρμός μπορεί να βρίσκεται σε διαφορετικές τοποθεσίες (“tm\_locations”). Κάθε τοποθεσία έχει κωδικό θέσης, περιγραφή θέσης, τύπο θέσης και συντεταγμένες (x,y).

Οι κινήσεις των ημισυρμών πρέπει να καταγράφονται (“tm\_train\_units\_moves”), όπως δηλαδή η κατάστασή του, η θέση που βρίσκεται, οι καμπίνες οδήγησης, ο χρήστης, η ημερομηνία και η ώρα της μετακίνησης αλλά με περιορισμούς, όπως το να μην βρίσκεται κάποιος άλλος ημισυρμός στην ίδια τροχιά ή το ότι για να αλλάξει αμαξοστάσιο πρέπει πρώτα να περάσει από την γραμμή. Επίσης, αλλάζει και η κατάσταση του συρμού με κάθε επιτρεπόμενη μετακίνηση.

Για κάθε ημισυρμό καταγράφονται τα χιλιόμετρα (“tm\_unitkms”) με πεδία την καμπίνα οδήγησης, τα χιλιόμετρα, την ημερομηνία και την ώρα εισαγωγής και τον τόπο καταγραφής.

Κάθε προγραμματισμένη συντήρηση (“tm\_prev\_maintenance”) έχει τον αριθμό ATS του οχήματος, την περιγραφή της συντήρησης, τα χιλιόμετρα που έχει διανύσει ο ημισυρμός, τη θέση και το όνομα χρήστη.

Κάθε καθαρισμός συρμού (“tm\_train\_washlog”) έχει τον αριθμό ATS του οχήματος, την περιγραφή του καθαρισμού, τυχόν σχόλια, θέση, ημερομηνία και ώρα και όνομα χρήστη.

## 4. ΕΡΓΑΛΕΙΑ

### ΛΟΓΙΣΜΙΚΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ ΓΙΑ ΤΗΝ ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Η υλοποίηση της εφαρμογής έγινε στο περιβάλλον του **Netbeans** και η δημιουργία της βάσης δεδομένων έγινε χρησιμοποιώντας την γλώσσα προγραμματισμού **MySQL** στο περιβάλλον του **MySQL Workbench**. Επίσης, χρησιμοποιήθηκε το plugin **easyUML** για την σχεδίαση των διαγραμμάτων κλάσεων της εφαρμογής.

#### 4.1. NetBeans IDE

Το πρόγραμμα **NetBeans** είναι ένα ανοιχτού κώδικα ολοκληρωμένο περιβάλλον ανάπτυξης (**IDE**) για συγγραφή κώδικα σε **Java**, **PHP**, **C++**, και άλλες γλώσσες προγραμματισμού. Το **NetBeans** προτιμάται επίσης σαν μια πλατφόρμα με πολλά στοιχεία που χρησιμοποιούνται για την ανάπτυξη εφαρμογών **Java**. Επίσης, είναι γραμμένο σε **Java** και τρέχει σε όλα σχεδόν τα λειτουργικά συστήματα που έχουν εικονικές μηχανές **Java (JVM)**, συμπεριλαμβανομένων των **Solaris**, **Mac OS**, και **Linux**.

##### 4.1.1. easyUML

Το **easyUML** είναι ένα εργαλείο **UML** διαγραμμάτων για το **NetBeans** το οποίο παρέχει δυνατότητες για την εύκολη δημιουργία και εργασία με τα διαγράμματα **UML**. Προς το παρόν υποστηρίζει μόνο τα διαγράμματα κλάσεων παρέχοντας έναν visual editor για την σχεδίαση των διαγραμμάτων, παραγωγή κώδικα Java από τα διαγράμματα και αντίστροφα τη δημιουργία διαγραμμάτων κλάσεων από κώδικα **Java**.

#### 4.2. MySQL Workbench

Το **MySQL Workbench** είναι ένα ενοποιημένο οπτικό εργαλείο για σχεδιαστές Βάσεων Δεδομένων, προγραμματιστές, και **DBAs**. Προσφέρει μοντελοποίηση δεδομένων, ανάπτυξη **SQL**, και γενικά εργαλεία διαχείρισης για διαμόρφωση διακομιστών, διαχείριση χρήστη, αντιγράφων και πολλά περισσότερα.

Παρακάτω περιγράφονται αναλυτικά τα εργαλεία που χρησιμοποιήθηκαν στο περιβάλλον του **NetBeans** για την υλοποίηση της παρούσας διπλωματικής εργασίας.

### 4.3. Αντικειμενοστραφής Γλώσσα Προγραμματισμού JAVA

Η γλώσσα προγραμματισμού JAVA δημιουργήθηκε από μια ομάδα επικεφαλής της οποίας ήταν ο James Gosling της Sun Microsystems. Η Sun Microsystems αγοράστηκε από την Oracle το 2010. Αρχικά ονομάστηκε Oak, η Java σχεδιάστηκε το 1991 για χρήση σε ενσωματωμένα τσιπ (embedded chips) σε ηλεκτρονικές συσκευές ευρείας κατανάλωσης. Το 1995, μετονομάστηκε σε Java, επανασχεδιάστηκε για την ανάπτυξη εφαρμογών Web.

Η JAVA έχει γίνει γνωστή σε μεγάλη κλίμακα. Η ταχεία ανάπτυξή της και η ευρεία αποδοχή της έχει να κάνει με τα σχεδιαστικά χαρακτηριστικά της, συγκεκριμένα η διαβεβαίωση ότι μπορεί να γραφτεί ένα πρόγραμμα μια φορά και να τρέξει παντού. Όπως έχει αναφέρει ο σχεδιαστής της, η JAVA είναι *απλή, αντικειμενοστραφής, διερμηνευτική, ανθεκτική, ασφαλής, ουδέτερη αρχιτεκτονικά, φορητή, υψηλών επιδόσεων, πολυνηματική, και δυναμική*.

Η Java είναι πλήρης, γενικού σκοπού γλώσσα προγραμματισμού η οποία μπορεί να χρησιμοποιηθεί για να κατασκευάσει υψίστης σημασίας δυνατές εφαρμογές. Σήμερα χρησιμοποιείται όχι μόνο για Web εφαρμογές αλλά και για μεμονωμένες εφαρμογές σε όλες τις πλατφόρμες των διακομιστών, σταθερών υπολογιστών και φορητών συσκευών.

### 4.4. JavaFX

Η JavaFX είναι ένα πλαίσιο υποστήριξης μέσων / γραφικών για ανάπτυξη GUI σε εφαρμογές Java. Το JavaFX API είναι ένα εξαιρετικό παράδειγμα του πως οι αρχές αντικειμενοστραφούς προγραμματισμού εφαρμόζονται. Η νέα GUI πλατφόρμα έχει έρθει για να αντικαταστήσει τα swing components.

Η JavaFX είναι ένα σύνολο γραφικών και πακέτων πολυμέσων που επιτρέπει στους προγραμματιστές να σχεδιάσουν, δημιουργήσουν, ελέγξουν, εντοπίσουν σφάλματα, και να αναπτύξουν πλούσιες εφαρμογές που λειτουργούν σταθερά σε διαφορετικές πλατφόρμες.

## Java API

Η JavaFX είναι μια βιβλιοθήκη η οποία είναι γραμμένη σε φυσική Java. Εντούτοις, θα είναι διαθέσιμη για τις γλώσσες που μπορούν να εκτελεστούν σε JVM.

## FXML και Scene Builder

Η FXML είναι μια βασισμένη σε XML δηλωτική γλώσσα σήμανσης για την δημιουργία μια JavaFX εφαρμογής διεπαφής χρήστη. Οι προγραμματιστές μπορούν να γράψουν κώδικα σε FXML ή να χρησιμοποιήσουν το Scene Builder για να σχεδιάσουν διαδραστικά ένα FXML GUI μέσω μια διεπαφής drag and drop.

## Swing διαλειτουργικότητα

Τα στοιχεία Java Swing μπορούν να ενσωματωθούν σε μια εφαρμογή JavaFX χρησιμοποιώντας την Swing Node κλάση. Παρομοίως, μια υπάρχουσα Swing εφαρμογή μπορεί επίσης να ενημερωθεί με JavaFX χαρακτηριστικά.

## Ενσωματωμένα UI controls και CSS

Η JavaFX παρέχει στους χρήστες όλα τα απαραίτητα UI controls για την ανάπτυξη ολοκληρωμένων εφαρμογών. Επιπλέον τα στοιχεία στην JavaFX εφαρμογή μπορούν να σχεδιαστούν χρησιμοποιώντας Cascading Style Sheet (CSS).

- **Πλούσια σετ από APIs**

Εκτός από ένα πλούσιο σετ από APIs για την ανάπτυξη GUI εφαρμογών, η **JavaFX** επίσης εμπεριέχει τις δυνατότητες της πλατφόρμας Java με στοιχεία όπως generic, σχολιασμό, πολυνηματισμό και εκφράσεις Λάμδα (Lambda).

- **Γραφικά σωλήνωσης επιτάχυνσης υλικού (Hardware-accelerated graphics pipeline)**

Τα γραφικά της JavaFX βασίζονται πάνω στην απόδοση γραφικών επιτάχυνσης υλικού (Hardware accelerated graphic rendering) γνωστή ως Prism. Η μηχανή Prism ομαλά και γρήγορα απεικονίζει τα **JavaFX** γραφικά όταν χρησιμοποιούνται με μια υποστηριζόμενη κάρτα γραφικών ή με μια μονάδα επεξεργασίας γραφικών (GPU). Εάν κάποιο σύστημα δεν διαθέτει κάποιο από αυτά, τότε το Prism έχει σαν προεπιλογή την στοίβα απεικόνισης λογισμικού (software-rendering stack).



Για την δημιουργία πλούσιων γραφικών, την εμφάνιση εικόνων αλλά και την εφαρμογή εφέ χρησιμοποιήθηκε το παρακάτω εργαλείο:

#### 4.5. CSS

Η **CSS (Cascading Style Sheet - Επάλληλα Φύλλα Στυλ)** είναι μια γλώσσα μορφής φύλλων που χρησιμοποιείται για να περιγράψει την παρουσίαση ενός εγγράφου γραμμένου σε μια γλώσσα σημείωσης όπως η **HTML**.

Η χρήση της **CSS** σε εφαρμογές **JavaFX** είναι παρόμοια με την χρήση **CSS** στην **HTML**. Χρησιμοποιείται για την δημιουργία προσαρμοσμένης εμφάνισης των εφαρμογών. Η **CSS** περιέχει ορισμούς εμφάνισης που ελέγχουν την απεικόνιση των στοιχείων διεπαφής χρήστη.

Για την διαχείριση των δεδομένων σε ηλεκτρονική μορφή χρησιμοποιήθηκαν τα παρακάτω εργαλεία:

#### 4.6. Βάση Δεδομένων

Η Βάση Δεδομένων είναι μια οργανωμένη συλλογή δομημένων πληροφοριών, ή δεδομένων, συνήθως αποθηκευμένων ηλεκτρονικά σε συστήματα υπολογιστών. Η Βάση Δεδομένων συνήθως ελέγχεται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων (**DBMS**). Όλα μαζί, τα δεδομένα και το **DBMS**, μαζί με τις εφαρμογές που σχετίζονται μαζί τους, αναφέρονται ως Σύστημα Βάσης Δεδομένων.

Τα δεδομένα στις κοινού τύπου βάσεις που λειτουργούν σήμερα είναι συνήθως μοντελοποιημένα σε γραμμές και στήλες σε μια σειρά από πίνακες για να κάνουν την επεξεργασία και την αναζήτηση αποτελεσματική. Τα δεδομένα μπορούν έτσι να είναι εύκολα προσβάσιμα, διαχειρίσιμα, ενημερώσιμα και οργανωμένα.

#### 4.7. MySQL

Η **MySQL** είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων βασισμένη στην **SQL** (Structured Query Language). Είναι η πιο δημοφιλής βάση δεδομένων ανοιχτού κώδικα, είναι σχεδιασμένη, διατίθεται και υποστηρίζεται από την Oracle.

**Η MySQL είναι ένα σύστημα διαχείρισης Βάσης Δεδομένων.**

Μια Βάση Δεδομένων είναι μια δομημένη συλλογή δεδομένων. Μπορεί να διαχειριστεί οτιδήποτε, από μια απλή λίστα για ψώνια μέχρι έναν τεράστιο όγκο πληροφοριών σένα εταιρικό δίκτυο. Για να προσθέσεις, να έχεις πρόσβαση και να επεξεργαστείς τα δεδομένα που είναι αποθηκευμένα σε μια βάση δεδομένων, χρειάζεται ένα σύστημα διαχείρισης βάσεων όπως η **MySQL**.

### **Οι MySQL βάσεις είναι σχεσιακές**

Μια σχεσιακή βάση «αποθηκεύει» δεδομένα σε ξεχωριστούς πίνακες παρά συγκεντρώνει όλα τα δεδομένα σε ένα μεγάλο αποθηκευτικό χώρο. Η οργάνωση των βάσεων είναι δομημένη σε φυσικά αρχεία για μεγαλύτερη ταχύτητα. Το λογικό μοντέλο, με αντικείμενα όπως βάσεις, πίνακες, όψεις, γραμμές και στήλες, προσφέρει ένα ευέλικτο προγραμματιστικό περιβάλλον.

Το πρόγραμμα **MySQL** είναι Ανοιχτού κώδικα, δηλαδή σημαίνει ότι είναι προσβάσιμο στον οποιοδήποτε θελήσει να χρησιμοποιήσει και να αλλάξει το λογισμικό.

Ο **MySQL** διακομιστής βάσης είναι πολύ γρήγορος, αξιόπιστος, διαθέτει επεκτασιμότητα και είναι εύκολος στην χρήση.

Ο διακομιστής μπορεί να τρέξει άνετα σε σταθερό ή φορητό υπολογιστή, παράλληλα με άλλες εφαρμογές, web servers, απαιτώντας λίγη ή καθόλου προσοχή.

Για να υπάρξει αμφίδρομη επικοινωνία της εφαρμογής με τη Βάση Δεδομένων χρησιμοποιήθηκαν τα παρακάτω εργαλεία:

## **4.8. JDBC**

Η **JDBC** (Java Database Connectivity ) σημαίνει Συνδεσιμότητα Βάσης Δεδομένων με την **Java**. Η **JDBC** είναι ένα Java API που χρησιμοποιείται για την σύνδεση και την εκτέλεση των ερωτημάτων με την βάση. Είναι μέρος της **JavaSE** (Java Standard Edition). Το **JDBC API** χρησιμοποιεί το **JDBC** πρόγραμμα οδήγησης για να συνδεθεί με την βάση δεδομένων.

## **4.9. MySQL Connector/J**

Η **MySQL** παρέχει σύνδεση για εφαρμογές πελάτη ανεπτυγμένες στην γλώσσα προγραμματισμού Java με το MySQL Connector/J, ένα πρόγραμμα

οδήγησης που εφαρμόζει τη Συνδεσιμότητα Java Βάσης Δεδομένων ( **JDBC**) API. Ο **MySQL Connector/J** είναι ένα πρόγραμμα οδήγησης **JDBC** Τύπου 4.

#### 4.10. Hibernate Framework

Το **Hibernate Framework** είναι ένα λογισμικό ανοιχτού κώδικα που σκοπό έχει να συνδέσει τα αντικείμενα που δημιουργούνται σε μια αντικειμενοστραφή γλώσσα προγραμματισμού (**Java**) με τους πίνακες μιας σχεσιακής βάσης δεδομένων. Η σύνδεση αυτή επιτυγχάνεται με την χρήση επιπρόσθετης πληροφορίας (metadata) που τοποθετείται κατάλληλα (μαζί με τον κώδικα **Java** ή σε ξεχωριστά xml αρχεία) και περιγράφει την αντιστοιχία μεταξύ των αντικειμένων και της Βάσης Δεδομένων. Γενικά το **Hibernate** προσφέρει την αυτόματη μετατροπή της μιας μορφής (αντικείμενα) στην άλλη (σχεσιακή βάση δεδομένων).

##### HQL

Η Hibernate χρησιμοποιεί μια ισχυρή γλώσσα ερωτημάτων (Γλώσσα Ερωτημάτων Hibernate - Hibernate Query Language) η οποία είναι παρόμοια στην εμφάνιση με την **SQL**. Συγκρίνοντάς την με την **SQL**, όμως, η **HQL** είναι πλήρως αντικειμενοστραφής και καταλαβαίνει τις έννοιες όπως **κληρονομικότητα, πολυμορφισμός και συσχέτιση**.

#### 4.11. Java Persistence API (JPA)

Το **JPA** είναι μια συλλογή από κλάσεις και μεθόδους που συνεχώς αποθηκεύουν δεδομένα στην βάση δεδομένων.

Απλοποιεί ιδιαίτερα την **Java persistence** και παρέχει μια προσέγγιση χαρτογραφημένου σχεσιακού-αντικειμένου που αφήνει στον χρήστη να προσδιορίσει πώς να καθορίσει αντικείμενα Java σε σχεσιακούς πίνακες βάσεων δεδομένων με έναν πρότυπο, φορητό τρόπο που λειτουργεί το ίδιο σε εφαρμογές **Java EE** και **Java SE**.

Χρησιμοποιώντας **JPA**, μπορεί να οριστεί οποιαδήποτε κλάση POJO ως μία οντότητα **JPA** – ένα αντικείμενο **Java** του οποίου τα μη μεταβατικά πεδία πρέπει να “αποθηκευτούν” σε μια σχεσιακή βάση χρησιμοποιώντας τις λειτουργίες ενός διαχειριστή οντοτήτων (**entity manager**) που αποκτήθηκαν από έναν **JPA persistence provider**.

## Entity Class

Μια **entity** είναι ένα ελαφρύ persistence domain αντικείμενο. Συνήθως μια entity αντιπροσωπεύει έναν πίνακα σε μια σχεσιακή βάση και κάθε στιγμιότυπο οντότητας ανταποκρίνεται σε μια γραμμή σ'αυτόν τον πίνακα. Το πρώτο προγραμματιστικό στοιχείο μιας οντότητας είναι η κλάση οντότητας (**Entity class**), αν και οι οντότητες μπορούν να χρησιμοποιήσουν βοηθητικές κλάσεις.

## JPQL (Java Persistence Query Language)

Η **JPQL** είναι μια ισχυρή γλώσσα ερωτημάτων που επιτρέπει στον χρήστη να ορίζει ερωτήματα στην Βάση Δεδομένων βάσει του μοντέλου της Οντότητας του. Η δομή και η σύνταξη είναι πολύ παρόμοια με την SQL. Αλλά υπάρχει μια σημαντική διαφορά, η **JPQL** χρησιμοποιεί το μοντέλο οντότητας αντικειμένου (entity object model) αντί των πινάκων των Βάσεων Δεδομένων για να προσδιορίσει τα Ερωτήματα. Αυτό την κάνει πολύ εύχρηστη για τους προγραμματιστές της **Java**, αν και η Βάση ακόμα χρησιμοποιεί **SQL**. Η Hibernate, ή οποιαδήποτε άλλη εφαρμογή **JPA**, πρέπει να μετατρέψει το ερώτημα **JPQL** σε **SQL**.

Η **JPQL** είναι η επέκταση της **EJBQL** (Γλώσσα Ερωτημάτων Οντότητας JavaBeans - Entity JavaBeans Query Language), προσθέτοντας χαρακτηριστικά όπως η εκτέλεση διαδικασιών συνένωση (join operations), η μαζική ανανέωση και η διαγραφή δεδομένων. Επίσης, οι συγκεντρωτικές λειτουργίες με προτάσεις ταξινόμησης και ομαδοποίησης και τα μονά ή πολλαπλά αποτελέσματα.

Η **JPQL** είναι ανεξάρτητη πλατφόρμα γλώσσας ερωτημάτων, είναι απλή και συνεκτική, μπορεί να χρησιμοποιηθεί με κάθε τύπο βάσης δεδομένων όπως η MySQL, Oracle. Επίσης, τα ερωτήματα **JPQL** μπορούν να δηλωθούν στατικά στα metadata ή να ενσωματωθούν δυναμικά στον κώδικα.

Για την εξαγωγή των δεδομένων σε μια μορφή που να μπορεί να τύχει επεξεργασίας από προγράμματα τρίτων, όπως είναι τα υπολογιστικά φύλλα (Excel) χρησιμοποιήθηκε το παρακάτω εργαλείο:

## 4.12. APACHE POI

Το πρότζεκτ Apache POI είναι το κύριο έργο για την ανάπτυξη αμιγώς θυρών **Java** μορφής αρχείου, βασισμένη στην Σύνθετη Μορφή Αρχείου Microsoft's OLE 2 (Microsoft's OLE 2 Compound Document Format). Η Σύνθετη Μορφή Αρχείου Microsoft's OLE 2 χρησιμοποιείται από τα αρχεία Microsoft Office, όπως και από προγράμματα που χρησιμοποιούν σύνολα ιδιοτήτων MFC για να σειριοποιήσουν τα δικά τους αντικείμενα αρχείων.

Το Apache POI είναι επίσης το κύριο πρότζεκτ για την ανάπτυξη αμιγώς θυρών Java μορφής αρχείου, βασισμένη στο Office Open XML (OOXML). Το **OOXML** είναι μέρος της προσπάθειας τυποποίησης ECMA / ISO.

### HSSF και XSSF για Έγγραφα Excel

Το **HSSF** είναι καθαρή εφαρμογή Java των αρχείων μορφής Microsoft Excel 97 (-2003) (BIFF8). Το **XSSF** είναι επίσης, καθαρή εφαρμογή **Java** των αρχείων μορφής Microsoft Excel XML (2007+) (OOXML) του POI προτζεκτ. Το **HSSF** και το **XSSF** παρέχουν τρόπους για ανάγνωση υπολογιστικών φύλλων και δημιουργία, τροποποίηση, ανάγνωση και συγγραφή XLS υπολογιστικών φύλλων.

Για την μοντελοποίηση των κλάσεων και των συσχετίσεων της εφαρμογής χρησιμοποιήθηκε το παρακάτω εργαλείο:

## 4.13. UML (Unified Modeling Language, Ενοποιημένη Γλώσσα Μοντελοποίησης)

### Διάγραμμα κλάσεων

Το διάγραμμα κλάσεων είναι ένα **UML** διάγραμμα δομής το οποίο απεικονίζει την δομή του σχεδιασμένου συστήματος στο επίπεδο των κλάσεων και διεπαφών, δείχνει τα χαρακτηριστικά τους, τους περιορισμούς και τις σχέσεις-συσχετίσεις, γενικεύσεις, εξαρτήσεις. Κάποιοι συνηθισμένοι τύποι διαγραμμάτων κλάσεων είναι:

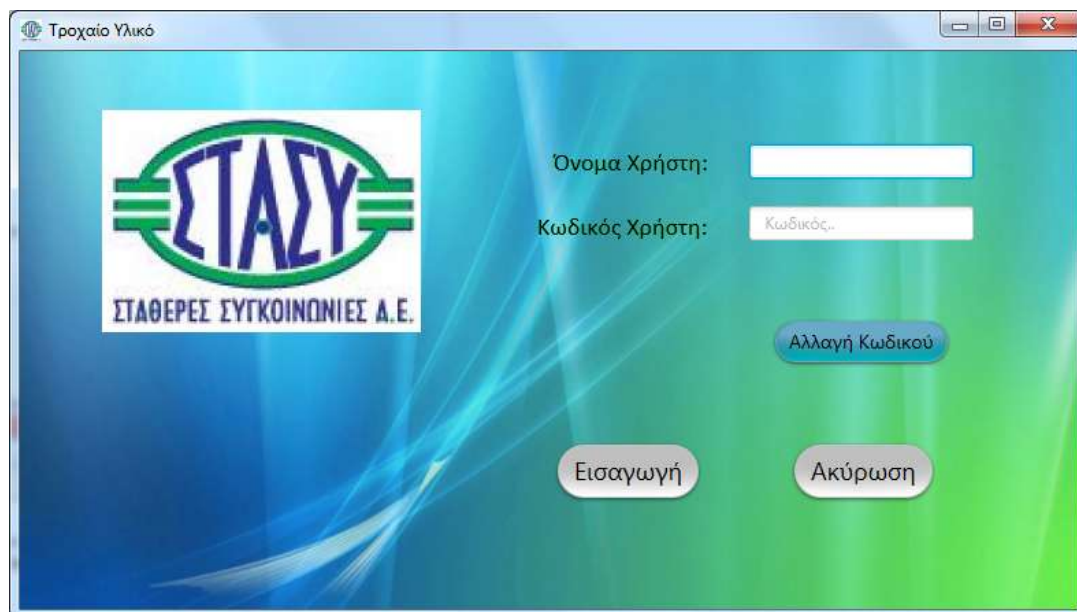
- Διάγραμμα δομής δεδομένων
- Διάγραμμα εφαρμογής κλάσεων.

## 5. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ

Σε αυτό το κεφάλαιο επιχειρείται να γίνει μια αναλυτική περιγραφή του “**Ημερολογίου Συμβάντων και της Απεικόνισης Θέσης**”. Η παρακάτω περιγραφή έγινε για έναν χρήστη που έχει πλήρη δικαιώματα.

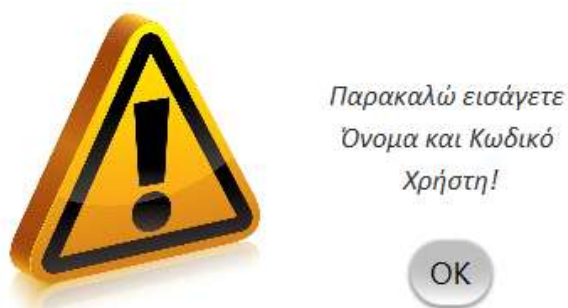
### 5.1. Αρχική Οθόνη

Στην αρχική οθόνη εισάγεται το όνομα χρήστη, ο κωδικός πρόσβασης και στην συνέχεια επιλέγει “**Εισαγωγή**” (**Εικόνα 1**).



**Εικόνα 1:** Αρχική οθόνη

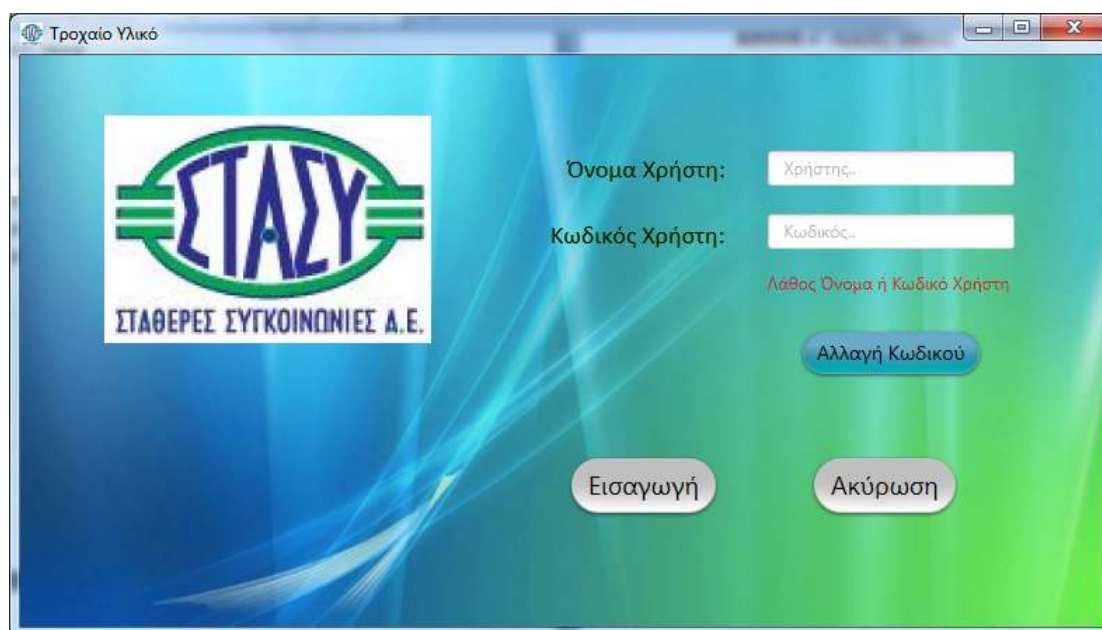
Αν δεν εισαχθεί κάποιο από τα πεδία τότε εμφανίζεται το παρακάτω μήνυμα (**Εικόνα 2**).



**Εικόνα 2:** Μήνυμα λάθους εισαγωγής



Αν έχουν εισαχθεί και τα δύο πεδία αλλά κάποιο είναι λάθος μετά από έλεγχο στην Βάση Δεδομένων, τότε εμφανίζεται ένα μήνυμα λάθους και διαγράφονται το όνομα χρήστη και ο κωδικός πρόσβασης (**Εικόνα 3**).



**Εικόνα 3:** Μήνυμα λάθους εισαγωγής

Επίσης, ο χρήστης έχει την δυνατότητα να αλλάξει και τον κωδικό πρόσβασης του. Εισάγει το όνομα χρήστη, τον παλιό κωδικό πρόσβασης, τον νέο κωδικό πρόσβασης και επαναλαμβάνει ξανά τον νέο κωδικό. Στην συνέχεια επιλέγει **“Εισαγωγή”**. Στην περίπτωση που ταυτοποιηθεί ο υπάρχων κωδικός πρόσβασης με το όνομα χρήστη, τότε μπορεί να ενημερωθεί ο κωδικός (**Εικόνα 4**).



**Εικόνα 4:** Αλλαγή κωδικού χρήστη

## 5.2. Κύριο Μενού

Στο κύριο μενού υπάρχουν τρεις επιλογές: “**Εργασίες**”, “**Αμαξοστάσια**” και “**Λογισμικά Συρμών**” (**Εικόνα 5**). Ο χρήστης μπορεί να επιλέξει όποιο επιθυμεί.

Στις “**Εργασίες**” ο χρήστης μπορεί να βρει τις Βλάβες (Δημιουργία, Τροποποίηση, Ενεργές και Ιστορικό), την Προληπτική Συντήρηση (Προγραμματισμός, Διόρθωση, Ιστορικό), την Κατάσταση των συρμών δηλαδή τα Χιλιόμετρα (Καταχώριση, Διόρθωση, Ιστορικό) και τους Καθαρισμούς (Νέος, Διόρθωση, Ιστορικό) και την Διαχείριση Χρηστών (Νέος Χρήστης, Διαχείριση Χρήστη, Όλοι οι Χρήστες). Στα “**Αμαξοστάσια**” εμφανίζεται μια λίστα με αυτά που υπάρχουν στην γραμμή 1. Φυσικά, ανάλογα με τα δικαιώματα του κάθε χρήστη είναι πιθανόν να μην εμφανιστούν κάποιες επιλογές. Τα πιθανά δικαιώματα φαίνονται στον **Πίνακας 1**.



**Εικόνα 5:** Κύριο μενού



### 5.3. Εργασίες

Επιλέγοντας τις “**Εργασίες**” στον χρήστη εμφανίζονται τέσσερις επιλογές: οι “**Βλάβες**”, η “**Συντήρηση**”, η “**Κατάσταση**” και η “**Διαχείριση Χρηστών**”. Τη “**Διαχείριση Χρηστών**” μπορούν να την έχουν ως επιλογή μόνο αυτοί που τους έχει δοθεί αυτό το δικαίωμα πρόσβασης (Πίνακας 1).

Στο παρακάτω GUI (**Εικόνα 6**) ο χρήστης έχει την δυνατότητα να διαχειριστεί την κατάσταση του συρμού.

Θα πρέπει να έχουν δοθεί τα ανάλογα δικαιώματα στον χρήστη για να δημιουργήσει ή να τροποποιήσει βλάβες και συντηρήσεις, να καταχωρίσει ή να τροποποιήσει τα χιλιόμετρα των συρμών. Στην περίπτωση που δεν έχουν δοθεί αυτά τα δικαιώματα από τον/τους διαχειριστή/ές της εφαρμογής τότε δεν εμφανίζονται οι επιλογές.



**Εικόνα 6:** Εργασίες

### 5.3.1. Βλάβες

Ο χρήστης εδώ μπορεί να δει τις ενεργές βλάβες των συρμών, να τροποποιήσει κάποια από αυτές, δηλαδή να αλλάξει την αιτία της ή να προσθέσει κάποια σχόλια που αφορούν την επίλυσή της. Επίσης, μπορεί να δει και το ιστορικό τους, δηλαδή ποιες ενέργειες έγιναν για την αποκατάσταση της κάθε βλάβης.

Συγκεκριμένα, επιλέγοντας το μενού **“Βλάβες”** από τις **“Εργασίες”** βλέπουμε τις παρακάτω επιλογές: **“Νέα”**, **“Ενεργές”** και **“Ιστορικό”**. Εδώ υπάρχει και μια κρυφή ακόμα επιλογή η **“Αποκατάσταση”**, που την βλέπουμε μόνο όταν έχει επιλεγεί κάποια ενεργή βλάβη από τον πίνακα ενεργών βλαβών που θα δούμε παρακάτω.

#### 5.3.1.1. Καταγραφή Νέας Βλάβης

Επιλέγοντας **“Νέα”** (Εικόνα 7), ο χρήστης εισάγει τα δεδομένα για μια νέα βλάβη που έχει δημιουργηθεί σε κάποιον συρμό. Επιλέγει από μια λίστα τον αριθμό Τ.Υ., την γραμμή και την ημερομηνία. Επιπλέον επιλέγει το βαγόνι ή τα βαγόνια όπου βρίσκεται η βλάβη και συμπληρώνει οποιαδήποτε πληροφορία θεωρεί αναγκαία για το συμβάν.

**Εικόνα 7:** Εισαγωγή Νέας Βλάβης

### 5.3.1.2. Ενεργές Βλάβες

Στην παρακάτω οθόνη φαίνονται οι **“Ενεργές”** βλάβες. Εδώ ο χρήστης έχει την δυνατότητα να κάνει εξαγωγή όλων των δεδομένων σε μορφή Excel μέσω του κομβίου **“Εξαγωγής Δεδομένων”**. Επίσης, μπορεί να χρησιμοποιήσει τα φίλτρα που υπάρχουν στις επικεφαλίδες των στηλών του πίνακα για να περιορίσει τα αποτελέσματα. Στη **“Συμπληρωματική πληροφορία”** μπορεί να εισαγάγει μέρος της λέξης ή φράσης και να περιοριστούν τα αποτελέσματα. Επίσης, δίνεται η δυνατότητα να αποκρυφτούν ολόκληρες στήλες ή να επανεμφανιστούν (πατώντας το +).

Επιπρόσθετα, δίνεται η δυνατότητα στον χρήστη με δεξί κλικ να κάνει αποκατάσταση ή διόρθωση της ενεργούς βλάβης.

Κατ.	Συμμετρία	Ημερ. Αναφοράς	S1	S2	S3	S4	S5	S6	Ερ. Αναφοράς	Σημειώσεις	Συμπληρωματική Πληροφορία	ΕΛΛΗΝ
170-171		20-01-2019 06:00:00	113	209					ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ5	#T1	new Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε:  Ο χρήστης null έγραψε: Δεν ανώθε. Ο χρήστης MIS έγραψε: Δεν λειτουργεί	0
110-111		24-07-2018 05:49:24	110						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ4	ΜΕΚΡΟΑΥΤΟΜΑΤΟΙ (ΜΕΒ)	new	0
122-123		25-07-2018 04:10:35	122						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ8	ΟΡΓΑΝΑ ΤΡΑΠΕΖΑΣ ΟΔΗΓΩ	new	0
122-129		25-07-2018 04:13:07	122						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ5	ΟΡΓΑΝΑ ΤΡΑΠΕΖΑΣ ΟΔΗΓΩ	new Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: ΟΧΧΧΧΧΧΧ	0
120-119		25-07-2018 04:15:01	120						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ8	ΣΠΡΟΣ ΕΠΙΒΑΤΩΝ - ΔΙΑΦΟΡΑ ΛΕΠΤΟΜΕΤΡΑ	new	0
120-110		30-08-2018 05:37:26	120						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ8	#T1	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: στασια	0
110-111		30-08-2018 05:43:17	120						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ4	ΑΝΔΡΕΥΣΗ ΚΑΠΝΟΥ	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: new new	0
3170-3180		30-08-2018 05:46:28	3170						ΠΕΡΑΙΑΣ ΕΝΑΛΟΩ. Τ7	ΑΕΡΟΣΥΜΠΙΕΣΤΗΣ	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: εμπει	0
113-114		05-09-2018 09:01:25	113						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ8	GRAFFITI (ΣΤΙΣΤΡ.-ΕΣΤΙΣΤΡΩΣ)	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: αδελ	2000
113-114		10-12-2018 12:54:23	113						ΠΕΡΑΙΑΣ ΑΠΟΒΑΘΡ. Τ3	CCTV - ΕΣΤΕΡΝΕΣ ΚΑΜΕΡΕΣ	Ο χρήστης null έγραψε: BNV	2000
105-104		10-12-2018 01:08:04	105						ΠΕΡΑΙΑΣ ΑΠΟΒΑΘΡ. Τ2	GRAFFITI	Ο χρήστης SAA έγραψε: SFSDGDFBNOGHYN	2000
108-107		11-12-2018 08:25:31	108	205					ΠΕΡΑΙΑΣ ΑΠΟΒΑΘΡ. Τ2	GRAFFITI	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: DNDFG	2000
110-111		04-03-2019 01:34:46	120						ΠΕΡΑΙΑΣ ΣΥΝΤΡ. Τ5	#T1	Ο χρήστης ΣΑΚΕΛΛΑΡΗ ΑΝΩΗ έγραψε: NEWW	2000
125-126		10-05-2020 02:02:36	125						ΠΕΡΑΙΑΣ ΕΠΙΣΤΕΙΝΗ Τ10a	ΑΕΡΟΣΥΜΠΙΕΣΤΗΣ	Ο χρήστης MIS έγραψε: ετοιμάστηκαν για λειτουργία.	0
117-118		10-05-2020 06:23:42	206						ΘΗΞΕΟ Τ17	CCTV - ΕΣΤΕΡΝΕΣ ΚΑΜΕΡΕΣ	Ο χρήστης MIS έγραψε:	0

**Εικόνα 8:** Ενεργές Βλάβες

### 5.3.1.3. Τροποποίηση Βλάβης

Επιλέγοντας “**Τροποποίηση**” εμφανίζεται η παρακάτω φόρμα όπου ο χρήστης μπορεί να τροποποιήσει κάποια βλάβη του συρμού ή/ και να προσθέσει συμπληρωματικές πληροφορίες.

Τροποποίηση Βλάβης

**Βαγόνια ημι-συρμού:**  
παλαιά υίτη-αυτήτων:

Ημι-συρμός 170-171	170 <input checked="" type="checkbox"/>	224 <input checked="" type="checkbox"/>	171 <input type="checkbox"/>
-----------------------	--	--	---------------------------------

Γραμμή: PT5      Χιλιόμετρα: 1627723      Κατάσταση: ME ΒΛΑΒΗ - ΔΙ...

Ημ/μηνία Συμβάντος: 2019-01-20 22:38

Συμπτωμα: PTI

Τροποποίηση      Ακύρωση

**Εικόνα 9:** Τροποποίηση Βλάβης



#### 5.3.1.4. Αποκατάσταση Βλάβης

Επιλέγοντας “**Αποκατάσταση**” εμφανίζεται η παρακάτω φόρμα όπου ο χρήστης μπορεί να κάνει κάποια διόρθωση στην γραμμή που έγινε το συμβάν, στο σύστημα και στο βαγόνι ή στα βαγόνια. Επίσης, μπορεί να γράψει τις διορθωτικές ενέργειες που έγιναν για την αποκατάσταση της βλάβης και να την βάλει στο Ιστορικό Βλαβών, δηλαδή να δοθεί ο συρμός στην κυκλοφορία.

Αποκατάσταση Βλάβης

**Βαγόνια ημι-συρμού:**  
παλαιά υφιστάμενα:

Ημι-συρμός	170	224	171
170-171	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Γραμμή: PT5 Χιλιόμετρα: 1627723 Κατάσταση: ΔΙΑΘΕΣΙΜΟΣ

Ημ/μηνία Συμβάντος: 2019-01-20 22:40 Ημ/μηνία Αποκατάστασης: 2020-05-10 22:40

Συμπτωμά: PTI

Αποκατάσταση Ακύρωση

**Εικόνα 10:** Αποκατάσταση Βλάβης



### 5.3.2. Συντήρηση

Στο μενού “**Συντήρηση**” μπορούμε να δούμε το ιστορικό των Προληπτικών συντηρήσεων, να το τροποποιήσουμε ή να δημιουργήσουμε μια νέα συντήρηση για κάποιο συρμό. Συγκεκριμένα υπάρχει η “**Προληπτική**” συντήρηση που έχει τρεις επιλογές: “**Προγραμματισμός**”, “**Διόρθωση**” και “**Ιστορικό**”.

#### 5.3.2.1. Δημιουργία Προληπτικής Συντήρησης

Επιλέγοντας το “**Προγραμματισμός**”, εμφανίζονται στον χρήστη οι παρακάτω φόρμες όπου έχει την δυνατότητα να κάνει εισαγωγή μιας νέας συντήρησης. Επιλέγοντας πρώτα τον **ημι-συρμό**, στην συνέχεια εμφανίζονται τα βαγόνια του, η ημερομηνία για εισαγωγή, τα χιλιόμετρα και μια λίστα όπου επιλέγει ο χρήστης τη συντήρηση που επιθυμεί να εισαγάγει για τον συγκεκριμένο ημισυρμό. Στη συνέχεια επιλέγει “**Εισαγωγή**”.



**Εικόνα 12:** Δημιουργία Προληπτικής Συντήρησης



Προληπτική Συντήρηση

Βαγόνια ημι-συρμού:  
σε λειτουργία υπερ-αοδήτου:

Ημι - συρμός  
3133-3134

3133

3134

Γραμμή  
L01

Χιλιόμετρα  
1637639

Ημ/μηνία  
2020-01-12 21:55

Συντήρηση:  
ΣΥΝΤΗΡΗΣΗ 11ης ΣΕΙΡΑΣ

Εισαγωγή

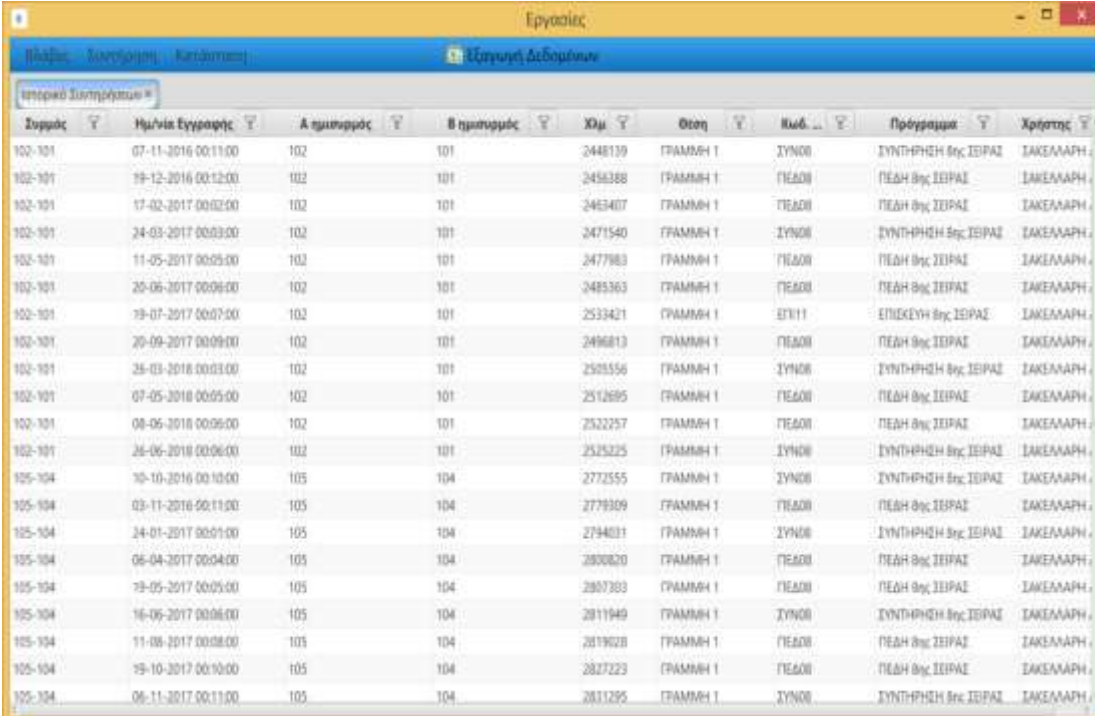
Ακύρωση

**Εικόνα 13:** Προληπτική συντήρηση μετά από επιλογή ημισυρμού



### 5.3.2.2. Ιστορικό Συντηρήσεων

Επιλέγοντας το “Ιστορικό” μπορούμε να δούμε το ιστορικό όλων των συντηρήσεων των συρμών (**Εικόνα 14**). Δίνονται οι ίδιες δυνατότητες με τους υπόλοιπους πίνακες, όπως το *Ιστορικό Βλαβών*. Δίνεται επίσης η δυνατότητα με δεξί κλικ, επιλέγοντας δηλαδή πάνω σε κάποια γραμμή του πίνακα να τροποποιηθεί η συντήρηση.



Συρμός	Ημέρα/Ώρα Εγγραφής	Α ημισυρμός	Β ημισυρμός	Χλμ.	Θεση	Κωδ.	Πρόγραμμα	Χρήστης
102-101	07-11-2016 00:11:00	102	101	2448139	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	19-12-2016 00:12:00	102	101	2456388	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	17-02-2017 00:02:00	102	101	2463407	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	24-03-2017 00:03:00	102	101	2471540	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	11-05-2017 00:05:00	102	101	2477983	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	20-06-2017 00:06:00	102	101	2485363	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	19-07-2017 00:07:00	102	101	2533421	ΓΡΑΜΜΗ 1	ΕΠΙ11	ΕΠΙΣΚΕΥΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	20-09-2017 00:09:00	102	101	2496813	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	26-03-2018 00:03:00	102	101	2505556	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	07-05-2018 00:05:00	102	101	2512695	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	08-06-2018 00:06:00	102	101	2522257	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
102-101	26-06-2018 00:06:00	102	101	2525225	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	10-10-2016 00:10:00	105	104	2772555	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	03-11-2016 00:11:00	105	104	2779309	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	24-01-2017 00:01:00	105	104	2794031	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	06-04-2017 00:04:00	105	104	2800820	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	19-05-2017 00:05:00	105	104	2807383	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	16-06-2017 00:06:00	105	104	2811949	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	11-08-2017 00:08:00	105	104	2819028	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	19-10-2017 00:10:00	105	104	2827223	ΓΡΑΜΜΗ 1	ΠΕΔΘ	ΠΕΔΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ
105-104	06-11-2017 00:11:00	105	104	2831295	ΓΡΑΜΜΗ 1	ΣΥΝΘ	ΣΥΝΤΗΡΗΣΗ θηξ ΣΕΡΑΣ	ΣΑΚΕΛΛΑΡΗ

**Εικόνα 14:** Ιστορικό Συντηρήσεων

### 5.3.2.3. Διόρθωση Προληπτικής Συντήρησης

Επιλέγοντας “**Διόρθωση**” ο χρήστης μπορεί να διορθώσει το είδος της συντήρησης ή να διορθώσει την ημερομηνία ή τη γραμμή που θα γίνει ή έγινε η συντήρηση του ημισυρμού (**Εικόνα 15**). Βασική προϋπόθεση είναι πρώτα ο χρήστης να έχει επιλέξει κάποια συντήρηση από τον πίνακα του “**Ιστορικό Συντηρήσεων**”, αλλιώς εμφανίζεται το παρακάτω *Μήνυμα λάθους*.



**Εικόνα 15:** Διόρθωση Προληπτικής Συντήρησης



Δεν έχετε επιλέξει κάποιο  
συρμό για ενημέρωση  
Προγραμματισμένης  
Συντήρησης!

OK

**Εικόνα 16:** Μήνυμα λάθους

### 5.3.3. Κατάσταση

Στο μενού της **Κατάστασης** βρίσκονται τα “**Χιλιόμετρα**” και οι “**Καθαρισμοί**” των συρμών.

#### 5.3.3.1. Χιλιόμετρα

Τα “**Χιλιόμετρα**” έχουν τις επιλογές: “**Καταχώριση**”, “**Διόρθωση**” και “**Ιστορικό**”.

##### 5.3.3.1.1. Καταχώριση

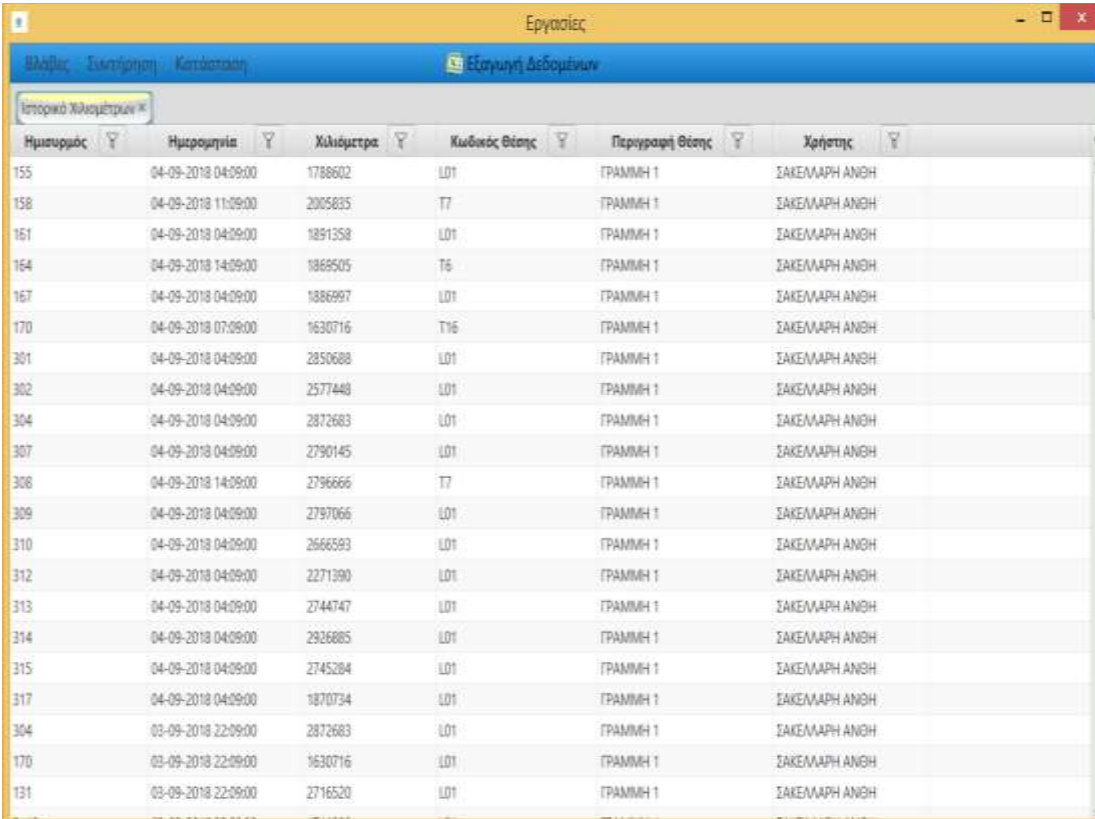
Επιλέγοντας “**Καταχώριση**” εμφανίζεται η παρακάτω φόρμα όπου μπορούμε να εισαγάγουμε τα τελευταία χιλιόμετρα που έχει διανύσει ο συρμός κατά την είσοδό του στο Αμαξοστάσιο.



**Εικόνα 17:** Εισαγωγή Διανυθέντων Χιλιομέτρων

### 5.3.3.1.2. Ιστορικό Χιλιομέτρων

Επιλέγοντας “Ιστορικό” ο χρήστης μπορεί να δει τα συνολικά χιλιόμετρα που έχουν εισαχθεί για όλους τους συρμούς. Ακόμα, έχει την δυνατότητα για **Διόρθωση Χιλιομέτρων** κάνοντας δεξί κλικ ή επιλέγοντας από το μενού “**Διόρθωση**”. Επίσης, σε αυτόν τον πίνακα υπάρχουν οι ίδιες δυνατότητες που βλέπουμε σε όλους τους παραπάνω πίνακες. Επιπλέον, εάν δεν έχει επιλεγεί κάποια καταχώριση εμφανίζεται το παρακάτω μήνυμα λάθους.



Ημισυρμός	Ημερομηνία	Χιλιόμετρα	Κωδικός Θέσης	Περιγραφή Θέσης	Χρήστης
155	04-09-2018 04:09:00	1788602	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
158	04-09-2018 11:09:00	2005835	T7	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
161	04-09-2018 04:09:00	1891358	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
164	04-09-2018 14:09:00	1869505	T6	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
167	04-09-2018 04:09:00	1886997	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
170	04-09-2018 07:09:00	1630716	T16	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
301	04-09-2018 04:09:00	2850688	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
302	04-09-2018 04:09:00	2577448	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
304	04-09-2018 04:09:00	2872683	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
307	04-09-2018 04:09:00	2790145	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
308	04-09-2018 14:09:00	2796666	T7	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
309	04-09-2018 04:09:00	2797066	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
310	04-09-2018 04:09:00	2666593	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
312	04-09-2018 04:09:00	2271390	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
313	04-09-2018 04:09:00	2744747	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
314	04-09-2018 04:09:00	2926885	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
315	04-09-2018 04:09:00	2745284	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
317	04-09-2018 04:09:00	1870734	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
304	03-09-2018 22:09:00	2872683	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
170	03-09-2018 22:09:00	1630716	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ
131	03-09-2018 22:09:00	2716520	L01	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΘΗ

Εικόνα 18: Ιστορικό χιλιομέτρων



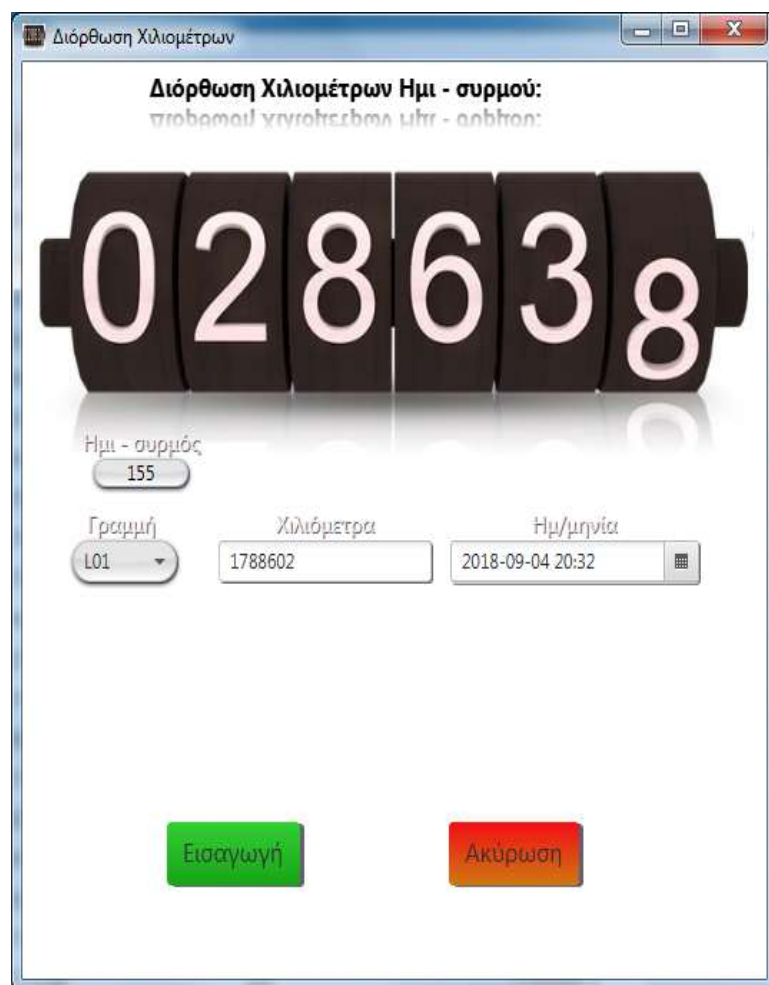
Δεν έχετε επιλέξει κάποιο  
συρμό για διόρθωση  
χιλιομέτρων!

OK

Εικόνα 19: Μήνυμα λάθους

### 5.3.3.1.3. Διόρθωση

Ο χρήστης έχει την δυνατότητα να επιλέξει από τον Πίνακα “*Ιστορικό χιλιομέτρων*” μια καταχώριση που έχει γίνει, και να την διορθώσει. Επιλέγοντας “**Διόρθωση**” το παρακάτω παράθυρο εμφανίζεται και μας δίνεται η δυνατότητα να διορθώσουμε τα χιλιόμετρα, την ημερομηνία εισαγωγής και την γραμμή που ήταν όταν καταχωρίστηκαν τα στοιχεία.



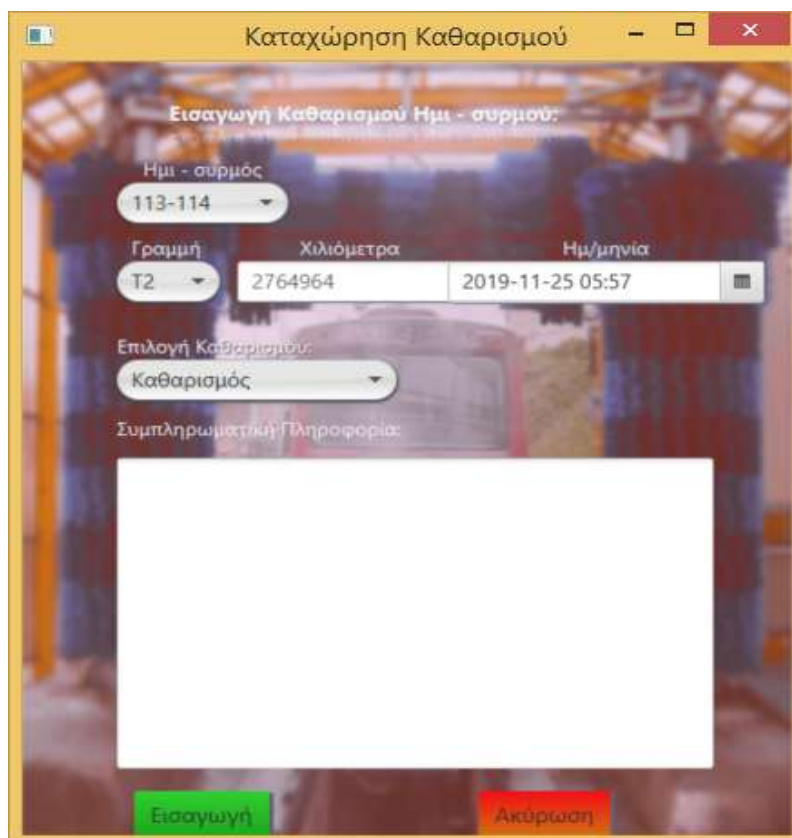
**Εικόνα 20:** Διόρθωση Χιλιομέτρων

### 5.3.3.2. Καθαρισμοί

Σε αυτό το μενού γίνεται η εισαγωγή, η τροποποίηση και εμφάνιση όλων των καθαρισμών που έχουν γίνει στους συρμούς. Συγκεκριμένα, υπάρχουν οι εξής επιλογές: **“Νέος”, “Διόρθωση”, “Ιστορικό”**.

#### 5.3.3.2.1. Νέος καθαρισμός

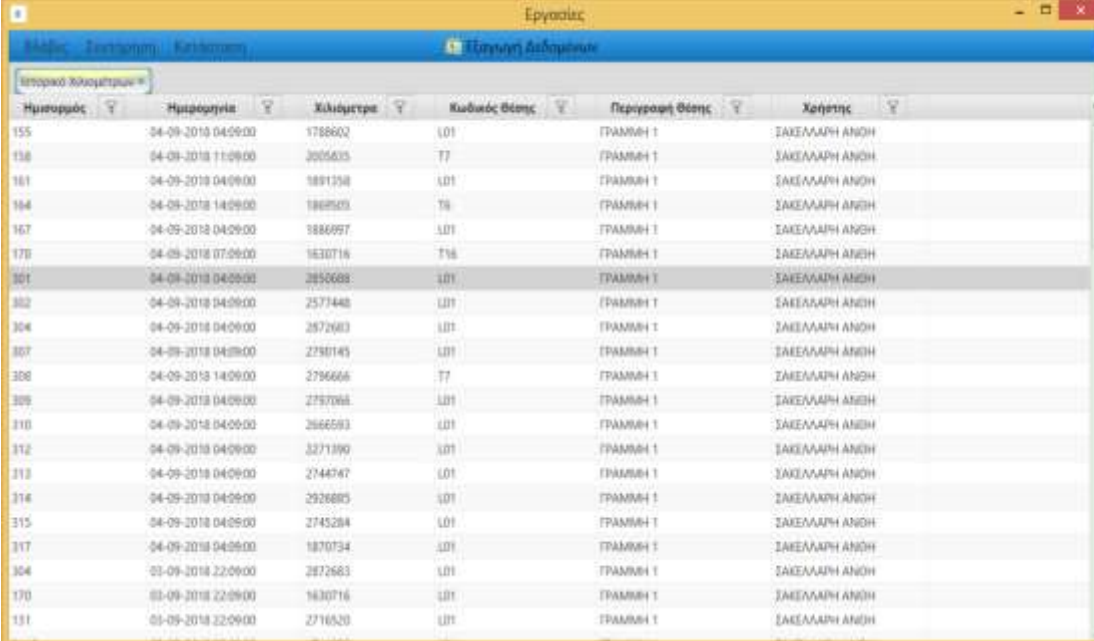
Επιλέγοντας **“Νέος”**, τότε ο χρήστης έχει την δυνατότητα να καταχωρίσει κάποιον νέο καθαρισμό (**Εικόνα 21**). Επιλέγει τον ημισυρμό, την γραμμή, την ημερομηνία και τον τύπο του καθαρισμού. Επιπλέον έχει την δυνατότητα να γράψει και κάποια συμπληρωματική πληροφορία.



**Εικόνα 21:** Καταχώριση νέου καθαρισμού

### 5.3.3.2.2. Ιστορικό Καθαρισμών

Στο “Ιστορικό” όπως φαίνεται παρακάτω, εμφανίζονται όλοι οι καθαρισμοί όλων των ημισυρμών. Εδώ υπάρχουν οι ίδιες δυνατότητες με τους προηγούμενους πίνακες, όπως επίσης και η δυνατότητα εξαγωγής των δεδομένων σε μορφή excel.



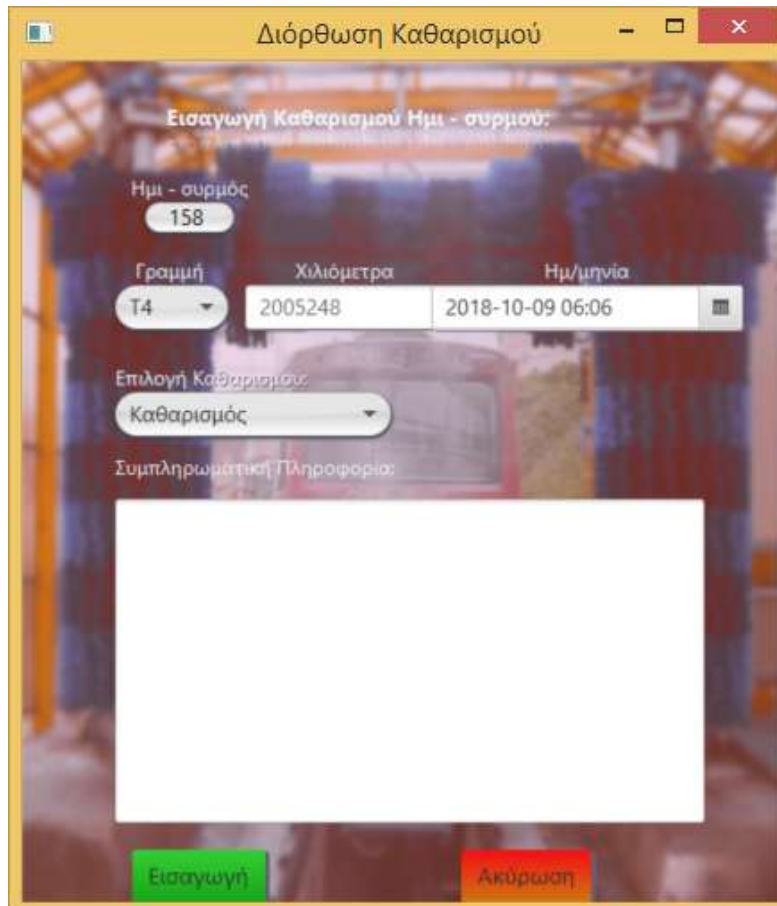
Ημισυρμός	Ημερομηνία	Χιλιόμετρα	Κωδικός Θέσης	Περιγραφή Θέσης	Χρήστης
155	04-09-2018 04:09:00	1788602	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
156	04-09-2018 11:09:00	2005605	77	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
161	04-09-2018 04:09:00	1891358	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
164	04-09-2018 14:09:00	1869509	78	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
167	04-09-2018 04:09:00	3886997	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
170	04-09-2018 07:09:00	1630716	716	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
301	04-09-2018 04:09:00	3850688	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
302	04-09-2018 04:09:00	2577448	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
304	04-09-2018 04:09:00	2872683	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
307	04-09-2018 04:09:00	2798145	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
308	04-09-2018 14:09:00	2796666	77	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
309	04-09-2018 04:09:00	2797066	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
310	04-09-2018 04:09:00	2666593	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
312	04-09-2018 04:09:00	3271390	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
313	04-09-2018 04:09:00	2744747	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
314	04-09-2018 04:09:00	2926885	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
315	04-09-2018 04:09:00	2745284	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
317	04-09-2018 04:09:00	1870734	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
304	03-09-2018 22:09:00	2872683	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
170	03-09-2018 22:09:00	1630716	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ
131	03-09-2018 22:09:00	2716520	101	ΓΡΑΜΜΗ 1	ΣΑΚΕΛΛΑΡΗ ΑΝΔΡΗ

Εικόνα 22: Ιστορικό καθαρισμών



### 5.3.3.2.3. Διόρθωση Καθαρισμού

Επιλέγοντας από το “Ιστορικό” κάποια εισαγωγή που έχει γίνει μπορούμε να την διορθώσουμε είτε με δεξί κλικ, είτε από το μενού των καθαρισμών επιλέγοντας “Διόρθωση” ( Εικόνα 23).



**Εικόνα 23:** Διόρθωση καθαρισμού



#### 5.3.4. Διαχείριση Χρηστών

Το μενού “**Διαχείριση χρηστών**” μπορεί να το επιλέξει μόνο ένας εξουσιοδοτημένος χρήστης και έχει την δυνατότητα να εισαγάγει έναν νέο χρήστη και να του προσθέτει ή και να του αφαιρεί δικαιώματα πρόσβασης στην εφαρμογή. Το συγκεκριμένο μενού **εμφανίζεται στους χρήστες που έχουν δικαιώματα διαχείρισης χρηστών** (Πίνακας 1).

Διαχείριση Πρόσβασης Χρηστών
Χρήστης Γραμμής 1
Χρήστης Γραμμής 2
Χρήστης Γραμμής 3
Καταχώριση ή /και διόρθωση καθαρισμών
Καταχώριση ή /και διόρθωση χιλιομέτρων
Καταχώριση ή /και διόρθωση προληπτικής συντήρησης
Εμφάνιση Αναφορών βλαβών
Καταχώριση νέας Βλάβης
Αποκατάσταση Βλάβης
Ιστορικό Αναφοράς Βλαβών
Μετακίνηση ή/ και Αλλαγή Κατάστασης Συρμού

**Πίνακας 1:** Πιθανά δικαιώματα των χρηστών

Στο συγκεκριμένο μενού οι επιλογές είναι “**Νέος Χρήστης**”, “**Διαχείριση Χρήστη**” και “**Όλοι οι Χρήστες**” όπου φαίνονται όλοι οι χρήστες με τα δικαιώματά τους.

#### 5.3.4.1. Νέος Χρήστης

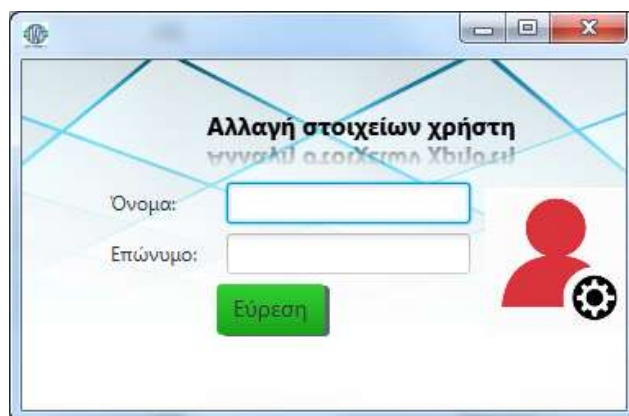
Επιλέγοντας από το μενού “**Νέος Χρήστης**” τότε μπορούμε να δημιουργήσουμε έναν νέο χρήστη με τα δικαιώματα που επιθυμούμε. Μπορούμε να του δώσουμε δυνατότητες “**READ**” ή “**READ/WRITE**” (Εικόνα 24).

Κωδικός Πρόσβασης	Περιγραφή Πρόσβασης	Τρόπος πρόσβασης	Επιλογή
R001	RSD, ΕΜΦΑΝΙΣΗ - ΔΙΟΡΘΩΣΗ ΑΝΑΦΟΡΑΣ ΒΛΑΒΗΣ ΣΥ...	READ/WRITE	<input type="checkbox"/>
R002	RSD, ΑΠΟΚΑΤΑΣΤΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ - ΒΛΑΒΗΣ	READ/WRITE	<input type="checkbox"/>
R003	RSD, ΚΑΤΑΧΩΡΗΣΗ ΩΡΩΝ ΛΕΙΤΟΥΡΓΙΑΣ LOCO	READ/WRITE	<input type="checkbox"/>
R004	RSD, ΔΙΑΧΕΙΡΙΣΗ ΔΕΝΔΡΩΝ ΕΠΙΠΕΔΩΝ ΕΓΚΑΤΑΣΤΑΣΕ...	READ/WRITE	<input type="checkbox"/>
R005	RSD, ΔΙΑΧΕΙΡΙΣΗ ΣΦΑΛΜΑΤΩΝ ΔΙΑΓΝΩΣΤΙΚΟΥ ΣΥΣΤΗ...	READ/WRITE	<input type="checkbox"/>
R006	RSD, ΔΙΑΧΕΙΡΙΣΗ ΣΥΜΠΤΩΜΑΤΩΝ	READ/WRITE	<input type="checkbox"/>
R007	RSD, ΔΙΑΧΕΙΡΙΣΗ ΚΑΘΑΡΙΣΜΩΝ	READ/WRITE	<input type="checkbox"/>
R008	RSD, ΔΙΑΧΕΙΡΙΣΗ ΧΙΛΟΜΕΤΡΩΝ	READ/WRITE	<input type="checkbox"/>
R009	RSD, ΔΙΑΧΕΙΡΙΣΗ ΤΡΕΝΩΝ	READ/WRITE	<input type="checkbox"/>
R010	RSD, ΚΑΤΑΧΩΡΗΣΗ ΝΕΑΣ ΒΛΑΒΗΣ	READ/WRITE	<input type="checkbox"/>
R011	RSD, ΚΑΤΑΧΩΡΗΣΗ ΠΡΟΛΗΠΤΙΚΗΣ	READ/WRITE	<input type="checkbox"/>
R012	RSD, ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΣΒΑΣΕΩΝ ΧΡΗΣΤΩΝ	READ/WRITE	<input type="checkbox"/>

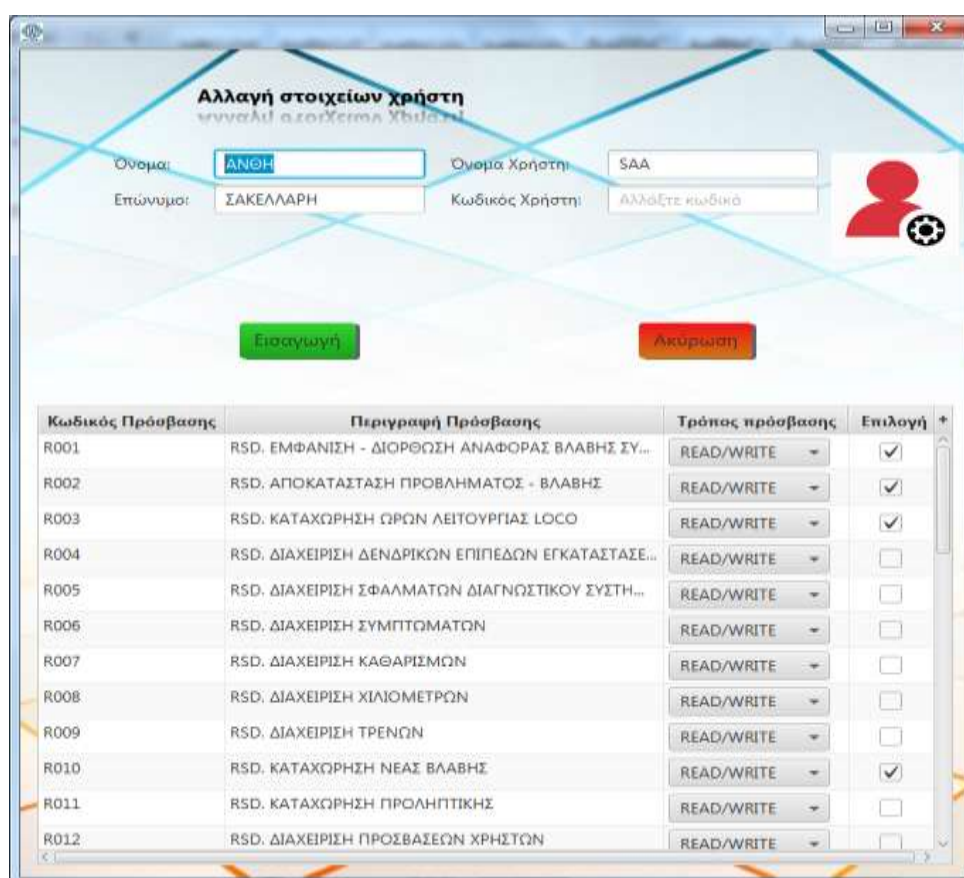
**Εικόνα 24:** Δημιουργία νέου χρήστη και επιλογή δικαιωμάτων

### 5.3.4.2. Διαχείριση Χρήστη

Σε αυτό το παράθυρο (**Εικόνα 25**) εισάγουμε το όνομα και το επώνυμο του χρήστη που επιθυμούμε να αλλάξουμε, τα δικαιώματα πρόσβασης για την εφαρμογή του Ημερολογίου και πατάμε “**Εύρεση**”. Στην συνέχεια εμφανίζεται η φόρμα της **Εικόνα 26**.



**Εικόνα 25:** Εισαγωγή Ονομ/νυμου και εύρεση χρήστη

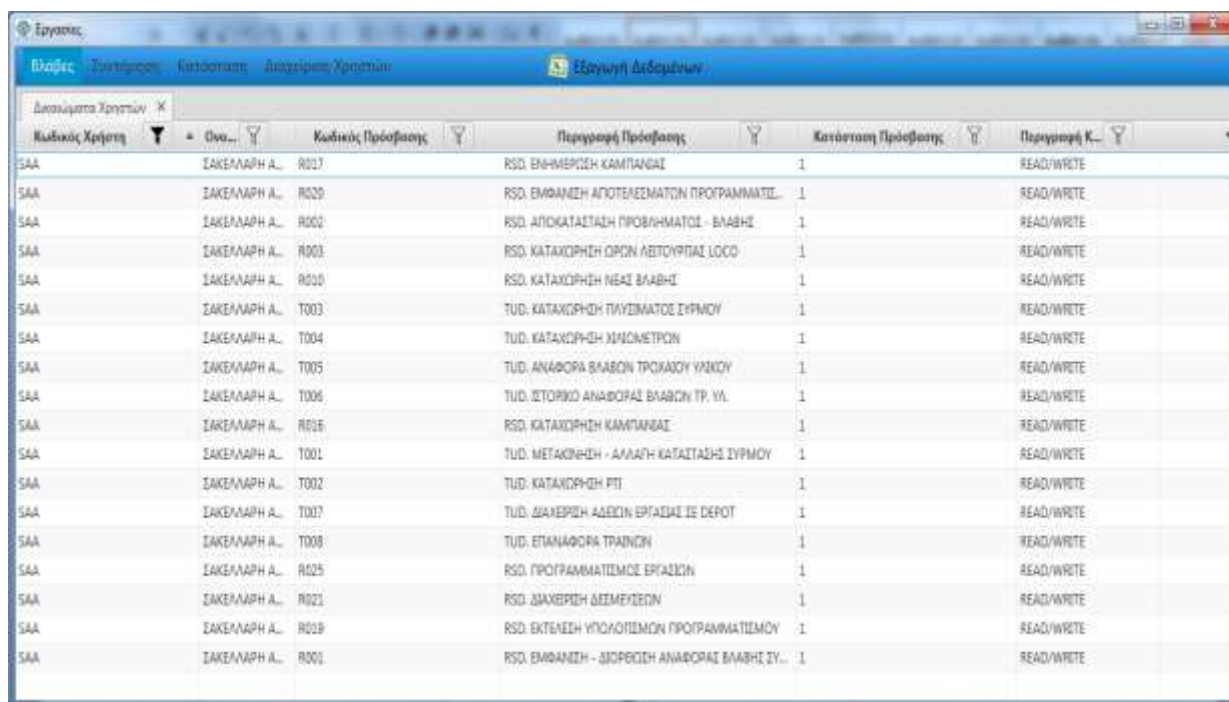


Κωδικός Πρόσβασης	Περιγραφή Πρόσβασης	Τρόπος πρόσβασης	Επιλογή
R001	RSD, ΕΜΦΑΝΙΣΗ - ΔΙΟΡΘΩΣΗ ΑΝΑΦΟΡΑΣ ΒΛΑΒΗΣ ΣΥ...	READ/WRITE	<input checked="" type="checkbox"/>
R002	RSD, ΑΠΟΚΑΤΑΣΤΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ - ΒΛΑΒΗΣ	READ/WRITE	<input checked="" type="checkbox"/>
R003	RSD, ΚΑΤΑΧΩΡΗΣΗ ΩΡΩΝ ΛΕΙΤΟΥΡΓΙΑΣ LOCO	READ/WRITE	<input checked="" type="checkbox"/>
R004	RSD, ΔΙΑΧΕΙΡΙΣΗ ΔΕΝΔΡΙΚΩΝ ΕΠΙΠΕΔΩΝ ΕΓΚΑΤΑΣΤΑΣΕ...	READ/WRITE	<input type="checkbox"/>
R005	RSD, ΔΙΑΧΕΙΡΙΣΗ ΣΦΑΛΜΑΤΩΝ ΔΙΑΓΝΩΣΤΙΚΟΥ ΣΥΣΤΗ...	READ/WRITE	<input type="checkbox"/>
R006	RSD, ΔΙΑΧΕΙΡΙΣΗ ΣΥΜΠΤΩΜΑΤΩΝ	READ/WRITE	<input type="checkbox"/>
R007	RSD, ΔΙΑΧΕΙΡΙΣΗ ΚΑΘΑΡΙΣΜΩΝ	READ/WRITE	<input type="checkbox"/>
R008	RSD, ΔΙΑΧΕΙΡΙΣΗ ΧΙΛΙΟΜΕΤΡΩΝ	READ/WRITE	<input type="checkbox"/>
R009	RSD, ΔΙΑΧΕΙΡΙΣΗ ΤΡΕΝΩΝ	READ/WRITE	<input type="checkbox"/>
R010	RSD, ΚΑΤΑΧΩΡΗΣΗ ΝΕΑΣ ΒΛΑΒΗΣ	READ/WRITE	<input checked="" type="checkbox"/>
R011	RSD, ΚΑΤΑΧΩΡΗΣΗ ΠΡΟΛΗΠΤΙΚΗΣ	READ/WRITE	<input type="checkbox"/>
R012	RSD, ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΣΒΑΣΕΩΝ ΧΡΗΣΤΩΝ	READ/WRITE	<input type="checkbox"/>

**Εικόνα 26:** Τροποποίηση δικαιωμάτων χρήστη

### 5.3.4.3. Όλοι οι Χρήστες

Επιλέγοντας από το μενού “Όλοι οι Χρήστες” εμφανίζεται ο παρακάτω πίνακας με όλους τους χρήστες και τα δικαιώματα που τους έχουν εκχωρηθεί. Εδώ, όπως και στους άλλους πίνακες στις κεφαλίδες των στηλών μπορεί ο χρήστης να φιλτράρει τα αποτελέσματα είτε τσεκάροντάς τα από μια λίστα, είτε γράφοντας κάποια λέξη ή γράμματα στην επικεφαλίδα της κάθε στήλης.



Κωδικός Χρήστη	Όνομα	Κωδικός Πρόσβασης	Περιγραφή Πρόσβασης	Κατάσταση Πρόσβασης	Περιγραφή Κ...
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R017	RSD. ΕΦΗΜΕΡΙΔΗ ΚΑΜΠΑΝΙΑΣ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R020	RSD. ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΠΡΟΓΡΑΜΜΑΤΙΣ...	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R002	RSD. ΑΠΟΚΑΤΑΣΤΑΣΗ ΠΡΟΒΛΗΜΑΤΟΣ - ΒΛΑΒΗΣ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R003	RSD. ΚΑΤΑΧΩΡΗΣΗ ΟΡΩΝ ΛΕΙΤΟΥΡΓΙΑΣ LOCO	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R010	RSD. ΚΑΤΑΧΩΡΗΣΗ ΝΕΑΣ ΒΛΑΒΗΣ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T003	TUD. ΚΑΤΑΧΩΡΗΣΗ ΠΛΥΣΜΑΤΟΣ ΣΥΡΜΩΝ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T004	TUD. ΚΑΤΑΧΩΡΗΣΗ ΧΙΛΙΟΜΕΤΡΩΝ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T005	TUD. ΑΝΑΦΟΡΑ ΒΛΑΒΩΝ ΤΡΟΧΑΚΟΥ ΥΛΙΚΟΥ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T006	TUD. ΣΤΟΙΧΩ ΑΝΑΦΟΡΑΣ ΒΛΑΒΩΝ ΤΡ. ΥΛ.	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R016	RSD. ΚΑΤΑΧΩΡΗΣΗ ΚΑΜΠΑΝΙΑΣ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T001	TUD. ΜΕΤΑΚΙΝΗΣΗ - ΑΛΛΑΓΗ ΚΑΤΑΣΤΑΣΗΣ ΣΥΡΜΩΝ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T002	TUD. ΚΑΤΑΧΩΡΗΣΗ ΡΤΙ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T007	TUD. ΔΙΑΧΕΡΙΣΗ ΑΔΕΙΩΝ ΕΡΓΑΣΙΑΣ ΣΕ DEPOT	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	T008	TUD. ΕΠΙΛΗΛΟΦΟΡΑ TRAINING	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R025	RSD. ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΡΓΑΣΙΩΝ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R021	RSD. ΔΙΑΧΕΡΙΣΗ ΔΕΣΜΕΥΣΕΩΝ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R019	RSD. ΕΚΤΕΛΕΣΗ ΥΠΟΛΟΓΙΣΜΩΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ	1	READ/WRITE
SAA	ΣΑΚΕΛΛΑΡΗ Α...	R001	RSD. ΕΜΦΑΝΙΣΗ - ΔΙΟΡΕΞΙΣΗ ΑΝΑΦΟΡΑΣ ΒΛΑΒΗΣ ΣΥ...	1	READ/WRITE

**Εικόνα 27:** Πίνακας δικαιωμάτων χρηστών

#### 5.4. Αμαξοστάσια

Επιλέγοντας ο χρήστης από το Κύριο μενού “**Αμαξοστάσια**” τότε εμφανίζεται μια λίστα με όλα τα αμαξοστάσια της Γραμμής 1 (**Εικόνα 28**). Σε όσα αμαξοστάσια δεν έχει δημιουργηθεί ακόμα η απεικόνιση, τότε εμφανίζεται το **Μήνυμα « Υπό κατασκευή »**.



**Εικόνα 28:** Επιλογή Αμαξοστασίου



**Εικόνα 29:** Μήνυμα « Υπό κατασκευή »



The screenshot shows the 'Αποκρίσεις Τρένα' (Train Responses) window in the Train Simulator. The window displays a detailed view of the railway track layout, including station names like 'ΑΜΑΖΩΝΙΣΤΑΣ ΠΕΙΡΑΙΑ' and 'ΠΕΙΡΑΙΑΣ'. It shows the positions and movements of various trains, with their respective schedules and destinations. The interface includes a top menu bar, a central track diagram, and several data tables on the right side.

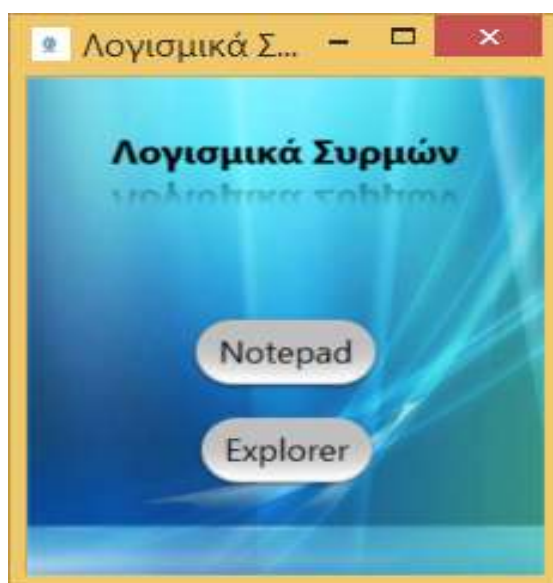
Η μετακίνηση των ημισυρμών επιτρέπεται μόνο από τροχιά σε τροχιά στο ίδιο αμαξοστάσιο και από οποιαδήποτε τροχιά προς ή από την Γραμμή 1. Δεν επιτρέπεται η κίνηση ενός ημισυρμού προς μια θέση που είναι ήδη κατειλημμένη παρά μόνο στην γραμμή. Επίσης, στην Γραμμή 1 μπορούν να είναι όλοι οι ημισυρμοί.



**Εικόνα 31:** Αλλαγή κατάσταση ημισυρμού

## 5.5. Λογισμικά Συρμών

Αν επιλέξει από το “**Κύριο Μενού**” το “**Λογισμικά Συρμών**”, τότε ο χρήστης μπορεί να επιλέξει κάποιες εφαρμογές όπως το Notepad ή τον Internet Explorer.



**Εικόνα 32:** Λογισμικά Συρμών



## **6. ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ**

### **6.1. ΣΥΜΠΕΡΑΣΜΑΤΑ**

Η παρούσα διπλωματική εργασία δημιουργήθηκε για να αποτυπωθεί η κατάσταση των συρμών χρησιμοποιώντας μια γλώσσα προγραμματισμού που είναι:

- εύκολη στη χρήση, στο γράψιμο, στη μεταγλώττιση, στην εύρεση σφαλμάτων,
- αντικειμενοστραφής, δημιουργεί προγράμματα με ιεραρχία και δυνατότητα επαναχρησιμοποίησης κώδικα,
- μη εξαρτώμενη από πλατφόρμα, δηλαδή μπορεί να μετακινηθεί η εφαρμογή από τον ένα υπολογιστή στον άλλον.

Επιπλέον, χρησιμοποιήθηκαν καλύτερα γραφικά έτσι ώστε να είναι η εφαρμογή πιο φιλική στον χρήστη, δηλαδή εύχρηστη και προσφέρεται για καλύτερη κατανόηση του στόλου των συρμών.

### **6.2. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ**

Η παρούσα εφαρμογή έχει αναπτυχθεί ώστε να παρουσιάζει ένα μικρό μέρος των δυνατοτήτων του Ημερολογίου του Τροχαίου Υλικού με δυνατότητα επέκτασης. Ακόμα, μπορεί να προστεθεί η δυνατότητα να ενώνονται οι ημισυρμοί μεταξύ τους με κάποιους περιορισμούς, εννοείται.

Επιπροσθέτως, η εφαρμογή έχει την δυνατότητα να επεκταθεί και ως προς τις υπόλοιπες γραμμές δηλαδή Γραμμή 2, 3 και TRAM.

Επιπλέον, είναι δυνατόν να υπάρχουν πιο αναλυτικές πληροφορίες σχετικά με την κατάσταση των συρμών, δηλαδή εκτός από τον χρωματισμό τους, θα υπάρχει η δυνατότητα, επιλέγοντας κάποιον συρμό, να βλέπει ο χρήστης:

- σε πιο υποσύστημα έχει βλάβη (στην περίπτωση που υπάρχει) ανοίγοντας ένα μεγαλύτερο παράθυρο που να δείχνει όλα τα υποσυστήματα του ημισυρμού με χρωματισμό για την κατάστασή του,
- αν υπάρχει κάποια διαφήμιση που τρέχει και πότε τελειώνει,
- τα τρέχοντα χιλιόμετρα του και
- ποια είναι η παρούσα ή/ και η επόμενη συντήρηση που πρέπει να περάσει ο ημισυρμός και σε πόσο χρόνο αναμένεται.

Θα μπορούσε επίσης να προστεθεί και η πρόβλεψη των μελλοντικών συντηρήσεων των συρμών βάσει των χιλιομέτρων τους.

Στην εφαρμογή θα μπορούσαν να ενσωματωθούν τα προγράμματα ανάγνωσης σφαλμάτων των συρμών και να γίνουν πιο εύχρηστα.

Επίσης, μπορούν να προστεθούν και άλλες δυνατότητες, όπως είναι η καταχώριση των υλικών που αλλάζονται στους συρμούς μέσω εργαλείων σάρωσης barcode πάνω στο υλικό. Επιπλέον, θα μπορεί ο χρήστης να επιλέγει μέσω εικόνων που θα δείχνουν τα μέρη του συρμού, πού να τοποθετηθεί το υλικό.

Μια ακόμα δυνατότητα θα μπορούσε να είναι η μεταφορά της εφαρμογής σε περιβάλλον Android. Εκεί οι χρήστες μέσω μιας φορητής συσκευής (tablet) την οποία θα μπορούν να την έχουν μαζί τους, θα μπορούν να βλέπουν την κατάσταση των ημισυρμών και να την ενημερώνουν άμεσα.

Η πιο σημαντική απ' όλες τις μελλοντικές επεκτάσεις θα ήταν η προσθήκη στο Ημερολόγιο Συρμών ενός διαγνωστικού εργαλείου για την ταχύτερη εύρεση βλαβών. Το σύστημα αυτό θα οδηγεί τους ρυθμιστές κυκλοφορίας με λογικά βήματα στην επίλυση της βλάβης που δημιουργήθηκε κατά τη διάρκεια του δρομολογίου, χωρίς να χάνεται χρόνος για απόσυρση του συρμού ή να χρειάζεται μεγαλύτερη διαθεσιμότητα συρμών. Θα χρησιμοποιείται επίσης και από το τεχνικό προσωπικό για την ταχύτερη εύρεση της βλάβης στους εκτός κυκλοφορίας συρμούς. Συγκεκριμένα, θα πρέπει να δημιουργηθούν δέντρα βλαβών που με κατάλληλες ερωτήσεις να φτάνει το προσωπικό στη λύση του προβλήματος.

## ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΑ

Ακρώνυμο	Περιγραφή
<b>SQL</b>	Structured Query Language
<b>CSS</b>	Cascading Style Sheet
<b>DBMS</b>	Database Management System
<b>JPQL</b>	Java Persistence Query Language
<b>JPA</b>	Java Persistence API
<b>JPQL</b>	Java Persistence Query Language
<b>JPA</b>	Java Persistence API
<b>JDBC</b>	Java DataBase Connectivity
<b>HQL</b>	Hibernate Query Language
<b>OOXML</b>	Office Open XML
<b>HSSF</b>	Horrible SpreadSheet Format
<b>XSSF</b>	XML SpreadSheet Format
<b>ATS</b>	Automatic Train Supervision
<b>T.Y.</b>	Τροχαίο Υλικό

## ΠΑΡΑΡΤΗΜΑ II

### ΔΟΜΗ ΕΦΑΡΜΟΓΗΣ

Παρακάτω φαίνεται η δομή της εφαρμογής σε **Java, JavaFX**:

➤ **MetroAthens**

- **Source Packages**

- **<default package>**
  - Log4properties
- **AllFrames**
  - DepotHsapPeiraias.java
  - Locations.java
  - UnitComponentSQL.java
- **AllFrames.components**
  - ChangeUnitColor.java
  - ColorConverter.java
  - ColorRenderer.java
  - DepotImage.java
  - StatusFontColorConverter.java
- **AllFrames.fx**
  - AccessControl.java
  - ChooserFX.java
  - LoginScreenFX.java
  - SoftwareFX.java
  - TasksFX.java
- **AllFrames.fx.tables**
  - TableAllUsersAccessRights.java
  - TableCleaning.java
  - TableFailures.java
  - TableKMS.java
  - TablePrevMaintenance.java

- TableUserRights.java
- **AllFrames.fx.tables.utils**
  - ColoredStatusTrainCell.java
  - EditCell.java
  - ExportExcel.java
  - ModifyCell.java
  - Status.java
  - colorAWTConvJavaFX.java
- **AllFrames.fx.tables.utils.TableFilters**
  - FilterColumns.java
  - FilterPopUpWithDatesRemarks.fxml
  - FilterPopUpWithDatesRemarks.java
  - TableFilter.java
- **AllFrames.fx.usf**
  - ChangeUnitState.java
  - DateTimePicker.java
  - DropDownErgasies.java
  - EditCleaning.java
  - EditKms.java
  - EditPrevMaintenance.java
  - MenuBarTasks.java
  - MenuFailures.java
  - MenuMaintenance.java
  - MenuState.java
  - MenuUser.java
  - NewCleaning.java
  - NewFaillure.java
  - NewKMS.java
  - NewPrevMaintenance.java
  - NewUser.java
  - PopUpWarning.java
  - RestoreEditFaillureClass.java

- **AllFrames.fx.usf.css**
  - FaillureCSS.css
  - LoginScreenCSS.css
  - MaintenanceCSS.css
  - StateCSS.css
  - UserCSS.css
  - popUpWarning.css
  - tableCSS.css
  - tstColors.css
  - windowsCSS.css
- **AllFrames.fx.usf.css.TableFilters**
  - filteredtable.css
  - historyFileChooser.css
- **META-INF**
  - Persistence.xml
- **SQLConnection.java**
  - MySQLConnection.java
  - QueryActiveUnits.java
  - QueryFleetStatus.java
  - QueryLastUnitMoves.java
  - QueryOccupiedLocationsAndStatus.java
- **SQLEntities**
  - TmFleetStatus.java
  - TmFleetSymptoms.java
  - TmLocations.java
  - TmPrevMaintenance.java
  - TmPrevMaintenanceStatus.java
  - TmTrain.java
  - TmTrainFontStatuses.java
  - TmTrainStatuses.java
  - TmTrainUnits.java

- TmTrainUnitsMoves.java
- TmTrainUnitsPK.java
- TmTrainWashlog.java
- TmUnitkms.java
- TmUnits.java
- TmUsers.java
- TmWashStatus.java
- UserAccessMode.java
- UserAccessModeAsl.java
- **SQLTables.JPAControllers**
  - TmFleetStatusJpaController.java
  - TmFleetSymptomsJpaController.java
  - TmLocationJpaController.java
  - TmPrevMaintenanceJpaController.java
  - TmPrevMaintenanceStatusJpaController.java
  - TmTrainFontStatusesJpaController.java
  - TmTrainJpaController.java
  - TmTrainStatusesJpaController.java
  - TmTrainUnitsJpaController.java
  - TmTrainUnitsMovesJpaController.java
  - TmTrainWashlogJpaController.java
  - TmUnitkmsJpaController.java
  - TmUnitsJpaController.java
  - TmUsersJpaController.java
  - TmWashStatusJpaController.java
  - UserAccessModeJpaController.java
- **SQLEntities.JPAControllers.exceptions**
  - IllegalOrphanException.java
  - NonexistentEntityException.java
  - PreexistingEntityException.java
- **images**
  - 3d-warning.jpg



- Eject\_Blue\_Button.jpg
- Excel.png
- HSAPDepot\_v4.jpg
- History.png
- Piraeus\_ISAP\_terminus\_building\_2.jpg
- Correction.png
- failure.png
- filtered.png
- hsap\_Piraeus\_station.jpg
- hsapold.jpg
- isapLOGO.png
- logging\_image.jpg
- metal-warning.jpg
- newUserBackground.jpg
- newUserBackground2.jpg
- newUserIcon.jpg
- not\_filtered.png
- odometer.jpg
- odometer\_icon.jpg
- rails.jpg
- restore-button.jpg
- sky.jpg
- speedometer.jpg
- stasyLOGO.jpg
- train\_blur.jpg
- under-maintenance.jpg
- washingplant.jpg

Η δομή της βάσης δεδομένων MySQL είναι η παρακάτω:

- metroathensdb
  - Tables
    - tm\_fleet\_status
    - tm\_fleet\_symptoms
    - tm\_locations
    - tm\_prev\_maintenance
    - tm\_prev\_maintenace\_status
    - tm\_train\_font\_statuses
    - tm\_train\_statuses
    - tm\_train\_units
    - tm\_train\_units\_moves
    - tm\_train\_washlog
    - tm\_unitkms
    - tm\_users
    - tm\_wash\_status
    - user\_access\_mode
    - user\_access\_mode\_asl

## ΑΝΑΦΟΡΕΣ

- [ 1 ] Introduction to Java Programming Tenth Edition by Daniel Liang
- [ 2 ] UML for Java Programmers by Robert Cecil Martin
- [ 3 ] JavaFX Essentials By Mohamed Taman
- [ 4 ] [www.oracle.com](http://www.oracle.com)
- [ 5 ] <https://netbeans.org/community/>
- [ 6 ] <https://www.w3.org>
- [ 7 ] [wikipedia.org](http://wikipedia.org)
- [ 8 ] [poi.apache.org](http://poi.apache.org)
- [ 9 ] [thoughts-on-java.org](http://thoughts-on-java.org)
- [ 10 ] <https://eclass.uoa.gr> (Ανάπτυξη Λογισμικού για Δίκτυα και Τηλεπικοινωνίες U. of Crete, Information Systems Analysis and Design by Yannis Tzitzikas, Fall 2005)
- [ 11 ] <https://www.uml-diagrams.org>
- [ 12 ] [en.wikipedia.org](http://en.wikipedia.org)
- [ 13 ] <https://etrenddot.wordpress.com/2017/03/19/javafx-8-a-fantastic-tool-to-create-beautiful-gui-for-your-java-application-easily-part-1>
- [ 14 ] <https://www.baeldung.com/>
- [ 15 ] <https://stackoverflow.com>
- [ 16 ] <https://www.railjournal.com/>