



# D.O.P.E.

Designing Online Personal Environment

Use D.O.P.E. to become a finer designer

## Απαιτήσεις Λογισμικού (Προδιαγραφές Λογισμικού)

### Del.1.2

Version 1.0

Χρήστος Παπακωνσταντίνου [papachri@ece.auth.gr](mailto:papachri@ece.auth.gr)

Αντώνης Μαυρομανώλης [antomavr@ece.auth.gr](mailto:antomavr@ece.auth.gr)

Γιώργος Μιχαηλίδης [geormich@auth.gr](mailto:geormich@auth.gr)

Θωμάς Πλιάκης [tpliakis@ece.auth.gr](mailto:tpliakis@ece.auth.gr)

7/05/2019

## Ιστορικό Αλλαγών

Όνομα	Ημερομηνία	Αλλαγή	Έκδοση
A. Μαυρομανώλης	20/04/2019	Δημιουργία εγγράφου. Προσαρμογή των προτύπων του K. E. Wiegers <sup>*</sup> και του M. Smialek's.	0.1
A. Μαυρομανώλης	1/05/2019	Προσαρμογή του εγγράφου.	0.1.1
A. Μαυρομανώλης	7/05/2019	Τελική έκδοση εγγράφου.	1.0

## Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Αντώνης Μαυρομανώλης	40	<a href="mailto:antomavr@ece.auth.gr">antomavr@ece.auth.gr</a>
Θωμάς Πλιάκης	40	<a href="mailto:tpliakis@ece.auth.gr">tpliakis@ece.auth.gr</a>
Χρήστος Παπακωνσταντίνου	40	<a href="mailto:papachri@ece.auth.gr">papachri@ece.auth.gr</a>
Γιώργος Μιχαηλίδης	40	<a href="mailto:geormich@auth.gr">geormich@auth.gr</a>

\* Copyright © 2002 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document. Original template is available at: <http://www.processimpact.com/>

## Πίνακας Περιεχομένων

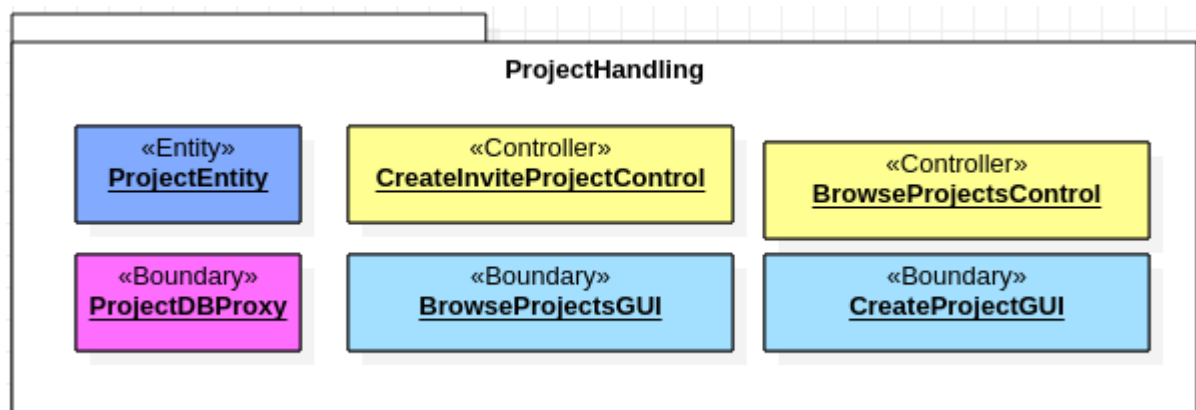
<b>1 Στατική Μοντελοποίηση</b>	<b>4</b>
<b>1.1 &lt;Πακέτα λεξιλογίου σεναρίων υψηλής προτεραιότητας&gt;</b>	<b>4</b>
1.1.1 Πακέτο ProjectHandling	4
1.1.2 Πακέτο FileHandling	10
<b>1.2 &lt;Πακέτο λεξιλογίου σεναρίων μεσαίας προτεραιότητας &gt;</b>	<b>17</b>
1.2.1 Πακέτο SalesHandling	17
<b>1.3 &lt;Πακέτα λεξιλογίου σεναρίων χαμηλής προτεραιότητας &gt;</b>	<b>25</b>
1.3.1 Πακέτο DeviceHandling	25
1.3.2 Πακέτο ChatHandling	33
<b>2 Μηλειτουργικές απαιτήσεις</b>	<b>38</b>
<b>2.1 Απαιτήσεις επίδοσης</b>	<b>38</b>
<ΜΛΑ-2>	38
<ΜΛΑ-6>	38
<b>2.2 Απαιτήσεις Χρηστικότητας (Usability)</b>	<b>38</b>
<ΜΛΑ-3>	38
2.3 Τεχνικές Απαιτήσεις περιβάλλοντος	39
<ΜΛΑ-1>	39
<ΜΛΑ-4>	39
<b>2.4 Απαιτήσεις πολιτικής (Policy - legal, corporate)</b>	<b>39</b>
<ΜΛΑ-5>	39
<b>3 Πρότυπα Σχεδιασμού που υιοθετήθηκαν</b>	<b>40</b>
3.2 Πρότυπα Συμπεριφοράς	40
3.2 Δομικά Πρότυπα	40

# 1 Στατική Μοντελοποίηση

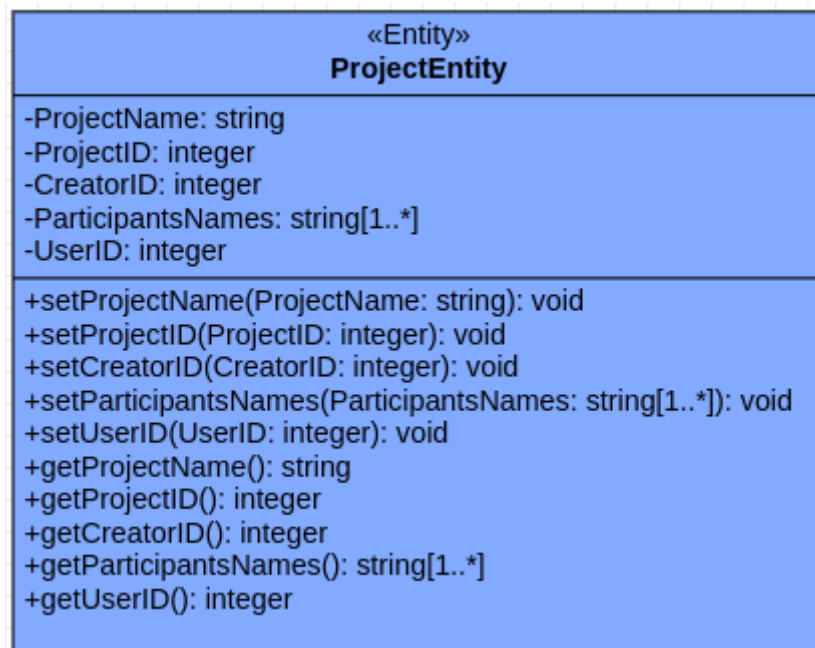
## 1.1 <Πακέτα λεξιλογίου σεναρίων υψηλής προτεραιότητας>

### 1.1.1 Πακέτο ProjectHandling

Το πακέτο αυτό είναι υπεύθυνο να επιτελέσει κάθε λειτουργία που αφορά την δημιουργία κάποιου καινούργιου Project, την πρόσθεση άλλων χρηστών στο Project και είναι υπεύθυνο για την περιήγηση σε όλα τα Projects.



### Entity ProjectEntity



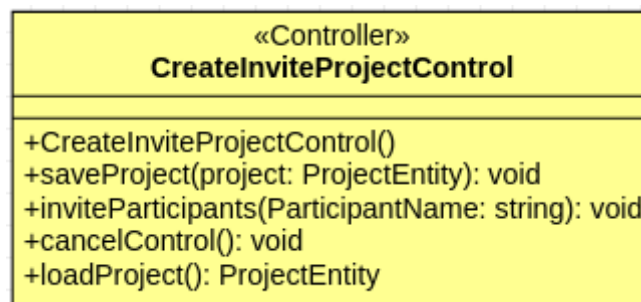
Αυτή είναι η κλάση οντότητας του project και περιέχει τα στοιχεία για τον ορισμό του.

**Χαρακτηριστικά της κλάσης:**

- **ProjectName:** String για το όνομα του project.
- **ProjectID:** Ο μοναδικός χαρακτηριστικός αριθμός του project.
- **CreatorID:** Ο μοναδικός χαρακτηριστικός αριθμός του δημιουργού του project.
- **ParticipantsNames:** Η λίστα με τα ονόματα των χρηστών που συμμετέχουν στο project.
- **UserID:** Ο μοναδικός χαρακτηριστικός αριθμός του συνδεδεμένου χρήστη.

**Μέθοδοι της κλάσης:**

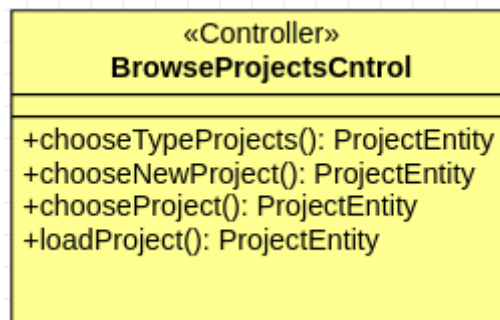
- **setProjectName(ProjectName: String):** Μέθοδος που θέτει το όνομα του Project.
- **setProjectID(ProjectID: integer):** Μέθοδος που θέτει το μοναδικό αναγνωριστικό του Project.
- **setCreatorID(CreatorID: integer):** Μέθοδος που θέτει το μοναδικό αναγνωριστικό του δημιουργού του Project.
- **SetParticipantsNames():** Μέθοδος που θέτει τους χρήστες που είναι συνδεδεμένοι στο Project.
- **setUserID(UserID: integer):** Μέθοδος που θέτει τον συνδεδεμένο χρήστη.
- **getProjectName():** Η μέθοδος επιστρέφει το όνομα του project.
- **getProjectID():** Η μέθοδος επιστρέφει το μοναδικό αναγνωριστικό του Project.
- **getCreatorID():** Η μέθοδος επιστρέφει το μοναδικό αναγνωριστικό του δημιουργού του Project.
- **getParticipantsNames():** Η μέθοδος επιστρέφει τους χρήστες που είναι συνδεδεμένοι στο Project.
- **getUserID():** Η μέθοδος επιστρέφει τον μοναδικός χαρακτηριστικός αριθμός του συνδεδεμένου χρήστη.

**Controller CreateInviteProjectControl**

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για τη δημιουργία ενός Project και την πρόσκληση να συμμετάσχουν άλλοι χρήστες σε αυτό.

**Μέθοδοι της κλάσης:**

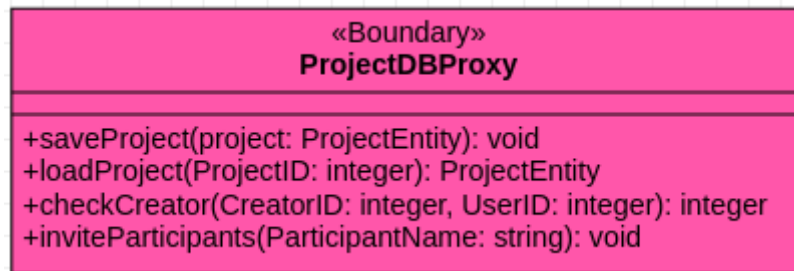
- **CreateInviteProjectControl():** Πρόκειται για μέθοδο δόμησης του ελεγκτή.
- **saveProject():** Η μέθοδος αυτή καλείται με σκοπό την αποθήκευση του αντίστοιχου project.
- **nviteParticipants():** Η μέθοδος αυτή καλείται με σκοπό την πρόσκληση άλλων χρηστών να συμμετάσχουν στο project.
- **loadProject():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση κάποιου project.

**Controller BrowseProjectsControl**

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για την περιήγηση στα διάφορα project.

**Μέθοδοι της κλάσης:**

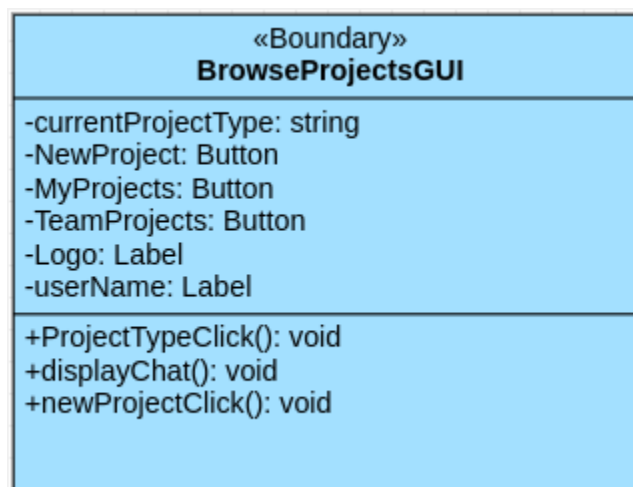
- **chooseTypeProjects():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση των αντίστοιχων(ομαδικών, ατομικών) project.
- **chooseNewProject():** Η μέθοδος αυτή καλείται με σκοπό την δημιουργία κάποιου καινούργιου project.
- **chooseProject():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση συγκεκριμένου project.
- **loadProject():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση των περιεχομένων ενός project.

**Boundary ProjectDBProxy**

Η κλάση αυτή επικοινωνεί με απομακρυσμένη βάση δεδομένων όπου είναι αποθηκευμένο οι φάκελοι των project.

**Μέθοδοι της κλάσης:**

- **saveProject():** Η μέθοδος αυτή αποθηκεύει ένα project, δηλαδή κλάση ProjectEntity, στη βάση δεδομένων.
- **loadProject():** Η μέθοδος αυτή επιστρέφει ένα project από τη βάση δεδομένων, μέσω του μοναδικού αναγνωριστικού της.
- **checkCreator():** Η μέθοδος αυτή ελέγχει αν ο χρήστης είναι ο δημιουργός του project.
- **inviteParticipants():** Η μέθοδος αυτή καλείται με σκοπό την αναζήτηση και πρόσκληση άλλων χρηστών στην βάση δεδομένων για συμμετοχή στο project.

**Boundary BrowseProjectsGUI**

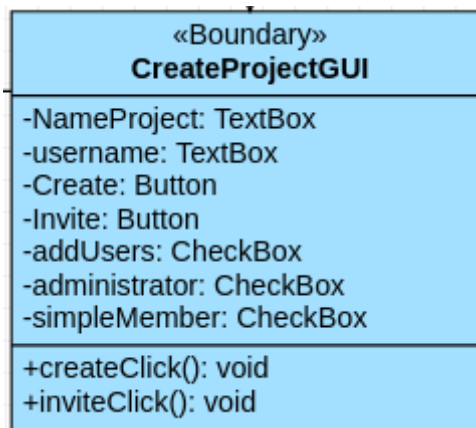
Η κλάση αυτή αναπαριστά τη διεπαφή για την εμφάνιση της σελίδας περιήγησης στα διάφορα Projects.

**Χαρακτηριστικά της κλάσης:**

- **currentProjectType:** String για το όνομα των project που εμφανίζονται.
- **NewProject:** Κουμπί για την δημιουργία νέου project.
- **MyProjects:** Κουμπί για την εμφάνιση των ατομικών project.
- **TeamProjects:** Κουμπί για την εμφάνιση των ομαδικών project.
- **Logo:** Επιγραφή εμφάνισης του logo της εφαρμογής.
- **UserName:** Επιγραφή εμφάνισης του ονόματος του χρήστη.

**Μέθοδοι της κλάσης:**

- **ProjectTypeClick():** Καλεί την συνάρτηση chooseTypeProjects() του ελεγκτή BrowseProjectsControl για την εμφάνιση κάποιου τύπου Project.
- **displayChat():** Καλεί την αντίστοιχο ελεγκτή για την εμφάνιση του chat.
- **newProjectClick():** Καλεί την συνάρτηση chooseNewProject() του ελεγκτή BrowseProjectsControl για την δημιουργία ενός καινούργιου project.

**Boundary BrowseProjectsGUI**

Η κλάση αυτή αναπαριστά τη διεπαφή για την εμφάνιση της σελίδας δημιουργία; ενός καινούργιου Project.



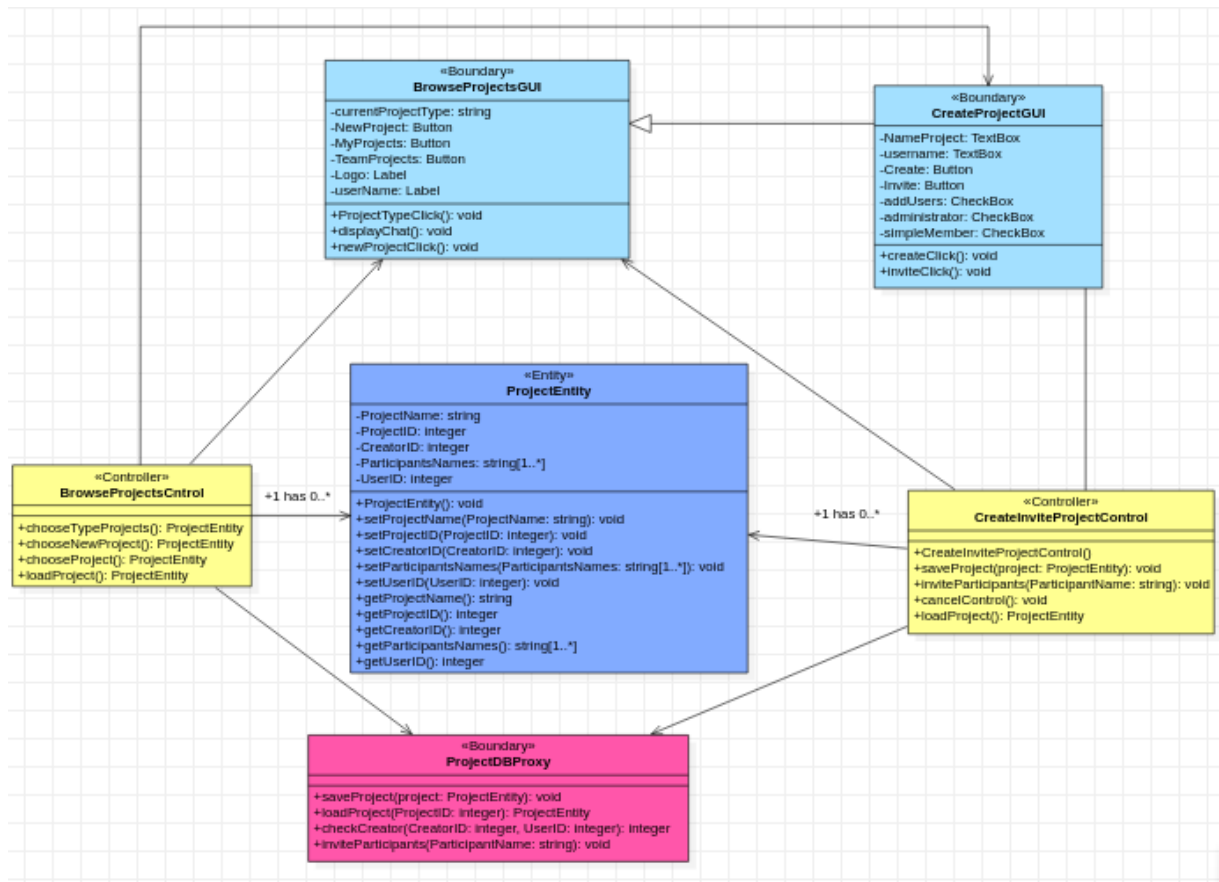
**Χαρακτηριστικά της κλάσης:**

- **NameProject:** Πεδίο εισαγωγής του ονόματος του καινούργιου project.
- **username:** Πεδίο εισαγωγής του ονόματος κάποιου χρήστη για να προσκληθεί να συμμετάσχει στο project.
- **Create:** Κουμπί για την δημιουργία καινούργιου project.
- **Invite:** Κουμπί για την πρόσκληση άλλων χρηστών στο project.
- **addUsers:** Πλαίσιο επιλογής για πρόσκληση άλλου χρήστη στο project.
- **administrator:** Πλαίσιο επιλογής για τα δικαιώματα του άλλου χρήστη στο project.
- **simpleMember:** Πλαίσιο επιλογής για τα δικαιώματα του άλλου χρήστη στο project.

**Μέθοδοι της κλάσης:**

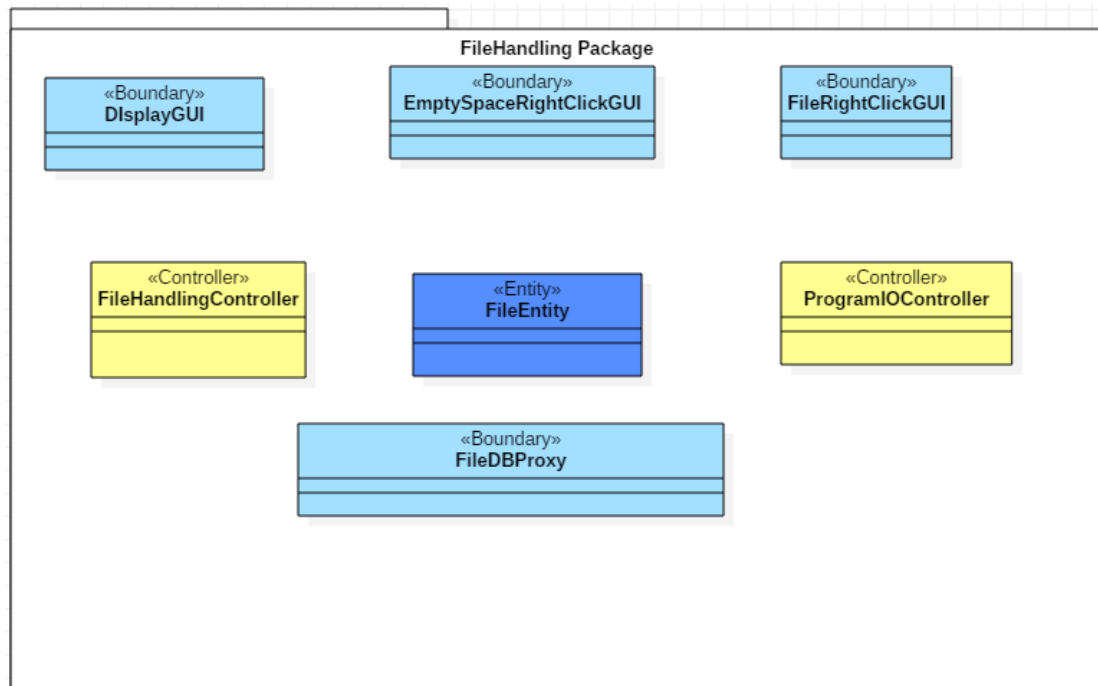
- **createClick():** Καλεί την συνάρτηση `saveProject()` του ελεγκτή `CreateInviteProjectControl` για την αποθήκευση του νέου Project.
- **inviteClick():** Καλεί την συνάρτηση `inviteParticipants()` του ελεγκτή `CreateInviteProjectControl` για την πρόσκληση άλλου χρήστη στο projec

## Διάγραμμα κλάσεων

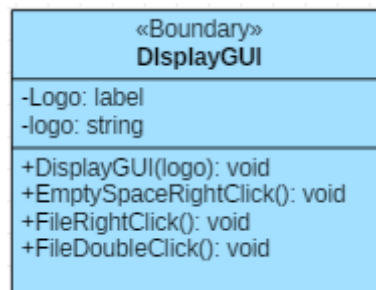


## 1.1.2 Πακέτο FileHandling

Το πακέτο αυτό είναι υπεύθυνο να επιτελέσει κάθε λειτουργία που αφορά κάποια ενέργεια του χρήστη πάνω σε ένα αρχείο ( Δημιουργία , Άνοιγμα, διαγραφή κ.α.).επίσης είναι υπεύθυνο για την εκτέλεση των προγραμμάτων που υποστηρίζει το σύστημα.



### Boundary DisplayGUI



Η κλάση αυτή εκφράζει τη γραφική διεπαφή του χρήστη με τα αρχεία και το περιβάλλον του συστήματος.

#### Χαρακτηριστικά της κλάσης:

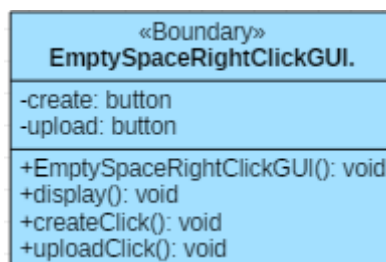
- **Logo** : Λογότυπο της εφαρμογής.
- **logo** : Το όνομα του λογότυπου.

#### Μέθοδοι της κλάσης:

- **DisplayGUI(logo)** : Αποτελεί τον Constructor της κλάσης.

- **EmptySpaceRightClick()**: Καλεί τη συνάρτηση display της κλάσης EmptySpaceRightClickGUI.
- **FileRightClick()** : Καλεί τη συνάρτηση display της κλάσης FileRightClickGUI.
- **FileDoubleClick()** : Καλεί τη συνάρτηση open() της κλάσης FileHandlingController.

### **Boundary EmptySpaceRightClickGUI**



Η κλάση αυτή είναι σχετίζεται με το μενού που εμφανίζεται όταν ο χρήστης κάνει δεξί κλικ στον κενό χώρο μεταξύ των αρχείων.

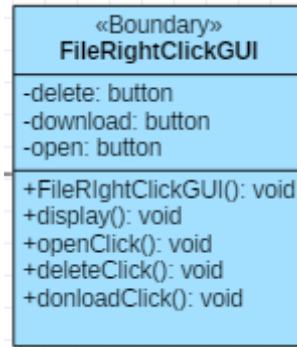
#### **Χαρακτηριστικά της κλάσης:**

- **create** : Κουμπί για την δημιουργία αρχείου.
- **upload** : Κουμπί για ανέβασμα αρχείου.

#### **Μέθοδοι της κλάσης:**

- **EmptySpaceRightClickGUI()** : Αποτελεί τον Constructor της κλάσης.
- **display()** : Η συνάρτηση αυτή εμφανίζει στο χρήστη το μενού όταν κάνει δεξί κλικ στο κενό
- **createClick()** : Καλεί τη συνάρτηση create() της κλάσης FileHandlingController.
- **UploadClick()** : Καλεί τη συνάρτηση upload() της κλάσης FileHandlingController.

### Boundary FileRightClick



Η κλάση αυτή είναι σχετίζεται με το μενού που εμφανίζεται όταν ο χρήστης κάνει δεξί κλικ πάνω σε ένα αρχείο.

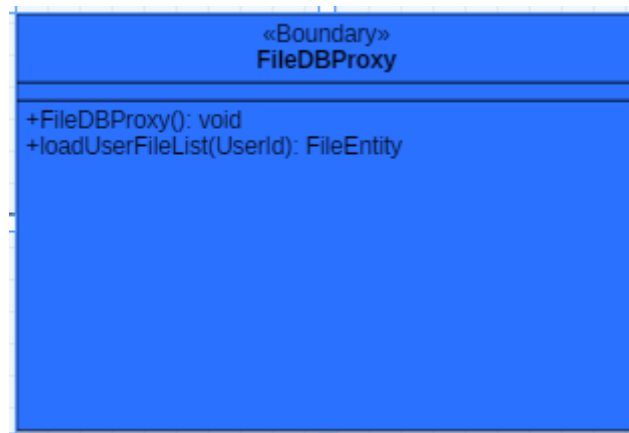
#### Χαρακτηριστικά της κλάσης:

- **open** : Κουμπί για την άνοιγμα αρχείου.
- **delete**: Κουμπί για διαγραφή αρχείου.
- **download**: Κουμπί για κατέβασμα αρχείου.

#### Μέθοδοι της κλάσης:

- **FileRightClickGUI()** : Αποτελεί τον Constructor της κλάσης.
- **display()** : Η συνάρτηση αυτή εμφανίζει στο χρήστη το μενού όταν κάνει δεξί κλικ πάνω σε αρχείο.
- **openClick()** : Καλεί τη συνάρτηση `open()` της κλάσης `FileHandlingController`.
- **deleteClick()** : Καλεί τη συνάρτηση `delete()` της κλάσης `FileHandlingController`.
- **downloadClick()** : Καλεί τη συνάρτηση `download()` της κλάσης `FileHandlingController`.

### Boundary FileDBProxy

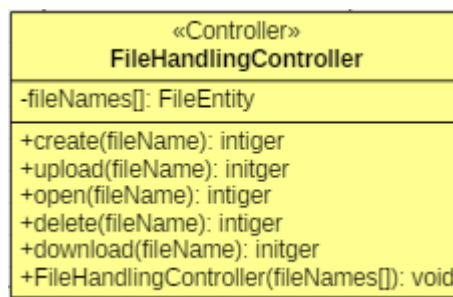


Η κλάση αυτή είναι υπεύθυνη για την επικοινωνία με τη βάση δεδομένων ώστε να φορτώνεται η λίστα με τα αρχεία του κάθε χρήστη.

#### Μέθοδοι κλάσεις :

- **fileDBProxy()** : Αποτελεί τον Constructor της κλάσης.
- **loadUserFileList(UserID)** : Η συνάρτηση αυτή βρίσκει και γορτώνει την λίστα των αρχείων του χρήστη από τη βάση δεδομένων .

### Controller FileHandlingController



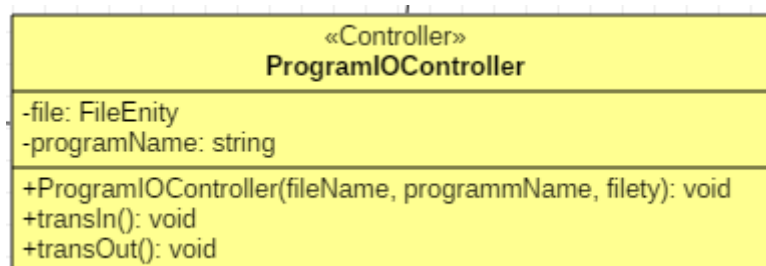
Η κλάση αυτή είναι υπεύθυνη για τον έλεγχο και την επιτέλεση των ενεργειών του χρήστη πάνω στα αρχεία.

**Χαρακτηριστικά της κλάσης:**

- **FileNames[]** : Μια λίστα με τα ονόματα των αρχείων του χρήστη.

**Μέθοδοι της κλάσης:**

- **FileHandlingController** : Αποτελεί τον Constructor της κλάσης.
- **create(fileName)** : Η συνάρτηση αυτή δημιουργεί ένα αρχείο με όνομα fileName και επιστρέφει έναν δείκτη στο αρχείο σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.
- **upload(fileName)** : Η συνάρτηση αυτή ανεβάζει στην πλατφόρμα ένα αρχείο με όνομα fileName και επιστρέφει έναν δείκτη στο αρχείο σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.
- **open(fileName)** : Η συνάρτηση αυτή ανοίγει ένα αρχείο με όνομα fileName και επιστρέφει έναν δείκτη στο αρχείο σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.
- **delete(fileName)** : Η συνάρτηση αυτή διαγράφει ένα αρχείο με όνομα fileName και επιστρέφει 1 σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.
- **download(fileName)** : Η συνάρτηση αυτή κατεβάζει στον τοπικό υπολογιστή ένα αρχείο με όνομα fileName και επιστρέφει 1 σε περίπτωση επιτυχίας ή -1 σε περίπτωση αποτυχίας.

**Controller ProgramIOController**

Η κλάση αυτή διαχειρίζεται τις εισόδους και τις εξόδους των προγραμμάτων που υποστηρίζει η πλατφόρμα.

**Χαρακτηριστικά της κλάσης:**

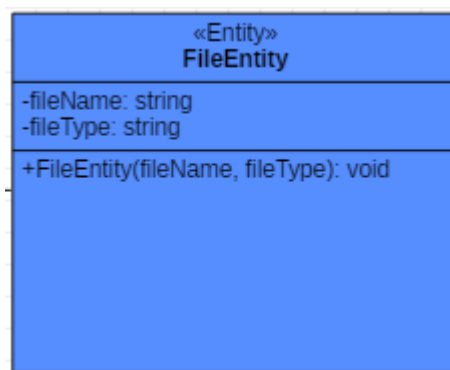
- **fileName** : Το όνομα του αρχείου που έχει ανοίξει ένα πρόγραμμα.

- **ProgramName** : Το όνομα του προγράμματος που άνοιξε το αρχείο.

#### Μέθοδοι της κλάσης:

- **ProgramIOController(fileName,programName)** : Αποτελεί τον Constructor της κλάσης.
- **tranIn()** : Διοχετεύει τις εισόδους του χρήστη στην είσοδο του προγράμματος όταν ο χρήστης βρίσκεται στο αντίστοιχο παράθυρο διεπαφής.
- **transOut()** : Διοχετεύει την έξοδο του προγράμματος στο κατάλληλο παράθυρο διεπαφής.

### Entity FileEntity



Αυτή είναι η κλάση οντότητας των αρχείων και περιέχει τα απαραίτητα στοιχεία για τον ορισμό τους.

#### Χαρακτηριστικά της κλάσης:

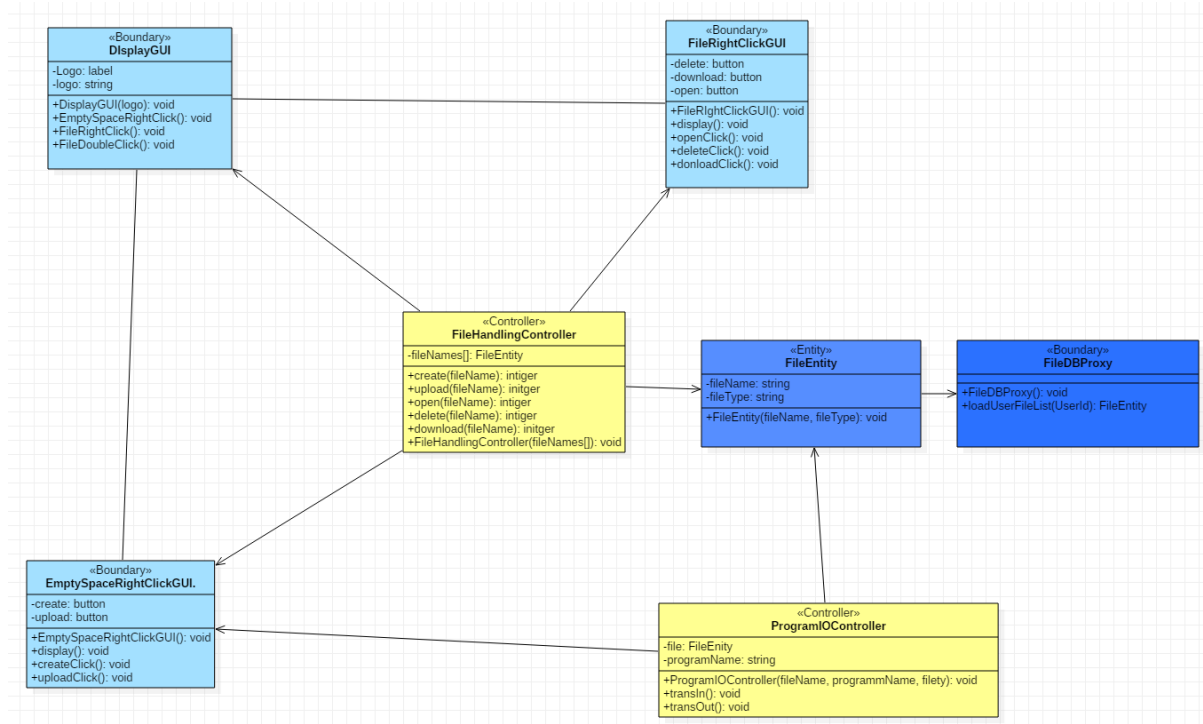
- **fileName** : Το όνομα του αρχείου.
- **fileType** : Ο τύπος του αρχείου.

#### Μέθοδοι κλάσης:

- **FileEntity(filename,filetype)** : Αποτελεί τον Constructor της κλάσης.



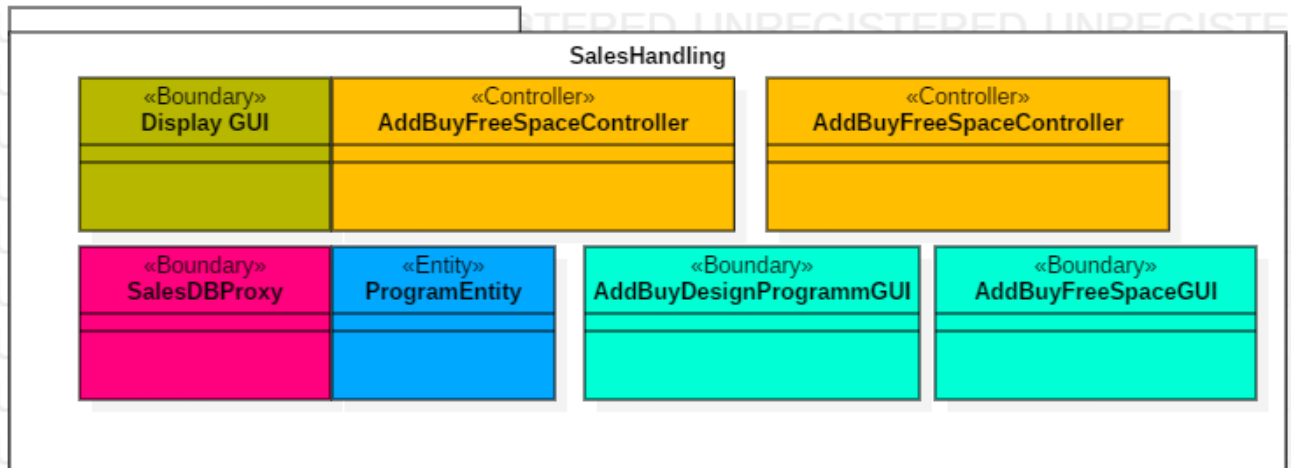
## Διάγραμμα κλάσεων



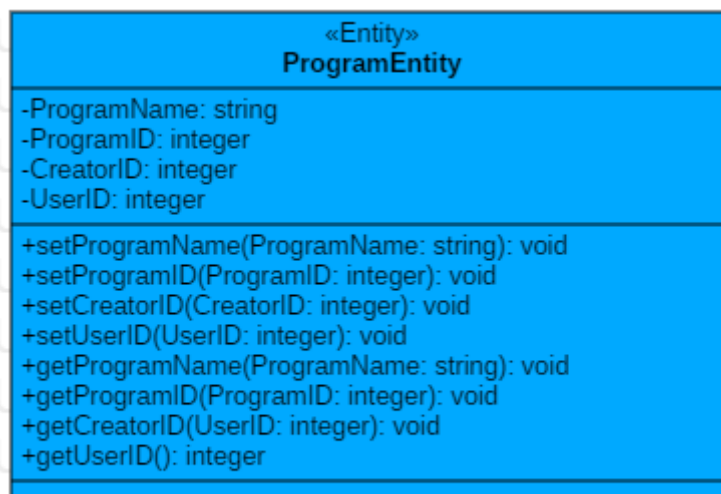
## 1.2 &lt;Πακέτο λεξιλογίου σεναρίων μεσαίας προτεραιότητας &gt;

## 1.2.1 Πακέτο SalesHandling

Το πακέτο αυτό είναι υπεύθυνο να επιτελέσει κάθε λειτουργία που αφορά την προσθήκη-αγορά νέων υποστηριζόμενων σχεδιαστικών προγραμμάτων από τον χρήστη και την αναβάθμιση του αποθηκευτικού του χώρου που έχει στην πλατφόρμα, δεδομένου ότι μπορεί κάποια στιγμή να τελειώσει ο ελεύθερος χώρος.



### Entity ProgramEntity



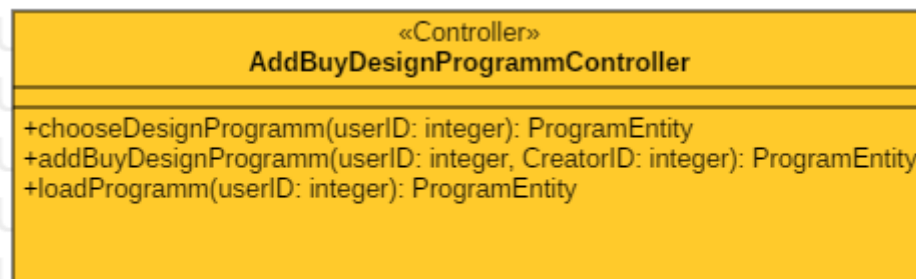
Αυτή είναι η κλάση οντότητας του σχεδιαστικού προγράμματος και περιέχει τα στοιχεία για τον ορισμό του.

#### Χαρακτηριστικά της κλάσης:

- **ProgramName:** String για το όνομα του program.
- **ProgramID:** Ο μοναδικός χαρακτηριστικός αριθμός του program.
- **CreatorID:** Ο μοναδικός χαρακτηριστικός αριθμός του δημιουργού του program.
- **UserID:** Ο μοναδικός χαρακτηριστικός αριθμός του συνδεδεμένου χρήστη.

**Μέθοδοι της κλάσης:**

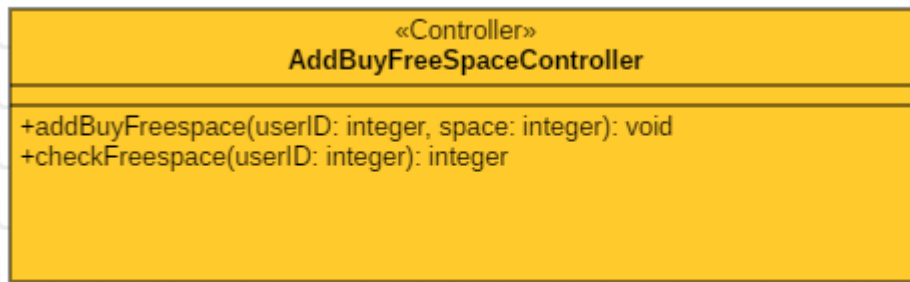
- **setProgramName(ProgramName: string):** Μέθοδος που θέτει το όνομα του Program.
- **setProgramID(ProgramID: integer):** Μέθοδος που θέτει το μοναδικό αναγνωριστικό του Program.
- **setCreatorID(CreatorID: integer):** Μέθοδος που θέτει το μοναδικό αναγνωριστικό του δημιουργού του Program.
- **setUserID(UserID: integer):** Μέθοδος που θέτει τον συνδεδεμένο χρήστη.
- **getProgramName(ProgramName: string):** Η μέθοδος επιστρέφει το όνομα του program.
- **getProgramID():** Η μέθοδος επιστρέφει το μοναδικό αναγνωριστικό του program.
- **getCreatorID():** Η μέθοδος επιστρέφει το μοναδικό αναγνωριστικό του δημιουργού του program.
- **getUserID():** Η μέθοδος επιστρέφει το μοναδικό χαρακτηριστικό αριθμό του συνδεδεμένου χρήστη.

**Controller AddBuyDesignProgrammController**

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για την πρόσθεση ή την αγορά ενός σχεδιαστικού προγράμματος ,την επιλογή του και την φόρτωσή του.

**Μέθοδοι της κλάσης:**

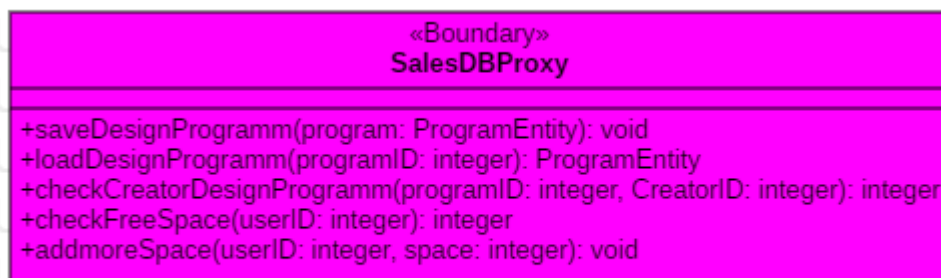
- **chooseDesignProgramm():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση των αντίστοιχων σχεδιαστικών programs.
- **addBuyDesignProgramm():** Η μέθοδος αυτή καλείται με σκοπό την πρόσθεση ή την αγορά του αντίστοιχου σχεδιαστικού προγράμματος.
- **loadProgramm():** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση κάποιου program.

**Controller BrowseProjectsControl**

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για την αγορά επιπλέον αποθηκευτικού χώρου.

**Μέθοδοι της κλάσης:**

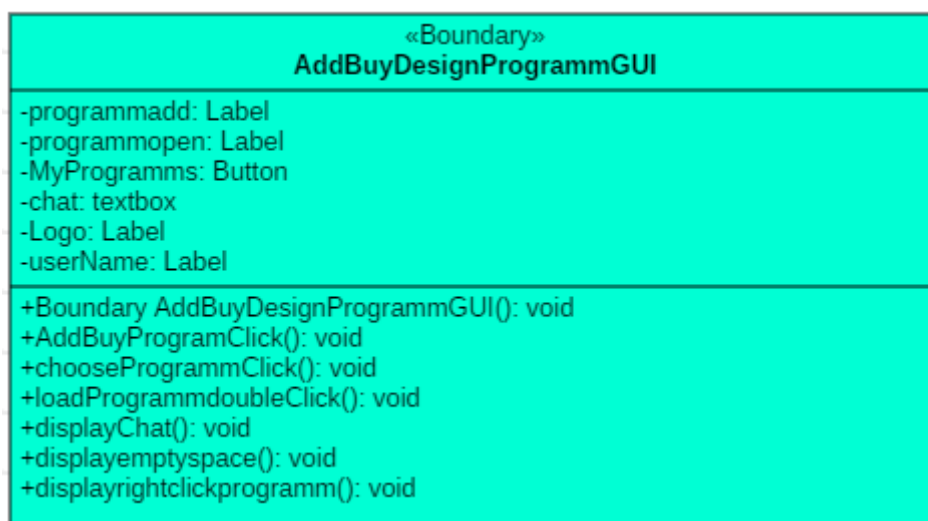
- **addBuyFreespace():** Η μέθοδος αυτή καλείται με σκοπό την αγορά συγκεκριμένου αποθηκευτικού χώρου από ένα χρήστη.
- **checkFreespace():** Η μέθοδος αυτή καλείται με σκοπό τον έλεγχο του διαθέσιμου αποθηκευτικού χώρου.

**Boundary SalesDBProxy**

Η κλάση αυτή αναπαριστά τη βάση δεδομένων με την οποία επικοινωνούν οι ελεγκτές για να κάνουν τις διάφορες λειτουργίες τους (είτε τραβάνε δεδομένα από αυτήν, είτε αποθηκεύουν).

**Μέθοδοι της κλάσης:**

- **saveDesignProgramm():** Η μέθοδος αυτή αποθηκεύει ένα σχεδιαστικό πρόγραμμα στη Βάση Δεδομένων.
- **loadDesignProgramm():** Η μέθοδος αυτή επιστρέφει ένα σχεδιαστικό πρόγραμμα από τη βάση δεδομένων, μέσω του μοναδικού αναγνωριστικού της.
- **checkDesignProgramm():** Η μέθοδος αυτή ελέγχει αν υπάρχει το συγκεκριμένο σχεδιαστικό πρόγραμμα στη Βάση Δεδομένων.
- **checkFreeSpace():** Η μέθοδος αυτή καλείται με σκοπό την εύρεση του διαθέσιμου αποθηκευτικού χώρου ενός χρήστη.
- **addmoreSpace():** Η μέθοδος αυτή καλείται με σκοπό την προσθήκη επιπλέον αποθηκευτικού χώρου στο λογαριασμό ενός χρήστη.

**Boundary AddBuyDesignProgrammGUI**

Η κλάση αυτή αναπαριστά τη διεπαφή για την εμφάνιση της προσθήκης ή αγοράς σχεδιαστικών προγραμμάτων αλλά βέβαια και της επιλογής σχεδιαστικού προγράμματος προς εκτέλεση.

**Χαρακτηριστικά της κλάσης:**

- **programmadd:** Εικόνα για το κουμπί-σταυρό με τον οποίο μπορείς να προσθέσεις σχεδιαστικά programs.
- **programmopen:** Εικόνα για το άνοιγμα ενός σχεδιαστικού program.
- **MyProgramms:** Κουμπί για την εμφάνιση των σχεδιαστικών programms.

- **chat:** textbox για το άνοιγμα του chat.
- **Logo:** Επιγραφή εμφάνισης του logo της εφαρμογής.
- **UserName:** Επιγραφή εμφάνισης του ονόματος του χρήστη.

#### Μέθοδοι της κλάσης:

- **AddBuyDesignProgrammGUI():** void Αποτελεί τον Constructor της κλάσης.
- **AddBuyProgramClick():** void Καλεί την συνάρτηση addBuyDesignProgramm() του ελεγκτή AddBuyDesignProgrammController για την προσθήκη ή την αγορά κάποιου σχεδιαστικού προγράμματος.(προσθήκη αν είναι δωρεάν,αγορά αν δεν είναι).
- **chooseProgramClick():** void Καλεί την συνάρτηση chooseDesignProgramm() του ελεγκτή AddBuyDesignProgrammController για την εμφάνιση των σχεδιαστικών προγραμμάτων που υπάρχουν.
- **loadProgrammdoubleClick():** void Καλεί την συνάρτηση loadProgramm() του ελεγκτή AddBuyDesignProgrammController για την εμφάνιση ενός συγκεκριμένου σχεδιαστικού προγράμματος.
- **displayChat():** void Καλεί τον αντίστοιχο ελεγκτή για την εμφάνιση του chat.
- **displayemptyspace():** void Η συνάρτηση αυτή εμφανίζει στο χρήστη το μενού όταν κάνει δεξί κλικ στο κενό.
- **displayrightclickprogramm():** void Η συνάρτηση αυτή εμφανίζει στο χρήστη το μενού όταν κάνει δεξί κλικ πάνω σε ένα σχεδιαστικό πρόγραμμα.

#### Boundary AddBuyFreeSpaceGUI

«Boundary» AddBuyFreeSpaceGUI
-freespace: Bar -Upgrade: Button -Available_Storage_Space: Button -chat: textbox -Logo: Label -userName: Label
+AddBuyFreeSpaceGUI(): void +addBuyFreespaceClick(): void +checkFreespaceClick(): void +displayChat(): void +displayemptyspace(): void

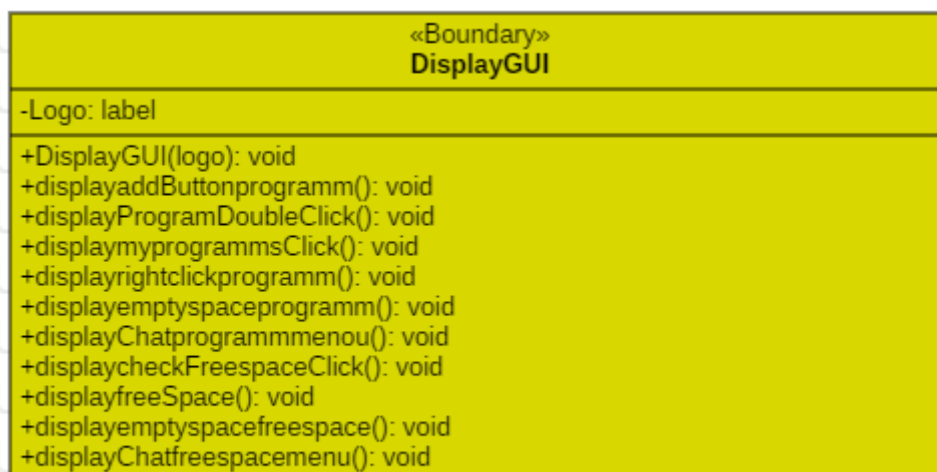
Η κλάση αυτή αναπαριστά τη διεπαφή για την αγορά επιπλέον αποθηκευτικού χώρου αλλά βέβαια και για να δεις τον αποθηκευτικό χώρο που έχεις.

**Χαρακτηριστικά της κλάσης:**

- **freespace:** Μπάρα όπου ο χρήστης βλέπει τον διαθέσιμο αποθηκευτικό χώρο.
- **Upgrade:** Κουμπί για να αγοράσει ο χρήστης επιπλέον αποθηκευτικό χώρο.
- **Available\_Storage\_Space:** Κουμπί για να μπει ο χρήστης στην καρτέλα όπου θα δει τον διαθέσιμο αποθηκευτικό χώρο του και θα μπορεί να αγοράσει επιπλέον.
- **chat:** textbox για το άνοιγμα του chat.
- **Logo:** Επιγραφή εμφάνισης του logo της εφαρμογής.
- **UserName:** Επιγραφή εμφάνισης του ονόματος του χρήστη.

**Μέθοδοι της κλάσης:**

- **AddBuyFreeSpaceGUI():void:** Καλεί τον Constructor της κλάσης.
- **addBuyFreespaceClick():void** Καλεί την συνάρτηση addBuyFreespace() του ελεγκτή AddBuyFreeSpaceController για την προσθήκη ή αγορά αποθηκευτικού χώρου.
- **checkFreespaceClick():void** Καλεί την συνάρτηση checkFreespace() του ελεγκτή AddBuyFreeSpaceController για τον έλεγχο του αποθηκευτικού χώρου του χρήστη.
- **displayChat():** Καλεί την αντίστοιχο ελεκτή για την εμφάνιση του chat.
- **displayemptyspace() :void** Η συνάρτηση αυτή εμφανίζει στο χρήστη το μενού όταν κάνει δεξί κλικ στο κενό.

**Boundary DisplayGUI**

Η κλάση αυτή εκφράζει τη γραφική διεπαφή του χρήστη με τα αρχεία και το περιβάλλον του συστήματος.

**Χαρακτηριστικά της κλάσης:**

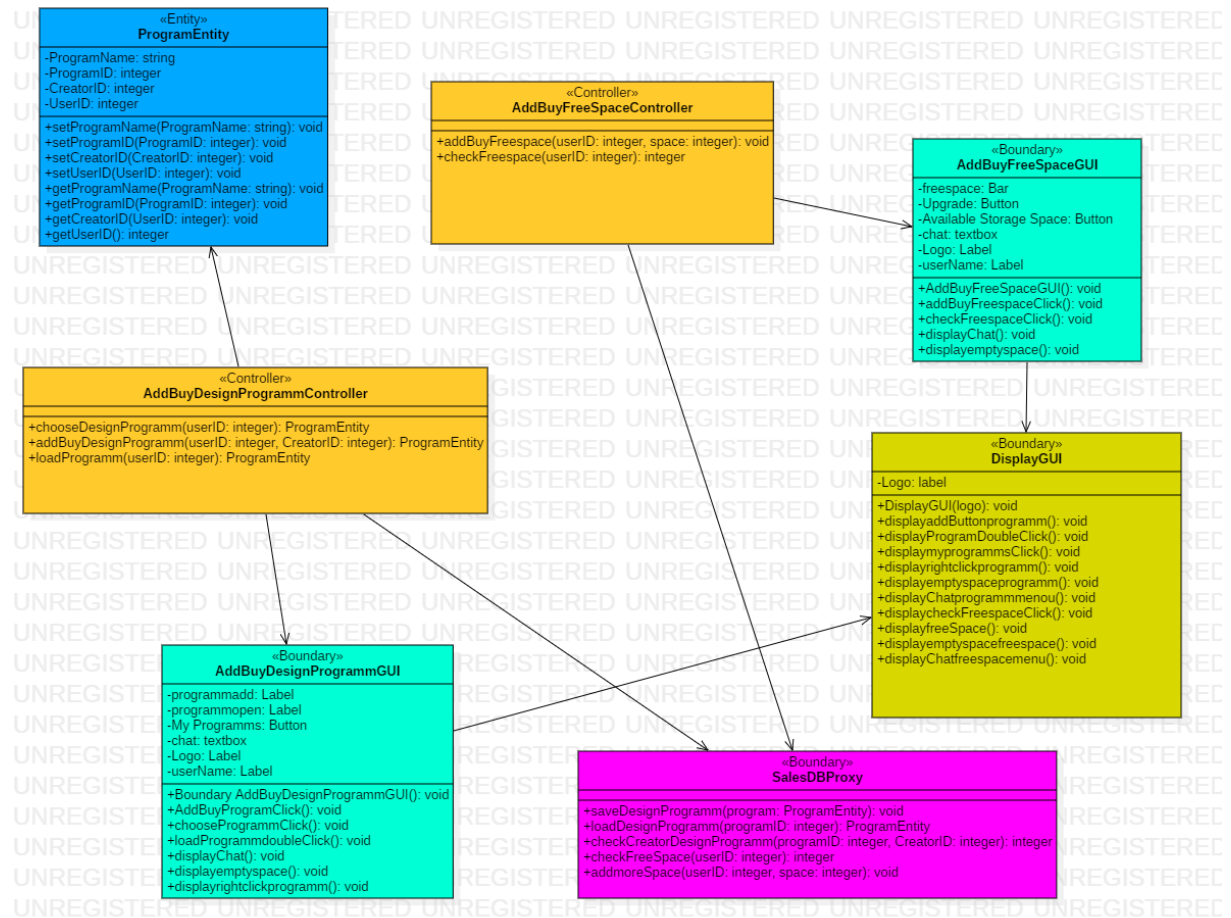
- **Logo:**label:Λογότυπο της εφαρμογής.

**Μέθοδοι της κλάσης:**

- **DisplayGUI(logo):void** Αποτελεί τον constructor της κλάσης.
- **displayaddButtonprogramm() :void:** καλεί τη συνάρτηση AddBuyProgramClick() της κλάσης AddBuyDesignProgrammGUI.
- **displayProgramDoubleClick(): void:** Καλεί τη συνάρτηση loadProgrammdoubleClick() της κλάσης AddBuyDesignProgrammGUI.
- **displaymyprogrammsClick(): void:** Καλεί τη συνάρτηση chooseProgramClick() της κλάσης AddBuyDesignProgrammGUI.
- **displayrightclickprogramm(): void:** Καλεί τη συνάρτηση displayrightclickprogramm() της κλάσης AddBuyDesignProgrammGUI.
- **displayemptyspaceprogramm(): void:** Καλεί τη συνάρτηση displayemptyspace() της κλάσης AddBuyDesignProgrammGUI.
- **displayChatprogrammmenu(): void:** Καλεί την συνάρτηση displayChat() της κλάσης AddBuyDesignProgrammGUI για την εμφάνιση του chat σε εκείνη τη σελίδα.
- **displaycheckFreespaceClick(): void:** καλεί τη συνάρτηση checkFreespaceClick() της κλάσης AddBuyFreeSpaceGUI για τον έλεγχο του αποθηκευτικού χώρου.
- **displayfreeSpace(): void:** Η συνάρτηση αυτή εμφανίζει στο χρήστη την επιλογή προσθήκης επιπλέον αποθηκευτικού χώρου όταν κάνει κλικ στο Upgrade καλώντας τη συνάρτηση addBuyFreespaceClick() της κλάσης AddBuyFreeSpaceGUI..
- **displayemptyspacefreespace(): void:** Η συνάρτηση αυτή καλεί τη συνάρτηση displayemptyspace() της κλάσης AddBuyFreeSpaceGUI.
- **displayChatfreespacemenu(): void:** Καλεί την συνάρτηση displayChat() της κλάσης AddBuyFreeSpaceGUI για την εμφάνιση του chat σε εκείνη τη σελίδα.



## Διαγράμμα κλάσεων

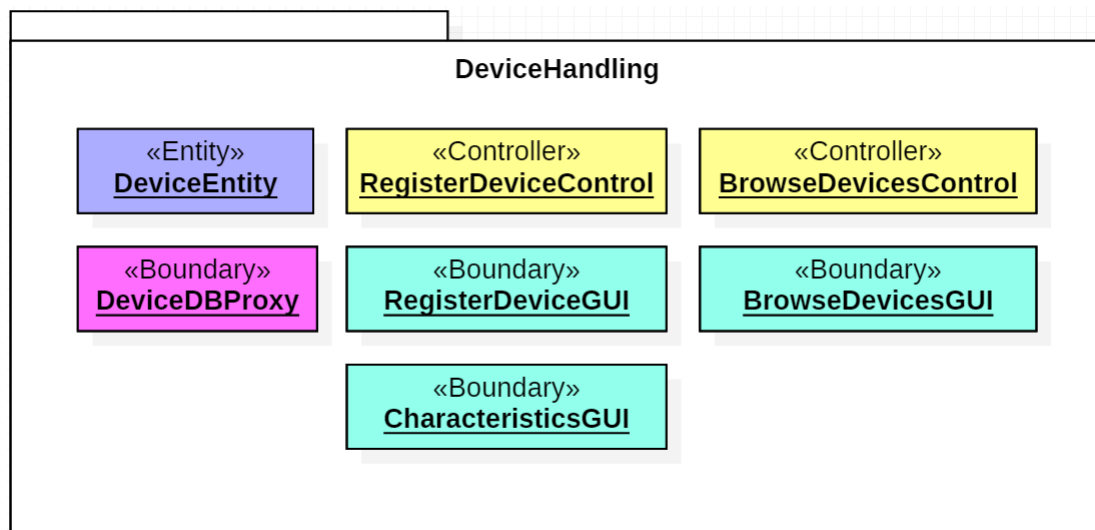


## 1.3 &lt;Πακέτα λεξιλογίου σεναρίων χαμηλής προτεραιότητας &gt;

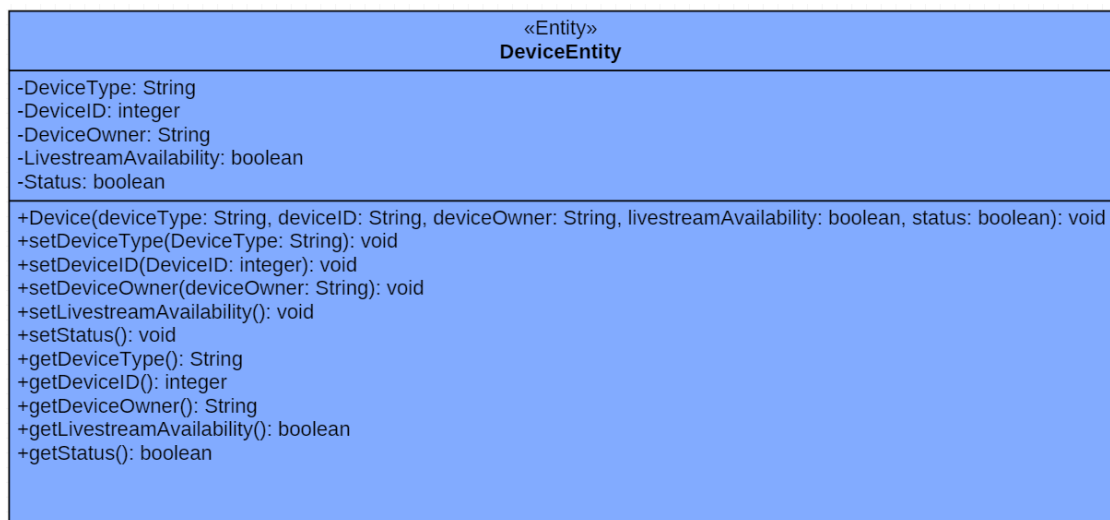
Σε αυτό το πακέτο εμπεριέχονται οι κλάσεις που εμπλέκονται στα σενάρια χρήσης χαμηλής προτεραιότητας (ΛΑ-8, ΛΑ-9, ΛΑ-10).

## 1.3.1 Πακέτο DeviceHandling

Το πακέτο αυτό είναι υπεύθυνο να επιτελέσει κάθε λειτουργία που αφορά την προσθήκη καινούργιας συσκευής στην πλατφόρμα και την περιήγηση του χρήστη στις συνδεδεμένες συσκευές.



### Entity DeviceEntity



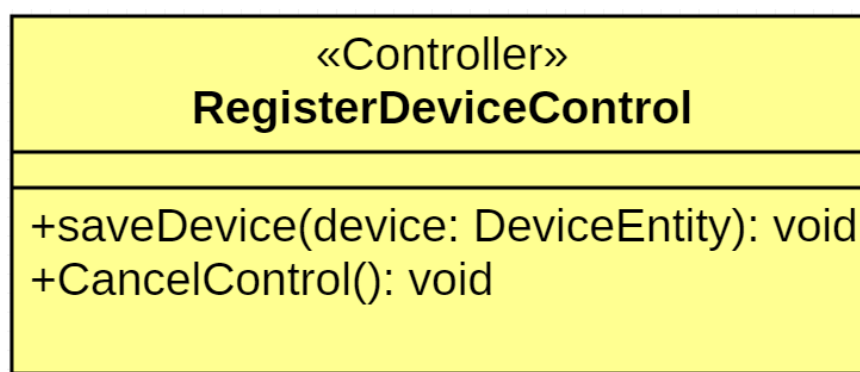
Αυτή είναι η κλάση οντότητας της εξωτερικής συσκευής και περιέχει τα στοιχεία για τον ορισμό της.

#### Χαρακτηριστικά της κλάσης:

- **DeviceType:** String με το είδος εξωτερικής συσκευής.
- **DeviceID:** Ο μοναδικός χαρακτηριστικός αριθμός της συσκευής.
- **DeviceOwner:** String με το όνομα του ιδιοκτήτη της συσκευής.
- **LivestreamAvailability:** Boolean μεταβλητή σχετικά με το αν έχει την δυνατότητα η συσκευή να παρέχει livestream video, τις στιγμές που είναι σε λειτουργία.
- **Status:** Boolean μεταβλητή με την κατάσταση στην οποία βρίσκεται η συσκευή (on/off).

**Μέθοδοι της κλάσης:**

- **Device(deviceType: String, deviceId: String, deviceOwner: String, livestreamAvailability: boolean, status: boolean): void:** Constructor της κλάσης DeviceEntity.
- **setDeviceType(DeviceType: String): void:** Μέθοδος που θέτει τον τύπο της εξωτερικής συσκευής.
- **setDeviceID(DeviceID: integer): void:** Μέθοδος που θέτει τον μοναδικό χαρακτηριστικό αριθμό της συσκευής.
- **setDeviceOwner(deviceOwner: String): void:** Μέθοδος που θέτει το όνομα του ιδιοκτήτη της συσκευής.
- **setLivestreamAvailability(): void:** Μέθοδος που ορίζει αν η συσκευή έχει δυνατότητα μετάδοσης livestream video.
- **setStatus(): void:** Μέθοδος που ορίζει την κατάσταση της συσκευής (on/off).
- **getDeviceType(): String:** Μέθοδος που επιστρέφει τον τύπο της συσκευής.
- **getDeviceID(): integer:** Μέθοδος που επιστρέφει τον μοναδικό χαρακτηριστικό αριθμό της συσκευής.
- **getDeviceOwner(): String:** Μέθοδος που επιστρέφει το όνομα του ιδιοκτήτη της συσκευής.
- **getLivestreamAvailability(): Boolean:** Μέθοδος που επιστρέφει το αν η συσκευή έχει την δυνατότητα μετάδοσης livestream video.
- **getStatus(): Boolean:** Μέθοδος που επιστρέφει την κατάσταση της συσκευής (on/off).

**Controller RegisterDeviceControl:**

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για τη καταχώρηση μιας νέας συσκευής στην πλατφόρμα.

**Μέθοδοι της κλάσης:**

- **ssaveDevice(device: DeviceEntity): void:** Η μέθοδος αυτή καλείται με σκοπό να αποθηκεύσει την συσκευή στην πλατφόρμα.
- **CancelControl(): void:** Πρόκειται για μέθοδο που ακυρώνει την καταχώρηση της συσκευής στην πλατφόρμα.

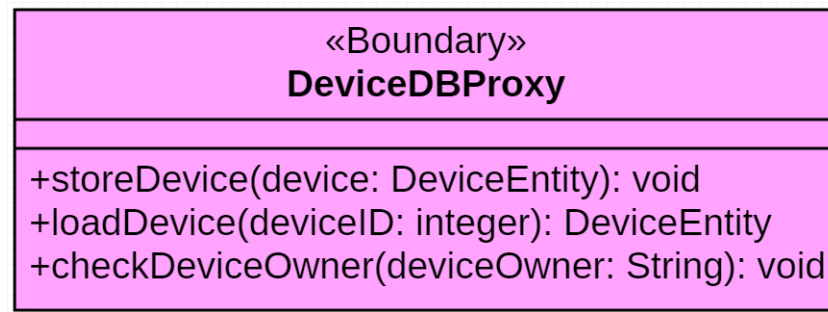
**Controller BrowseDevicesControl:**

«Controller» <b>BrowseDevicesControl</b>
+showDevices(): DeviceEntity +addNewDevice(): DeviceEntity +loadDevice(): DeviceEntity +showLivestreamingGUI(livestreamingAvailability: boolean): void

Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για την περιήγηση στη λίστα των συνδεδεμένων συσκευών στην πλατφόρμα.

**Μέθοδοι της κλάσης:**

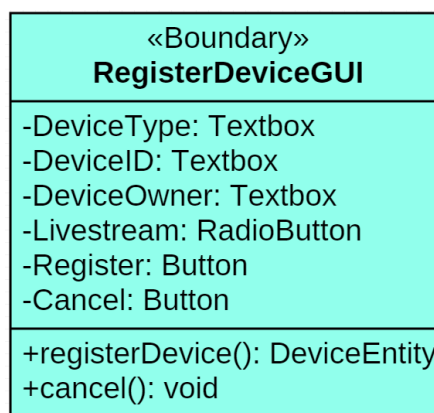
- **showDevices(): DeviceEntity:** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση των καταχωρημένων συσκευών στην πλατφόρμα.
- **addNewDevice(): DeviceEntity:** Η μέθοδος αυτή καλείται με σκοπό την καταχώρηση καινούριας συσκευής.
- **loadDevice(): DeviceEntity:** Η μέθοδος αυτή καλείται με σκοπό την εμφάνιση των χαρακτηριστικών μιας συσκευής.
- **showLivestreamingGUI(LivestreamingAvailability: boolean): void:** Η μέθοδος αυτή εμφανίζει το περιβάλλον διεπαφής livestreaming video, όταν η LivestreamingAvailability είναι true.

**Boundary DeviceDBProxy:**

Η κλάση αυτή χρησιμοποιείται για επικοινωνία με τη βάση δεδομένων όπου είναι αποθηκευμένα στοιχεία των συσκευών.

**Μέθοδοι της κλάσης:**

- **storeDevice(in device:DeviceEntity): void:** Η μέθοδος αυτή αποθηκεύει μια συσκευή στη βάση δεδομένων.
- **loadDevice(in deviceID:integer): DeviceEntity:** Η μέθοδος αυτή επιστρέφει μια συσκευή από την βάση δεδομένων, μέσω του μοναδικού χαρακτηριστικού αριθμού της συσκευής.
- **checkDeviceOwner(in deviceOwner:String): void:** Η μέθοδος αυτή ελέγχει αν ο χρήστης είναι ο ιδιοκτήτης της συσκευής.

**Boundary RegisterDeviceGUI:**

Η κλάση αυτή εκφράζει τη διεπαφή της φόρμας καταχώρησης της εξωτερικής συσκευής.

**Χαρακτηριστικά της κλάσης:**

- **DeviceType: Textbox:** Πεδίο συμπλήρωσης του τύπου της συσκευής.
- **DeviceID: Textbox:** Πεδίο συμπλήρωσης του μοναδικού χαρακτηριστικού αριθμού της συσκευής.
- **DeviceOwner: Textbox:** Πεδίο συμπλήρωσης του ονόματος του ιδιοκτήτη της συσκευής.
- **Livestream: RadioButton:** RadioButton για το αν έχει δυνατότητα μετάδοσης livestream video η συσκευή(yes/no).
- **Register: Button:** Κουμπί για την ολοκλήρωση καταχώρησης της συσκευής.
- **Cancel: Button:** Κουμπί για την ακύρωση της καταχώρησης της συσκευής.

**Μέθοδοι της κλάσης:**

- **registerDevice(): DeviceEntity:** Η μέθοδος αυτή καλεί την συνάρτηση saveDevice(device: DeviceEntity) για την καταχώρηση της συσκευής στην πλατφόρμα.
- **cancel(): void:** Η μέθοδος αυτή ακυρώνει την καταχώρηση της συσκευής στην πλατφόρμα.

**Boundary BrowseDevicesGUI:**

«Boundary» <b>BrowseDevicesGUI</b>
-registeredDevice: String -DeviceTypeImage: Image -AddDevice: Button
+addDeviceClick(): void +loadDeviceClick(): void

Η κλάση αυτή εκφράζει την διεπαφή για την περιήγηση μεταξύ των συνδεδεμένων συσκευών στην πλατφόρμα.

**Χαρακτηριστικά της κλάσης:**

- **registeredDevice:** String: String με το όνομα της συσκευής.
- **DeviceTypeImage:** Image: Εικόνα του τύπου της συσκευής.
- **AddDevice:** Button: Κουμπί για την καταχώρηση νέας συσκευής στην πλατφόρμα.

**Μέθοδοι της κλάσης:**

- **addDeviceClick(): void:** Η μέθοδος αυτή καλεί την συνάρτηση addNewDevice() του ελεγκτή BrowseDevicesControl για την προσθήκη νέας συσκευής στην πλατφόρμα.
- **loadDeviceClick(): void:** Η μέθοδος αυτή καλεί την συνάρτηση loadDevice() του ελεγκτή BrowseDevicesControl για την εμφάνιση των χαρακτηριστικών της συσκευής.

**Boundary CharacteristicsGUI:**

«Boundary» CharacteristicsGUI
-DeviceType: String -DeviceID: String -DeviceOwner: String -Livestreaming: String -Status: String
+editDevice(device: DeviceEntity): void +exit(): void

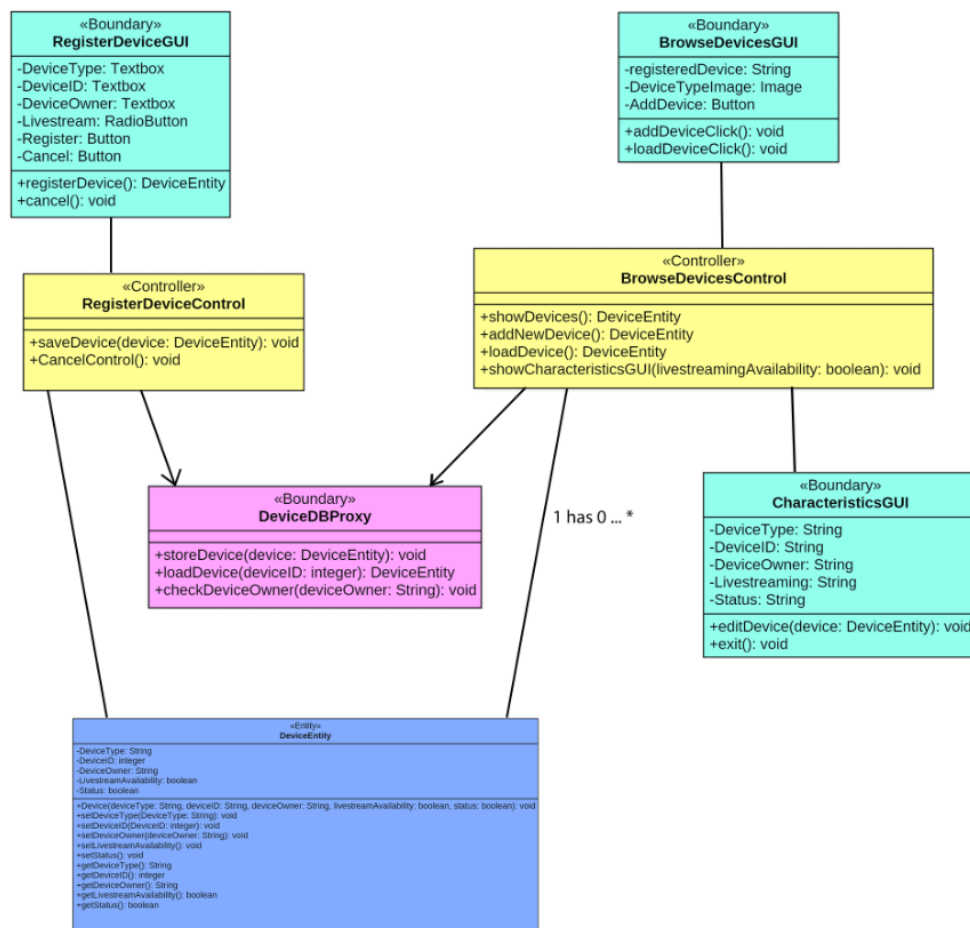
Η κλάση αυτή εκφράζει την διεπαφή για τα χαρακτηριστικά κάθε συνδεδεμένης συσκευής στην πλατφόρμα.

**Χαρακτηριστικά της κλάσης:**

- **DeviceType: String:** String με το όνομα του τύπου συσκευής.
- **DeviceID: String:** String με τον μοναδικό χαρακτηριστικό αριθμό της συσκευής.
- **DeviceOwner: String:** String με το όνομα
- **Livestreaming: String:** String με το αν η συσκευή έχει την δυνατότητα livestream video.
- **Status: String:** String με το status της συσκευής.

**Μέθοδοι της κλάσης:**

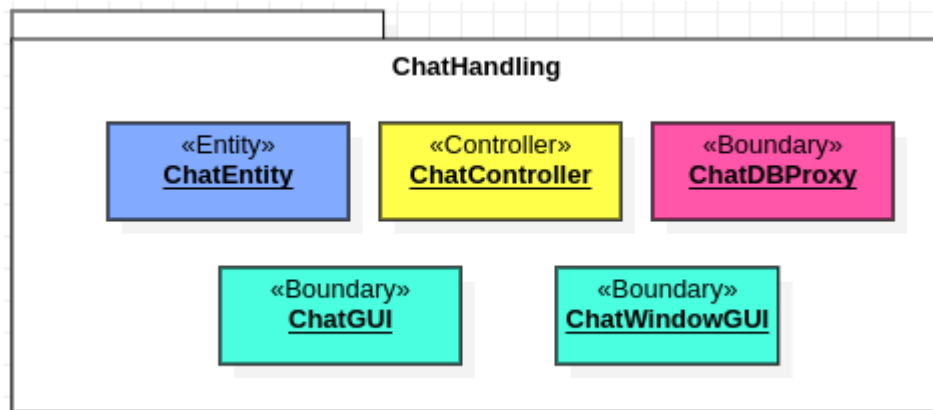
- **editDevice(device: DeviceEntity): void:** Μέθοδος για την επεξεργασία των στοιχείων της συσκευής.
- **exit(): void:** Μέθοδος για έξοδο από το CharacteristicsGUI.

**Διάγραμμα κλάσεων**

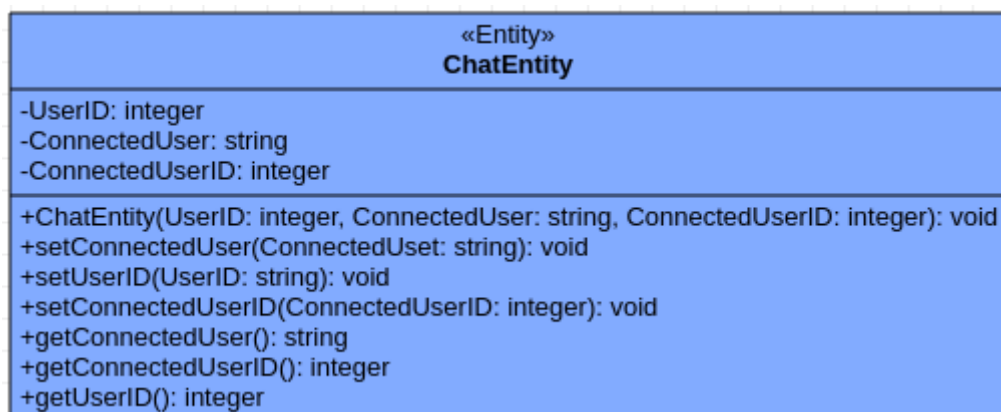


### 1.3.2 Πακέτο ChatHandling

Το πακέτο αυτό είναι υπεύθυνο να επιτελέσει κάθε λειτουργία που αφορά την επικοινωνία των χρηστών μεταξύ τους.



#### Entity ChatEntity



Αυτή είναι η κλάση οντότητας του Chat και περιέχει τα στοιχεία για τον ορισμό του.

#### Χαρακτηριστικά της κλάσης:

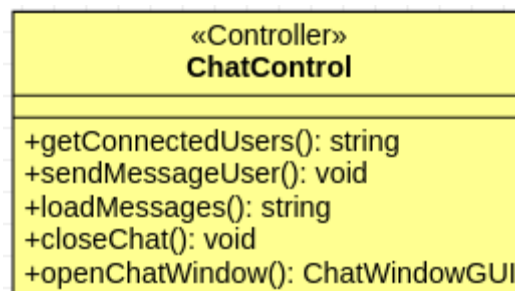
- **UserID:** Ο μοναδικός χαρακτηριστικός αριθμός του συνδεδεμένου χρήστη.
- **ConnectedUser:** String για το όνομα ενός συνδεδεμένου χρήστη.

- **ConnectedUserID:** Ο μοναδικός χαρακτηριστικός αριθμός του συνδεδεμένου χρήστη.

#### Μέθοδοι της κλάσης:

- **ChatEntity():** Constructor της κλάσης ChatEntity.
- **setUserID(UserID: integer):** Μέθοδος που θέτει τον μοναδικό χαρακτηριστικό αριθμό του συνδεδεμένου χρήστη.
- **setConnectedUser(ConnectedUser: string):** Μέθοδος που θέτει το όνομα ενός άλλου συνδεδεμένου χρήστη στην πλατφόρμα.
- **setConnectedUserID(ConnectedUserID: integer):** Μέθοδος που θέτει το μοναδικό αναγνωριστικό αριθμό του άλλου συνδεδεμένου χρήστη στην πλατφόρμα.
- **getConnectedUser():** Η μέθοδος επιστρέφει το όνομα ενός άλλου συνδεδεμένου χρήστη.
- **getProjectID():** Η μέθοδος επιστρέφει το μοναδικό αναγνωριστικό αριθμό του άλλου συνδεδεμένου χρήστη.
- **getUserID():** Η μέθοδος επιστρέφει τον μοναδικό χαρακτηριστικό αριθμό του συνδεδεμένου χρήστη.

### Controller ChatControl



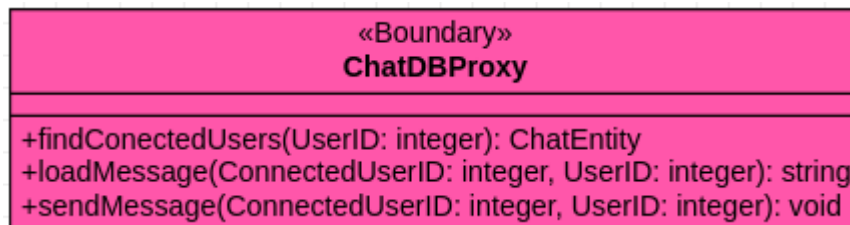
Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις για τη λειτουργία του Chat.

#### Μέθοδοι της κλάσης:

- **GetConnectedUsers():** Η μέθοδος αυτή καλείται με σκοπό να βρεθούν ποιοι άλλοι χρήστες είναι συνδεδεμένοι.
- **sendMessageUser():** Η μέθοδος αυτή καλείται με σκοπό την αποστολή μηνύματος σε άλλων χτήστη.
- **loadMessages():** Η μέθοδος αυτή καλείται με σκοπό την να φορτώσει παλιότερα μηνύματα.
- **loadProject():** Η μέθοδος αυτή καλείται με σκοπό την κλείσιμο του chat.

- **OpenChatWindow():** Η μέθοδος αυτή καλείται με σκοπό το άνοιγμα παραθύρου επικοινωνίας με κάποιον χρήστη.

### Boundary ChatDBProxy

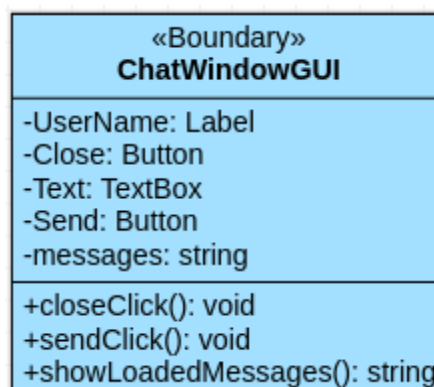


Η κλάση αυτή επικοινωνεί με απομακρυσμένη βάση δεδομένων όπου είναι αποθηκευμένο οι υπόλοιποι χρήστες και μπορεί να γίνει επικοινωνία μαζί τους.

#### Μέθοδοι της κλάσης:

- **findConnectedUsers():** Η μέθοδος αυτή επιστρέφει ένα άλλο συνδεδεμένο χρήστη, δηλαδή κλάση ChatEntity, από βάση δεδομένων.
- **loadMessage():** Η μέθοδος αυτή επιστρέφει όλα τα μηνύματα μεταξύ των 2 χρηστών από τη βάση δεδομένων, μέσω των μοναδικών αναγνωριστικων τους.
- **sendMessage():** Η μέθοδος αυτή στέλνει όλα τα μηνύματα μεταξύ των 2 χρηστών μέσω της βάσης δεδομένων και μέσω των μοναδικών αναγνωριστικων τους.

### Boundary ChatWindowGUI



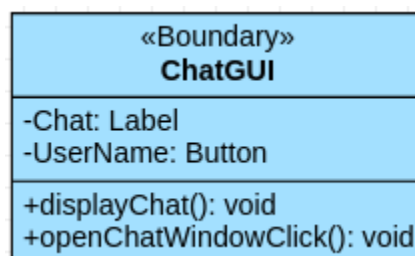
Η κλάση αυτή αναπαριστά τη διεπαφή για την εμφάνιση του παραθύρου επικοινωνίας 2 χρηστών.

**Χαρακτηριστικά της κλάσης:**

- **UserName:** Επιγραφή εμφάνισης του ονόματος του χρήστη.
- **Close:** Κουμπί για το κλείσιμο του παραθύρου.
- **Text:** Πεδίο εισαγωγής του μηνύματος μεταξύ των 2 χρηστών.
- **Send:** Κουμπί για την αποστολή του μηνύματος.
- **messages:** Παλαιότερα μηνύματα.

**Μέθοδοι της κλάσης:**

- **closeClick():** Καλεί την συνάρτηση closeChat() του ελεγκτή ChatControl για την κλείσιμο του παραθύρου.
- **sendClick():** Καλεί την συνάρτηση sendMessageUser() ελεγκτή ChatControl για την αποστολή του μηνύματος.
- **showLoadMessages():** Καλεί την συνάρτηση loadMessages() του ελεγκτή ChatControl για την εμφάνιση παλαιότερων μηνυμάτων.

**Boundary BrowseProjectsGUI**

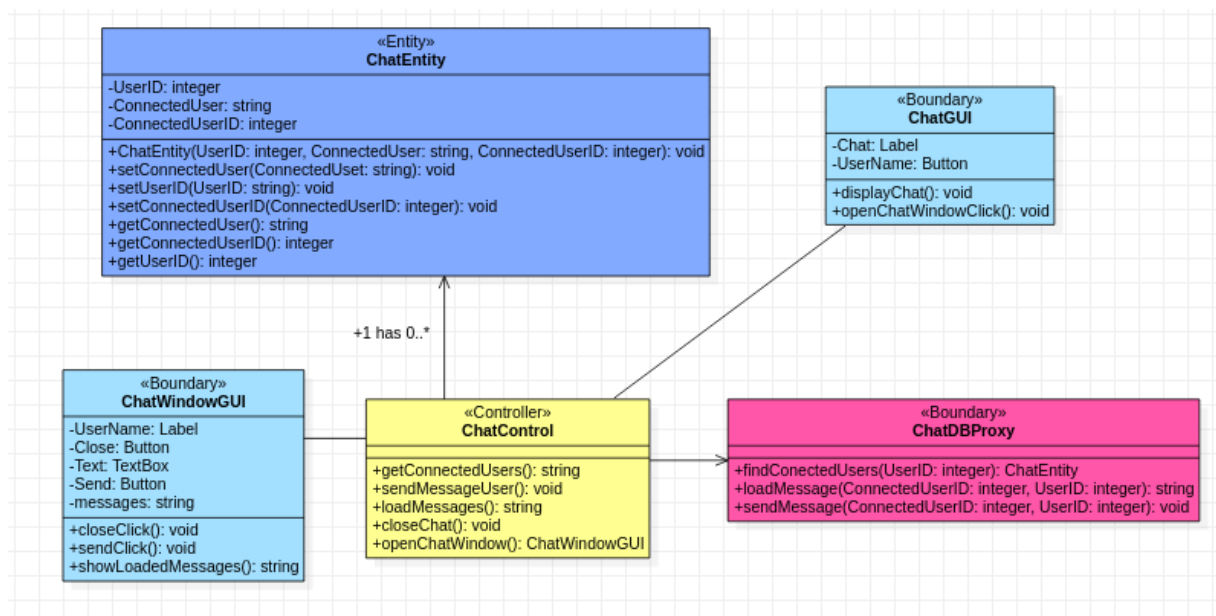
Η κλάση αυτή αναπαριστά τη διεπαφή για την εμφάνιση του chat με όλους τους συνδεδεμένους χρήστες.

**Χαρακτηριστικά της κλάσης:**

- **Chat:** Επιγραφή εμφάνισης της θέσης του chat.
- **username:** Κουμπί για την άνοιγμα ενός παραθύρου επικοινωνίας μεταξύ 2 χρηστών.

**Μέθοδοι της κλάσης:**

- **displayChat():** Καλεί την συνάρτηση GetConnectedUsers() του ελεγκτή ChatControl για να εμφανιστούν οι συνδεδεμένοι χρήστες.
- **openChatWindowClick():** Καλεί την συνάρτηση openChatWindow() του ελεγκτή ChatControl, όταν πατήσει ο χρήστης πάνω στο όνομα κάποιου συνδεδεμένου χρήστη, για την εμφάνιση ενός παραθύρου επικοινωνίας 2 χρηστών.

**Διαγράμμα κλάσεων**

## Μη λειτουργικές απαιτήσεις

### 2.1 Απαιτήσεις επίδοσης

#### <ΜΛΑ-2>

Το σύστημα πρέπει να υποστηρίζει την ταυτόχρονη σύνδεση έως και 100.000 χρηστών.

**Περιγραφή :** Το σύστημα πρέπει να λειτουργεί χωρίς πρόβλημα και να καλύπτει τις ανάγκες μέχρι και 1000.000 χρηστών.

**User Priority (4/5):** Είναι σημαντικό να εξυπηρετούνται πολύ χρήστες ταυτόχρονα καθώς το ώστε να μποεί το σύστημα να παρέχει τις υπηρεσίες του σε κάθε χρήστη ανά πάσα στιγμή.

**Technical Priority (4/5):** Η απαίτηση είναι αρκετά σημαντική για να καταστήσει λειτουργικό το σύστημα.

**Stability(3/5) :** Το σύστημα ίσως να χρειαστεί να υποστηρίζει περισσότερους χρήστες στο μέλλον.

#### <ΜΛΑ-6>

Το site θα πρέπει να έχει χρόνο απόκρισης 100 msec.

**Περιγραφή :** Το site θα πρέπει να αποκρίνεται στα 100 msec.

**User Priority (3/5):** Είναι σχετικά σημαντική απαίτηση για το χρήστη καθώς είναι χρήσιμη για την εύκολη παροχή υπηρεσιών και την εύκολη εξυπηρέτηση.

**Technical Priority (3/5):** Η απαίτηση είναι περίπου το ίδιο σημαντική για το σύστημα ειδικά για μεγάλο αριθμό πελατών.

**Stability(5/5):** Η απαίτηση αυτή δεν ενδέχεται να αλλάξει.

### 2.2 Απαιτήσεις Χρηστικότητα (Usability)

#### <ΜΛΑ-3>

Το σύστημα πρέπει να υποστηρίζει τη χρήση συγκεκριμένων σχεδιαστικών προγραμμάτων.

**Περιγραφή :** Το σύστημα πρέπει να υποστηρίζει τη χρήση συγκεκριμένων σχεδιαστικών προγραμμάτων , όπως Solidworks, 3Ds Max,blender κ.α.

**User Priority (5/5):** Είναι πολύ σημαντική για το χρήστη απαίτηση αφού για το χρήστη καθώς περιγράφει την βασική υπηρεσία που του παρέχει το σύστημα.

**Technical Priority (5/5):** Η απαίτηση είναι εξίσου σημαντική για το σύστημα.

**Stability(5/5) :** Η απαίτηση αυτή δεν ενδέχεται να αλλάξει.

## 2.3 Τεχνικές Απαιτήσεις περιβάλλοντος

### <ΜΛΑ-1>

Το σύστημα θα πρέπει να υποστηρίζει τις γλώσσες : Ελληνικά, Αγγλικά, Ρωσικά.

**Περιγραφή:** Το σύστημα θα πρέπει να δίνει στο χρήστη την δυνατότητα να επιλέξει τη γλώσσα που θα εμφανίζεται στο σύστημα.

**User Priority (5/5):** Είναι απαραίτητο να υπάρχει η γλώσσα με την οποία είναι πιο εξοικειωμένος ο χρήστης για να μπορεί να χειριστεί το σύστημα.

**Technical Priority (3/5):** Η απαίτηση είναι αρκετά σημαντική για να καταστήσει λειτουργικό το σύστημα.

**Stability(3/5) :** Μπορεί να προστεθούν επιπλέον γλώσσες με την πάροδο του χρόνου.

### <ΜΛΑ-4>

Το σύστημα πρέπει να παρέχει σε κάθε χρήστη προσωπική μνήμη 3Gb με δυνατότητα επέκτασης.

**Περιγραφή :** Το σύστημα πρέπει να παρέχει σε κάθε χρήστη προσωπική μνήμη 3Gb ,για να αποθηκεύει τα προσωπικά του αρχεία ή αρχεία του project το οποίο έχει δημιουργήσει ,και θα πρέπει να έχει την δυνατότητα να αγοράσει περισσότερη .

**User Priority (4/5):** Η συγκεκριμένη απαίτηση είναι σημαντική αφού δίνει τη δυνατότητα στον κάθε χρήστη να αποθηκεύσει σημαντική ποσότητα αρχείων

**Technical Priority (4/5):** Η απαίτηση είναι εξίσου σημαντική για το σύστημα.

**Stability(4/5):** Ενδέχεται σε βάθος χρόνου να αυξηθεί αυτό το μέγεθος μνήμης.

## 2.4 Απαιτήσεις πολιτικής (Policy - legal, corporate)

### <ΜΛΑ-5>

Ο χρήστης θα μπορεί να πραγματοποιήσει πληρωμές μέσω PayPal, πιστωτικής κάρτας ή Stripe.

**Περιγραφή :** Ο χρήστης θα μπορεί να πραγματοποιεί πληρωμές μέσω των παραπάνω για να αποκτήσει δικαίωμα χρήσης κάποιου προγράμματος ή για να επεκτείνει τη μνήμη του.

**User Priority (5/5):** Είναι σημαντική αίτηση για το χρήστη καθώς είναι απαραίτητη για την παροχή υπηρεσιών.

**Technical Priority (4/5):** Η απαίτηση είναι εξίσου σημαντική για το σύστημα.

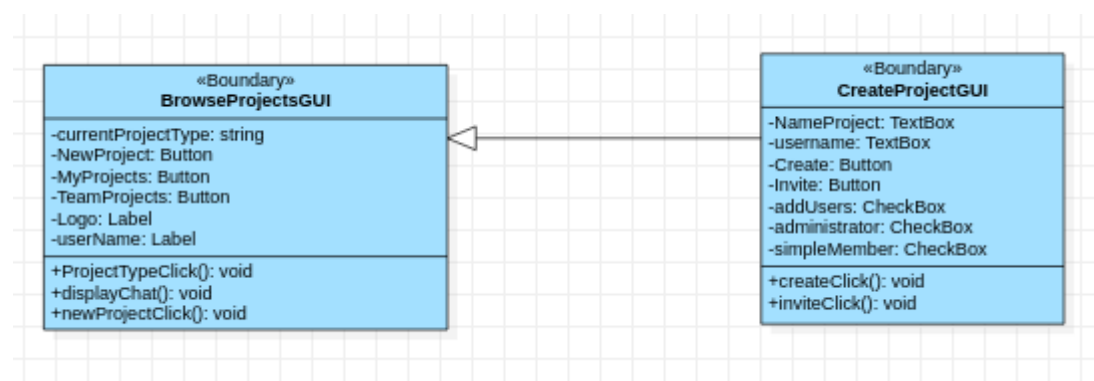
## 3 Πρότυπα Σχεδιασμού που υιοθετήθηκαν

### 3.1 Πρότυπα Συμπεριφοράς

#### Template Design Pattern

Το συγκεκριμένο πρότυπο παρέχει μια ομαδοποίηση πολλών κλάσεων που μοιράζονται τον ίδιο αλγόριθμο, αλλά διαφέρουν στις λεπτομέρειες. Έτσι τα κοινά βήματα συγκεντρώνονται σε μια κλάση ώστε οι υπόλοιπες να μπορούν να τα κληρονομούν και να διαφέρουν μόνο στις επιμέρους λεπτομέρειες. Αυτή η ομαδοποίηση βοηθάει στην εύκολη επέκταση του συστήματος.

Ένα πρόβλημα που αντιμετωπίστηκε με τη χρήση του συγκεκριμένου προτύπου ήταν ότι στην διάρκεια περιήγησης στα project ενός χρήστη στην πλατφόρμα η διεπαφή παρουσιάζει αρκετά κοινά χαρακτηριστικά. Με την χρήση του προτύπου αυτού ομαδοποιήθηκαν τα κοινά βήματα σε μια κλάση ώστε να μην επαναλαμβάνονται και περιπλέκουν την διεπαφή.



### 3.2 Δομικά πρότυπα

Τα δομικά πρότυπα εμπεριέχουν σύνθετες δομές, εισάγουν μια abstract κλάση για μελλοντικές επεκτάσεις του συστήματος και επιτυγχάνουν τη μείωση της σύζευξης ανάμεσα σε κλάσεις.

#### Facade Design Pattern

Το συγκεκριμένο πρότυπο παρέχει μια ενοποιημένη διεπαφή σε ένα σετ από αντικείμενα σε ένα υποσύστημα, ορίζοντας μια υψηλού επιπέδου διεπαφή που κάνει το υποσύστημα πιο εύκολο στη χρήση και μας παρέχει κλειστή αρχιτεκτονική.

Ένα πρόβλημα που αντιμετωπίστηκε με τη χρήση του συγκεκριμένου προτύπου ήταν



η επεξεργασία ή η δημιουργία ενός αρχείου στην πλατφόρμα με κάποιο από τα προγράμματα που υποστηρίζονται. Συγκεκριμένα τα προγράμματα αυτά θεωρήθη-

καν εξωτερικό υποσύστημα και έγινε χρήση αδιαφανούς αρχιτεκτονικής. Έτσι στην ουσία χρειάστηκε να υλοποιήσουμε έναν οδηγό που να πραγματοποιεί αυτή την σύνδεση.

### Διάγραμμα κλάσεων

