



Μηχανική Λογισμικού I  
Τομέας Ηλεκτρονικής και Υπολογιστών  
Τμήμα HMMY  
Α.Π.Θ

8<sup>ο</sup> Εξάμηνο  
Άνοιξη 2023



# Team Tactix

**Where Speed Meets Community!**

## Απαιτήσεις Χρηστών και Προδιαγραφές Συστήματος

**Version 0.1**

Παναγιώτης Καρβουνάρης [panakarv@ece.auth.gr](mailto:panakarv@ece.auth.gr)

Έλενα Μαχμούτάι [elenmach@ece.auth.gr](mailto:elenmach@ece.auth.gr)

Παναγιώτης Σιταρίδης [psitarid@ece.auth.gr](mailto:psitarid@ece.auth.gr)

Άννα Τσιτσάνου [tsitsanou@ece.auth.gr](mailto:tsitsanou@ece.auth.gr)

**07/05/2023**



## Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Α. Συμεωνίδης	*	<a href="mailto:asymeon@issel.ee.auth.gr">asymeon@issel.ee.auth.gr</a>
Π. Σιταρίδης	40	<a href="mailto:psitarid@ece.auth.gr">psitarid@ece.auth.gr</a>
Π. Καρβουνάρης	40	<a href="mailto:panakarv@ece.auth.gr">panakarv@ece.auth.gr</a>
Ε. Μαχμουτάτη	40	<a href="mailto:elenmach@ece.auth.gr">elenmach@ece.auth.gr</a>
Α. Τσιτσάνου	40	<a href="mailto:tsitsanou@ece.auth.gr">tsitsanou@ece.auth.gr</a>



<b>1 Απαιτήσεις Συστήματος</b>	<b>5</b>
1.1 Λειτουργικές απαιτήσεις (Σενάρια χρήσης)	5
1.2 Χρήστες και εξωτερικά συστήματα	8
1.2.1 Unassigned User	8
1.2.2 Assigned User	8
1.2.3 Hands-On Engineer	8
1.2.4 Chief Engineer	8
1.2.5 Engineer	8
1.2.6 TeamTactixDatabase	8
1.2.7 GoogleDatabase	8
1.3 Σημαντικές Μη Λειτουργικές Απαιτήσεις	9
1.3.1 Απαιτήσεις επίδοσης	9
1.3.2 Απαιτήσεις ασφάλειας (Security) και πολιτικής (Policy - legal, corporate)	9
1.3.3 Απαιτήσεις Συμβατότητας (Compatibility)	9
1.3.4 Απαιτήσεις Αξιοπιστίας (Reliability)	9
1.3.5 Απαιτήσεις Χρηστικότητας (Usability)	10
1.4 Λεξικογραφικοί Προσδιορισμοί	10
<b>2 Σενάρια Χρήσης</b>	<b>12</b>
2.1 Διάγραμμα σεναρίων χρήσης	12
2.2 Role Assignment (Feature με βάση την ορολογία Gherkin)	12
2.3 Edit personal profile (Feature με βάση την ορολογία Gherkin)	13
2.4 Send Message to Chat (Feature με βάση την ορολογία Gherkin)	14
2.5 Accept or Decline a proposal (Feature με βάση την ορολογία Gherkin)	15
2.6 Select and View Telemetry Data Chart (Feature με βάση την ορολογία Gherkin)	15
2.7 View Weather Report (Feature με βάση την ορολογία Gherkin)	16
2.8 Make Proposal (Feature με βάση την ορολογία Gherkin)	17
2.9 Mark Proposal as DONE (Feature με βάση την ορολογία Gherkin)	18
2.10 View Vehicle SetUp (Feature με βάση την ορολογία Gherkin)	19
2.11 Modify Vehicle Setup Values (Feature με βάση την ορολογία Gherkin)	19
2.12 Send Telemetry Data Charts on Chat (Feature με βάση την ορολογία Gherkin)	20
2.13 Add New Elements To Vehicle Setup(Feature με βάση την ορολογία Gherkin)	21
2.14 Zoom in the Telemetry Data Chart (Feature με βάση την ορολογία Gherkin)	22
2.15 Create Chat Room(Feature με βάση την ορολογία Gherkin)	22
2.16 Upload Data from Sensors(Feature με βάση την ορολογία Gherkin)	24
<b>3 Επιδεικτικά γραφικά παράθυρα διεπαφής</b>	<b>25</b>
3.1 Homepage	25
3.2 Admin Panel - Chief Engineer	26
3.3 Proposals - Chief Engineer	27
3.4 Proposals - Hands-On Engineers	27



3.5 Make Proposal - Engineer	28
3.6 Vehicle Setup	28
3.7 Chat	30
3.8 Personal Profile	31
3.9 Charts	31
3.10 Weather	33
<b>4 Στατική Μοντελοποίηση</b>	<b>34</b>
4.1 <Πακέτα λεξιλογίου σεναρίων υψηλής προτεραιότητας>	35
4.1.1 Πακέτο User Handle Package	35
4.1.2 Πακέτο Admin Panel	39
4.1.3 Πακέτο Proposal Package	42
4.1.4 Πακέτο Chart	47
4.1.5 Πακέτο Vehicle Setup Package	54
4.2 <Πακέτο λεξιλογίου σεναρίων μέσης προτεραιότητας>	61
4.2.1 Πακέτο Chat	61
4.3 <Πακέτα λεξιλογίου σεναρίων χαμηλής προτεραιότητας >	67
4.3.1 Πακέτο Personal Profile	67
4.3.2 Πακέτο Weather Package	70
<b>Παράρτημα I – Γλωσσάριο</b>	<b>75</b>

## Λίστα Σχημάτων

Figure 1: Home Page των χρηστών	25
Figure 2: Admin panel του χρήστη Chief Engineer	26
Figure 3: Proposals page του χρήστη Chief Engineer	27
Figure 4: Proposals Page του χρήστη Hands-on Engineer	27
Figure 5: Proposal Page του χρήστη Engineer	28
Figure 6: Vehicle setup	28
Figure 7: Σελίδες chat όλων των χρηστών	30
Figure 8: Προσωπικό προφίλ κάθε χρήστη	31
Figure 9: Σελίδες chart όλων των χρηστών	31
Figure 10: Σελίδα Weather Report των χρηστών	33



# 1 Απαιτήσεις Συστήματος

## 1.1 Λειτουργικές απαιτήσεις (Σενάρια χρήσης)

### <ΛΑ- 1>

Ο Chief Engineer πρέπει να μπορεί να αναθέτει ρόλους.

**Περιγραφή:** Ο Chief Engineer θα πρέπει μέσω της οθόνης “control panel” να βλέπει μια λίστα με τους εγγεγραμμένους χρήστες και τους ρόλους και να μπορεί να τους αλλάξει.

**User Priority: 5/5** Είναι πολύ σημαντικό για τον χρήστη ώστε να έχει πρόσβαση στις διαφορετικές λειτουργίες κάθε ρόλου.

**Technical Priority: 5/5** Είναι πολύ σημαντικό για το σύστημα ώστε να δίνει στον κάθε χρήστη τις ανάλογες δυνατότητες.

### <ΛΑ- 2>

Όλοι οι χρήστες πρέπει να μπορούν να επεξεργάζονται το προσωπικό τους προφίλ.

**Περιγραφή:** Όλοι οι χρήστες θα πρέπει να μπορούν να επεξεργάζονται την εικόνα προφίλ, το όνομα χρήστη, τη σύντομη περιγραφή, σε ποιο department ανήκουν και τα social media τους.

**User Priority: 2/5** Είναι σημαντικό για το χρήστη γιατί μπορεί να κάνει πιο προσωπικό και ευχάριστο το προφίλ του και να ξεχωρίζει από τους υπόλοιπους χρήστες.

**Technical Priority: 1/5** Δεν είναι σημαντικό για το σύστημα καθώς μπορεί να λειτουργήσει κανονικά και χωρίς αυτή.

### <ΛΑ- 3>

Όλοι οι εγγεγραμμένοι χρήστες πρέπει να μπορούν να στέλνουν μήνυμα στο chat.

**Περιγραφή:** Το σύστημα πρέπει να επιτρέπει στους χρήστες την αλληλεπίδραση μέσω μηνυμάτων στο chat.

**User Priority: 3/5:** Είναι σημαντικό διότι ο χρήστης μπορεί να επικοινωνεί, να ενημερώνεται άμεσα και να γίνεται πιο αποτελεσματικός.

**Technical Priority: 3/5** Είναι σχετικά σημαντικό διότι η άμεση επικοινωνία δεν είναι απαραίτητη για την υπόλοιπη λειτουργικότητα του συστήματος.

### <ΛΑ- 4>

Ο Chief Engineer πρέπει να μπορεί να αποδέχεται ή να απορρίπτει προτάσεις.

**Περιγραφή:** Ο Chief Engineer θα πρέπει να μπορεί να βλέπει τις προτάσεις που πραγματοποιούνται από τους Engineers και να τις απορρίπτει ή να τις δέχεται αναλόγως με το αν πιστεύει ότι θα ωφελήσουν την ομάδα και θα βελτιώσουν το όχημα.

**User Priority: 5/5** Είναι σημαντικό για τον χρήστη γιατί του δίνει την δυνατότητα να ελέγχει ποιές αλλαγές θα πραγματοποιούνται στο όχημα και να ενημερώνει τους υπόλοιπους χρήστες κατάλληλα.

**Technical Priority: 5/5** Είναι πολύ σημαντικό διότι εξαρτώνται και άλλες σημαντικές λειτουργίες του συστήματος από αυτήν.

### <ΛΑ- 5>

Όλοι οι assigned users πρέπει να μπορούν να δουν Data Charts.

**Περιγραφή:** Όλοι οι assigned users μπορούν να δουν τα δεδομένα τηλεμετρίας του οχήματος, μέσω διαφόρων διαγραμμάτων Data Charts, μπορούν να επιλέξουν συγκεκριμένη ημερομηνία, πίστα, γύρο και αισθητήρα, ώστε να έχουν πλήρη επίγνωση της κατάστασης του οχήματος σε όλη την διάρκεια του session.

**User Priority: 4/5** Είναι αρκετά σημαντικό για τον χρήστη ώστε να έχει επίγνωση της κατάστασης του οχήματος από τα δεδομένα που στέλνουν οι αισθητήρες, ώστε να βελτιώνεται οι ποιότητα των αποφάσεών τους.



**Technical Priority: 3/5** Είναι σημαντικό για το σύστημα ώστε να δίνει στον χρήστη διάφορες πληροφορίες για την κατάσταση του οχήματος.

#### <ΛΑ- 6>

Όλοι οι εγγεγραμμένοι χρήστες πρέπει να μπορούν να δουν τον καιρό.

**Περιγραφή:** Το σύστημα πρέπει να είναι ικανό να δώσει πληροφορίες σχετικά με τις καιρικές συνθήκες όπως θερμοκρασία, υγρασία και ταχύτητα του ανέμου σε κάθε εγγεγραμμένο χρήστη.

**User Priority: 2/5:** Είναι σχετικά σημαντικό, διότι είναι καλό ο χρήστης να γνωρίζει τις καιρικές συνθήκες σε περίπτωση έκθεσης του οχήματος σε εξωτερικό χώρο, για την ασφάλειά του.

**Technical Priority: 1/5:** Δεν είναι ιδιαίτερα σημαντικό για το σύστημα, καθώς η υπόλοιπη λειτουργικότητά του δεν επηρεάζεται από τις καιρικές συνθήκες.

#### <ΛΑ- 7>

Ο Engineer πρέπει να μπορεί να κάνει προτάσεις.

**Περιγραφή:** Ο Engineer θα πρέπει να μπορεί να δημιουργήσει προτάσεις για αλλαγή στο setup το οχήματος προς τον Chief Engineer. Θα πρέπει να μπορεί να επιλέξει το συγκεκριμένο μέρος ενός υποσυστήματος του συστήματος που επιθυμεί και να εισάγει την αλλαγή που θέλει στην επιλεγμένη μονάδα μέτρησης καθώς και να προσθέσει τον λόγο και μια μικρή περιγραφή που τον οδήγησε στην απόφαση αυτή.

**User Priority: 5/5** Είναι πολύ σημαντικό για το χρήστη γιατί με αυτόν τον τρόπο ενημερώνει τον Chief Engineer για την αλλαγή που πιστεύει ότι πρέπει να γίνει στο setup του οχήματος.

**Technical Priority: 4/5** Οι προτάσεις που δημιουργούνται δεν είναι απαραίτητες για τη λειτουργία του συστήματος.

#### <ΛΑ- 8>

Οι hands-on engineers πρέπει να μπορούν να αλλάξουν το status της πρότασης σε DONE.

**Περιγραφή:** οι Hands-On Engineers θα πρέπει να μπορούν να αλλάξουν το status της πρότασης σε DONE όταν ολοκληρώσουν τις αλλαγές που προτάθηκαν για το όχημα.

**User Priority: 5/5** Είναι πολύ σημαντικό για το χρήστη γιατί με αυτή τη λειτουργία μπορεί να γνωρίζει ανά πάσα στιγμή ποιες αλλαγές έχουν ολοκληρωθεί και ποιες όχι.

**Technical Priority: 5/5** Είναι αρκετά σημαντικό για τη λειτουργία του συστήματος καθώς συνδέεται και με άλλες λειτουργίες.

#### <ΛΑ- 9>

Όλοι οι assigned users πρέπει να μπορούν να δουν το vehicle setup.

**Περιγραφή:** Όλοι οι assigned users μπορούν να δουν το τρέχον vehicle setup, δηλαδή διάφορες πληροφορίες του οχήματος.

**User Priority: 5/5** Είναι πολύ σημαντικό για τον χρήστη ώστε να έχει πρόσβαση στις διάφορες πληροφορίες του οχήματος.

**Technical Priority: 5/5** Είναι πολύ σημαντικό για το σύστημα ώστε να δίνει στον κάθε χρήστη τις διάφορες πληροφορίες του οχήματος.

#### <ΛΑ- 10>

Ο Chief Engineer πρέπει να μπορεί να τροποποιήσει τις τιμές του setup του οχήματος

**Περιγραφή:** Ο Chief Engineer πρέπει να μπορεί να επιλέξει το συγκεκριμένο part του υποσυστήματος του συστήματος που επιθυμεί και να τροποποιήσει την τιμή του καθώς και την μονάδα μέτρησης του.

**User Priority: 5/5** Είναι πολύ σημαντικό για το χρήστη γιατί ο ίδιος θέλει να βλέπει το ενημερωμένο setup του οχήματος ανα πάσα στιγμή.



**Technical Priority: 3/5** Το setup του οχήματος δεν είναι απαραίτητα για τη λειτουργία του συστήματος.

#### <ΛΑ- 11>

Όλοι οι assigned users πρέπει να μπορούν στείλουν Data Charts.

**Περιγραφή:** Όλοι οι Assigned Users μπορούν στείλουν συγκεκριμένα διαγράμματα, ώστε να ενημερώνουν τους υπόλοιπους και να βελτιώνουν την απόδοση της ομάδας.

**User Priority: 5/5** Είναι πολύ σημαντικό για τον χρήστη ώστε να μοιράζεται γραφικά δεδομένα και να βελτιώνει την επικοινωνία με τους συναδέλφους του.

**Technical Priority: 5/5** Είναι πολύ σημαντικό για το σύστημα ώστε να δίνει στον χρήστη την δυνατότητα καλύτερης επικοινωνίας.

#### <ΛΑ- 12>

Ο chief engineer πρέπει να μπορεί να προσθέσει νέα στοιχεία στο Vehicle setup.

**Περιγραφή:** Ο Chief Engineer μπορεί να προσθέσει parts ή subsystem ή systems στο vehicle setup.

**User Priority: 4/5** Είναι αρκετά σημαντικό για τον χρήστη ώστε να μπορεί να προσθέσει καινούργια μέλη στο vehicle setup ανά πάσα στιγμή.

**Technical Priority: 4/5** Είναι σημαντικό για το σύστημα επειδή προσφέρει μεγάλη ευελιξία ώστε να χρησιμοποιηθεί η εφαρμογή για διαφορετικά οχήματα.

#### <ΛΑ- 13>

Όλοι οι assigned users πρέπει να μπορούν να κάνουν zoom στα Data Charts.

**Περιγραφή:** Όλοι οι assigned users μπορούν να δουν συγκεκριμένα χρονικά διαστήματα (format: mm,ss) των charts ώστε να βλέπουν λεπτομερώς τις τιμές των αισθητήρων και βελτιώνουν τις αποφάσεις τους.

**User Priority: 4/5** Είναι σημαντικό για τον χρήστη ώστε να έχει καλύτερη επίγνωση της κατάστασης του οχήματος από τα δεδομένα που στέλνουν οι αισθητήρες, καθώς τα βλέπει με μεγαλύτερη λεπτομέρεια.

**Technical Priority: 3/5** Είναι σημαντικό για το σύστημα ώστε να δίνει στον χρήστη διάφορες πληροφορίες, με μεγάλη λεπτομέρεια για την κατάσταση του συστήματος οχήματος.

#### <ΛΑ- 14>

Όλοι οι εγγεγραμμένοι χρήστες πρέπει να μπορούν να δημιουργήσουν chat rooms.

**Περιγραφή:** Το σύστημα πρέπει να επιτρέπει σε όλους τους εγγεγραμμένους χρήστες να δημιουργήσουν chat rooms με σκοπό την κατάλληλη οργάνωση της επικοινωνίας.

**User Priority: 3/5:** Είναι σχετικά σημαντικό καθώς χρειάζεται να φίλτραρονται οι παραλήπτες των μηνυμάτων σε κάθε περίπτωση.

**Technical Priority: 4/5:** Δεν είναι ιδιαίτερα σημαντικό για το σύστημα, καθώς μπορεί να είναι λειτουργικό και χωρίς την δημιουργία chat rooms.

#### <ΛΑ- 15>

Οι Engineers πρέπει να μπορούν να ανεβάσουν αρχείο που περιέχει δεδομένα από τους αισθητήρες.

**Περιγραφή:** Οι Engineers θα πρέπει να μπορούν να ανεβάσουν αρχείο με δεδομένα που προσκομίζουν από τους αισθητήρες ώστε να είναι ορατά από τους υπόλοιπους χρήστες και να χρησιμοποιηθούν.

**User Priority: 5/5** Είναι πολύ σημαντικό για το χρήστη γιατί με αυτή τη λειτουργία μπορεί να ανεβάζει νέα δεδομένα και να κρατά ενημερωμένους τους χρήστες για την κατάσταση του οχήματος.

**Technical Priority: 5/5** Είναι αρκετά σημαντικό για τη λειτουργία του συστήματος καθώς συνδέεται και με άλλες λειτουργίες.



## 1.2 Χρήστες και εξωτερικά συστήματα

Η συγκεκριμένη εφαρμογή στοχεύει στην εξυπηρέτηση ομάδων που ασχολούνται με τον μηχανοκίνητο αθλητισμό. Χρησιμοποιείται για την γρήγορη και λεπτομερή επικοινωνία των μηχανικών της ομάδας με την χρήση μηνυμάτων και διαγραμμάτων. Η βασικότερη λειτουργία που προσφέρει, απευθύνεται στην εύκολη και γρήγορη λήψη αποφάσεων μέσω ενός συστήματος που μεταβιβάζει προτάσεις για αλλαγές από τους Engineers στον Chief Engineer και έπειτα από έγκριση του τελευταίου στους Hand-On Engineers, οι οποίοι πραγματοποιούν τις τροποποιήσεις. Αυτές μπορούν να γίνουν σε διάφορα κομμάτια του οχήματος, ανάλογα με τις προτάσεις των engineers, οι οποίοι βλέπουν τα δεδομένα από τους αισθητήρες μέσω τηλεμετρίας.

### 1.2.1 Unassigned User

Κάθε χρήστης που διαθέτει λογαριασμό αλλά δεν του έχει ανατεθεί ακόμα ρόλος από τον Chief Engineer. Η μοναδική λειτουργία στην οποία έχει πρόσβαση είναι η τροποποίηση του προφίλ του.

### 1.2.2 Assigned User

Κάθε χρήστης που διαθέτει λογαριασμό ενώ του έχει ανατεθεί ρόλος από τον Chief Engineer και έχει πρόσβαση στις βασικές λειτουργίες της εφαρμογής, δηλαδή χρήση του chat, προβολή weather report, πρόσβαση στα δεδομένα του Vehicle Setup και στα δεδομένα της τηλεμετρίας μέσω διαφόρων διαγραμμάτων.

### 1.2.3 Hands-On Engineer

Μία κατηγορία χρήστη που μπορεί να λαμβάνει προτάσεις για τροποποιήσεις στο όχημα, εγκεκριμένες από τον Chief Engineer. Μόλις διεκπεραιωθεί η εφαρμογή τους μπορεί με το κουμπί “Done” να γνωστοποιήσει την επιτυχή τροποποίηση στο όχημα.

### 1.2.4 Chief Engineer

Ως επικεφαλής μηχανικός έχει εξουσιοδότηση για την ανάθεση ρόλων στους υπόλοιπους χρήστες. Στο Vehicle Setup είναι ο μόνος να προσθέσει καινούργια System, Subsystem και Part στο Vehicle, καθώς και να τροποποιήσει τις τιμές τους. Ακόμη, μπορεί να αποδέχεται ή να απορρίπτει τις προτάσεις που του στέλνουν οι Engineers.

### 1.2.5 Engineer

Μία κατηγορία χρήστη που βλέπει τα δεδομένα της τηλεμετρίας και κάνει προτάσεις για βελτίωση του οχήματος προς τον Chief Engineer. Υποστηρίζει αυτές τις προτάσεις με περιγραφές και με διαγράμματα.

### 1.2.6 TeamTactixDatabase

Ένα αναγκαίο εξωτερικό σύστημα για την ύπαρξη της συγκεκριμένης εφαρμογής αποτελεί μία βάση δεδομένων, η οποία διατηρεί όλα τα στοιχεία των χρηστών και του οχήματος, τα δεδομένα της τηλεμετρίας και τα μηνύματα του Chat.

### 1.2.7 GoogleDatabase

Ένα εξωτερικό σύστημα ως βάση δεδομένων που εξυπηρετεί την λήψη δεδομένων που αφορούν το δελτίο καιρού.

## 1.3 Σημαντικές Μη Λειτουργικές Απαιτήσεις

### 1.3.1 Απαιτήσεις επίδοσης

<ΜΛΑ- 1>



Το σύστημα θα πρέπει να έχει χρόνο απόκρισης λίγα ms.

**Περιγραφή:** Το σύστημα πρέπει να σχεδιαστεί για να παρέχει μια εμπειρία χρήστη με μέσο χρόνο απόκρισης μόλις μερικών ms. Αυτή η απαίτηση ισχύει για όλες τις λειτουργίες του συστήματος, συμπεριλαμβανομένης της ανάκτησης δεδομένων, της επεξεργασίας και των αλληλεπιδράσεων διεπαφής χρήστη.

**User Priority: 5/5:** Είναι πολύ σημαντικό για τον χρήστη καθώς η εφαρμογή έχει σκοπό να χρησιμοποιείται για αγωνιστικό και για επαγγελματικό σκοπό όπου η φύση του αθλήματος είναι τέτοια ώστε κάθε κλάσμα δευτερολέπτου να μετράει για το αποτέλεσμα.

**Technical Priority: 4/5:** Το να μπορεί το σύστημα να ανταποκρίνεται σε πολύ σύντομο χρονικό διάστημα είναι πολύ σημαντικό για την ομαλή λειτουργία της εφαρμογής.

### 1.3.2 Απαιτήσεις ασφάλειας (Security) και πολιτικής (Policy - legal, corporate)

#### <ΜΛΑ- 2>

Το σύστημα θα πρέπει να ικανοποιεί τους περιορισμούς ασφαλείας που θέτει το πρωτόκολλο GDPR.

**Περιγραφή:** Με βάση την τρέχουσα Ευρωπαϊκή νομοθεσία θα πρέπει το σύστημα να είναι κατάλληλα σχεδιασμένο ώστε να τηρούνται οι κανονισμοί που τίθενται από το πρωτόκολλο GDPR.

**User Priority: 5/5:** Είναι πολύ σημαντικό για τον χρήστη καθώς αφορά την προστασία των προσωπικών του δεδομένων.

**Technical Priority: 5/5:** Είναι πολύ σημαντικό για το σύστημα για να αποφευχθούν περιπτώσεις υποκλοπής προστασίας προσωπικών δεδομένων.

### 1.3.3 Απαιτήσεις Συμβατότητας (Compatibility)

#### <ΜΛΑ- 3>

Το σύστημα θα πρέπει να είναι συμβατό με πολλαπλές πλατφόρμες.

Όπως το γράφετε στην Περιγραφή, υποδηλωνει πως υποστηρίζετε και άλλες πέρα από αυτές. Εδώ θα πρέπει να είστε ακριβείς. Σκεφτείτε, αν εσείς προγραμματίζατε το σύστημα με βάση αυτή την απαίτηση, θα ξέρατε ποιες είναι οι πλατφόρμες με τις οποίες πρέπει να είστε συμβατοί;

**Περιγραφή:** Η εφαρμογή πρέπει να έχει σχεδιαστεί ώστε να είναι συμβατή με πολλές πλατφόρμες, πράγμα που σημαίνει ότι θα πρέπει να μπορεί να εκτελείται σε πολλαπλά λειτουργικά συστήματα και τύπους συσκευών, συμπεριλαμβανομένων, ενδεικτικά, των Windows, macOS, iOS και Android. Επιπλέον, η εφαρμογή πρέπει να είναι συμβατή με διάφορες εκδόσεις αυτών των λειτουργικών συστημάτων, όπως Windows (7 έως πιο πρόσφατα), macOS Big Sur, iOS (12 εως και πιο πρόσφατη) και Android (από 10 έως πιο πρόσφατη).

**User Priority: 4/5:** Η συμβατότητα είναι σημαντική διότι επιτρέπει στο χρήστη να χρησιμοποιεί την εφαρμογή σε όλες του τις συσκευές ανα πάσα στιγμή. Επίσης σε όσο περισσότερες συσκευές και εκδόσεις είναι διαθέσιμη εφαρμογή, τόσο περισσότεροι χρήστες έχουν πρόσβαση σε αυτή.

**Technical Priority: 5/5:** Η συμβατότητα είναι σημαντική αφενός για θέματα εμφάνισης και αφετέρου για θέματα σωστής λειτουργίας της εφαρμογής.

### 1.3.4 Απαιτήσεις Αξιοπιστίας (Reliability)

#### <ΜΛΑ- 4>

Το σύστημα θα πρέπει να είναι συμβατό με την εφαρμογή Google Weather.

**Περιγραφή:** Το σύστημα θα πρέπει να υποστηρίζει την υπηρεσία Google Weather καθώς στην εφαρμογή θα χρησιμοποιούνται τα δεδομένα του καιρού από την υπηρεσία της Google.

**User Priority: 4/5:** Είναι αρκετά σημαντικό για τον χρήστη καθώς η ανά πάσα στιγμή σωστή γνώση των καιρικών συνθηκών όπως η υγρασία, ταχύτητα αέρα είναι πολύ σημαντική κατά τη διάρκεια του αγώνα.

**Technical Priority: 1/5:** Δεν είναι σημαντικό για την λειτουργία του συστήματος.

### 1.3.5 Απαιτήσεις Χρηστικότητας (Usability)

#### <ΜΛΑ- 5>



Το γραφικό περιβάλλον χρήστη του συστήματος πρέπει να είναι φιλικό προς τον χρήστη.

**Περιγραφή:** Το σύστημα πρέπει να έχει εύκολη πλούγηση, προβολή προφίλ, εύκολη επεξεργασία, εύκολους συνδέσμους μεταβίβασης από μια σελίδα σε άλλη κλπ.

**User Priority: 4/5:** Η ευχρηστία καθιστά ένα από τους σημαντικότερους παράγοντες ικανοποίησης των χρηστών.

**Technical Priority: 2/5:** Δεν είναι απαραίτητο για την ορθή λειτουργία της εφαρμογής.

## 1.4 Λεξικογραφικοί Προσδιορισμοί

**Χρήστης:** User. Το άτομο το οποίο χρησιμοποιεί τις βασικές λειτουργίες της υπό ανάπτυξη εφαρμογής.

**Εγγεγραμμένος χρήστης:** Registered user. Ο χρήστης ο οποίος είναι εγγεγραμμένος στη βάση δεδομένων της εφαρμογής. Μπορεί να χρησιμοποιήσει τις επιπρόσθετες λειτουργίες της εφαρμογής, αναλόγως με τον ρόλο που έχει.

**Σύστημα:** Ένα σύνολο, τα στοιχεία του οποίου αποτελούν οντότητες που έχουν το χαρακτηριστικό ότι κάθε μία από αυτές αλληλεπιδρά ή σχετίζεται με τουλάχιστον μίαν άλλη από το ίδιο σύνολο, έτσι ώστε να επιτυγχάνεται η ομαλή λειτουργία της εφαρμογής.

**Engineer:** Μηχανικός. Ένα άτομο που είναι υπεύθυνο για να προτείνει λύσεις και τροποποιήσεις στον Chief Engineer για το όχημα, ώστε να αυξηθεί οι απόδοση του. Οι λύσεις αυτές προκύπτουν με βάση την ανάλυση των δεδομένων που κάνει μέσω των διαγραμμάτων που υπάρχουν στην εφαρμογή.

**Chief Engineer:** Αρχιμηχανικός. Είναι ένας βασικός τεχνικός ρόλος υπεύθυνος για το σχεδιασμό, την ανάπτυξη, τη βελτιστοποίηση και τη λήψη των σημαντικών αποφάσεων των αγωνιστικών οχημάτων. Ο Αρχιμηχανικός είναι συνήθως μέρος μιας μεγαλύτερης ομάδας μηχανικών και είναι συχνά ένας πολύ έμπειρος και εξειδικευμένος μηχανικός που έχει εξειδικευμένες γνώσεις στη μηχανική μηχανοκίνητου αθλητισμού.

**Hands on Engineer:** Μηχανικός Πεδίου. Το άτομο που είναι υπεύθυνο για την υλοποίηση των αποφασισμένων αλλαγών στο όχημα ώστε να διασφαλιστεί ότι το όχημα λειτουργεί με κορυφαίες επιδόσεις και είναι σε θέση να ανταγωνιστεί αποτελεσματικά άλλες ομάδες. Πρέπει να έχει ισχυρό τεχνικό υπόβαθρο στη μηχανική του οχήματος, καθώς και άριστες δεξιότητες επικοινωνίας, επίλυσης προβλημάτων και διαχείρισης ομάδας για να συντονιστούν αποτελεσματικά με άλλα μέλη της αγωνιστικής ομάδας.

**Βαση δεδομένων:** Η βάση στην οποία αποθηκεύονται όλες οι πληροφορίες των χρηστών, καθώς και τα δεδομένα του οχήματος.

**Chart:** Ο τρόπος αναπαράστασης των δεδομένων τηλεμετρίας του οχήματος μέσω διαγράμματος. Τα γραφήματα τηλεμετρίας απεικονίζουν δεδομένα σε πραγματικό χρόνο που μεταδίδονται ασύρματα από ένα όχημα. Εμφανίζουν διάφορες παραμέτρους και αισθητήρες, όπως η ταχύτητα και η κατανάλωση καυσίμου, σε ένα γραμμικό γράφημα. Τα διαγράμματα τηλεμετρίας χρησιμοποιούνται από τους Engineers για την παρακολούθηση της απόδοσης του οχήματος και τον γρήγορο εντοπισμό προβλημάτων. Τα διαγράμματα τηλεμετρίας είναι ένα κρίσιμο εργαλείο στον μηχανοκίνητο αθλητισμό για τη βελτιστοποίηση της απόδοσης.

**Proposals:** Προτάσεις που κάνουν οι Engineers στον Chief Engineer με βάση τα δεδομένα από τα διαγράμματα τηλεμετρίας του οχήματος. Ο Chief Engineer στην συνέχεια αποφασίσει ποιές από τις προτάσεις θα απορρίψει και ποιές θα δεχτεί. Οι προτάσεις τις οποίες δέχεται μεταφέρονται στους Hands on Engineers για υλοποίηση, οι οποίοι μόλις τις υλοποιήσουν αναφέρουν ότι έχουν πραγματοποιηθεί.

**Proposal Status:** Η κατάσταση στην οποία βρίσκεται μια πρόταση. Μια πρόταση μπορεί να έχει 4 καταστάσεις. New: Νέα κατάσταση- Η πρώτη κατάσταση που παίρνει μια πρόταση μόλις γίνει. Accepted: Η κατάσταση στην οποία βρίσκεται όταν ο Chief Engineer έχει δεχτεί την υλοποίηση της πρότασης. Declined: Η κατάσταση στην οποία βρίσκεται όταν ο Chief Engineer έχει απορρίψει την υλοποίηση της πρότασης. Done: Η κατάσταση στην οποία βρίσκεται όταν οι Hands-on Engineers έχουν υλοποιήσει την πρόταση.



**Πληροφορίες χρηστη:** Οι πληροφορίες του κάθε χρήστη. Αυτές περιλαμβάνουν το username, το password, το ονοματεπώνυμο, περιγραφή, τμήμα, ρόλος, προσωπική εικόνα και συνδέσμους για κάποιους λογαριασμούς του στα υπόλοιπα μέσα κοινωνικής δικτύωσης. Όλες οι πληροφορίες αυτές εκτός του password είναι εμφανείς μόνο στον διαχειριστή. Οι χρήστες αλληλεπιδρούν αποκλειστικά με τη χρήση των usernames.

**Vehicle:** Όχημα. Είναι ένα μηχάνημα που έχει σχεδιαστεί για τη μεταφορά ανθρώπων ή αγαθών από το ένα μέρος στο άλλο. Τα οχήματα μπορούν να πάρουν πολλές μορφές, από αυτοκίνητα και φορτηγά μέχρι τρένα, βάρκες και αεροπλάνα. Μπορούν να τροφοδοτούνται από διάφορες πηγές, όπως βενζίνη, ντίζελ, ηλεκτρική ενέργεια ή ακόμα και ανθρώπινη ενέργεια.

**Vehicle Setup:** Οι ρυθμίσεις του οχήματος. Είναι οι συγκεκριμένες διαμορφώσεις συστημάτων όπως της ανάρτησης, της αεροδυναμικής, της πίεσης των ελαστικών και άλλων εξαρτημάτων ενός αγωνιστικού οχήματος, τα οποία μπορούν να ρυθμιστούν για να βελτιστοποιήσουν την απόδοση και την ταχύτητά του.

**System:** Στο πλαίσιο ενός οχήματος, ένα σύστημα αναφέρεται σε ένα σύνολο εξαρτημάτων που συνεργάζονται για να εκτελέσουν μια συγκεκριμένη λειτουργία ή ένα σύνολο λειτουργιών. Υπάρχουν πολλά διαφορετικά συστήματα σε ένα όχημα, το καθένα με το δικό του σκοπό και σύνολο εξαρτημάτων.

**Subsystem:** Στο πλαίσιο ενός οχήματος, ένα υποσύστημα είναι ένα μικρότερο σύστημα που αποτελεί μέρος ενός μεγαλύτερου συστήματος. Μπορεί να θεωρηθεί ως ένα σύστημα μέσα σε ένα σύστημα. Για παράδειγμα, το σύστημα πέδησης είναι ένα μεγαλύτερο σύστημα σε ένα όχημα, αλλά αποτελείται από πολλά μικρότερα υποσυστήματα, όπως τα τακάκια των φρένων, τις δαγκάνες και τους ρότορες.

**Part:** Εξάρτημα. Σε ένα όχημα, αναφέρεται σε ένα μόνο εξάρτημα ή στοιχείο που χρησιμοποιείται για να αποτελέσει ολόκληρο το όχημα. Τα εξαρτήματα μπορεί να είναι μεγάλα ή μικρά και μπορεί να είναι είτε μηχανικής είτε ηλεκτρονικής φύσης. Μερικά παραδείγματα μηχανικών μερών σε ένα όχημα περιλαμβάνουν τον κινητήρα, το κιβώτιο ταχυτήτων και τα φρένα.

**Chat Room:** Δωμάτιο συνομιλίας. Μία σελίδα όπου οι χρήστες μπορούν να ανταλλάσσουν μηνύματα κειμένου σε πραγματικό χρόνο μεταξύ τους σε δωμάτια συνομιλίας που θα αφορούν πιθανές τροποποιήσεις του οχήματος. Οι χρήστες μπορούν επίσης να στέλνουν στιγμιότυπα από διαγράμματα που θεωρούν ότι είναι χρήσιμα στο να παρθεί μια απόφαση.

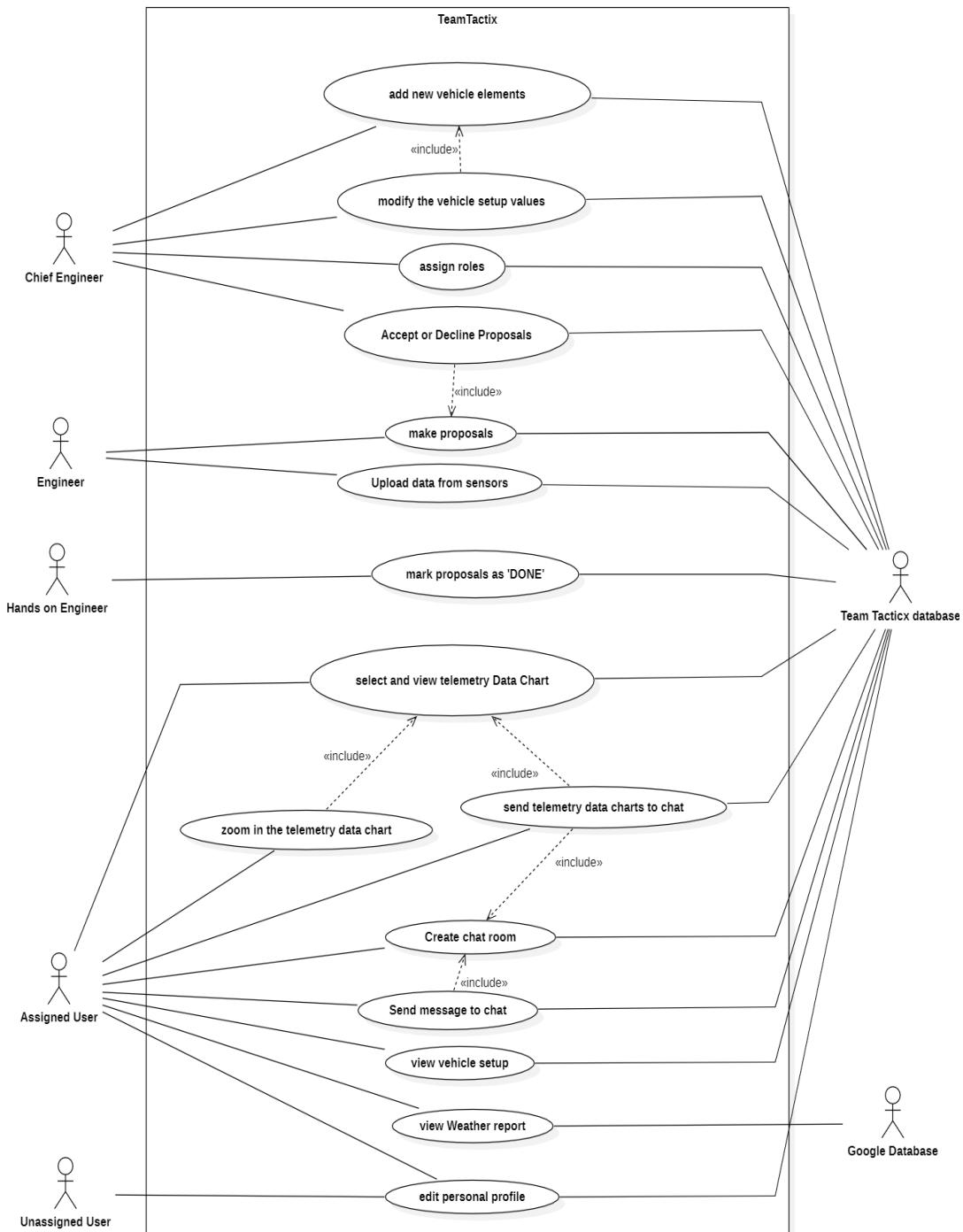
**Chat Page:** Σελίδα συνομιλίας. Μία σελίδα που δείχνει τα διαθέσιμα δωμάτια συνομιλίας στα οποία ανήκει ο χρήστης σε μορφή λίστας.

**Admin Panel:** Πίνακας Διαχειριστή. Ένας πίνακας στον οποίο έχει πρόσβαση μόνο ο Chief Engineer και μπορεί να βλέπει ορισμένα δεδομένα των χρηστών της εφαρμογής και να τροποποιεί τον ρόλο τους στην εφαρμογή και το τμήμα της ομάδας στο οποίο βρίσκονται.



## 2 Σενάρια Χρήσης

### 2.1 Διάγραμμα σεναρίων χρήσης



### 2.2 Role Assignment (Feature με βάση την ορολογία Gherkin)

**Feature:** Role Assignment

**Background:**

Given that I am a chief engineer

And that there are other users registered



And I am logged in

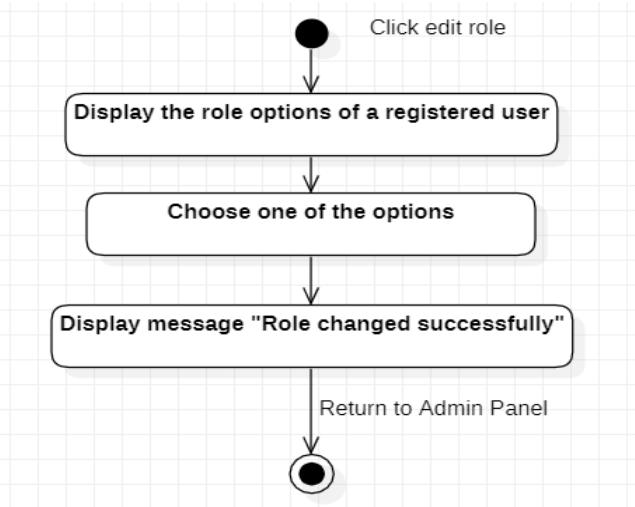
**Scenario:** Successful Role Assignment

When I click on edit role

Then I should be able to see the role options

When I choose one of the options

Then I can see the message "Role changed successfully"



## 2.3 Edit personal profile (Feature με βάση την ορολογία Gherkin)

**Feature:** Edit personal profile

**Background:**

Given that I am a user

And I am logged in

And I am in my profile page

**Scenario:** Successfully editing personal profile

When I click edit personal profile

Then I should be asked enter my new username, my photo, a short description, my department and my social media

When I enter my new username, my photo, a short description, my department and my social media And the size of the photo is: [0,1MB-5MB]

Then I should see a message saying "Profile updated successfully"

**Scenario:** Unsuccessfully editing personal Profile

When I click edit personal profile

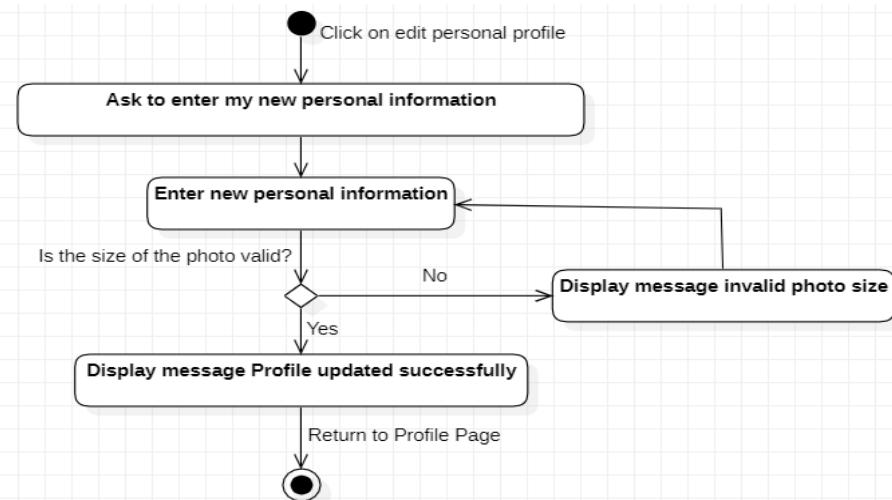
Then I should be asked enter my new username, my photo, a short description, my department and my social media

When I enter my new username, my photo, a short description, my department and my social media

But the size of the photo is invalid

Then I should see a message saying "Invalid photo size. Please import a new photo."

And I should be prompted to import a new photo



## 2.4 Send Message to Chat (Feature με βάση την ορολογία Gherkin)

**Feature:** Send Message To Chat

**Background:**

Given that I am an assigned user

And I am logged in

And I am on Chat page

And there is at least one chat room

And I have chosen a chat room

**Scenario:** Successful Sending Message To Chat

When I type a message

And I click send

but And the message is not blank

Then I should see my message sent in chat

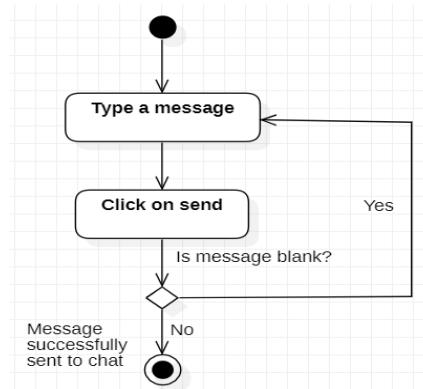
**Scenario:** Unsuccessful Sending Message To Chat

When I type a message

And I click send

But the message doesn't include any character but space

Then I should not see a message sent in chat





## 2.5 Accept or Decline a proposal (Feature με βάση την ορολογία Gherkin)

### Background:

Given I am logged in

And I am Chief Engineer

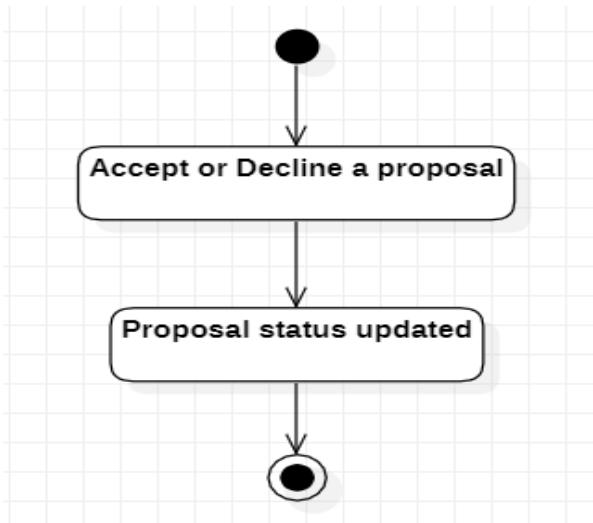
And I have clicked on Dashboard

And there is at least one proposal

### Scenario: Successfully Accept or Decline a proposal

When I click Accept or Decline a proposal

Then Its status should be updated for everyone to see



## 2.6 Select and View Telemetry Data Chart (Feature με βάση την ορολογία Gherkin)

### Feature: Select and view telemetry data charts

### Background:

Given I am logged in

And I am an assigned user

And I am on the Data Chats page

### Scenario: Successfully choosing telemetry Data Chart

When I enter the date, the track and the lap of the chart you want to view

And There is a lap on this date

And there is a lap on this track

Then I should see a message saying "Data Chart is loading..."

And I can view the specific Data Chart

### Scenario: Unsuccessfully choosing telemetry Data Chart due to invalid date

When I enter the date, the track and the lap of the chart you want to view

But there is no lap on this date

Then I should see a message saying "Invalid date"

And I should be prompted to enter the date again

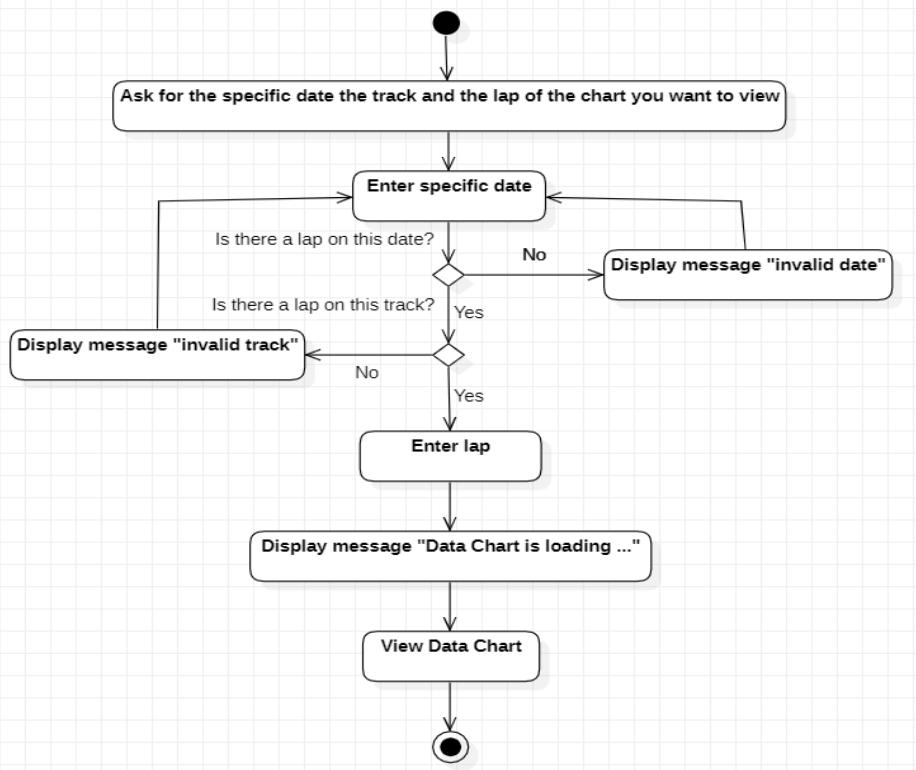
### Scenario: Unsuccessfully choosing telemetry Data Chart due to invalid date

When I enter the date, the track and the lap of the chart you want to view

But there is no lap on this track



**Then** I should see a message saying "Invalid track"  
**And** I should be prompted to enter the track again



## 2.7 View Weather Report (Feature με βάση την ορολογία Gherkin)

**Feature:** View Weather Report

**Background:**

Given that I am an assigned user

And I am logged in

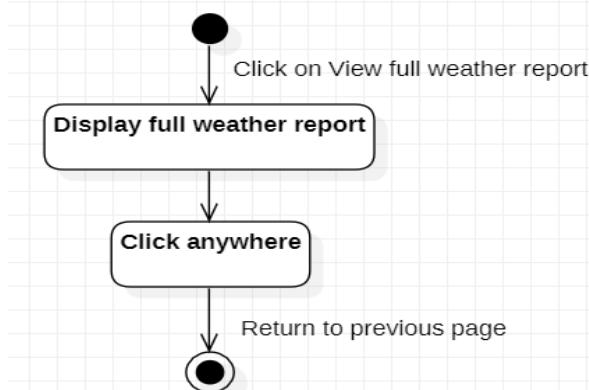
**Scenario:** View Weather Report

**When** I click on View full weather report

**Then** I should be able to view the full weather report

**When** I click anywhere

**Then** I should return to my previous page



## 2.8 Make Proposal (Feature με βάση την ορολογία Gherkin)



**Feature:** Make a proposal

**Background:**

Given I am logged in

And I am Engineer

**Scenario:** Successfully making a proposal

When I click on Make proposal

Then I should be able to choose the system, subsystem and part I want to make the proposal for

When I chose the specific part I want

Then I should be able to see the current setup value

And I should be able to type the new value I want to add

And I should be able to type the reason I want this change

And I should be able to type a short description

When I fill in those

Then I should be able to click on Make proposal

And I should be able to see a message saying "New Proposal added"

**Scenario:** Unsuccessfully making a proposal due to invalid setup value

When I click on Make proposal

Then I should be able to choose the system, subsystem and part I want to make the proposal for

When I chose the specific part I want

Then I should be able to see the current setup value

And Type the new value I want to add

And Type the reason I want this change

And Type a short description

When I enter the new value

But The new value is negative

Then I should see a message saying "Invalid value"

And I should be prompted to enter a valid value

**Scenario:** Unsuccessfully making a proposal due to lack of reason

When I click on Make proposal

Then I should be able to choose the system, subsystem and part I want to make the proposal for

When I chose the specific part I want

Then I should be able to see the current setup value

And Type the new value I want to add

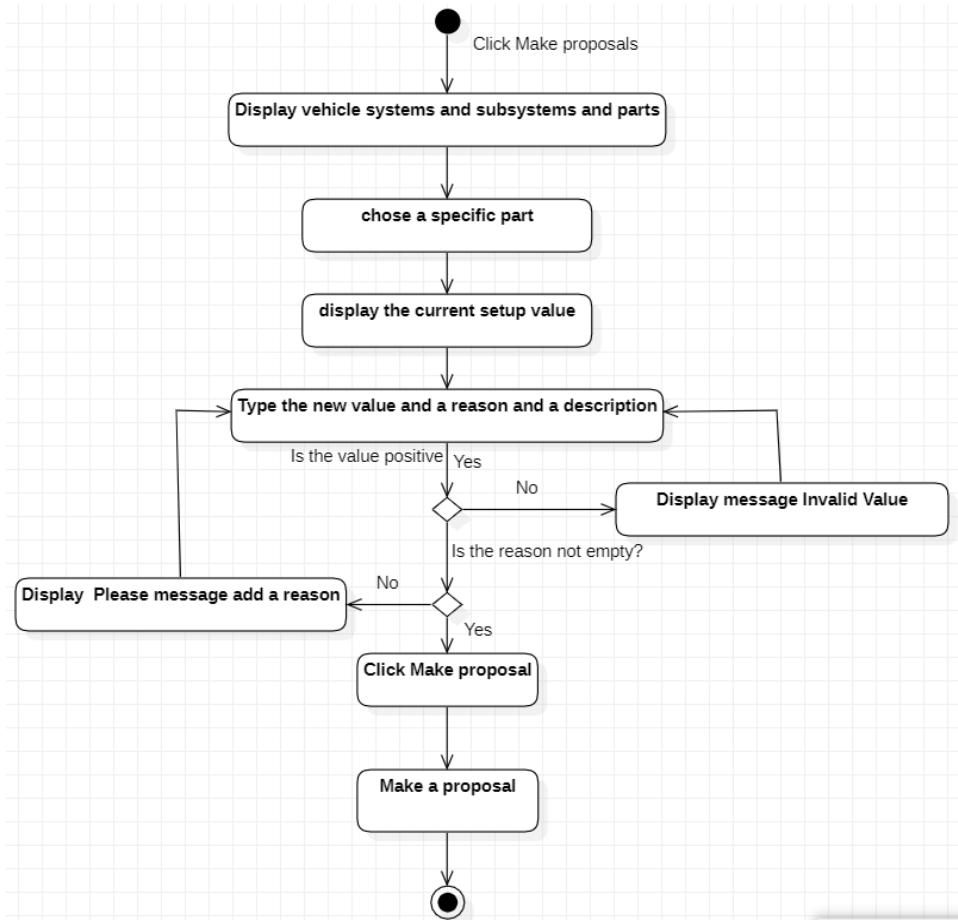
And Type the reason I want this change

And Type a short description

But The reason is empty

Then I should see a message saying "Please add a reason"

And I should be prompted to type a reason



## 2.9 Mark Proposal as DONE (Feature με βάση την ορολογία Gherkin)

**Feature:** Mark Proposal as 'DONE'

**Background:**

Given that I am a hands-on engineer

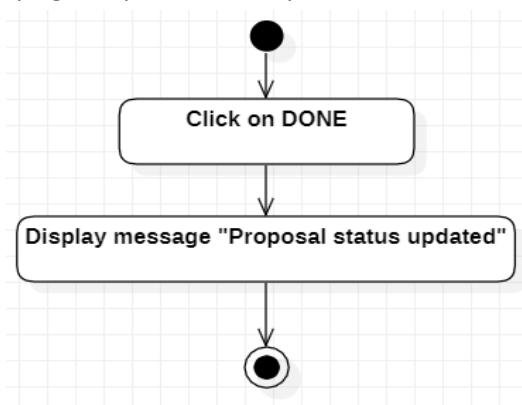
And I am logged in

And I can see the list of accepted proposals

**Scenario:** Successful status update

When I click the button DONE of an accepted proposal

Then I should see a message saying "Proposal status updated"





## 2.10 View Vehicle SetUp (Feature με βάση την ορολογία Gherkin)

**Feature:** View Vehicle Setup

**Background:**

Given that I am logged in

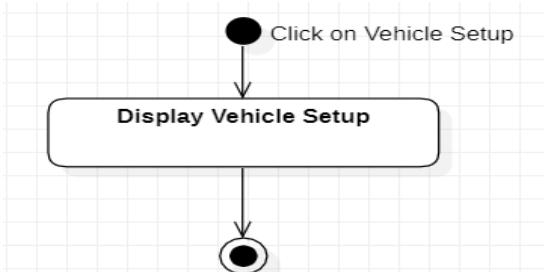
And I am an assigned user

**Scenario:** Successful View Setup

When I click on Vehicle Setup

Then I can see the Vehicle Setup

Εχετε μερικά τέτοια μικρά Gherkin Features/Σενάρια Χρήσης.  
Αν και τυπικά δεν είναι λάθος, ουσιαστικά το να έχετε τόσο μικρά Σενάρια Χρήσης, είναι μία ένδειξη πως ίσως πρέπει να αναδιοργανώσετε τη λειτουργικότητά σας σε σενάρια χρήσης με διαφορετικό τρόπο.



## 2.11 Modify Vehicle Setup Values (Feature με βάση την ορολογία Gherkin)

**Feature:** Modify a Vehicle Setup Value

**Background:**

Given I am logged in

And I am Chief Engineer

And I am on Vehicle Setup Page

**Scenario:** Successfully modifying a Setup Value

When I click on the part I want to change

Then I should be able to change its value

And I should be able to change its measurement unit

When I fill in the new value

And I fill in the new measurement unit

Then I should be able to click on "Apply Changes"

And I should see a message saying "Setup changed successfully"

**Scenario:** Unsuccessfully modifying a Setup Value due to invalid value

When I click on the Part I want to change

Then I should be able to change its value

And I should be able to change its measurement unit

When I fill in the new value

And I fill in the new measurement unit

But the value is negative

Then I should see a message saying "Invalid value"

And I should be prompted to enter a valid value

**Scenario:** Unsuccessfully modifying a Setup Value due to invalid measurement unit

When I click on the Part I want to change

Then I should be able to change its value



And I should be able to change its measurement unit

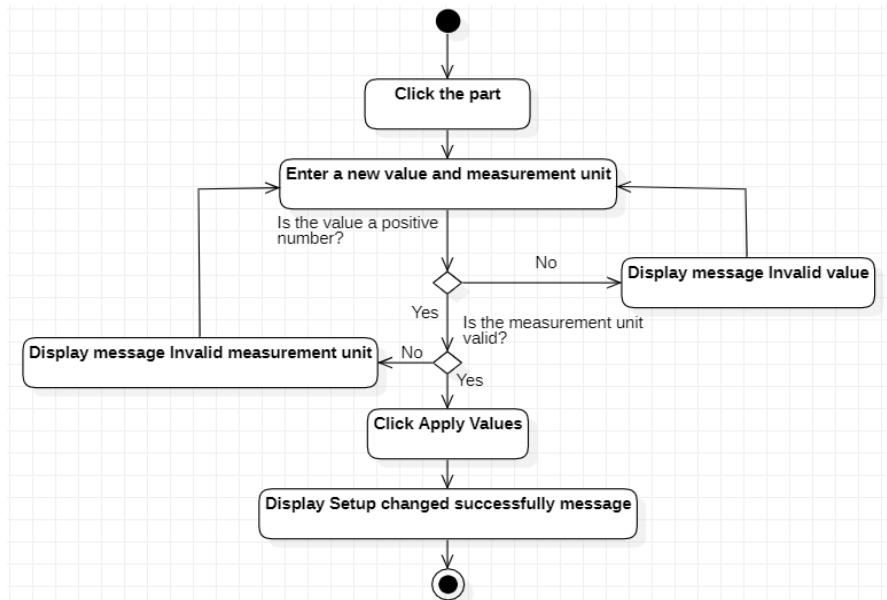
When I fill in the new value

And I fill in the new measurement unit

But the measurement unit is invalid

Then I should see a message saying "Invalid measurement unit"

And I should be prompted to enter a valid measurement unit



## 2.12 Send Telemetry Data Charts on Chat (Feature με βάση την ορολογία Gherkin)

**Feature:** Send Telemetry Data Charts to Chat

**Background:**

Given I am logged in

And I am an assigned user

And I have chosen a specific telemetry Data Chart to view

And I am in at least one chat room

**Scenario:** Successfully Sending Telemetry Data Charts to Chat

When I click on send Chart

Then I should be asked "Are you sure you want to send this chart?"

When I click "Yes"

Then I should see a message "Chart send successfully"

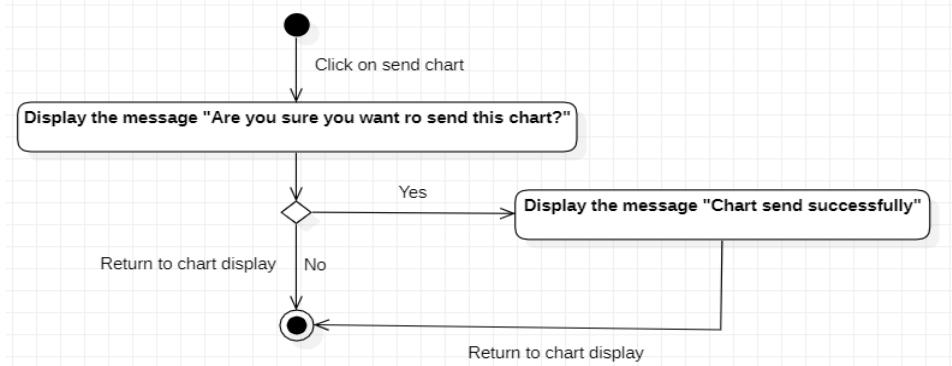
**Scenario:** Unsuccessfully Sending Telemetry Data Charts to Chat

When I click on send Chart

Then I should be asked "Are you sure you want to send this chart?"

When I click "No"

Then I should return to the display of the chart



## 2.13 Add New Elements To Vehicle Setup(Feature με βάση την ορολογία Gherkin)

**Feature:** Add New Elements To Vehicle Setup

**Background:**

Given I am a Chief Engineer

And I am logged in

And I have entered the vehicle setup display

**Scenario:** Successful Adding New Elements To Vehicle Setup

When I click add Element

Then I should be able to choose among part, subsystem and system

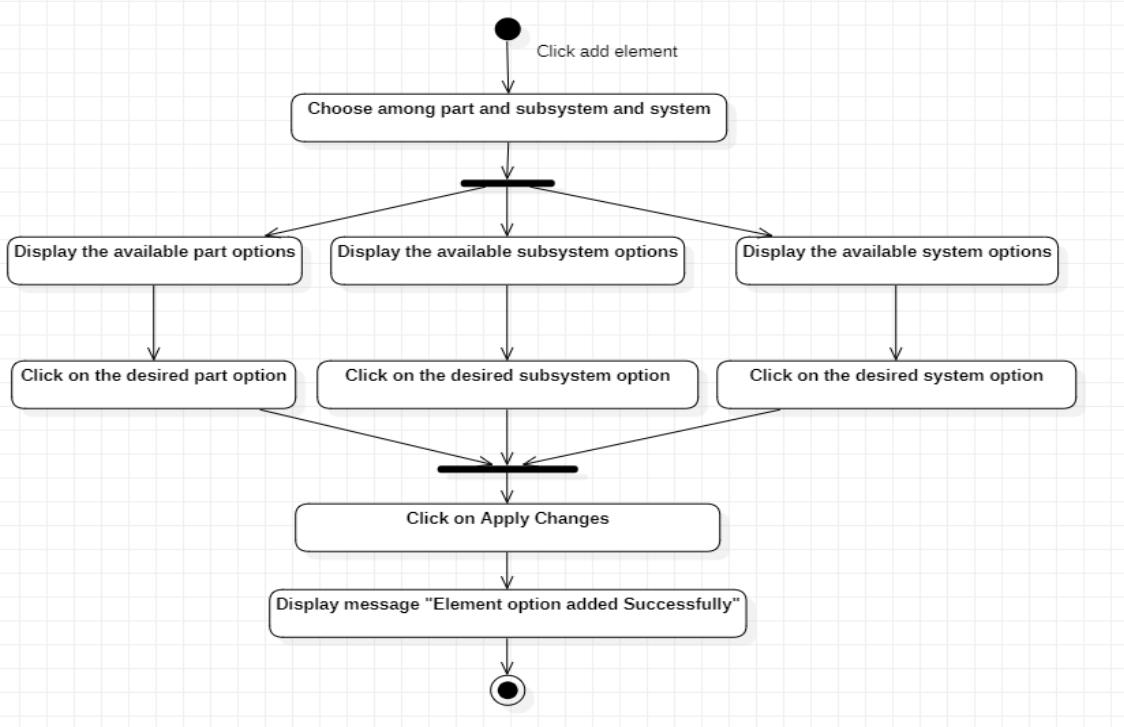
When I click on the desired element (part or subsystem or system) type

Then I should be able to see the available specific element options

When I click on the desired element option

And I click on Apply Changes

Then I should see a message "Element option added Successfully"





## 2.14 Zoom in the Telemetry Data Chart (Feature με βάση την ορολογία Gherkin)

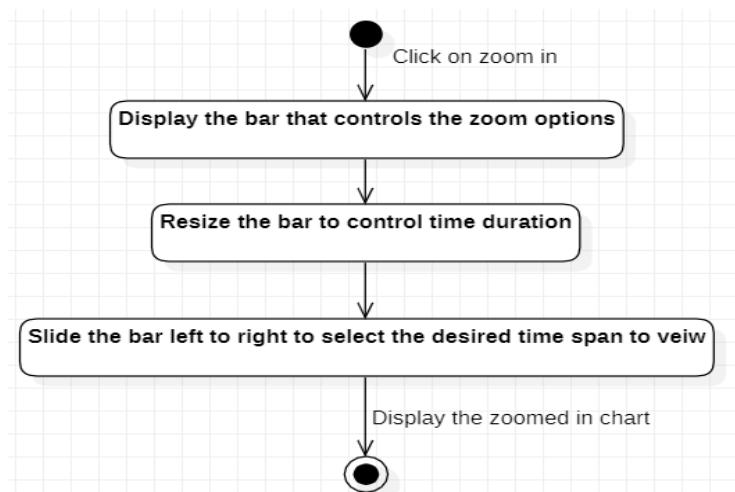
**Feature:** Zoom in the Telemetry Data Chart

**Background:**

Given I am an assigned user  
And I am logged in  
And I have already chosen one Telemetry Data Chart

**Scenario:** Successfully Zooming in

When I click on zoom in  
Then I should be able to see the bar that controls the zoom options  
When I resize the bar  
Then I should be able to choose the desired time duration  
When I slide the bar left to right  
Then I should be able to select the desired time span to view  
And I should see the zoomed in chart



## 2.15 Create Chat Room(Feature με βάση την ορολογία Gherkin)

**Feature:** Create Chat Room

**Background:**

Given that I am logged in  
And I am an assigned user  
And I am on the Chat page

**Scenario:** Successful Create Chat Room

When I click on the create chat room icon  
Then I should be able to choose the users I want to add in the group  
And I should be able to enter chat room name  
When I choose the users I want to add in the group  
And I enter the name of the chat room  
Then I should see the create chat room option  
When I click on the Create chat room button  
And at least one user is added in the chat room



And the chat room name is not blank

Then I should see a message saying "Chat room successfully created."

**Scenario:** Unsuccessful due to invalid name

When I click on the plus icon

Then I should be able to choose the users I want to add in the group

And I should be able to enter chat room name

When I choose the users I want to add in the group

And I enter the name of the chat room

Then I should see the create chat option

When I click on the Create chat room button

And at least one user is added in the chat room

But the chat room name is blank

Then I should see a message saying "Chat room cannot be created without a chat room name"

**Scenario:** Unsuccessful due to no users added

When I click on the plus icon

Then I should be able to choose the users I want to add in the group

And I should be able to enter chat room name

When I choose the users I want to add in the group

And I enter the name of the chat room

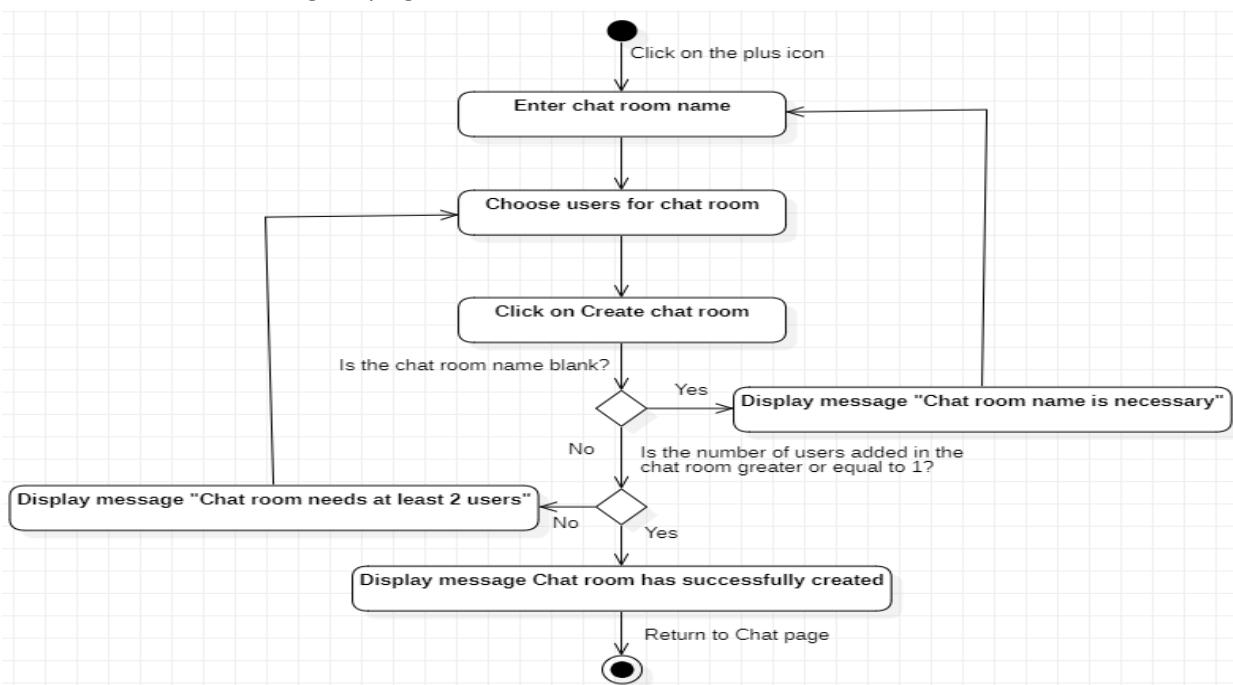
Then I should see the create chat option

When I click on the Create chat room button

But there are no users added in chat room

And the chat room name is not blank

Then I should see a message saying "Chat room cannot be created without chat room users."



## 2.16 Upload Data from Sensors(Feature με βάση την ορολογία Gherkin)



**Feature:** Upload a file containing data from the sensors

**Background:**

Given I am an engineer

And I am logged in

And I have a file containing data from the sensors

**Scenario:** Successfully uploading data file

When I click on add data file

Then I should be asked to upload a data file

When I upload a data file

And the size of the file is: [0,1MB-10MB]

Then I should see a message saying "Data file successfully uploaded"

**Scenario:** Unsuccessfully uploading data file

When I click on add data file

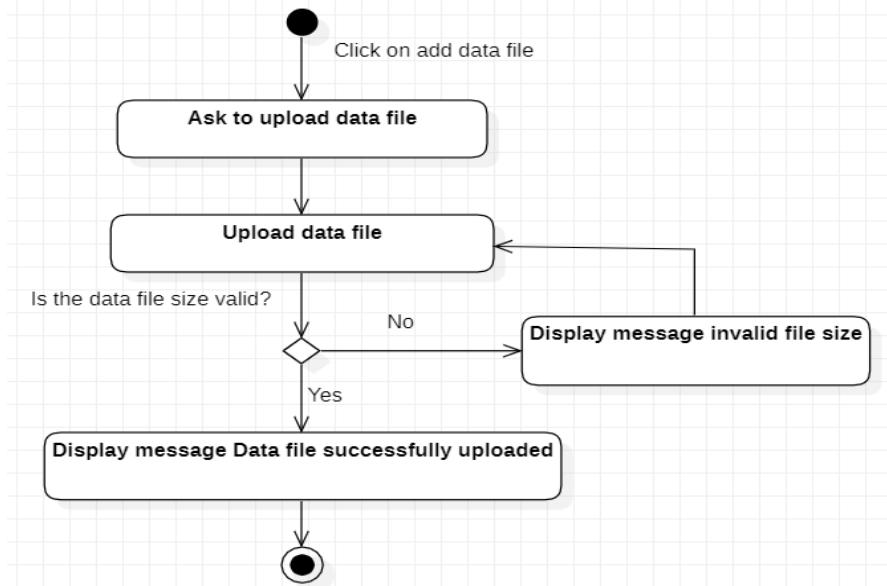
Then I should be asked to upload a data file

When I upload a data file

But the size of the file is invalid

Then I should see a message saying "File size invalid. Please upload new file"

And I should be prompted to upload a new file





### 3 Επιδεικτικά γραφικά παράθυρα διεπαφής

Ακολουθούν τα παράθυρα διεπαφής των χρηστών Engineer, Hands-on Engineer και Chief Engineer.

#### 3.1 Homepage

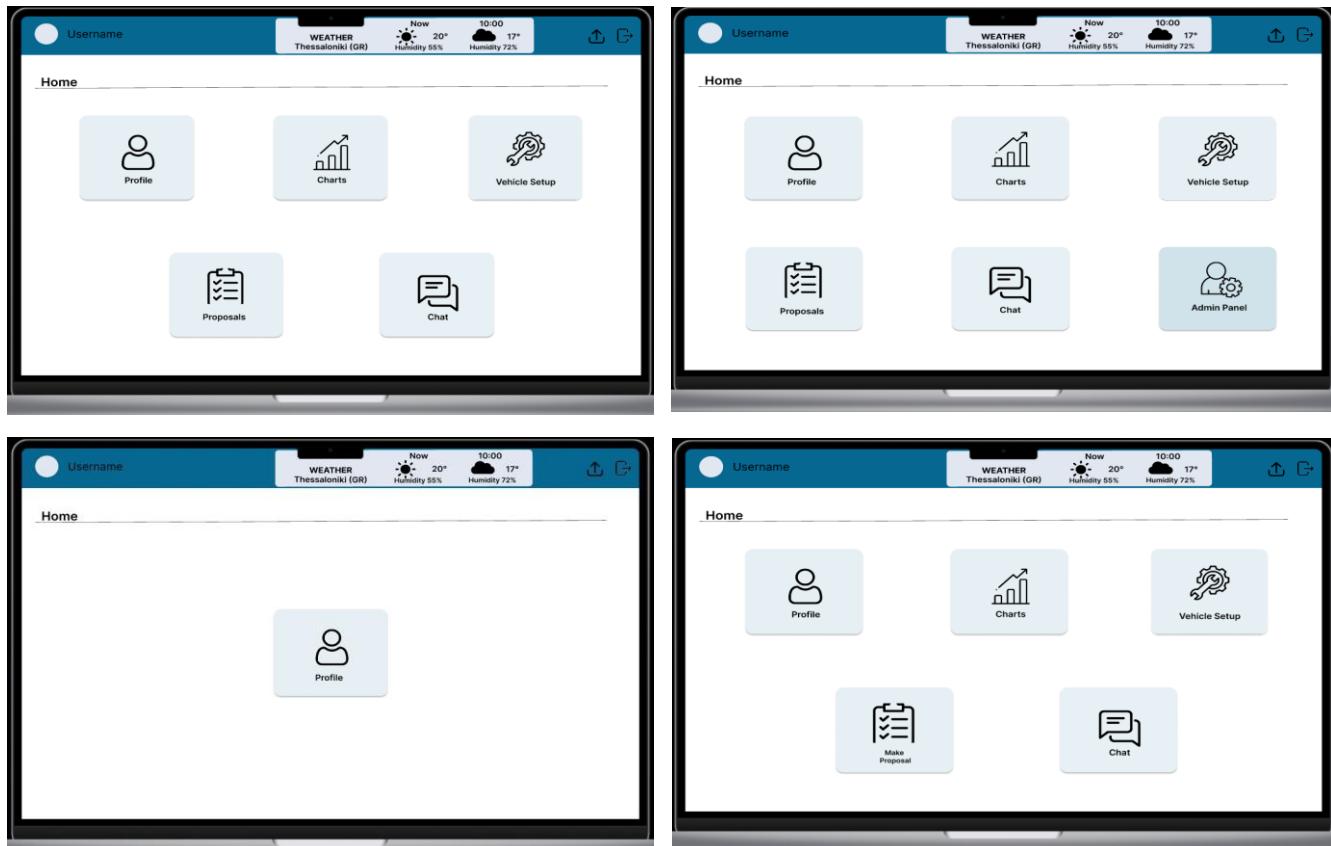


Figure 1: Homepage των χρηστών

Οι αρχικές οθόνες των χρηστών Hands-on Engineer, Chief Engineer, Engineer και Default αντίστοιχα.



### 3.2 Admin Panel - Chief Engineer

The Admin Panel interface for the Chief Engineer consists of two main components:

- Main Dashboard (Top Screenshot):** This is the primary view showing a list of users, current weather conditions for Thessaloniki (GR), and various navigation links. The user list includes:

No	Name	Department	Role	Joined	Last modified
1	Panos Karvounaris	Cleaning	Default	31/01/2022	15/04/23 06:37:58
2	Panos Sitaridis	Electronics	Engineer	30/02/2020	28/03/23 11:28:45

- User Profile Edit Dialog (Bottom Screenshot):** A modal window for editing the profile of Panos Karvounaris. It displays a placeholder image, the user's name, and input fields for "Role" and "Department". At the bottom are "Cancel" and "Save" buttons.

Figure 2: Admin panel του χρήστη Chief Engineer

Πατώντας Admin Panel στο Homepage ο Chief Engineer μεταφέρεται στην πρώτη οθόνη του Figure 2, όπου φαίνονται τα στοιχεία των υπόλοιπων χρηστών. Πατώντας στο μολύβι του κάθε χρήστη, ο Chief Engineer μπορεί να μεταβάλλει τον ρόλο και το τμήμα του χρήστη (δεύτερη οθόνη του figure 2).



### 3.3 Proposals - Chief Engineer

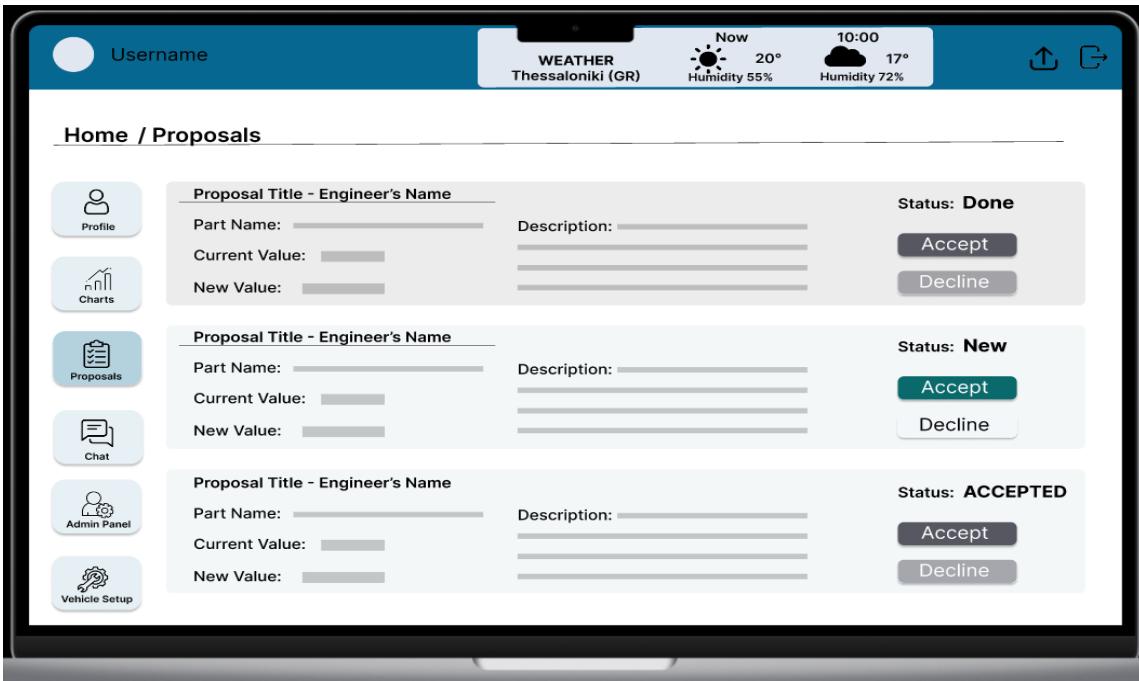


Figure 3: Proposals page του χρήστη Chief Engineer

Πατώντας Proposals στο Homepage του Chief Engineer ο χρήστης μεταφέρεται στην οθόνη του figure 3, όπου μπορεί να δει τις προτάσεις των Engineers και να κάνει Accept ή Decline σε κάθε πρόταση.

### 3.4 Proposals - Hands-On Engineers

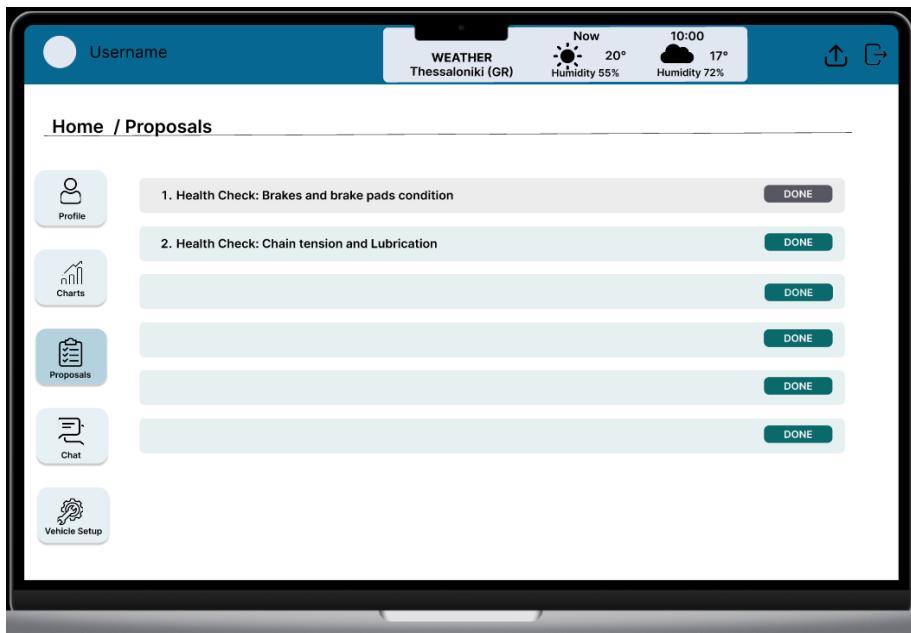


Figure 4: Proposals Page του χρήστη Hands-on Engineer

Κάνοντας click στο Homepage του Hands-on Engineer ο χρήστης μεταφέρεται στη σελίδα proposals του Hands-on Engineer, όπου μπορεί να δεις τις προτάσεις που έχουν γίνει αποδεκτές απ' τον Chief Engineer, να κάνει τις απαραίτητες αλλαγές στο όχημα και να πατήσει DONE όταν ολοκληρωθούν.



### 3.5 Make Proposal - Engineer

**New Proposal**

Proposal Title

Part Name

Descriprion

Current Value:  mm      New Value:  mm

Cancel      Save

Figure 5: Proposal Page του χρήστη Engineer

Πατώντας Make Proposal στο Homepage του Engineer ο χρήστης μεταφέρεται στη σελίδα Make Proposal, όπου του δίνεται η δυνατότητα να κάνει μια καινούργια πρόταση.

### 3.6 Vehicle Setup

**Black Panther**  
Our 2nd prototype for MotoStudent VI @ Panther Racing AUTH

**Electronic Control Unit**

Traction Control (Set)	1
Power Mapping (Set)	1
Engine Break (Set)	1

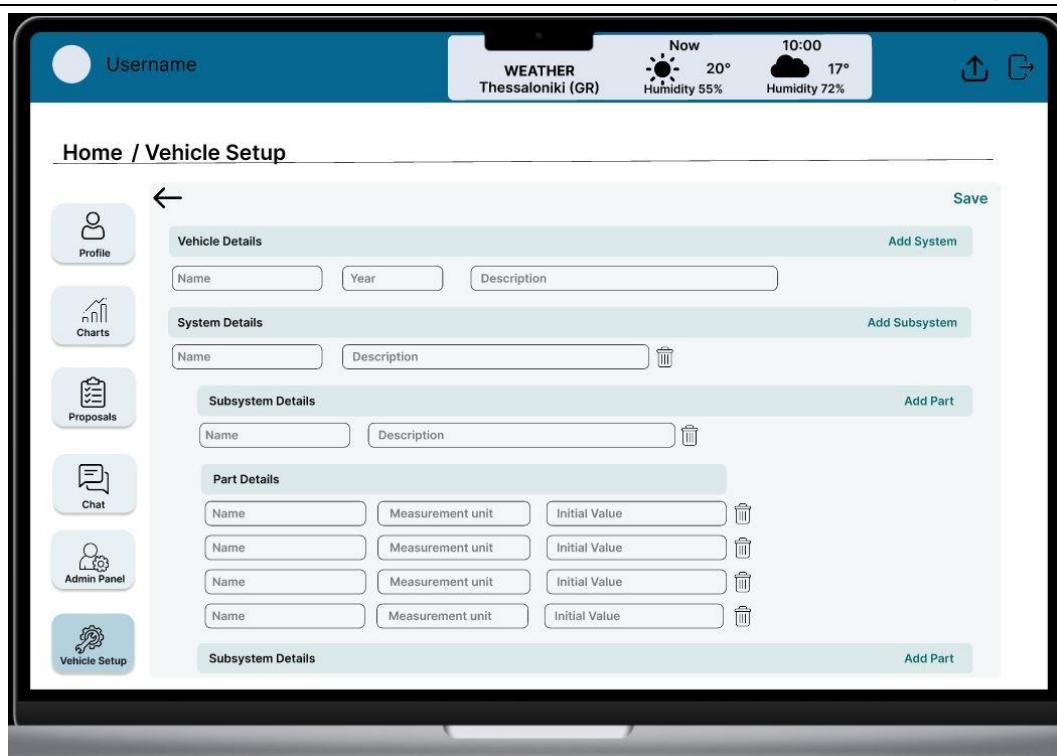
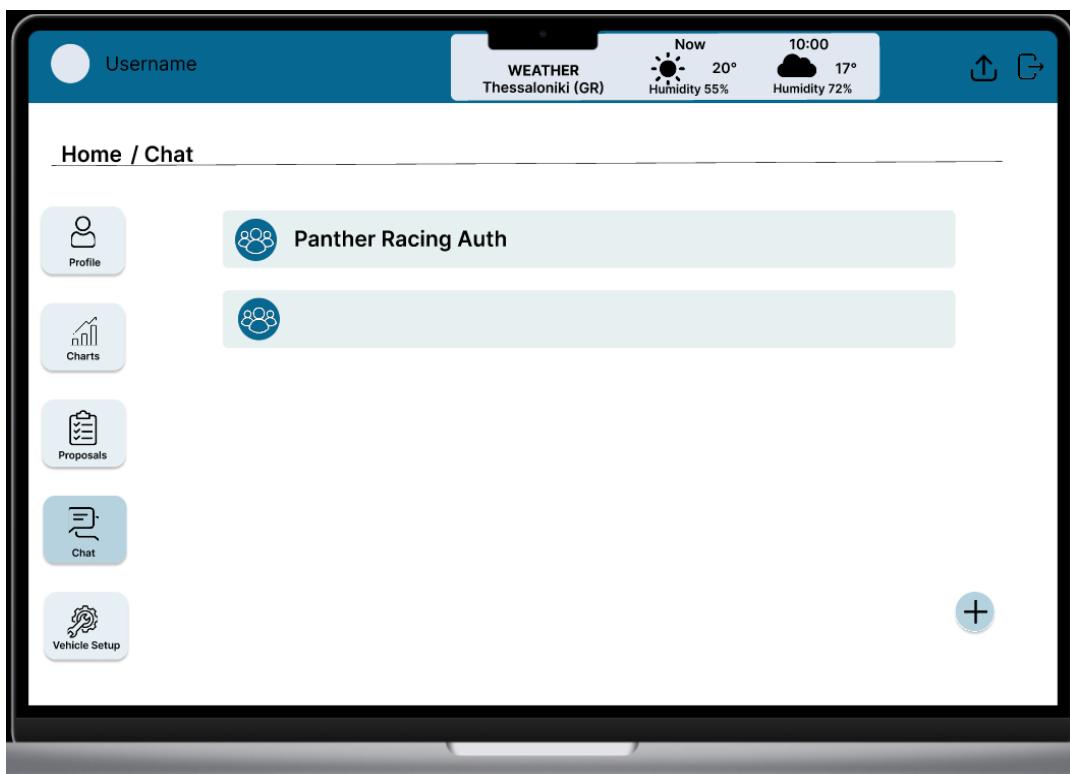


Figure 6: Vehicle setup

Κάνοντας click στο Vehicle Setup του Homepage, ο χρήστης μπορεί να δει το setup του οχήματος του(πρώτη οθόνη του figure 6). Πατώντας το μολύβι μεταβαίνει στη δεύτερη σελίδα του figure 6, όπου μπορεί να πεξεργαστεί κάθε part, subsystem και system, καθώς και να προσθέσει νέα.

### 3.7 Chat



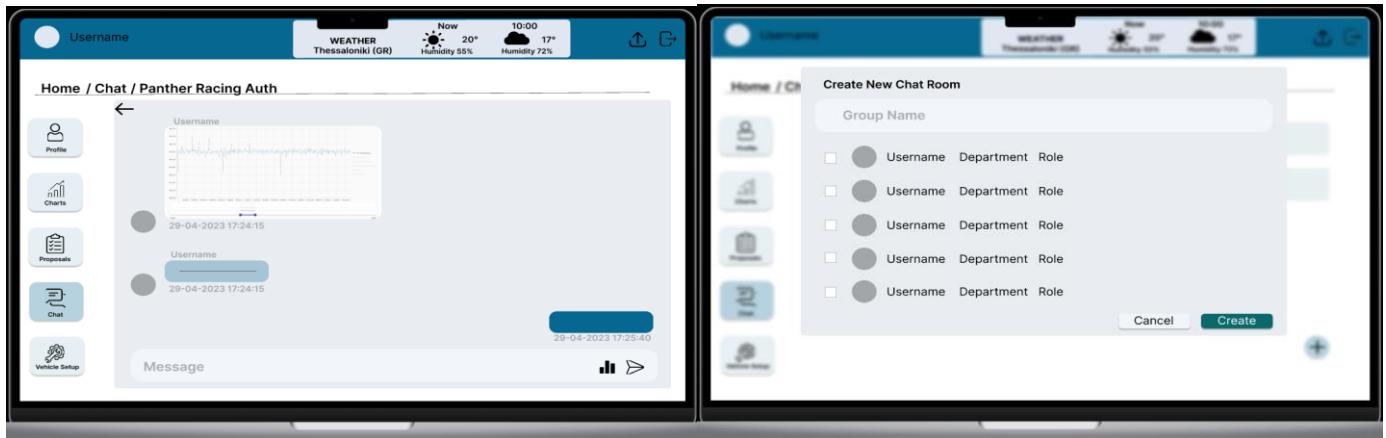


Figure 7: Σελίδες chat όλων των χρηστών

Πατώντας Chat στο Homepage του, ο χρήστης μεταβαίνει στη πρώτη οθόνη του figure 7, όπου μπορεί να δει μια λίστα με τα chat rooms. Αν επιλέξει κάποιο από αυτά μεταβαίνει στη δεύτερη οθόνη του figure 7, όπου μπορεί να δει και να στείλει μηνύματα και charts. Αν επιλέξει το plus icon στο κάτω μέρος της σελίδας chat, μεταβαίνει στη τρίτη οθόνη, όπου μπορεί να δημιουργήσει ένα καινούργιο chat room.

### 3.8 Personal Profile

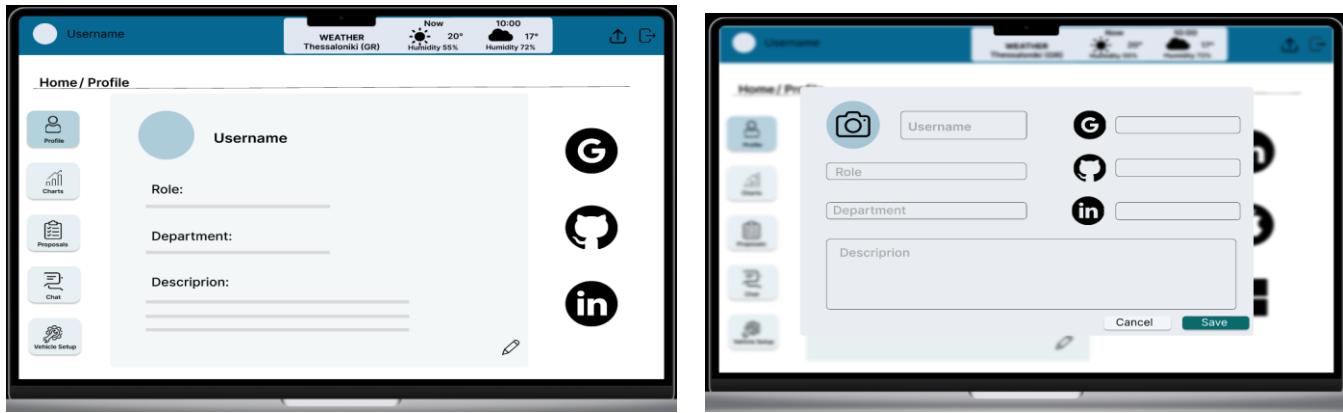


Figure 8: Προσωπικό προφίλ κάθε χρήστη

Πατώντας το Profile στην αρχική του σελίδα, ο χρήστης μπορεί να δει το όνομα, τη φωτογραφία, τον ρόλο, το τμήμα, την περιγραφή και τα social του. Κάνοντας click στο μολύβι στο κάτω μέρος της σελίδας, μεταφέρεται στη δεύτερη οθόνη του figure 8 και μπορεί να επεξεργαστεί τα στοιχεία του.



### 3.9 Charts

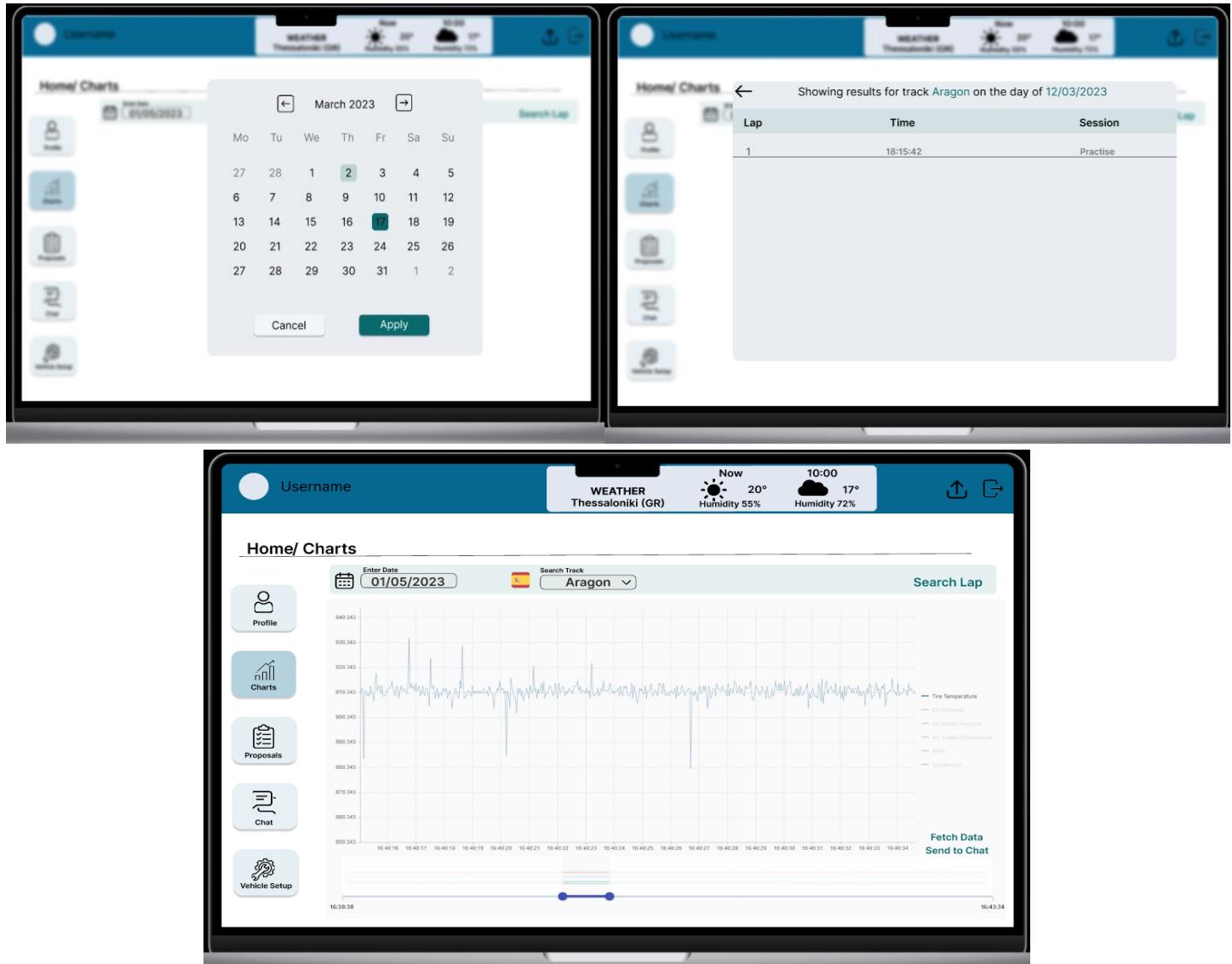


Figure 9: Σελίδες chart όλων των χρηστών

Πατώντας Charts στην αρχική σελίδα του, ο χρήστης πηγαίνει στην πρώτη οθόνη του figure 9, όπου του ζητείται να επιλέξει την ημερομηνία που πραγματοποιήθηκαν οι δοκιμές που θέλει να μελετήσει. Αφού επιλέξει, εμφανίζεται μια λίστα με τα lap. Επιλέγοντας ένα από αυτά μεταφέρεται στην τρίτη οθόνη του figure 9. Από το δεξιό μέρος της οθόνης μπορεί να επιλέξει το είδος δεδομένων που θέλει να δει και να προσκομίσει. Τέλος, μπορεί να στείλει το Chart στο Chat πατώντας Send to Chat.



### 3.10 Weather

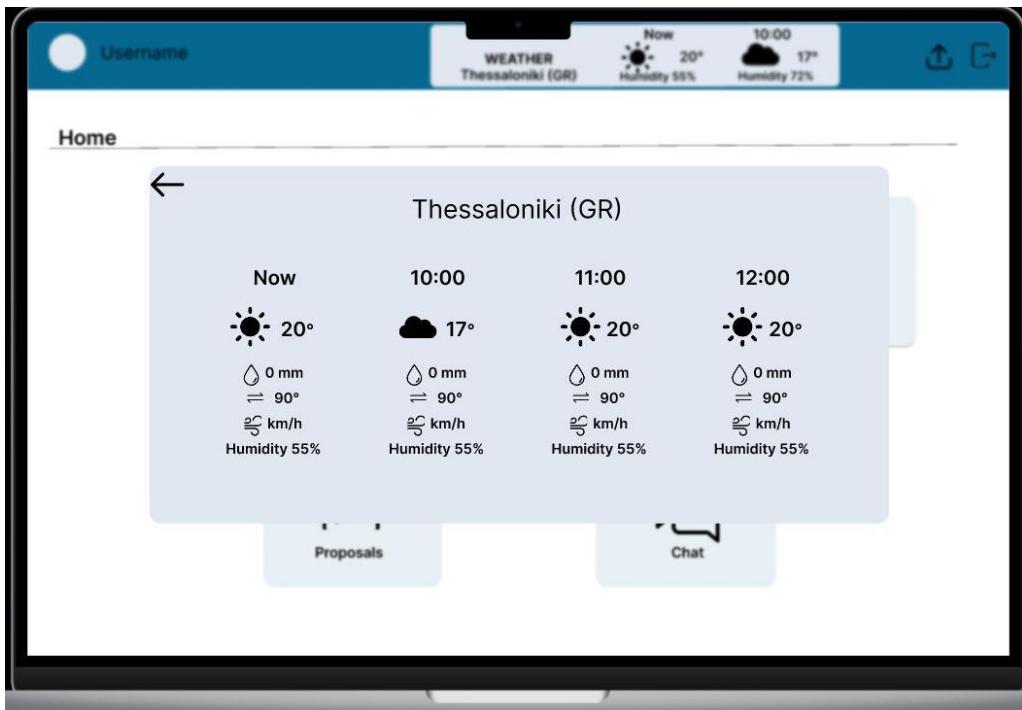
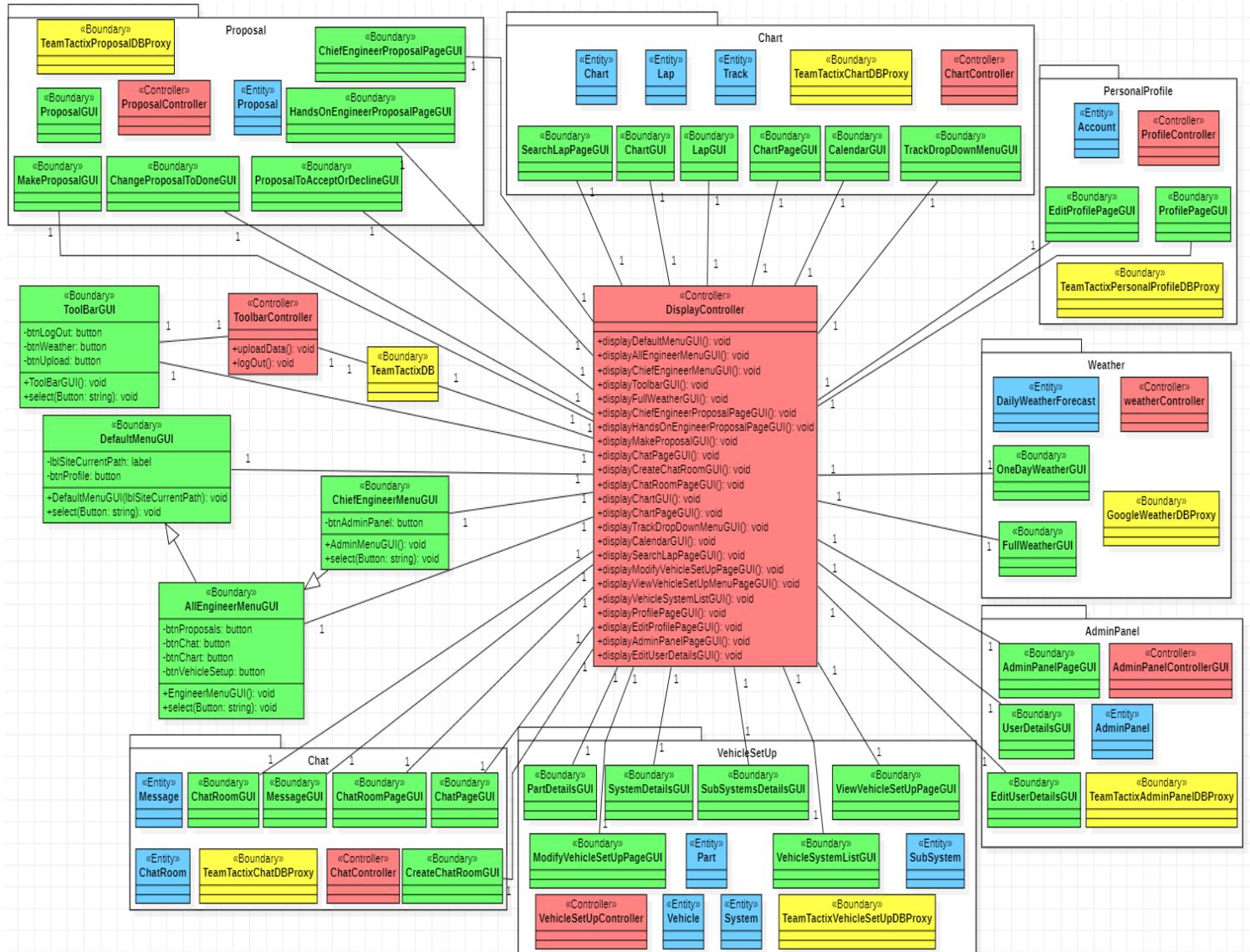


Figure 10: Σελίδα Weather Report των χρηστών

Πατώντας τον καιρό που εμφανίζεται στο Toolbar, ο χρήστης πηγαίνει στην οθόνη του figure 10, όπου μπορεί να δει πιο αναλυτικά στοιχεία για τον καιρό στην περιοχή του.



## 4 Στατική Μοντελοποίηση

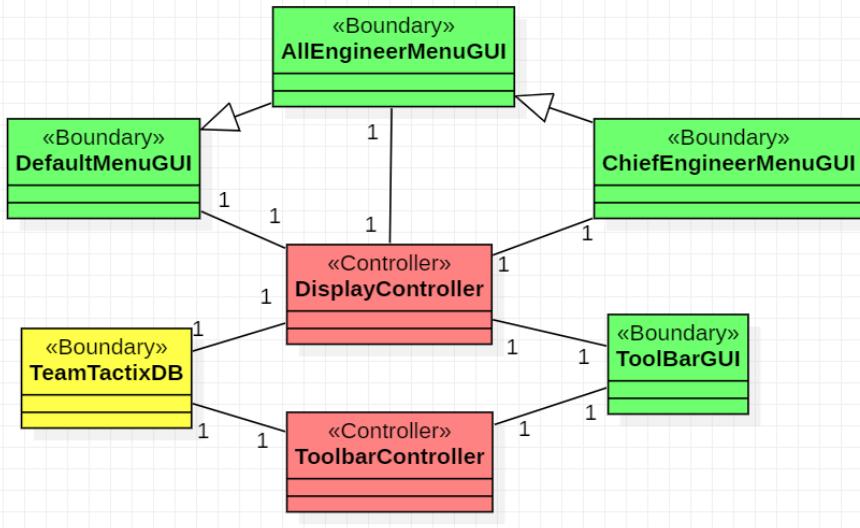




## 4.1 <Πακέτα λεξιλογίου σεναρίων υψηλής προτεραιότητας>

### 4.1.1 Πακέτο User Handle Package

#### User Handle Package



#### DefaultMenuGUI

«Boundary» DefaultMenuGUI	
-lblSiteCurrentPath: label	
-btnProfile: button	

+DefaultMenuGUI(lblSiteCurrentPath): void
+select(Button: string): void

Η κλάση αυτή εκφράζει τη διεπαφή του Menu του Unassigned User και του δίνει μόνο την δυνατότητα να τροποποιήσει το προφίλ του.

#### Χαρακρηριστικά κλάσης

- lblSiteCurrentPath: label: Ετικέτα με το τρέχον path του χρήστη.
- btnProfile: button: Κουμπί διεπαφής για ανακατεύθυνση στο προφίλ του χρήστη.

#### Μέθοδοι κλάσης

- DefaultMenuGUI(lblSiteCurrentPath): void: Αποτελεί τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

#### AllEngineerMenuGUI

«Boundary» AllEngineerMenuGUI	
-btnProposals: button	
-btnChat: button	
-btnChart: button	
-btnVehicleSetup: button	

+AllEngineerMenuGUI(): void
+select(Button: string): void

Η κλάση αυτή αφορά την διεπαφή του Menu των χρηστών με ρόλο Hands-On Engineer και Engineer. Κληρονομεί την κλάση Default



## Χαρακρητικά κλάσης

- btnProposals: button: Κουμπί διεπαφής για ανακατεύθυνση στην διεπαφή των προτάσεων.
- bntChat: button: Κουμπί διεπαφής για ανακατεύθυνση στην διεπαφή του Chat.
- btnChart: button: Κουμπί διεπαφής για ανακατεύθυνση στην διεπαφή των Data Charts.
- btnVehicleSetup: button: Κουμπί διεπαφής για ανακατεύθυνση στην διεπαφή του Vehicle Setup.

## Μέθοδοι κλάσης

- EngineerMenuGUI(): void: Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

## ChiefEngineerMenuGUI

«Boundary»
<b>ChiefEngineerMenuGUI</b>
-btnAdminPanel: button
+AdminMenuGUI(): void +select(Button: string): void

Η προκείμενη κλάση αφορά την διεπαφή του Chief Engineer. Κληρονομεί την κλάση AllEngineerMenuGUI και δίνει την επιπλέον δυνατότητα για μετάβαση στην διεπαφή του Admin Panel ώστε να μπορεί ο Chief Engineer να αναθέτει ρόλους στους χρήστες.

## Χαρακτηριστικά κλάσης

- btnAdminPanel: button: Κουμπί διεπαφής για ανακατεύθυνση στην διεπαφή του Admin Panel.

## Μέθοδοι κλάσης

- AdminPanelGUI(): void: Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

## ToolBarGUI

«Boundary»
<b>ToolBarGUI</b>
-btnLogOut: button -btnWeather: button -btnUpload: button
+ToolBarGUI(): void +select(Button: string): void

Η κλάση αυτή σχετίζεται με την διεπαφή της μπάρας και των στοιχείων στο πάνω μέρος της εφαρμογής. Παραμένει σταθερή για όλη τη διάρκεια χρήσης και για όλους τους χρήστες και δίνει δυνατότητα για



αποσύνδεσης, προβολής του καιρού σε όλους, ενώ και για τους Engineers επιτρέπει και το ανέβασμα αρχείων.

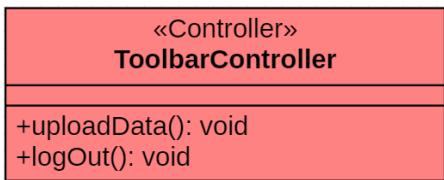
### Χαρακτηριστικά κλάσης

- btnLogOut: button: Κουμπί που εκτελεί την αποσύνδεση του χρήστη.
- btnWeather: button: Clickable Widget που προβάλλει συνοπτικά πληροφορίες για τον καιρό, ενώ αν πατηθεί αναπτύσσει ένα παράθυρο με το πλήρες δελτίο καιρού.
- btnUpload: button: Κουμπί που ανακατευθύνει τους Engineers σε φάκελο των αρχείων τους ώστε να ανεβάσουν αυτά που επιθυμούν.

### Μέθοδοι κλάσης

- ToolBarGUI(): void: Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ToolBarController

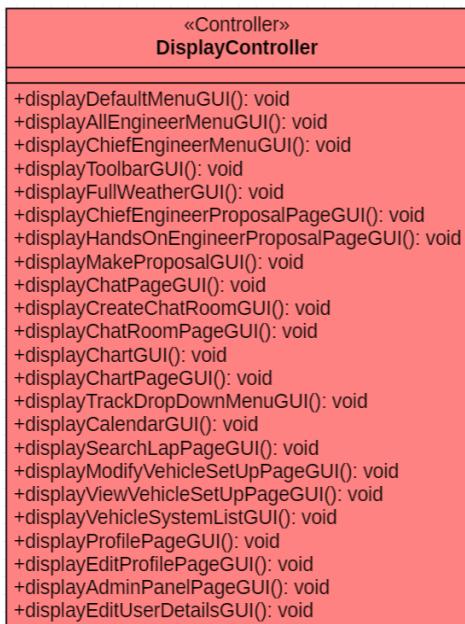


Η κλάση ToolbarController είναι υπεύθυνη για την λειτουργία της αποσύνδεσης και του ανεβάσματος των επιθυμητών αρχείων που εκτελούνται με το πάτημα των αντίστοιχων κουμπιών στην κλάση ToolBarGUI.

### Μέθοδοι κλάσης

- uploadData(): void: Λειτουργία που επιτρέπει στους Engineers να ανεβάσουν τα επιθυμητά αρχεία τους στην βάση δεδομένων της ομάδας τους.
- logOut(): void: Λειτουργία που αποσυνδέει τον χρήστη από την εφαρμογή.

### DisplayController



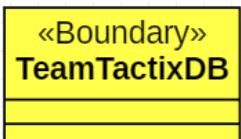
Η κλάση DisplayController έχει ρόλο να ανακατευθύνει το χρήστη στην οθόνη που επιλέγει με το πάτημα των αντίστοιχων κουμπιών.



## Μέθοδοι κλάσης

- `displayDefaultMenuGUI(): void`: Λειτουργία που προβάλλει το μενού του Unassigned User.
- `displayAllEngineerMenuGUI(): void`: Λειτουργία που προβάλλει το μενού των μηχανικών εκτός του Chief Engineer.
- `displayChiefEngineerMenuGUI(): void`: Λειτουργία που προβάλλει το μενού του Chief Engineer.
- `displayToolBarGUI(): void`: Λειτουργία που προβάλλει το ToolBar.
- `displayFullWeatherGUI(): void`: Λειτουργία που προβάλλει το πλήρες δελτίο καιρού.
- `displayChiefEngineerProposalPageGUI(): void`: Λειτουργία που προβάλλει τις προτάσεις στον Chief Engineer
- `displayHandsOnEngineerProposalPageGUI(): void`: Λειτουργία που προβάλλει τις προτάσεις στον Hands-On Engineer.
- `displayMakeProposalGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για την δημιουργία πρότασης στους Engineers.
- `displayChatPage(): void`: Λειτουργία που προβάλλει τις διαθέσιμες συνομιλίες για επικοινωνία.
- `displayCreateChatRoomGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για την δημιουργία συνομιλίας.
- `displayChatRoomPageGUI(): void`: Λειτουργία που προβάλλει την επιλεγμένη συνομιλία.
- `displayChartGUI(): void`: Λειτουργία που προβάλλει ένα Data Chart.
- `displayChartPageGUI(): void`: Λειτουργία που προβάλλει την σελίδα για προβολή Charts.
- `displayTrackDropDownMenuGUI(): void`: Λειτουργία που αναπτύσσει την λίστα με τις διαθέσιμες πίστες.
- `displayCalendarGUI(): void`: Λειτουργία που προβάλλει το ημερολόγιο για την επιλογή ημερομηνίας προβολής των δεδομένων των Data Charts.
- `displaySearchLapPageGUI(): void`: Λειτουργία που προβάλλει την διεπαφή της αναζήτησης γύρου.
- `displayModifyVehicleSetupPageGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για την τροποποίηση των μεγεθών του Vehicle Setup.
- `displayViewVehicleSetupPageGUI(): void`: Λειτουργία που προβάλλει τη διεπαφή του Vehicle Setup.
- `displayVehicleSystemListGUI(): void`: Λειτουργία που προβάλλει το μενού με τα συστήματα του οχήματος.
- `displayProfilePageGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για την προβολή του προφίλ.
- `displayEditProfilePageGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για την τροποποίηση του προφίλ του χρήστη.
- `displayAdminPanelPageGUI(): void`: Λειτουργία που προβάλλει την διεπαφή για το Admin Panel στον Chief Engineer.
- `displayEditUserDetailsGUI(): void`: Λειτουργία που προβάλλει την διεπαφή με τις πληροφορίες του χρήστη στην διεπαφή AdminPanelPageGUI.

## TeamTactixDB



Μια κλάση που δεν έχει ούτε attributes ούτε methods σπάνια είναι χρήσιμη (και ακόμη και τότε συνήθως είναι κακό design).

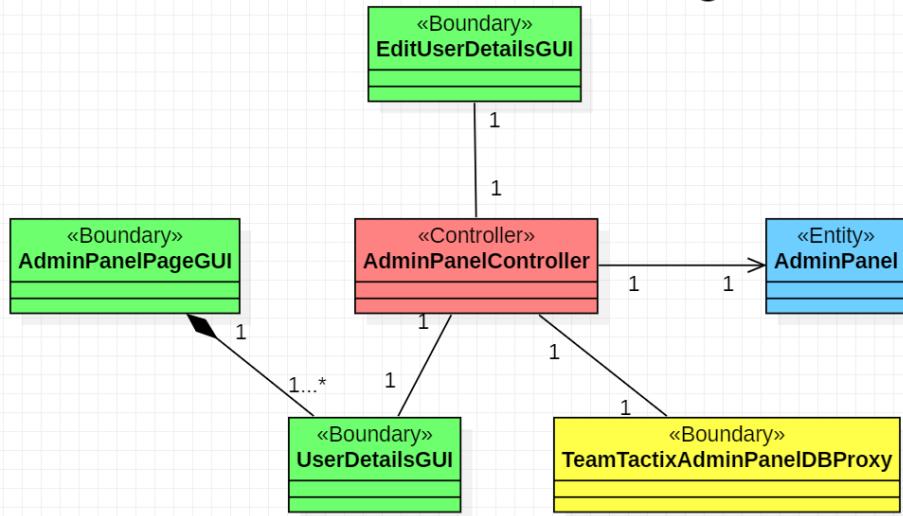
Η κλάση αυτή αντιπροσωπεύει την κύρια βάση δεδομένων της ομάδας.





#### 4.1.2 Πακέτο Admin Panel

### Admin Panel Package



#### Admin Panel

«Entity» AdminPanel	
-accountList:	List<Account>
+AdminPanel(	accountList): void
+setAccountList(	accountList: List<Account>): void
+getAccountList()	: List<Account>

Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν το Admin Panel.

#### Χαρακτηριστικά κλάσης

- accountList: list<Account> : Λίστα που περιέχει τα accounts των χρηστών.

#### Μέθοδοι κλάσης

- AdminPanel( accountList): void : Πρόκειται για τον constructor της κλάσης
- setAccountList( accountList:list<Account>): void : Μέθοδος που θέτει τη λίστα με τα accounts των χρηστών.
- getAccountList(): list<Account> : Μέθοδος που επιστρέφει τη λίστα με τα accounts των χρηστών.

#### AdminPanelPageGUI

«Boundary» AdminPanelPageGUI	
-lblNumber:	label
-lblProfile:	label
-lblDepartment:	label
-lblRole:	label
-lblCreatedAtTime:	label
-lblLastModified:	label
-userDetailsGUIList:	List<UserDetailsGUI>
+AdminPanelPageGUI(	lblNumber, lblProfile, lblDepartment, lblRole, lblCreatedAtTime, lblLastModified, userDetailsGUIList): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας Admin Panel του Chief Engineer. Δίνει τη δυνατότητα στο χρήστη να δει όλους τους χρήστες, τους ρόλους και τα τμήματά τους.



## Χαρακτηριστικά κλάσης

- IblNumber: label : Ετικέτα με το τίτλο No.
- IblProfile: label : Ετικέτα με το τίτλο Profile.
- IblDepartment: label : Ετικέτα με τον τίτλο Department.
- IblRole: label : Ετικέτα με τον τίτλο Role.
- IblCreatedAtTime: label : Ετικέτα με τον τίτλο Created At Time.
- IblLastModified: label : Ετικέτα με τον τίτλο Last Modified..
- userDetailsGUIList: list<UserDetailsGUI> : Λίστα με αντικείμενα τις διεπαφές userDetails των χρηστών.

## Μέθοδοι κλάσης

AdminPanelPageGUI(IblNumber,IblProfile,IblDepartment,IblRole,IblCreatedAtTime,IblLastModified, userDetailsGUIList): void : Πρόκειται για τον constructor της κλάσης.

## UserDetailsGUI

«Boundary»	
UserDetailsGUI	
-IblUserNumber: label	
-IblUserName: label	
-IblUserDepartment: label	
-IblUserRole: label	
-timeCreated: timestamp	.
-timeLastModified: timestamp	
-btnEdit: button	
+UserDetailGUI(IblUserNumber, IblUserName, IblUserDepartment, IblUserRole, timeCreated, timeLastModified): void	
+select(Button: string): void	

Η κλάση αυτή αφορά τη διεπαφή των πληροφοριών του χρήστη.

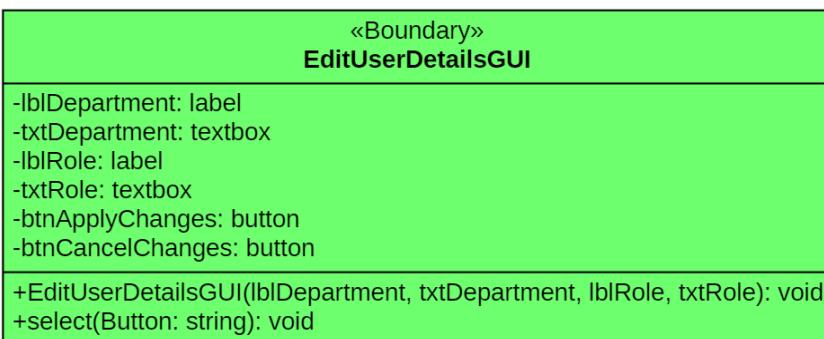
## Χαρακτηριστικά κλάσης

- IblUserNumber: label : Ετικέτα με τον αριθμό του χρήστη.
- IblUserName: label : Ετικέτα με το όνομα του χρήστη.
- IblUserDepartment: label : Ετικέτα με το τμήμα του χρήστη.
- IblUserRole: label : Ετικέτα με το ρόλο του χρήστη.
- timeCreated: timestamp : Χρονική σήμανση με το χρόνο δημιουργίας του χρήστη.
- timeLastModified: timestamp : Χρονική σήμανση με το χρόνο τελευταίας επεξεργασίας του χρήστη.
- btnEdit: button : Κουμπί που επιτρέπει την επεξεργασία του χρήστη.

## Μέθοδοι κλάσης

- UserDetailGUI(IblUserNumber, IblUserName, IblUserDepartment, IblUserRole, timeCreated, timeLastModified): void : Πρόκειται για τον constructor της κλάσης.
- select(Button:string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

## EditUserDetailsGUI



Η κλάση αυτή αφορά τη διεπαφή επεξεργασίας του ρόλου και του τμήματος του χρήστη.

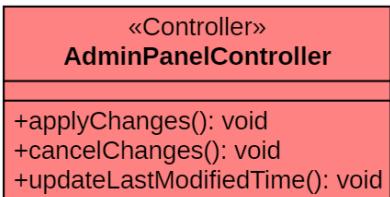
### Χαρακτηριστικά κλάσης

- lblDepartment: label : Ετικέτα με τίτλο Department.
- txtDepartment: textbox : Πεδίο εισαγωγής του τμήματος του χρήστη.
- lblRole: label : Ετικέτα με τίτλο Role.
- txtRole: textbox : Πεδίο εισαγωγής του ρόλου του χρήστη.
- btnApplyChanges: button : Κουμπί που αποθηκεύει τις αλλαγές που έκανε ο χρήστης.
- btnCancelChanges: button : Κουμπί που ακυρώνει τις αλλαγές που έκανε ο χρήστης.

### Μέθοδοι κλάσης

- EditUserDetailsGUI(lblDepartment, txtDepartment, lblRole, txtRole): void : Πρόκειται για τον constructor της κλάσης.
- select(in Button:string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### AdminPanelController

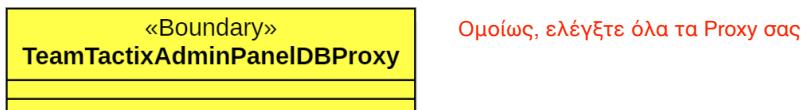


Η κλάση αυτή είναι υπεύθυνη για τη λειτουργίας προβολής και επεξεργασίας στο Admin Panel.

### Μέθοδοι κλάσης

- applyChanges(): void : Μέθοδος που αποθηκεύει τις αλλαγές που έκανε ο χρήστης.
- cancelChanges(): void : Μέθοδος που ακυρώνει τις αλλαγές που έκανε ο χρήστης.
- updateLastModifiedTime(): void : Μέθοδος που ενημερώνει τις πληροφορίες που φαίνονται στο Admin Panel, συμφωνα με τις αλλαγές που έκανε ο χρήσης.

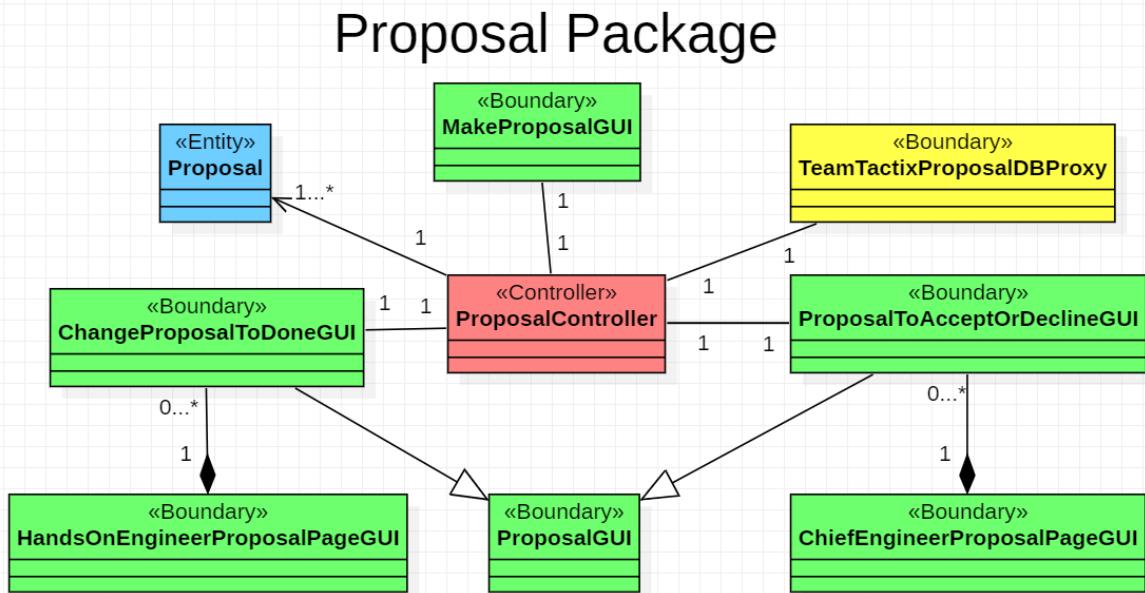
### TeamTactixAdminPanelDBProxy



Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί.



#### 4.1.3 Πακέτο Proposal Package



#### Proposal

<<Entity>> Proposal	
-proposalID: integer	
-partID: integer	
-userID: integer	
-status: string	
-title: string	
-description: string	
-currentValue: float	
-newValue: float	
+Proposal(proposalID, partID, userID, status, title, description, currentValue, newValue): void	
+setProposalID(proposalID: integer): void	
+getProposalID(): integer	
+setPartID(partID: integer): void	
+getPartID(): integer	
+setUserID(userID: integer): void	
+getUserID(): integer	
+setStatus(status: string): void	
+getStatus(): string	
+setTitle(title: string): void	
+getTitle(): string	
+setDescription(description: string): void	
+getDescription(): string	
+setCurrentValue(currentValue: float): void	
+getCurrentValue(): float	
+setNewValue(newValue: float): void	
+getNewValue(): float	

Πρόκειται για την οντότητα της πρότασης που κάνει ο Engineer προς τον Chief Engineer για οποιαδήποτε τροποποίηση στο όχημα.

#### Χαρακτηριστικά κλάσης

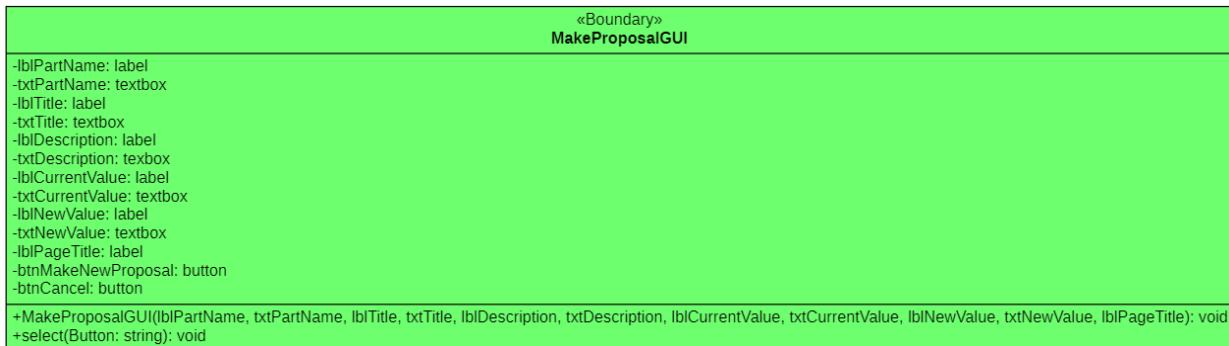
- proposalID: integer: Ακέραιος που αντιπροσωπεύει το ID της κάθε πρότασης.
- partID: integer: Ακέραιος που αντιπροσωπεύει το ID του Part του οχήματος που αφορά η πρόταση.
- status: string: Συμβολοσειρά που αποτελεί την κατάσταση της πρότασης.
- title: string: Συμβολοσειρά που αντιπροσωπεύει τον τίτλο της πρότασης.
- description: string: Συμβολοσειρά που αποτελεί την περιγραφή της πρότασης.
- currentValue: float: Αριθμός κινητής υποδιαστολής που αποτελεί την τρέχουσα τιμή της παραμέτρου που προτείνεται για αλλαγή.
- newValue: float: Αριθμός κινητής υποδιαστολής που αποτελεί την νέα τιμή της παραμέτρου που προτείνεται για αλλαγή.



## Μέθοδοι κλάσης

- Proposal(id, proposalID, partID, userID, status, title, description, currentValue, newValue): void: Πρόκειται για τον Constructor της κλάσης.
- setProposalID(ProposalID:integer ): void: Λειτουργία που θέτει το ID της πρότασης ίσο με τον integer στο όρισμα.
- getProposalID(): integer: Λειτουργία που επιστρέφει το ID της πρότασης.
- setPartID(partID:integer): void: Λειτουργία που θέτει το partID ενός Part ίσο με τον integer στο όρισμα.
- getPartID(): integer: Λειτουργία που επιστρέφει το partID ενός Part.
- setUserID(userID: integer): void: Λειτουργία που θέτει το ID του χρήστη ίσο με τον integer στο όρισμα.
- getPartID(): integer: Λειτουργία που επιστρέφει το ID ενός χρήστη.
- setStatus(status:string): void: Λειτουργία που θέτει το status μίας πρότασης ίσο με το status στο όρισμα.
- getStatus(): string: Λειτουργία που επιστρέφει το status ενός status.
- setTitle(title: string): void: Λειτουργία που θέτει τον Τίτλο της πρότασης ίσο με την συμβολοσειρά στο όρισμα.
- getTitle(): string: Λειτουργία που επιστρέφει τον Τίτλο της πρότασης.
- setDescription(description:string): void: Λειτουργία που θέτει την περιγραφή της πρότασης ίση με την συμβολοσειρά στο όρισμα.
- getDescription(): string: Λειτουργία που επιστρέφει την περιγραφή της πρότασης.
- setCurrentValue(currentValue:float): void: Λειτουργία που θέτει την τρέχουσα τιμή μίας παραμέτρου ίση με τον αριθμό κινητής υποδιαστολής στο όρισμα
- getCurrentValue(): float: Λειτουργία που επιστρέφει την τρέχουσα τιμή μίας παραμέτρου.
- setNewValue(newValue:float): void: Λειτουργία που θέτει την νέα τιμή μίας παραμέτρου ίση με τον αριθμό κινητής υποδιαστολής στο όρισμα.
- get NewValue(): float: Λειτουργία που επιστρέφει την νέα τιμή της παραμέτρου.

## MakeProposalGUI



Η κλάση MakeProposalGUI αφορά την διεπαφή της δημιουργίας προτάσεων από τους Engineers.

## Χαρακτηριστικά κλάσης

- lblPartName: label: Ετικέτα που περιέχει την συμβολοσειρά “Part: ”.
- txtPartName: textbox: Πεδίο εισαγωγής για το όνομα του Part.
- lblTitle: label: Ετικέτα που περιέχει την συμβολοσειρά “Title: ”.
- txtTitle: textbox: Πεδίο εισαγωγής για τον τίτλο της πρότασης.
- lblDescription: label: Ετικέτα που περιέχει την συμβολοσειρά “Description: ”.
- txtDescription: textbox: Πεδίο εισαγωγής για την περιγραφή της πρότασης.
- lblCurrentValue: label: Ετικέτα που περιέχει την συμβολοσειρά “Current Value: ”

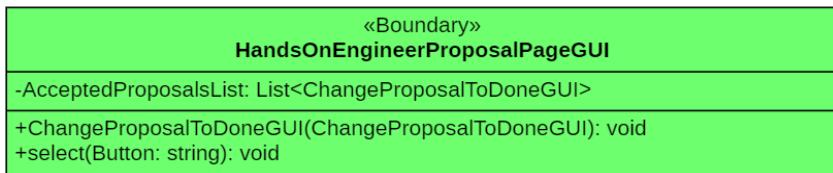


- txtCurrentValue: textbox: Πεδίο εισαγωγής για την τρέχουσα τιμή της παραμέτρου που προτείνεται για αλλαγή.
- lblNewValue: label: Ετικέτα που περιέχει την συμβολοσειρά “New Value: ”.
- txtnewValue: textbox: Πεδίο εισαγωγής για την νέα τιμή της παραμέτρου που προτείνεται για αλλαγή.
- lblPageTitle: label: Ετικέτα που περιέχει τον τίτλο της σελίδας “New Proposal”.
- btnMakeNewProposal: button: Κουμπί που όταν πατηθεί δημιουργείται η πρόταση που έχει εισαχθεί.
- btnCancel: button: Κουμπί που όταν πατηθεί ακυρώνει την δημιουργία της νέας πρότασης.

### Μέθοδοι κλάσης

- MakeProposalGUI(lblPartName, txtPartName, lblTitle, txtTitle, lblDescription, txtDescription, lblCurrentValue, txtCurrentValue, lblNewValue, txtnewValue, lblPageTitle): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### HandsOnEngineerProposalPageGUI



Η κλάση αυτή αφορά την διεπαφή των Hands-On Engineers όσον αφορά την σελίδα με τις προτάσεις που έχουν γίνει αποδεκτές από τον Chief Engineer ώστε όταν αυτές εφαρμοστούν να υπάρχει η επιλογή να αλλάξουν σε status σε “DONE”.

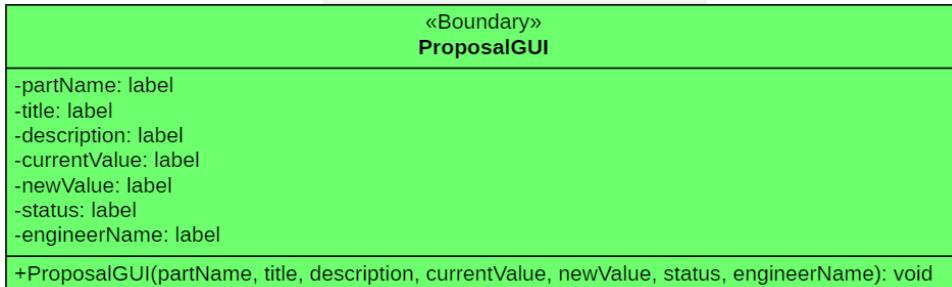
### Χαρακτηριστικά κλάσης

- AcceptedProposalsList: List<ChangeProposalToDoneGUI>: Λίστα με αντικείμενα διεπαφής τύπου ChangeProposalToDoneGUI.

### Μέθοδοι κλασης

- ChangeProposalToDoneGUI(ChangeProposalToDoneGUI): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ProposalGUI



Η κλάση ProposalGUI αντιπροσωπεύει την διεπαφή της πρότασης ως απλή έκδοση και βάση για άλλες κλάσεις που την κληρονομούν.

### Χαρακτηριστικά κλάσης

- partName: label: Ετικέτα που περιέχει το όνομα του Part που αφορά η πρόταση.
- title: label: Ετικέτα που περιέχει τον τίτλο της προκείμενης πρότασης.



- descriprion: label: Ετικέτα που περιέχει την συμβολοσειρά της περιγραφής της πρότασης.
- currentValue: label: Ετικέτα που περιέχει την τρέχουσα τιμή της παραμέτρου που προτείνεται για τροποποίηση.
- newValue: label: Ετικέτα που περιέχει την νέα τιμή της παραμέτρου που προτείνεται για τροποίηση.
- status: label: Ετικέτα που περιέχει το status της πρότασης. Οι δυνατές τιμές του είναι “New”, “Accepted”, “Declined”, “DONE”
- engineerName: label: Ετικέτα που περιέχει το όνομα του Engineer ο οποίος προτείνει την τροποποίηση.

### Μέθοδοι κλάσης

- Proposalgui(partName, title, description, currentValue, newValue, status, engineerName): void: Πρόκειται για τον Constructor της κλάσης.

### ChangeProposalToDoneGUI

«Boundary»	
ChangeProposalToDoneGUI	
-btnDone: button	
+ChangeProposalToDoneGUI(): void	
+Operation1()	

Η κλάση αυτή κληρονομεί την κλάση Proposal και προσθέτει ένα επιπλέον κουμπί. Σε αυτή την διεπαφή, όταν οι εγκεκριμένες προτάσεις εφαρμοστούν από τους Hands-On Engineers υπάρχει η επιλογή να αλλάξουν οι ίδιοι το status των προτάσεων σε “DONE”.

### Χαρακτηριστικά κλάσης

- btnDone: button: Κουμπί που όταν πατηθεί αλλάζει το status των προτάσεων σε “DONE”.

### Μέθοδοι κλάσης

- ChangeProposalToDoneGUI(): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ProposalToAcceptOrDeclineGUI

«Boundary»	
ProposalToAcceptOrDeclineGUI	
-btnAccept: button	
-btnDecline: button	
+ProposalGUI(): void	
+select(Button: string): void	

Η κλάση αυτή αποτελεί την διεπαφή της πρότασης που έχει δημιουργηθεί από τον Engineer και αναμένει αποδοχής από τον Chief Engineer. Κληρονομεί την κλάση Proposal και προσθέτει επιπλέον δύο κουμπιά για την δυνατή αποδοχή και την απόρριψη.

### Χαρακτηριστικά κλάσης

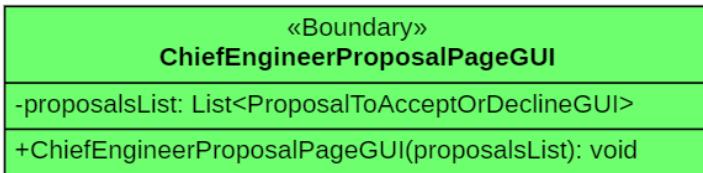
- btnAccept: button: Κουμπί που αν πατηθεί αλλάζει το status της πρότασης από “New” σε “Accepted”.
- btnDecline: button: Κουμπί που αν πατηθεί αλλάζει το status της πρότασης από “New” σε “Declined”.

### Μέθοδοι κλάσης

- Proposalgui(): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.



### ProposalToAcceptOrDeclineGUI



Η προκείμενη κλάση αφορά την διεπαφή της σελίδας του Chief Engineer και περιλαμβάνει μία λίστα με τις προτάσεις που χρήζουν έγκρισης.

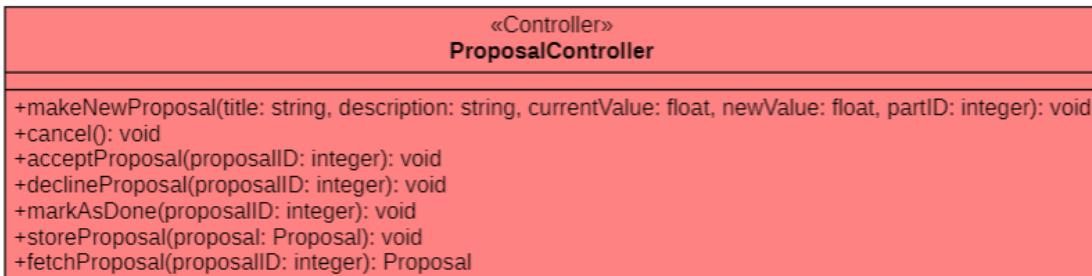
#### Χαρακτηριστικά κλάσης

- proposalsList: List<ProposalToAcceptOrDeclineGUI>: Λίστα με αντικείμενα διεπαφής τις προτάσεις με status “New” που περιμένουν έγκριση.

#### Μέθοδοι κλάσης

- ChiefEngineerProposalPageGUI(proposalList): void: Πρόκειται για τον Constructor της κλάσης.

### ProposalController

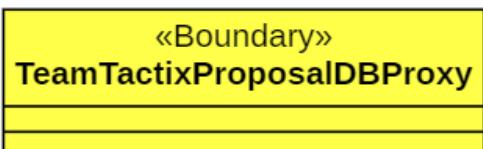


Με την κλάση ProposalController διεκπεραιώνονται οι λειτουργίες που αφορούν το πακέτο Proposals.

#### Μέθοδοι κλάσης

- makeNewProposal(title: string, description: string, currentValue: float, newValue: float, partID: integer): void: Λειτουργία η οποία δημιουργεί μία νέα πρόταση.
- cancel(): void: Λειτουργία που ακυρώνει την δημιουργία πρότασης.
- acceptProposal(proposalID: integer): void: Λειτουργία που αλλάζει το status της πρότασης από “New” σε “Accepted” όταν το επιλέξει ο Chief Engineer.
- declineProposal(proposalID: integer): void: Λειτουργία που αλλάζει το status της πρότασης με το ID του από “New” σε “Accepted” όταν το επιλέξει ο Chief Engineer.
- markAsDone(proposalID: integer): void: Λειτουργία που αλλάζει το status της πρότασης από “Accepted” σε “DONE” όταν το επιλέξει ο Hands-On Engineer.
- storeProposal(proposal: Proposal): void: Λειτουργία που αποθηκεύει την πρόταση του ορίσματος στην βάση δεδομένων.
- fetchProposal(proposalID: integer): Proposal: Λειτουργία που προσκομίζει μία πρόταση από την βάση δεδομένων.

### TeamTactixProposalDBProxy

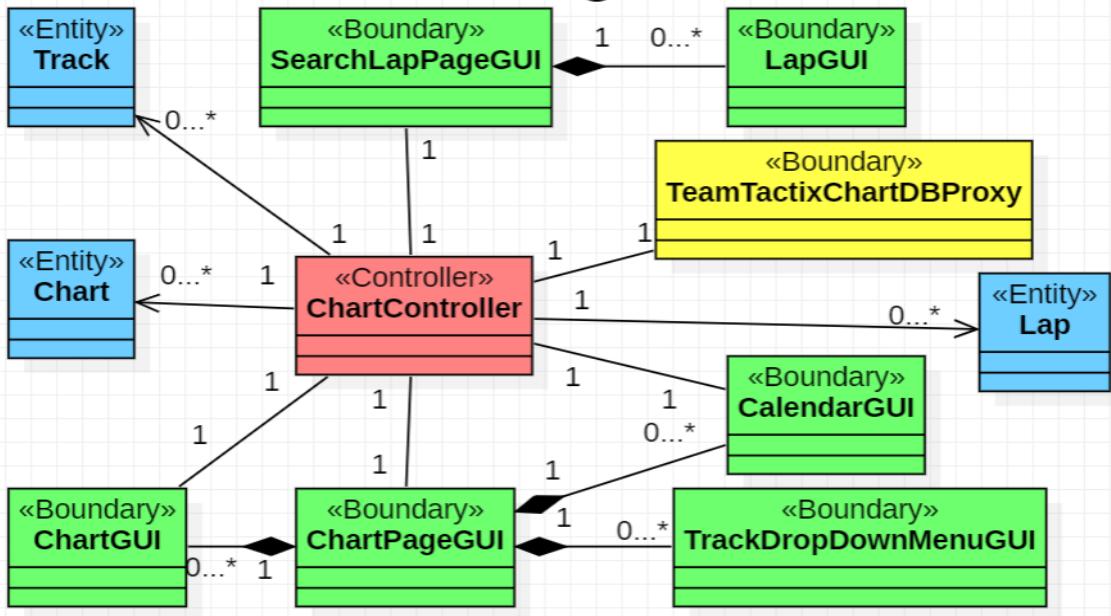




Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί σχετικά με τις προτάσεις.

#### 4.1.4 Πακέτο Chart

## Chart Package



### Chart

<code>&lt;&lt;Entity&gt;&gt;</code>
<code>Chart</code>  - <code>name: string</code> - <code>dataCategory: string</code> - <code>date: timestamp</code> - <code>track: string</code> - <code>lap: integer</code> - <code>data: float</code>  + <code>Chart(name, dataCategory, date, track, lap, data): void</code> + <code>setName(name: string): void</code> + <code>getName(): string</code> + <code>setDataCategory(dataCategory: string): void</code> + <code>getDataCategory(): string</code> + <code> setDate(date: timestamp): void</code> + <code>getDate(): timestamp</code> + <code>setTrack(track: string): void</code> + <code>getTrack(): string</code> + <code>setLap(lap: integer): void</code> + <code>getLap(): integer</code> + <code>setData(data: float): void</code> + <code>getData(): float</code>

Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν το Chart.

### Χαρακτηριστικά κλάσης

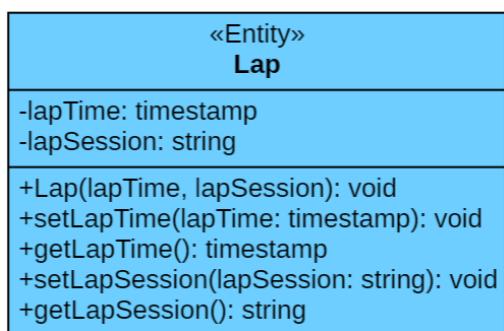


- name: string : Συμβολοσειρά που αντιπροσωπεύει το όνομα του γραφήματος.
- dataCategory: string : Συμβολοσειρά που αντιπροσωπεύει την κατηγορία δεδομένων στο γράφημα.
- date: timestamp : Χρονική σήμανση που αντιπροσωπεύει την ημερομηνία και την ώρα δημιουργίας του γραφήματος.
- track: string : Συμβολοσειρά που αντιπροσωπεύει το όνομα της πίστας που σχετίζεται με το γράφημα.
- lap: integer : Ακέραιος αριθμός που αντιπροσωπεύει τον αριθμό γύρου που σχετίζεται με το γράφημα.
- data: float : Δεκαδικός αριθμός που αντιπροσωπεύει τα πραγματικά δεδομένα του γραφήματος.

### Μέθοδοι κλάσης

- Chart(name, dataCategory, date, track, lap, data): void : Πρόκειται για τον Constructor της κλάσης.
- setName(name: string): void : Μέθοδος που θέτει το όνομα του γραφήματος.
- getName(): string : Μέθοδος που επιστρέφει το όνομα του γραφήματος.
- setDataCategory(dataCategory: string): void : Μέθοδος που θέτει την κατηγορία δεδομένων του γραφήματος.
- getDataCategory(): string : Μέθοδος που επιστρέφει την κατηγορία των δεδομένων του γραφήματος.
- setDate(dae: timestamp): void : Μέθοδος που θέτει την ημερομηνία του γραφήματος.
- getDate(): timestamp : Μέθοδος που επιστρέφει την ημερομηνία του γραφήματος.
- setTrack(track: string): void : Μέθοδος που θέτει το όνομα της πίστας στην οποία ανήκει το γράφημα.
- getTrack(): string : Μέθοδος που επιστρέφει το όνομα της πίστας στην οποία ανήκει το γράφημα.
- setLap(lap: integer): void : Μέθοδος που θέτει τον αριθμό του γύρου του γραφήματος.
- getLap(): integer : Μέθοδος που επιστρέφει τον αριθμό του γύρου του γραφήματος.
- setData(data: float): void : Μέθοδος που θέτει τις τιμές των δεδομένων στους άξονες του γραφήματος.
- getData(): float : Μέθοδος που επιστρέφει τις τιμές των δεδομένων στους άξονες του γραφήματος.

### Lap



Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν το Lap.

### Χαρακτηριστικά κλάσης

- lapTime: timestamp: Χρονική σήμανση που αντιπροσωπεύει την ημερομηνία και την ώρα που δημιουργήθηκε ο γύρος.
- lapSession: string: Συμβολοσειρά που αντιπροσωπεύει το όνομα του Session στον οποίο ανήκει ο γύρος.

### Μέθοδοι κλάσης

- Lap(lapTime, lapSession): void: Πρόκειται για τον Constructor της κλάσης.
- setLapTime(lapTime: timestamp): void: Μέθοδος που θέτει την χρονική σήμανση του γύρου.
- getLapTime(): timestamp: Μέθοδος που επιστρέφει την χρονική σήμανση του γύρου.
- setLapSession(lapSession: string): void: Μέθοδος που θέτει το όνομα του Session στον οποίο ανήκει ο γύρος.
- getLapSession(): string: Μέθοδος που επιστρέφει το όνομα του Session στον οποίο ανήκει ο γύρος.



## Track

«Entity»
Track
-trackName: string
-trackNationImage: image
+Track(trackName, trackNationImage): void
+setTrackName(track: string): void
+getTrackName(): string
+setTrackNationImage(trackNationImage: image): void
+getTrackNationImage(): image

Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν την πίστα.

### Χαρακτηριστικά κλάσης

- trackName: string: Συμβολοσειρά που αντιπροσωπεύει το όνομα της πίστας.
- trackNationImage: image: Εικόνα που αντιπροσωπεύει την σημαία της χώρας στην οποία βρίσκεται η πίστα.

### Μέθοδοι κλάσης

- Track(trackName, trackNationImage): void: Πρόκειται για τον Constructor της κλάσης.
- setTrackName(track: string): void: Μέθοδος που θέτει το όνομα της πίστας.
- getTrackName(): string: Μέθοδος που επιστρέφει το όνομα της πίστας.
- setTrackNationImage(trackNationImage: Image): void: Μέθοδος που θέτει την σημαία της χώρας της πίστας.
- getTrackNationImage(): image: Μέθοδος που επιστρέφει την σημαία της χώρας της πίστας.

## ChartGUI

«Boundary»
ChartGUI
-isVisible: boolean
-dataCategory: string
-timeList: List<timestamp>
-dataList: List<float>
-btnZoomInBar: button
+ChartGUI(isVisible, dataCategory, timeList, dataList, btnZoomInBar): void
+select(Button: string): void

Η κλάση αυτή αφορά την διεπαφή του Chart των δεδομένων της τηλεμετρίας με τον χρήστη.

### Χαρακτηριστικά Κλάσης

- isVisible: boolean: Τύπος δεδομένων Αλήθειας που αντιπροσωπεύει αν το διάγραμμα είναι ορατό.
- dataCategory: string: Συμβολοσειρά που αντιπροσωπεύει την κατηγορία των δεδομένων.
- timeList: List<timestamp>: Λίστα από αντικείμενα χρονικής σήμανσης που αντιπροσωπεύουν τον άξονα x του διαγράμματος.
- dataList: List<float>: Λίστα από αντικείμενα float των τιμών των δεδομένων που αντιπροσωπεύουν τον άξονα y του διαγράμματος.
- btnZoomInBar: button: Κουμπί που αν πατηθεί κάνει Zoom In και Zoom Out στο διάγραμμα.

### Μέθοδοι Κλάσης

- ChartGUI(isVisible, dataCategory, timeList, dataList, btnZoomInBar): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.



## ChartPageGUI

«Boundary»
ChartPageGUI
-lblLap: label
-DataCategoryList: List <label>
-isCheckedChangeListener: List<boolean>
-chartGUI: ChartGUI
-btnFetchData: button
-btnSendDataToChat: button
-btnSearchLap: button
-btnSelectDate: button
-btnSelectTrack: button
+ChartPageGUI(lblLap, txtLap, lblDataCategoryList, isCheckedList, chartGUI, btnFetchData, btnSendDataToChart): void
+select(Button: string): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας Chart από την οποία ο χρήστης μπορεί να επιλέξει και να δει τα δεδομένα τηλεμετρίας του οχήματος με την μορφή διαγράμματος.

### **Χαρακτηριστικά Κλάσης**

- lblLap: label: Ετικέτα με τίτλο Lap.
- DataCategoryList: List <label>: Λίστα από ετικέτες των κατηγοριών των δεδομένων τηλεμετρίας του διαγράμματος.
- isCheckedList: List<boolean>: Λίστα από αντικείμενα boolean για την εμφάνιση των επιλεγμένων κατηγοριών των δεδομένων στο διάγραμμα.
- chartGUI: ChartGUI: Αντικείμενο τύπου ChartGUI.
- btnFetchData: button: Κουμπί που αν πατηθεί προσκομίζει τα δεδομένα τηλεμετρίας από την βάση δεδομένων.
- btnSendDataToChat: button: Κουμπί που αν πατηθεί στέλνει το διάγραμμα στο επιλεγμένο τσατ.
- btnSearchLap: button: Κουμπί που αν πατηθεί εμφανίζει τα διαθέσιμα Lap της επιλεγμένης ημερομηνίας και πίστας
- btnSelectDate: button: Κουμπί που αν πατηθεί επιτρέπει τον χρήστη να διαλέξει ημερομηνία από το ημερολόγιο.
- btnSelectTrack: button: Κουμπί που αν πατηθεί εμφανίζεται μια λίστα με τις πίστες που μπορεί ο χρήστης να επιλεξει.

### **Μέθοδοι Κλάσης**

- ChartPageGUI(lblLap, txtLap, lblDataCategoryList, isCheckedList, chartGUI, btnFetchData, btnSendDataToChart): void: Πρόκειται για τον Contructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

## LapGUI

«Boundary»
LapGUI
-lblLapNumber: label
-lblLapTime: label
-lblLapSession: label
-btnSelectLapToViewDataFrom: button
-btnBackChartPageGUI: button
+LapGUI(lblLapNumber, lblLapTime, lblLapSession): void
+select(Button: string): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας LapGUI που αφορά την επιλογή του γύρου (Lap) από τον χρήστη προκειμένου να δει τα δεδομένα τηλεμετρίας αυτού.

### **Χαρακτηριστικά Κλάσης:**



- lblLapNumber: label: Ετικέτα με τίτλο τον αριθμό του Lap.
- lblLapTime: label: Ετικέτα με τίτλο την χρονική στιγμή που δημιουργήθηκε το Lap.
- lblLapSession: label: Ετικέτα με τίτλο την συνεδρία (Session) στην οποία ανήκει το Lap.
- btnSelectLapToViewDataFrom: button: Κουμπί που αν πατήσει ο χρήστης θα έχει επιλέξει το Lap που θέλει να εμφανίσει.
- btnBackChartPageGUI: button: Κουμπί που αν πατηθεί ο χρήστης επιστρέφει στην προηγούμενη σελίδα και οι αλλαγές αναιρούνται.

### **Μέθοδοι Κλάσης:**

- LapGUI(lblLapNumber, lblLapTime, lblLapSession): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### **SearchLapGUI**

«Boundary»
SearchLapPageGUI
-lblTrackName: label -lblDate: label -lblLapNumber: label -lblLapTime: label -lblLapSession: label -btnLapGUIList: List<LapGUI>
+SearchLapPageGUI(lblTrackName, lblDate, lblLapNumber, lblLapTime, lblLapSession, LapGUIList): void +select(Button: string): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας αναζήτησης ενός lap στη σελίδα Chart Page.

### **Χαρακτηριστικά κλάσης**

- lblTrackName: label : Ετικέτα με τίτλο Track Name.
- lblDate: label : Ετικέτα με τίτλο Date.
- lblLapNumber: label : Ετικέτα με τίτλο Lap Number.
- lblLapTime: label : Ετικέτα με τίτλο Time.
- lblLapSession: label : Ετικέτα με τίτλο Lap Session.
- btnLapGUIList: list<LapGUI> : Λίστα με κουμπιά. Κάθε κουμπί σημαίνει την επιλογή του συγκεκριμένου Lap.

### **Μέθοδοι κλάσης**

- SearchLapPageGUI(lblTrackName, lblDate, lblLapNumber, lblLapTime, lblLapSession, LapGUIList): void : Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### **TrackDropDownMenuGUI**

«Boundary»
TrackDropDownMenuGUI
-countryImage: List<image> -btnTrackList: List<button>
+TrackDropDownMenuGUI(countryImage): void +select(Button: string): void

Η κλάση αυτή αφορά την διεπαφή ενός Drop Down μενού στο οποίο ο χρήστης βλέπει όλες τις πιθανές χώρες και πίστες που υπάρχουν και μπορεί να επιλέξει αυτή που τον ενδιαφέρει.

### **Χαρακτηριστικά κλάσης**

- countryImage: List<image>: Λίστα με αντικείμενα εικόνες των σημαιών των χωρών.



- btnTrackList: List<button>: Λίστα από κουμπιά που αν πατηθούν ο χρήστης θα έχει επιλέξει την χώρα στην οποία βρίσκεται η πίστα του.

### Μέθοδοι κλάσης

- TrackDropDownMenuGUI(countryImage): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### CalendarGUI

«Boundary»	CalendarGUI
-lblMonthName: label	
-lblNameOfTheDayList: List<label>	
-lblDayIndexList: List<label>	
-dayList: List<boolean>	
-btnNextMonth: button	
-btnPreviousMonth: button	
-btnCancelCalendar: button	
-btnOkayCalendar: button	
+CalendarGUI(lblMonthName, lblNameOfTheDayList, lblDayIndexList, dayList): void	
+select(Button: string): void	

Η κλάση αυτή αφορά τη διεπαφή της σελίδας Calendar που προκύπτει όταν ο χρήστης πατήσει το κουμπί για να επιλέξει την ημερομηνία που θέλει να δει τα δεδομένα της τηλεμετρίας του οχήματος.

### Χαρακτηριστικά κλάσης

- lblMonthName: label: Ετικέτα με τίτλο το όνομα του μήνα.
- lblNameOfTheDayList: List<label>: Λίστα από ετικέτες που έχουν όνομα τις ημέρες της εβδομάδας.
- lblDayIndexList: List<label>: Λίστα από ετικέτες που έχουν όνομα τους αριθμούς που έχει κάθε μέρα του μήνα.
- dayList: List<boolean>: Λίστα από boolean. Κάθε αντικείμενο είναι μια μέρα του επιλεγμένου μήνα.
- btnNextMonth: button: Κουμπί που αν πατηθεί ο χρήστης μπορεί να δει τις ημερομηνίες του επόμενου μήνα.
- btnPreviousMonth: button: Κουμπί που αν πατηθεί ο χρήστης μπορεί να δει τις ημερομηνίες του προηγούμενου μήνα.
- btnCancelCalendar: button: Κουμπί που αν πατηθεί ο χρήστης επιστρέφει στην προηγούμενη σελίδα και οι αλλαγές αναιρούνται.
- btnOkayCalendar: button: Κουμπί που αν πατηθεί έχει επιλεχθεί η ημερομηνία που θέλει ο χρήστης.

### Μέθοδοι κλάσης

- CalendarGUI(lblMonthName, lblNameOfTheDayList, lblDayIndexList, dayList): void: Πρόκειται για τον Constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ChartController



**«Controller»**  
**ChartController**

```
+fetchChart(): Chart
+sendChartToChat(): void
+zoomIn(): void
+zoomOut(): void
+searchLap(): void
+selectLapToViewDataFrom(): void
+saveTheDateSelected(): void
+selectTrack(): void
+selectDay(): void
+changeMonth(): void
```

Η κλάση αυτή αφορά τη λειτουργία προβολής και επεξεργασίες των Charts.

**Μέθοδοι κλάσης**

- fetchChart(): Chart : Λειτουργία που προσκομίζει το Chart.
- sendChartToChat(): void : Λειτουργία που αποστέλλει το Chart στο Chat.
- zoomIn(): void : Λειτουργία που κάνει zoom in στο Chart.
- zoomOut(): void : Λειτουργία που κάνει zoom out στο Chart.
- searchLap(): void : Λειτουργία που κάνει αναζήτηση του Lap.
- selectLapToViewDataFrom(): void : Λειτουργία που κάνει την επιλογή του Lap.
- saveTheDateSelected(): void : Λειτουργία που αποθηκεύει την ημερομηνία που επιλέχθηκε.
- selectTrack(): void : Λειτουργία που κάνει την επιλογή της πίστας.
- selectDay(): void : Λειτουργία που κάνει την επιλογή της ημέρας.
- changeMonth(): void : Λειτουργία που κάνει την αλλαγή του μήνα.

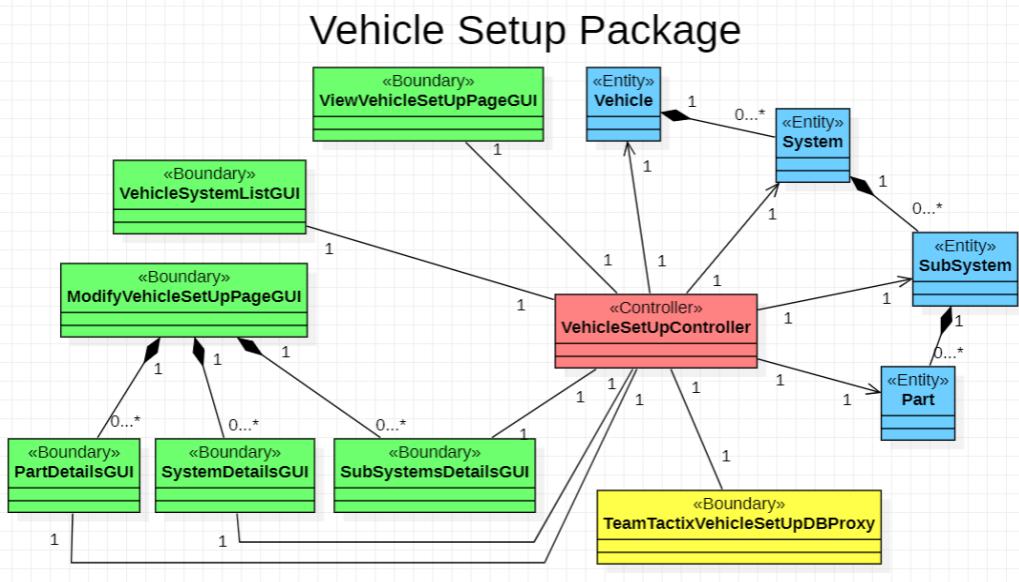
**TeamTactixChartDBProxy**

**«Boundary»**  
**TeamTactixChartDBProxy**

Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί σχετικά με τις προτάσεις.



#### 4.1.5 Πακέτο Vehicle Setup Package



#### Vehicle

<b>«Entity» Vehicle</b>	
-vehicleName: string	
-vehicleYear: integer	
-vehicleDescription: string	
-systemList: List<System>	
+Vehicle(vehicleName, vehicleYear, vehicleDescription, systemList): void	
+setVehicleName(vehicleName: string): void	
+getVehicleName(): string	
+setVehicleYear(vehicleYear: integer): void	
+getVehicleYear(): integer	
+setVehicleDescription(vehicleDescription: string): void	
+getVehicleDescription(): string	
+setSystemList(systemList: List<System>): void	
+getSystemList(): List<System>	

Η κλάση αυτή έχει όλες τις πληροφορίες που αφορούν ένα όχημα.

#### Χαρακτηριστικά κλάσης

- vehicleName: string: Αλφαριθμητικό που αντιπροσωπεύει το όνομα του οχήματος.
- vehicleYear: integer: Ακέραιος που αντιπροσωπεύει το έτος δημιουργίας του οχήματος.
- vehicleDescription: string: Αλφαριθμητικό που περιγράφει το όχημα.
- systemList: List<System>: Λίστα με αντικείμενα της οντότητας System.

#### Μέθοδοι κλάσης

- Vehicle(vehicleName, vehicleYear, vehicleDescription, systemList): void: Πρόκειται για τον constructor της κλάσης.
- setVehicleName(vehicleName:string): void: Λειτουργία που θέτει το όνομα του οχήματος.
- getVehicleName(): string: Λειτουργία που επιστρέφει το όνομα του οχήματος.
- setVehicleYear(vehicleYear:integer): void: Λειτουργία που θέτει το έτος δημιουργίας του οχήματος.
- getVehicleYear(): integer: Λειτουργία που επιστρέφει το έτος δημιουργίας του οχήματος.
- setVehicleDescription(vehicleDescription:string): void: Λειτουργία που θέτει την περιγραφή του οχήματος.
- getVehicleDescription(): string: Λειτουργία που επιστρέφει την περιγραφή του οχήματος.



- `setSystemList(systemList>List<System>): void`: Λειτουργία που θέτει μία λίστα με αντικείμενα τύπου System.
- `getSystemList(): List<System>`: Λειτουργία που επιστρέφει μία λίστα με αντικείμενα τύπου System.

## System

«Entity»
System
-systemName: string
-systemDescription: string
-subSystemList: List<SubSystem>
+System(systemName, systemDescription, subSystemList): void
+setSystemName(systemName: string): void
+getSystemName(): string
+setSystemDescription(systemDescription: string): void
+getSystemDescription(): string
+setSubSystemList(subSystemList: List<SubSystem>): void
+getSubSystemList(): List<SubSystem>

Η κλάση αυτή έχει όλες τις πληροφορίες που αφορούν ένα σύστημα του οχήματος.

### Χαρακτηριστικά κλάσης

- `systemName: string`: Αλφαριθμητικό που αντιπροσωπεύει το όνομα του συστήματος.
- `systemDescription: string`: Αλφαριθμητικό που περιγράφει το σύστημα.
- `subSystemList: List<SubSystem>`: Λίστα με αντικείμενα της οντότητας SubSystem.

### Μέθοδοι κλάσης

- `System(systemName,systemDescription,subSystemList): void`: Πρόκειται για τον constructor της κλάσης.
- `setSystemName(systemName:string): void`: Λειτουργία που θέτει το όνομα του συστήματος.
- `getSystemName(): string`: Λειτουργία που επιστρέφει το όνομα του συστήματος.
- `setSystemDescription(systemDescription:string): void`: Λειτουργία που θέτει την περιγραφή του συστήματος.
- `getSystemDescription(): string`: Λειτουργία που επιστρέφει την περιγραφή του συστήματος.
- `setSubSystemList(subSystemList>List<SubSystem>): void`: Λειτουργία που θέτει μία λίστα με αντικείμενα τύπου SubSystem.
- `getSubSystemList(): List<SubSystem>`: Λειτουργία που επιστρέφει μία λίστα με αντικείμενα τύπου SubSystem.

## SubSystem

«Entity»
SubSystem
-subSystemName: string
-subSystemDescription: string
-partList: List<Part>
+SubSystem(subSystemName, subSystemDescription, partList): void
+setSubSystemName(subSystemName: string): void
+getSubSystemName(): string
+setSubSystemDescription(subSystemDescription: string): void
+getSubSystemDescription(): string
+setPartList(partList: List<Part>): void
+getPartList(): List<Part>

Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν ένα υποσύστημα του οχήματος.

### Χαρακτηριστικά κλάσης

- `subSystemName: string`: Αλφαριθμητικό που αντιπροσωπεύει το όνομα του υποσυστήματος.

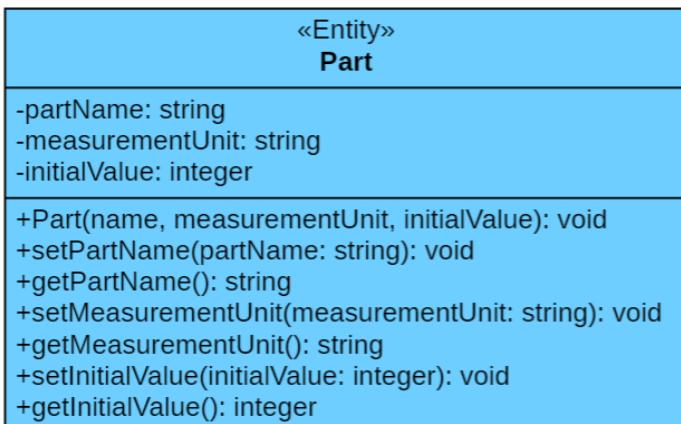


- subSystemDescription: string: Ακέραιος που περιγράφει το υποσύστημα.
- partList: List<Part>: Λίστα με αντικείμενα της οντότητας Part.

### Μέθοδοι κλάσης

- SubSystem(subSystemName,subSystemDescription,partList): void: Πρόκειται για τον constructor της κλάσης.
- setSubSystemName(subSystemName:string):void: Λειτουργία που θέτει το όνομα του υποσυστήματος.
- getSubSystemName(): string: Λειτουργία που επιστρέφει το όνομα του υποσυστήματος.
- setSubSystemDescription(subSystemDescription:string): void: Λειτουργία που θέτει την περιγραφή του υποσυστήματος.
- getSubSystemDescription(): string: Λειτουργία που επιστρέφει την περιγραφή του υποσυστήματος.
- setPartList(partList>List<Part>): void: Λειτουργία που θέτει μία λίστα με αντικείμενα τύπου Part.
- getPartList(): List<Part>: Λειτουργία που επιστρέφει μία λίστα με αντικείμενα τύπου Part

### Part



Η κλάση αυτή περιέχει όλες τις πληροφορίες που αφορούν ένα κομμάτι του οχήματος.

### Χαρακτηριστικά κλάσης

- subSystemName: string: Αλφαριθμητικό που αντιπροσωπεύει το όνομα του Part.
- measurementUnit: integer: Αλφαριθμητικό που αντιπροσωπεύει την μονάδα μέτρησης.
- initialValue: integer: Ακέραιος που αντιπροσωπεύει την τωρινή τιμή του Part.

### Μέθοδοι κλάσης

- Part(name,measurementUnit,initialValue): void: Πρόκειται για τον constructor της κλάσης.
- setPartName(partName:string): void: Λειτουργία που θέτει το όνομα στο Part.
- getPartName(): string: Λειτουργία που επιστρέφει το όνομα του Part.
- setMeasurementUnit(measurementUnit:string): void: Λειτουργία που θέτει την μονάδα μέτρησης.
- getMeasurementUnit(): string: Λειτουργία που επιστρέφει την μονάδα μέτρησης.
- setInitialValue(initialValue:integer): void: Λειτουργία που θέτει την τωρινή τιμή του Part.
- getInitialValue(): integer: Λειτουργία που επιστρέφει την τωρινή τιμή του Part



## PartDetailsGUI

«Boundary» PartDetailsGUI	
-lblName: label	
-txtName: textbox	
-lblMeasurementUnit: label	
-txtMeasurementUnit: textbox	
-lblInitialValue: label	
-txtInitialValue: textbox	
-btnDeletePart: button	
+PartDetailsGUI(lblName, txtName, lblMeasurementUnit, txtMeasurementUnit, lblInitialValue, txtInitialValue): void	
+select(Button: string): void	

Η κλάση αυτή αφορά την διεπαφή μεταξύ της σελίδα επεξεργασίας ενός κομματιού του οχήματος και του χρήστη.

### Χαρακτηριστικά κλάσης

- lblName: label: Ετικέτα που έχει το όνομα του Part.
- txtName: textbox: Πεδίο εισαγωγής για το όνομα του Part.
- lblMeasurement: label: Ετικέτα που γράφει Description.
- txtMeasurement: textbox: Πεδίο εισαγωγής για την μονάδα μέτρησης.
- lblInitialValue: label: Ετικέτα για την τωρινή τιμή του Part.
- txtInitialValue: textbox: Πεδίο εισαγωγής για την τωρινη τιμή του Part.
- btnDeletePart: button: Κουμπί που αν πατηθεί ενεργοποιεί το delete action για το συγκ

### Μέθοδοι κλάσης

- PartDetailsGUI(lblName, txtName, lblMeasurementUnit, txtMeasurementUnit, lblInitialValue, txtInitialValue): void: Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

## SubSystemDetailsGUI

«Boundary» SubSystemsDetailsGUI	
-lblName: label	
-txtName: textbox	
-lblDescription: label	
-txtDescription: textbox	
-lblPartDetailsList: List<PartDetailsGUI>	
-btnAddPart: button	
-btnDeleteSubSystem: button	
+SubSystemDetailsGUI(lblName, txtName, lblDescription, txtDescription, lblPartDetailsList): void	
+select(Button: string): void	

Η κλάση αυτή αφορά την διεπαφή μεταξύ της σελίδα επεξεργασίας ενός υποσυστήματος του οχήματος και του χρήστη.

### Χαρακτηριστικά κλάσης

- lblName: label: Ετικέτα που έχει το όνομα του SubSystem.
- txtName: textbox: Πεδίο εισαγωγής για το όνομα του SubSystem.
- lblDescription: label: Ετικέτα description.
- txtDescription: textbox: Πεδίο εισαγωγής για την περιγραφή του SubSystem.
- lblPartDetailsList: List<PartDetailsGUI>: Λίστα με αντικείμενα διεπαφής τύπου Part.



- btnAddPart: button: Κουμπί που αν πατηθεί δημιουργείται καινούργιο SubSystem.
- btnDeleteSubSystem: button: Κουμπί που αν πατηθεί ενεργοποιεί το delete action για το SubSystem.

### Μέθοδοι κλάσης

- SubSystemDetailsGUI(lblName,txtName,lblDescription,txtDescription): void: Πρόκειται για τον constructor της κλάσης.
- select(Button:string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### SystemDetailsGUI

«Boundary» SystemDetailsGUI	
-lblName: label	
-txtName: textbox	
-lblDescription: label	
-txtDescription: textbox	
-lblSubSystemDetailsList: List<SubSystemDetailsGUI>	
-btnAddSubsystem: button	
-btnDeleteSystem: button	
+SystemDetailsGUI(lblName, txtName, lblDescription, txtDescription, lblSubSystemDetailsList): void	
+select(Button: string): void	

Η κλάση αυτή αφορά την διεπαφή μεταξύ της σελίδα επεξεργασίας ενός συστήματος του οχήματος και του χρήστη.

### Χαρακτηριστικά κλάσης

- lblName: label: Ετικέτα που έχει το όνομα του System.
- txtName: textbox: Πεδίο εισαγωγής για το όνομα του System.
- lblDescription: label: Ετικέτα description.
- txtDescription: textbox: Πεδίο εισαγωγής για την περιγραφή του System.
- lblSubSystemDetailsList: list<SubSystemDetailsGUI> : Λίστα με αντικείμενα διεπαφής με τις λεπτομέρειες του SubSystem.
- btnAddSubSystem: button: Κουμπί που αν πατηθεί δημιουργείται καινούργιο SubSystem.
- btnDeleteSystem: button: Κουμπί που αν πατηθεί ενεργοποιεί το delete action για το System

### Μέθοδοι κλάσης

- SystemDetailsGUI(in lblName, in txtName, in lblDescription, in txtDescription): void
- select(Button:string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ModifyVehicleSetUpPageGUI

«Boundary» ModifyVehicleSetUpPageGUI	
-lblVehicleDetails: label	
-txtVehicleName: textbox	
-txtYear: textbox	
-txtVehicleDescription: textbox	
-systemList: List<SystemDetailsGUI>	
-btnAddSystem: button	
-btnBackToVehiclePageGUI: button	
-btnSaveChanges: button	
+ModifyVehicleSetUpPageGUI(lblVehicleDetails, txtVehicleName, txtYear, txtVehicleDescription, systemList): void	
+select(Button: string): void	



Η προκείμενη κλάση αφορά την διεπαφή της σελίδας που αφορά την τροποποίηση των παραμέτρων του Vehicle Setup. Αφορά μόνο τον Chief Engineer.

### Χαρακτηριστικά κλάσης

- lblVehicleDetail: label: Ετικέτα που περιέχει την συμβολοσειρά “VehicleDetails: ”.
- txtVehicleName: textbox: Πλαίσιο εισαγωγής του ονόματος του οχήματος.
- txtYear: textbox: Πλαίσιο εισαγωγής της χρονολογίας.
- txtVehicleDescription: textbox: Πλαίσιο εισαγωγής της περιγραφής του οχήματος.
- systemList: List<SystemDetailsGUI>: Λίστα με αντικείμενα διεπαφής τύπου SystemDetailsGUI.
- btnAddSystem: button: Κουμπί που αν πατηθεί προσθέτει ένα σύστημα στο Vehicle Setup.
- btnBackToVehiclePageGUI: button: Κουμπί που αν πατηθεί ανακατευθύνει τον χρήστη στην διεπαφή της σελίδας του Vehicle Setup.
- btnSaveChanges: button: Κουμπί που αν πατηθεί αποθηκεύει τις αλλαγές στις τροποποιήσεις του Vehicle Setup.

### Μέθοδοι κλάσης

- ModifyVehicleSetUpPageGUI(lblVehicleDetails, txtVehicleName, txtYear, txtVehicleDescription, systemList): void: Πρόκειται για τον Constructor της κλάσης.

### ViewVehicleSetupPageGUI

«Boundary»	
ViewVehicleSetupPageGUI	
-lblVehicleName: label	
-lblVehicleYear: label	
-lblVehicleDescription: label	
-lblSubSystemNameList: List<label>	
-lblPartNameList: List<label>	
+ViewVehicleSetUpMenuPageGUI(lblVehicleName, lblVehicleYear, lblVehicleDescription, lblSubSystemList): void	

### Χαρακτηριστικά κλάσης

- lblVehicleName: label: Ετικέτα που απεικονίζει το όνομα του οχήματος.
- lblVehicleYear: label: Ετικέτα που απεικονίζει το έτος δημιουργίας του οχήματος.
- lblVehicleDescription: label: Ετικέτα που δείχνει την περιγραφή του οχήματος.
- lblSubSystemNameList: List<label>: Λίστα με ετικέτες που έχουν τα ονόματα των SubSystem.
- lblPartNameList: List<label>: Λίστα με ετικέτες που έχουν τα ονόματα των Part.

### Μέθοδοι κλάσης

- ViewVehicleSetUpMenuPageGUI(lblVehicleName, lblVehicleYear, lblVehicleDescription, lblSubSystemList): void: Πρόκειται για τον constructor της κλάσης.

### VehicleSystemListGUI

«Boundary»	
VehicleSystemListGUI	
-btnVehicleSystemList: List<button>	
+VehicleSystemListGUI(): void	
+select(Button: string): void	

Η κλάση αυτή αφορά την διεπαφή μεταξύ της σελίδας επεξεργασίας ενός υποσυστήματος του οχήματος και του χρήστη.

### Χαρακτηριστικά κλάσης

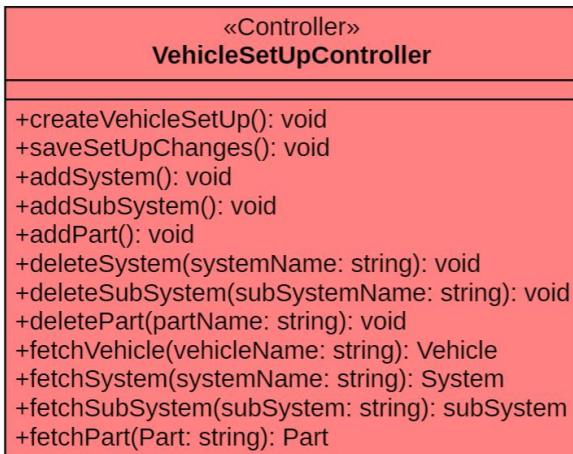


- btnVehicleSystemList: List<button>: Λίστα με κουμπιά με τα οποία ο χρήστης περιηγείται μεταξύ των διαφόρων συστημάτων του οχήματος.

### Μέθοδοι κλάσης

- VehicleSystemListGUI(): void: Πρόκειται για τον constructor της κλάσης.
- select(in Button:string): void: Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### VehicleSetUpController

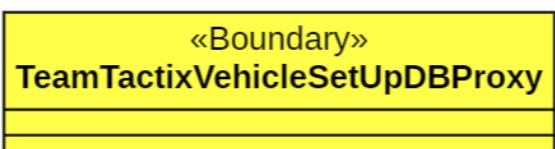


Ο Controller αυτός είναι υπεύθυνος για τις λειτουργίες στο Vehicle Setup πακέτο.

### Μέθοδοι κλάσης

- createVehicleSetUp(): void: Λειτουργία που δημιουργεί ένα Vehicle Setup.
- saveSetUpChanges(): void: Λειτουργία που σώζει τις αλλαγές στο Vehicle.
- addSystem(): void: Λειτουργία που προσθέτει ένα νέο System.
- addSubSystem(): void: Λειτουργία που προσθέτει ένα νέο SubSystem.
- addPart(): void: Λειτουργία που προσθέτει ένα νέο Part.
- deleteSystem(systemName:string): void: Λειτουργία που έχει ως όρισμα το όνομα του System και το διαγράφει.
- deleteSubSystem(subSystemName:string): void: Λειτουργία που έχει ως όρισμα το όνομα του SubSystem και το διαγράφει.
- deletePart(partName:string): void: Λειτουργία που έχει ως όρισμα το όνομα του Part και το διαγράφει.
- fetchVehicle(vehicleName:string): Vehicle: Λειτουργία που έχει ως όρισμα το όνομα του Vehicle και προσκομίζει ένα αντικείμενο τύπου Vehicle από το database.
- fetchSystem(systemName:string): System: Λειτουργία που έχει ως όρισμα το όνομα του System και προσκομίζει ένα αντικείμενο τύπου System από το database.
- fetchSubSystem(subSystem:string): subSystem: Λειτουργία που έχει ως όρισμα το όνομα του SubSystem και προσκομίζει ένα αντικείμενο τύπου SubSystem από το database.
- fetchPart(Part:string): Part: Λειτουργία που έχει ως όρισμα το όνομα του Part και προσκομίζει ένα αντικείμενο τύπου Part από το database.

### TeamTactixVehicleSetUpDBProxy



.....



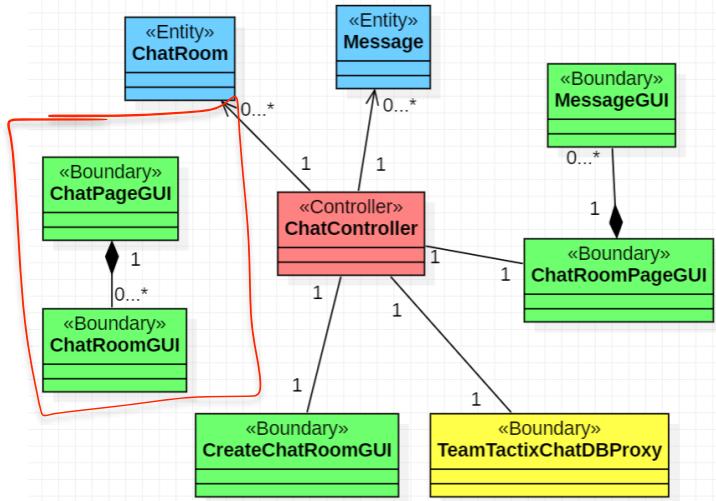
Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί σχετικά με τις προτάσεις.

## 4.2 <Πακέτο λεξιλογίου σεναρίων μέσης προτεραιότητας>

### 4.2.1 Πακέτο Chat

#### Chat Package

Αν κάποιες κλάσεις εμφανίζονται αποκομιμένες από τις υπόλοιπες στο διάγραμμα, τότε μάλλον θα έπρεπε να ανήκουν σε άλλο διάγραμμα.



#### ChatRoom

«Entity» ChatRoom
-chatRoomName: string
-userList: List<Account>
-messageList: List<Message>
-chatRoomID: integer
-chatRoomIcon: image
+ChatRoom(chatRoomName, userList, messageList, chatRoomID): void
+setChatRoomName(chatRoomName: string): void
+getChatRoomName(): string
+setUserList(userList: List<Account>): void
+getUserList(): List<Account>
+setMessageList(messageList: List<Message>): void
+getMessageList(): List<Message>
+setChatRoomID(chatRoomID: integer): void
+getChatRoomID(): integer
+setChatRoomIcon(chatRoomIcon: image): void
+getChatRoomIcon(): image

Η κλάση αυτή αφορούν όλες τις πληροφορίες που αφορούν τα Chat Rooms.

#### Χαρακτηριστικά κλάσης

- chatRoomName: string : Αλφαριθμητικό που αντιπροσωπεύει το όνομα του Chat Room.
- userList: list<Account> : Λίστα που περιέχει το σύνολο των χρηστών και των πληροφοριών τους.
- messageList: List<Message> : Λίστα που περιέχει το σύνολο των μηνυμάτων του Chat Room.
- chatRoomID: integer : Ακέραιος που αντιπροσωπεύει το ID του Chat Room.
- chatRoomIcon: image : Εικόνα του Chat Room.

#### Μέθοδοι κλάσης

- ChatRoom(chatRoomName, userList, messageList, chatRoomID): void : Πρόκειται για τον constructor της κλάσης.
- setChatRoomName(in chatRoomName:string): void : Λειτουργία που θέτει το όνομα του Chat Room.



- getChatRoomName(): string : Λειτουργία που επιστρέφει το όνομα του Chat Room.
- setUserList(in userList>List<Account>): void : Λειτουργία που θέτει τη λίστα με τους χρήστες του Chat Room.
- getUserList(): List<Account> : Λειτουργία που επιστρέφει τη λίστα με τους χρήστες του Chat Room.
- setMassageList(in massageList>List<Massage>): void : Λειτουργία που θέτει τη λίστα με τα μηνύματα του Chat Room.
- getMessageList(): List<Massage> : Λειτουργία που επιστρέφει τη λίστα με τα μηνύματα του Chat Room.
- setChatRoomID(in chatRoomID:integer): void : Λειτουργία που θέτει το ID του Chat Room.
- getChatRoomID(): integer : Λειτουργία που επιστρέφει το ID του Chat Room.
- setChatRoomIcon(in chatRoomIcon:image): void : Λειτουργία που θέτει την εικόνα του Chat Room.
- getChatRoomIcon(): image : Λειτουργία που επιστρέφει την εικόνα του Chat Room.

## Message

«Entity»
Message
-text: string
-time: timestamp
-senderID: integer
-messageID: string
-senderImage: image
+Message(text, time, userID, messageID): void
+setText(text: string): void
+getText(): string
+setTime(time: Timestamp): void
+getTime(): Timestamp
+setSenderId(userID: integer): void
+getSenderId(): integer
+setMessageID(messageID: string): void
+getMessageID(): string
+setSenderImage(senderImage: image): void
+getSenderImage(): image

Η κλάση αυτή περιλαμβάνει όλες τις πληροφορίες ενός μηνύματος.

### **Χαρακτηριστικά κλάσης**

- text: string : Αλφαριθμητικό που αντιπροσωπεύει το κείμενο του μηνύματος.
- time: timestamp : Χρονική σήμανση του χρόνου αποστολής του μηνύματος.
- senderID: integer : Ακέραιος που αντιπροσωπεύει το ID του αποστολέα.
- messageID: string : Αλφαριθμητικό που αντιπροσωπεύει το ID του μηνύματος.
- senderImage: image : Εικόνα προφίλ του αποστολέα.

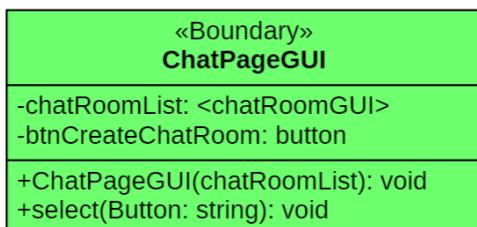
### **Μέθοδοι κλάσης**

- Message(text, time, userID, messageID): void : Πρόκειται για τον constructor της κλάσης.



- setText(in text:string): void : Λειτουργία που θέτει το κείμενο το μηνύματος.
- getText(): string : Λειτουργία που επιστρέφει το κείμενο του μηνύματος.
- setTime(in time:Timestamp): void : Λειτουργία που θέτει το χρόνο αποστολής του μηνύματος.
- getTime(): Timestamp : Λειτουργία που επιστρέφει το χρόνο αποστολής του μηνύματος.
- setSenderId(in userID:integer): void : Λειτουργία που θέτει το ID του αποστολέα.
- getSenderId(): integer : Λειτουργία που επιστρέφει το ID του αποστολέα.
- setMessageID(in messageID:string): void : Λειτουργία που θέτει το ID του μηνύματος.
- getMessageID(): string : Λειτουργία που επιστρέφει το ID του μηνύματος.
- setSenderImage(in senderImage:image): void : Λειτουργία που θέτει την εικόνα του αποστολέα.
- getSenderImage(): image : Λειτουργία που επιστρέφει την εικόνα του αποστολέα.

### ChatPageGUI



Η κλάση αυτή αφορά τη διεπαφή της σελίδας του Chat, όπου ο χρήστης μπορεί να δει τη λίστα με τα Chat Rooms.

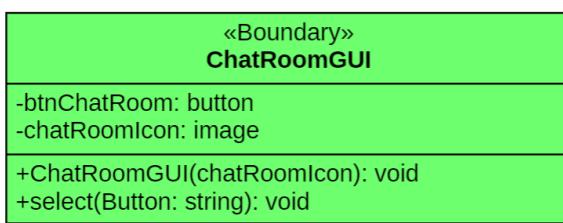
#### Χαρακτηριστικά κλάσης

- chatRoomList: <chatRoomGUI> : Λίστα με τα διαθέσιμα Chat Rooms.
- btnCreateChatRoom: button : Κουμπί που δημιουργεί το νέο Chat Room.

#### Μέθοδοι κλάσης

- ChatPageGUI(chatRoomList): void : Πρόκειται για τον constructor της κλάσης.
- select(in Button:string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ChatRoomGUI



Η κλάση αυτή αφορά τη διεπαφή ένος Chat Room στη σελίδα του Chat.

#### Χαρακτηριστικά κλάσης

- btnChatRoom: button : Κουμπί που μεταφέρει το χρήστη στο συγκεκιμένο Chat Room.
- chatRoomIcon: image : Εικόνα του Chat Room.

#### Μέθοδος κλάσης

- ChatRoomGUI(chatRoomIcon): void : Πρόκειται για τον constructor της κλάσης.
- select(in Button:string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### CreateChartRoomGUI



«Boundary» CreateChatRoomGUI	
-lblName: label	
-txtName: textbox	

-userNameList: List<label>

-isCheckedList: List<boolean>

-btnCreateNewChatRoom: button

-btnCancel: button

+CreateChatRoomGUI(lblName, txtName, userNameList, isCheckedList): void

+select(Button: string): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας δημιουργίας ενός νέου Chat Room.

### Χαρακτηριστικά κλάσης

- lblName: label : Ετικέτα με τίτλο Name.
- txtName: textbox : Πεδίο εισαγωγής του ονόματος του νέου Chat Room.
- userNameList: list<label> : Λίστα με τα ονόματα χρηστών για προσθήκη.
- isCheckedList: list<boolean> : Λίστα με checkbox για την ένδειξη επιλογής ενός ονόματος χρήστη.
- btnCreateNewChatRoom: button : Κουμπί που δημιουργεί ένα νέο Chat Room.
- btnCancel: button : Κουμπί που ακυρώνει τη δημιουργία ενός νέου Chat Room.

### Μέθοδοι κλάσης

- CreateChatRoomGUI( lblName, txtName, userNameList, isCheckedList): void : Πρόκειται για τον constructor της κλάσης.
- select(in Button: string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### ChatRoomPageGUI

«Boundary» ChatRoomPageGUI	
-txtBox: textbox	
-MessageList: List<MessageGUI>	
-lblChatRoomName: label	
-btnSend: button	
-btnBackToChatPageGUI: button	
-btnGoToChartPageGUI: button	

+ChatRoomPageGUI(txtBox, lblMessageList, lblChatRoomName): void

+select(Button: string): void

Η κλάση αυτή αφορά τη διεπαφή της σελίδας ενός Chat Room.

### Χαρακτηριστικά κλάσης

- txtBox: textbox : Πεδίο εισαγωγής κειμένου για αποστολή.
- MessageList: list<MessageGUI> : Λίστα με τα μηνύματα του Chat Room.
- lblChatRoomName: label : Ετικέτα με το όνομα του Chat Room.
- btnSend: button : Κουμπί που αποστέλλει το μήνυμα.
- btnBackToChatPageGUI: button : Κουμπί που επιστρέφει τον χρήστη στη σελίδα του Chat.
- btnGoToChartPageGUI: button : Κουμπί που μεταφέρει το χρήστη στο Chart Page για την επιλογή Chart προς αποστολή.

### Μέθοδοι κλάσης

- ChatRoomPageGUI(txtBox, lblMessageList, lblChatRoomName): void : Πρόκειται για τον constructor της κλάσης.
- select(in Button: string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### MessageGUI



«Boundary» MessageGUI
-lbltext: label
-lblTime: label
-iconSender: image
-lblNameSender: label
+MessageGUI(lbltext, lblTime, iconSender, lblNameSender): void

Η κλάση αυτή αφορά τη διεπαφή ενός μηνύματος στη σελίδα του Chat Room.

### Χαρακτηριστικά κλάσης

- lbltext: label : Ετικέτα με το κείμενο του μηνύματος.
- lblTime: label : Ετικέτα με το χρόνο αποστολής του μηνύματος
- iconSender: image : Εικόνα προφίλ του αποστολέα.
- lblNameSender: label : Ετικέτα με το όνομα του αποστολέα.

### Μέθοδοι κλάσης

- MessageGUI(lbltext, lblTime, iconSender, lblNameSender): void : Πρόκειται για τον constructor της κλάσης.

## ChatController

«Controller» ChatController
+sendMessage(): void
+updateMessageList(): void
+createNewChatRoom(): void
+cancelCreationOfChatRoom(): void
+fetchChatRoom(chatRoomID: integer): ChatRoom

Η κλάση αυτή αφορά τη λειτουργία του Chat.

### Μέθοδοι κλάσης

- sendMessage(): void : Λειτουργία που αποστέλλει ένα μήνυμα.
- updateMessageList(): void : Λειτουργία που ενημερώνει τη λιστά των μηνυμάτων του Chat Room.
- createNewChatRoom(): void : Λειτουργία που δημιουργεί ένα νέο Chat Room.
- cancelCreationOfChatRoom(): void : Λειτουργία που ακυρώνει τη δημιουργία του νέου Chat Room.
- fetchChatRoom(chatRoomID:integer): ChatRoom : Λειτουργία που προσκομίζει ένα Chat Room.

## TeamTactixChatDBProxy

«Boundary» TeamTactixChatDBProxy

Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί.





## 4.3 <Πακέτα λεξιλογίου σεναρίων χαμηλής προτεραιότητας >

### 4.3.1 Πακέτο Personal Profile

#### Account

«Entity»	
Account	
-name: string	
-userID: integer	
-password: string	
-userRole: string	
-userImage: image	
-descriprion: string	
-googleMailAccountName: string	
-linkedInAccountName: string	
-gitHubAccountName: string	
+Account(Name, userID, password, userType): void	
+setName(Name: string): void	
+getName(): string	
+setUserID(userID: integer): void	
+getUserID(): integer	
+setPassword(password: string): void	
+getPassword(): string	
+setUserRole(userRole: string): void	
+getUserRole(): string	
+setUserImage(userImage: image): void	
+getUserImage(): image	
+setDescription(description: string): void	
+getDescription(): string	
+setGoogleMailAccountName(googleMailAccountName: string): void	
+getGoogleMailAccountName(): string	
+setLinkedInAccountName(linkedInAccountName: string): void	
+getLinkedInAccountName(): string	
+setGitHubAccountName(gitHubAccountName: string): void	
+getGitHubAccountName(): string	

Η κλάση αυτή περιέχει όλα τα στοιχεία του λογαριασμού του χρήστη.

#### Χαρακτηριστικά Κλάσης

- name: string : Αλφαριθμητικό που αντιπροσωπεύει το όνομα του χρήστη.
- userID: integer : Ακέραιος που αντιπροσωπεύει το ID του χρήστη.
- password: string : Αλφαριθμητικό που αντιπροσωπεύει τον κωδικό του χρήστη.
- userRole: string : Αλφαριθμητικό που αντιπροσωπεύει το ρόλο του χρήστη.
- userImage: image : Εικόνα προφίλ του χρήστη.
- description: string : Αλφαριθμητικό που αντιπροσωπεύει την περιγραφή του χρήστη.
- googleMailAccountName: string : Αλφαριθμητικό που αντιπροσωπεύει το google mail account του χρήστη.
- linkedInAccountName: string : Αλφαριθμητικό που αντιπροσωπεύει το LinkedIn account του χρήστη.
- gitHubAccountName: string : Αλφαριθμητικό που αντιπροσωπεύει το GitHub account του χρήστη.

#### Μέθοδοι κλάσης

- Account(Name, userID, password, userType): void : Πρόκειται για τον constructor της κλάσης.
- setName(Name:string): void : Λειτουργία που θέτει το όνομα χρήστη.
- getName(): string : Λειτουργία που επιστρέφει το όνομα του χρήστη.
- setUserID(userID:integer): void : Λειτουργία που θέτει το ID του χρήστη.
- getUserID(): integer : Λειτουργία που επιστρέφει το ID του χρήστη.
- setPassword(password:string): void : Λειτουργία που θέτει τον κωδικό του χρήστη.
- getPassword(): string : Λειτουργία που επιστρέφει τον κωδικό του χρήστη.
- setUserRole(userRole:string): void : Λειτουργία που θέτει τον ρόλο του χρήστη.
- getUserRole(): string : Λειτουργία που επιστρέφει τον ρόλο του χρήστη.
- setUserImage(userImage:image): void : Λειτουργία που θέτει την εικόνα προφίλ του χρήστη.



- `getUserImage(): image` : Λειτουργία που επιστρέφει την εικόνα προφίλ του χρήστη.
- `setDescription(Description:string): void` : Λειτουργία που θέτει την περιγραφή του χρήστη.
- `getDescription(): string` : Λειτουργία που επιστρέφει την περιγραφή του χρήστη.
- `setGoogleMailAccountName(googleMailAccountName:string): void` : Λειτουργία που θέτει το google mail account του χρήστη.
- `getGoogleMailAccountName(): string` : Λειτουργία που επιστρέφει το google mail account του χρήστη.
- `setLinkedInAccountName(linkedInAccountName:string): void` : Λειτουργία που θέτει το LinkedIn account του χρήστη.
- `getLinkedInAccountName(): string` : Λειτουργία που επιστρέφει το LinkedIn account του χρήστη.
- `setGitHubAccountName(gitHubAccountName:string): void` : Λειτουργία που θέτει το GitHub account του χρήστη.
- `getGitHubAccountName(): string` : Λειτουργία που επιστρέφει το GitHub account του χρήστη.

### ProfilePageGUI

«Boundary» ProfilePageGUI	
-userImage: image	
-lblName: label	
-lblUserRole: label	
-lblDescription: label	
-btnGoogleMail: button	
-btnGitHub: button	
-btnLinkedIn: button	
-btnEditProfile: button	
+ProfilePageGUI(userImage, lblFirstName, lblLastName, lblUserType): void	
+select(Button: string): void	

Η κλάση αυτή αφορά τη διεπαφή του προφίλ όλων των χρηστών. Δίνει τη δυνατότητα στον χρήστη να δει τα προσωπικά του στοιχεία και να επιλέξει να τα επεξεργαστεί.

### **Χαρακτηριστικά κλάσης**

- `userImage: image` : Εικόνα προφίλ του χρήστη.
- `lblName: label` : Ετικέτα με το όνομα του χρήστη.
- `lblUserRole: label` : Ετικέτα με το ρόλο του χρήστη.
- `lblDescription: label` : Ετικέτα με την περιγραφή του χρήστη.
- `btnGoogleMail: button` : Κουμπί που οδηγεί στο Google Mail του χρήστη.
- `btnGitHub: button` : Κουμπί που οδηγεί στο GitHub του χρήστη.
- `btnLinkedIn: button` : Κουμπί που οδηγεί στο LinkedIn του χρήστη.
- `btnEditProfile: button` : Κουμπί που οδηγεί στη σελίδα επεξεργασίας προφίλ του χρήστη.

### **Μέθοδοι κλάσεις**

- `ProfilePageGUI(userImage, lblFirstName, lblLastName, lblUserType): void` : Πρόκειται για τον constructor της κλάσης.
- `select(Button: string): void` : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

### EditProfilePageGUI



«Boundary» EditProfilePageGUI	
-lblName: label	
-txtName: textbox	
-lblUserRole: label	
-txtUserRole: textbox	
-lblDescription: label	
-txtDescription: textbox	
-lblUserImage: label	
-lblGoogleMailAccountName: label	
-txtGoogleMailAccountName: textbox	
-lblGitHubAccountName: label	
-txtGitHubAccountName: textbox	
-lblLinkedInAccountName: label	
-txtLinkedInAccountName: textbox	
-btnUploadImage: button	
-btnSaveProfileDetails: button	
-btnCancelProfileChanges: button	
+EditProfilePageGUI(lblName, txtName, lblUserType, txtUserType, lblDescription, txtDescription, lblUserImage, lblGoogleMail, txtGoogleMail, lblGitHubAccount, txtGitHubAccount, lblLinkedInAccount, txtLinkedInAccount): void	
+select(Button: string): void	

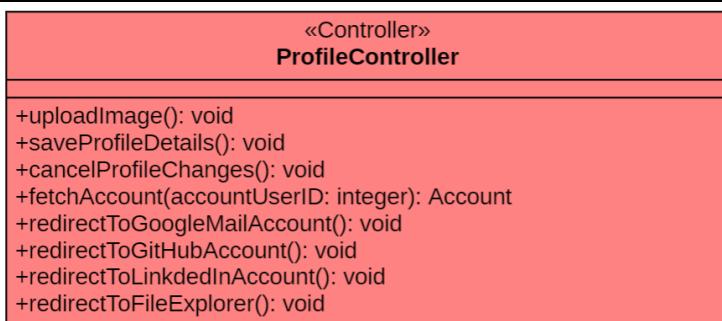
Η κλάση αυτή σχετίζεται με την διεπαφή επεξεργασίας προφύλ του χρήστη. Δίνει τη δυνατότητα στον χρήστη να επεξεργαστεί τα προσωπικά του στοιχεία.

### Χαρακτηριστικά κλάσης

- lblName: label : Ετικέτα με τίτλο Username.
- txtName: textbox : Πεδίο εισαγωγής ονόματος του χρήστη.
- lblUserRole: label : Ετικέτα με τίτλο Role.
- txtUserRole: textbox: Πεδίο εισαγωγής ρόλου του χρήστη.
- lblDescription: label : Ετικέτα με τίτλο Description.
- txtDescription: textbox : Πεδίο εισαγωγής περιγραφής του χρήστη.
- lblUserImage: label : Ετικέτα με την εικόνα προφύλ του χρήστη.
- lblGoogleMailAccountName: label : Ετικέτα google account.
- txtGoogleMailAccountName: textbox : Πεδίο εισαγωγής link του google account του χρήστη.
- lblGitHubAccountName: label : Ετικέτα GitHub.
- txtGitHubAccountName: textbox : Πεδίο εισαγωγής link του GitHub του χρήστη.
- lblLinkedInAccountName: label : Ετικέτα LinkedIn.
- txtLinkedInAccountName: textbox : Πεδίο εισαγωγής link του LinkedIn του χρήστη.
- btnUploadImage: button : Κουμπί που επιτρέπει στο χρήστη να ανεβάσει εικόνα προφύλ.
- btnSaveProfileDetails: button : Κουμπί που αποθηκεύει τις αλλαγές που έκανε ο χρήστης.
- btnCancelProfileChanges: button : Κουμπί που ακυρώνει τις αλλαγές που έκανε ο χρήστης.

### Μέθοδοι Κλάσης

- EditProfilePageGUI(lblName, txtName, lblUserType, txtUserType, lblDescription, txtDescription, lblUserImage, lblGoogleMail, txtGoogleMail, lblGitHubAccount, txtGitHubAccount, lblLinkedInAccount, txtLinkedInAccount): void : Πρόκειται για τον constructor της κλάσης.
- select(Button: string): void : Συνάρτηση για επιλογή κουμπιού από την προκείμενη διεπαφή.

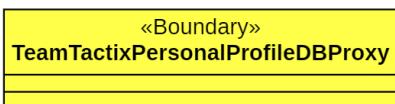


Η κλάση αυτή είναι υπεύθυνη για την λειτουργία της προβολής και επεξεργασίας του προσωπικού προφίλ.

### Μέθοδοι κλάσης

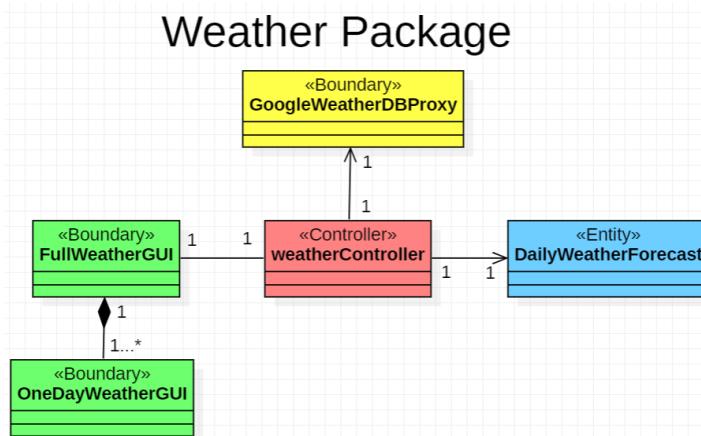
- uploadImage(): void : Λειτουργία που ανεβάζει την εικόνα προφίλ του χρήστη.
- saveProfileDetails(): void : Λειτουργία που αποθηκεύει τις αλλαγές που έκανε ο χρήστης.
- cancelProfileChanges(): void : Λειτουργία που ακυρώνει τις αλλαγές που έκανε ο χρήστης.
- fetchAccount(accountUserID:integer): Account : Λειτουργία που προσκομίζει το λογαριασμό του χρήστη.
- redirectToGoogleMailAccount(): void : Λειτουργία που κατευθύνει το χρήστη στο Google mail Account του.
- redirectToGitHubAccount(): void : Λειτουργία που κατευθύνει το χρήστη στο GitHub Account του.
- redirectToLinkedInAccount(): void : Λειτουργία που κατευθύνει το χρήστη στο LinkedIn Account του.
- redirectToFileExplorer(): void : Λειτουργία που ανοίγει το File Explorer για το upload της εικόνας προφίλ.

### TeamTactixPersonalProfileDBProxy



Η κλάση αυτή χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων της εφαρμογής ώστε ο χρήστης να μπορεί να δει τα δεδομένα και να αποθηκεύει τις νέες πληροφορίες που καταχωρεί.

### 4.3.2 Πακέτο Weather Package



### Daily Weather Forecast



«Entity» <b>DailyWeatherForecast</b>
<pre>-temperature: integer -humidity: float -windSpeed: float -windDirection: integer -chanceOfRain: integer  +WeatherWidget(temperature, humidity, windSpeed, windDirection, description): void +setTemperature(temperature: integer): void +getTemperature(): integer +setHumidity(humidity: float): void +getHumidity(): float +setWindSpeed(windSpeed: float): void +getWindSpeed(): float +setWindDirection(windDirection: integer): void +getWindDirection(): integer +setChanceOfRain(chanceOfRain: integer): void +getChanceOfRain(): integer</pre>

Η κλάση αυτή περιέχει όλα τα στοιχεία που αφορούν το Weather Forecast..

### Χαρακτηριστικά κλάσης

- temperature: integer: Ακέραιος αριθμός που δηλώνει την θερμοκρασία.
- humidity: float: Αριθμός κινητής υποδιαστολής που δηλώνει την υγρασία.
- windSpeed: float: Αριθμός κινητής υποδιαστολής που δηλώνει την ταχύτητα του αέρα.
- windDirection: integer: Ακέραιος αριθμός που δηλώνει την κατεύθυνση του αέρα.
- chanceOfRain: integer: Ακέραιος αριθμός που δηλώνει την πιθανότητα βροχής.

### Μέθοδοι κλάσης

- WeatherWidget(temperature,humidity,windSpeed,windDirection,description): void: Πρόκειται για τον constructor της κλάσης.
- setTemperature(temperature:integer): void: Η λειτουργία αυτή θέτει την τιμή της θερμοκρασίας.
- getTemperature(): integer: Η λειτουργία αυτή επιστρέφει την τιμή της θερμοκρασίας.
- setHumidity(humidity:float): void: Η λειτουργία αυτή θέτει την τιμή της υγρασίας.
- getHumidity(): float: Η λειτουργία αυτή επιστρέφει την τιμή της υγρασίας.
- setWindSpeed(windSpeed:float): void: Η λειτουργία αυτή θέτει την τιμή της ταχύτητας του αέρα.
- getWindSpeed(): float: Η λειτουργία αυτή επιστρέφει την τιμή της ταχύτητας του αέρα.
- setWindDirection(windDirection:integer): void: Η λειτουργία αυτή θέτει την τιμή της κατεύθυνσης του αέρα.
- getWindDirection(): integer: Η λειτουργία αυτή επιστρέφει την τιμή της κατεύθυνσης του αέρα.
- setChanceOfRain(chanceOfRain:integer): void: Η λειτουργία αυτή θέτει την τιμή της πιθανότητας βροχής.
- getChanceOfRain(): integer: Η λειτουργία αυτή επιστρέφει την τιμή της πιθανότητας βροχής.

### OneDayWeatherGUI

«Boundary» <b>OneDayWeatherGUI</b>
<pre>-lblTemperature: label -lblHumidity: label -lblWindSpeed: label -lblWindDirection: label -lblChanceOfRain: label  +OneDayWeatherGUI(lblTemperature, lblHumidity, lblWindSpeed, lblWindDirection, lblChanceOfRain): void</pre>

Η κλάση αυτή αφορά την διεπαφή του χρήστη με τις πληροφορίες του καιρού για μία μέρα.



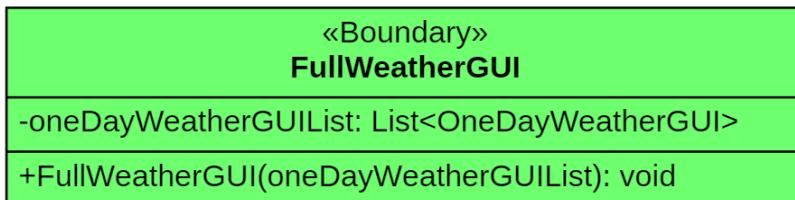
### Χαρακτηριστικά Κλάσης

- IblTemperature: label: Ετικέτα με την θερμοκρασία για την τοπική περιοχή.
- IblHumidity: label: Ετικέτα με την υγρασία για την τοπική περιοχή.
- IblWindSpeed: label: Ετικέτα με την ταχύτητα του αέρα για την τοπική περιοχή.
- IblWindDirection: label: Ετικέτα με την κατεύθυνση του αέρα για την τοπική περιοχή.
- IblChanceOfRain: label: Ετικέτα με την πιθανότητα βροχής για την τοπική περιοχή.

### Μέθοδοι κλάσης

- OneDayWeatherGUI(IblTemperature,IblHumidity,IblWindSpeed,IblWindDirection,IblChanceOfRain):void  
oid: Πρόκειται για τον constructor της κλάσης.

### FullWeatherGUI



Η κλάση αυτή αφορά την διεπαφή του χρήστη με τις πληροφορίες του καιρού για μερικές μέρες.

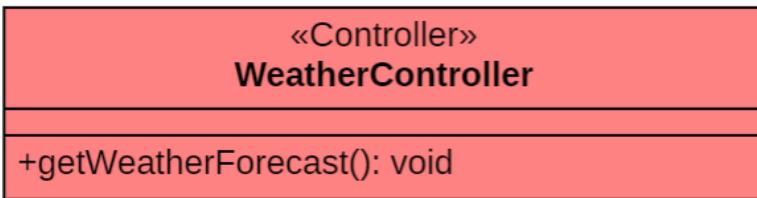
### Χαρακτηριστικά Κλάσης

- oneDayWeatherGUIList: List<OneDayWeatherGUI>: Λίστα με αντικείμενα τις διεπαφές oneDayWeather.

### Μέθοδοι κλάσης

- FullWeatherGUI(oneDayWeatherGUIList): void: Πρόκειται για τον constructor της κλάσης.

### WeatherController

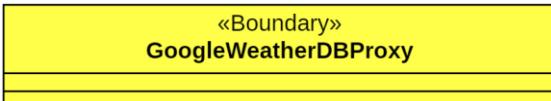


Η κλάση αυτή είναι υπεύθυνη για την ενημέρωση των καιρικών συνθηκών.

### Μέθοδοι κλάσης

- getWeatherForecast(): void: Η μέθοδος αυτή ενημερώνει τις καιρικές συνθήκες που επικρατούν στην περιοχή.

### GoogleWeatherDBProxy



Η κλάση αυτή χρησιμοποιείται για την επικοινωνία της βάσης δεδομένων της google με την εφαρμογή, με στόχο να ανανεώνονται οι τιμές που περιγράφουν τον καιρό.



## Παράρτημα I – Γλωσσάριο

ΛΑ-xx Λειτουργική Απαίτηση xx

ΜΛΑ-xx Μη Λειτουργική Απαίτηση xx

ΟΑ Ομάδα Εργασίας

DB Βάση Δεδομένων