

---

## ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ

---

---

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ ΓΙΑ  
ΤΟ ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2022-2023

---

---

ΟΜΑΔΑ 4970-4993

ΙΩΑΝΝΗΣ ΤΣΟΧΛΑΣ, ΑΜ:4993

ΙΩΑΝΝΗΣ ΓΙΑΝΝΑΚΟΣ, ΑΜ:4970

---

ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ

ΔΕΚΕΜΒΡΙΟΣ 2022

## ΙΣΤΟΡΙΚΟ ΕΚΔΟΣΕΩΝ ΤΗΣ ΠΑΡΟΥΣΑΣ ΑΝΑΦΟΡΑΣ

Ημερομηνία	Έκδοση	Περιγραφή	Συγγραφείς
2022/11/10	v.01	Οργάνωση απαιτήσεων σε use cases, complete oreos, uml diagramms	4993,4970
2022/11/20	v.01.5	Διορθώσεις στις uses cases, προσθήκη OtherComments ώστε να συσχετιστεί με τον κώδικα	4993,4970
2022/11/26	v.02	Αρχική σχεδίαση κλάσεων και ελέγχων,	4993,4970
2022/12/05	v.03	Διορθώσεις στις uses cases, επεκτάσεις στη σχεδίαση κλάσεων και ελέγχων	4993,4970
2022/12/08	v.04	Υλοποίηση Test Case	4993,4970
2022/12/13	v.05	ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ	4993,4970

## 1 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ – USE CASES

---

Στην παρούσα ενότητα, παρατίθενται οι περιγραφές των use cases με βάση τις καταγεγραμμένες απαιτήσεις.

### ΦΟΡΤΩΣΕ

---

#### ID: UC 1

#### DESCRIPTION AND GOAL

Η use case «Φόρτωση» εμφανίζει ταξινομημένο το αρχείο κειμένου που έχουμε επιλέξει .

#### ACTORS

Ο Χρήστης

#### BASIC FLOW

1. Το use case ξεκινάει όταν ο χρήστης επιλέξει από το μενού File την επιλογή «Load TSV».
2. Το σύστημα εμφανίζει ταξινομημένο το αρχείο στην οθόνη.

#### EXTENSIONS / VARIATIONS

Στην περίπτωση κατά την οποία δεν υπάρχει το αρχείο εμφανίζεται ένα μήνυμα που ενημερώνει ότι δεν υπάρχει ή δεν μπορεί να ανοιχτεί .

#### POST CONDITIONS

Ο χρήστης παραλαμβάνει φορτωμένο και ταξινομημένο το αρχείο

#### OTHER COMMENTS

Σε αντιστοιχία με το project αντιπροσωπεύει το πακέτο filemanagment και κατά συνέπεια τις κλάσεις FileOpen.java η οποία είναι υπεύθυνη για να ανοίξει ένα αρχείο tsv από το σύστημα ώστε να μπορεί να το χειριστεί η εφαρμογή μας και την κλάση Classification.java η οποία επιστρέφει ταξινομημένο το αρχείο (διάγραμμα gant )

# ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ

---

ID: UC 2

## DESCRIPTION AND GOAL

Η use case «ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ» εμφανίζει ταξινομημένες, μόνο τις εργασίες που η περιγραφή τους ξεκινά (prefix) με ένα δοθέν string .

## ACTORS

Ο Χρήστης

## PRECONDITIONS

Να έχει φορτωθεί ταξινομημένο το αρχείο

## BASIC FLOW

1. Το use case ξεκινάει όταν ο χρήστης επιλέξει από το μενού Filters την επιλογή «Filter by Prefix».
2. Στο παράθυρο αναζήτησης ο χρήστης πληκτρολογεί το επιθυμητό string .
3. Το σύστημα εμφανίζει τα αποτελέσματα στην οθόνη.

## POST CONDITIONS

Ανάκτηση όλων των εργασιών που πληρούν το ίδιο κριτήριο περιγραφής

## OTHER COMMENTS

Σε αντιστοιχία με το project αντιπροσωπεύει την κλάση TasksByPrefix.java του πακέτου searchmethods

# ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID

---

## ID: UC 3

### DESCRIPTION AND GOAL

Η use case «ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID» εμφανίζει την εργασία που έχει το επιθυμητό Id

### ACTORS

Ο Χρήστης

### PRECONDITIONS

Να έχει φορτωθεί ταξινομημένο το αρχείο

### BASIC FLOW

1. Το use case ξεκινάει όταν ο χρήστης επιλέξει από το μενού Filters την επιλογή «Filter by ID».
2. Στο παράθυρο αναζήτησης ο χρήστης πληκτρολογεί το επιθυμητό taskID .
3. Το σύστημα εμφανίζει το αποτέλεσμα στην οθόνη.

### POST CONDITIONS

Ανάκτηση εργασίας με συγκεκριμένο taskID

### OTHER COMMENTS

Σε αντιστοιχία με το project αντιπροσωπεύει την κλάση TaskById.java του πακέτου searchmethods

# ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ

---

## ID: UC 4

### DESCRIPTION AND GOAL

Η use case «ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ» εμφανίζει τις εργασίες κορυφαίου επιπέδου(MamaID=0).

### ACTORS

Ο Χρήστης

### PRECONDITIONS

Να έχει φορτωθεί ταξινομημένο το αρχείο

### BASIC FLOW

1. Το use case ξεκινάει όταν ο χρήστης επιλέξει από το μενού Filters την επιλογή «Top Level Tasks».
2. Το σύστημα εμφανίζει το αποτέλεσμα στην οθόνη.

### POST CONDITIONS

Ανάκτηση των εργασιών κορυφαίου επιπέδου

### OTHER COMMENTS

Σε αντιστοιχία με το project αντιπροσωπεύει την κλάση TopLevelTasks.java του πακέτου searchmethods

# ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ

ID: UC 5

## DESCRIPTION AND GOAL

Η use case «ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ» εξάγει τον τύπο αρχείου που έχουμε επιλέξει

## ACTORS

Ο Χρήστης

## PRECONDITIONS

Να έχει φορτωθεί ταξινομημένο το αρχείο

## BASIC FLOW

1. Το use case ξεκινάει όταν ο χρήστης επιλέξει το μενού Reports
2. Το σύστημα αποθηκεύει το αρχείο στον υπολογιστή.

## EXTENSIONS/VARIATIONS

1α. Όταν ο χρήστης επιλέξει την μορφή "Report Txt" η αναφορά αποθηκεύεται στον υπολογιστή σε μορφή κειμένου

1α. Όταν ο χρήστης επιλέξει την μορφή "Report Md" η αναφορά αποθηκεύεται στον υπολογιστή σε μορφή markdown

1α. Όταν ο χρήστης επιλέξει την μορφή "Report Html" η αναφορά αποθηκεύεται στον υπολογιστή σε μορφή Html (Hypertext Markup Language)

## POST CONDITIONS

Αποθήκευση σε απλό κείμενο, html και markdown

## OTHER COMMENTS

Σε αντιστοιχία με το project αντιπροσωπεύει το πακέτο report και κατά αντιστοιχία τις ακόλουθες κλάσεις TextReport.java , MarkdownReport.java , HtmlReport.java επιπρόσθετα να σημειωθεί ότι υπάρχει μια βοηθητική κλάση FilterReport.java στο πακέτο η οποία είναι υπεύθυνη για να ενεργοποιήσει την κατάλληλη κλάση (βάση το αρχείο που θέλουμε να εξάγουμε (.txt,.md,.html))

## 2 ΣΧΕΔΙΑΣΗ ΕΛΕΓΧΩΝ

Οι έλεγχοι που σχεδιάσθηκαν και εντάχθηκαν στην υλοποίηση περιγράφονται παρακάτω.

### 2.1 ΕΛΕΓΧΟΣ USE CASES VIA SYSTEM TESTS

#### 2.1.1 USE CASE UC1: ΦΟΡΤΩΣΕ (FILEMANAGEMENTTEST)

##### Test cases

Description	ON	any context
	RECEIVING	Request to load a specific txt file
	ENSURE	That the System
	OUTPUTS	Stored data from the file as sorted (Tasks) objects in arraylist
	SUCH THAT	state is intact

ID	T1_V0_01	HappyDayScenario for loadfile() //load a file WITHOUT BEEING SORTED
Pre-cond.		No specific precond constructed
Input		Input a EggsScrambled.tsv, a small valid file
Output		The total lines form the file that we have just open
Post-cond.		No state properties tested
Method To test		fileOpenToTest.loadfile();

ID	T1_V0_02	HappyDayScenario for load() //load and sort the file
		...identical setup with T1_V0_01
Method To test		testClassification.load("path","delimiter")

ID	T1_V0_03	Test if the file is not null testfileOpenNull()
		...identical setup with T1_V0_01
Method To test		testfileOpenNull()

ID	T1_V0_04	test if the constructor prevents creation with null delimiter testfileOpenNoDelimiter()
		...identical setup with T1_V0_01
Method To test		testfileOpenNoDelimiter()

##### Involved methods

FileOpen()

**Not designed yet:** T1\_V0\_04 That test fails so we must improve FileOpen to prevent creation with null delimiter



---

### 2.1.2 USE CASE UC2: ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ (TASKBYPREFIXTEST)

---

#### Test cases

<i>Description</i>	<i>ON</i>	<i>A loaded file</i>
	<i>RECEIVING</i>	<i>Request to return tasks by prefix</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>a set of tasks that match the prefix</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T2_V0_01	HappyDayScenario for getByPrefixTest()
Pre-cond.		Load the sorted file
Input		The prefix
Output		A SimpleTableModel with the matching results
Method To test		appController.getTasksByPrefix(Prefix)

#### Involved methods

MainControllerFactory()

---

### 2.1.3 USE CASE UC3: ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID (TASKBYIDTEST)

---

#### Test cases

<i>Description</i>	<i>ON</i>	<i>A loaded file</i>
	<i>RECEIVING</i>	<i>Request to return tasks by id</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>The task that match the id</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T3_V0_01	HappyDayScenario for getIdTest()
Pre-cond.		Load the sorted file
Input		The id
Output		A SimpleTableModel with the matching results
Method To test		appController.getId(Id);

#### Involved methods

MainControllerFactory()

---

#### 2.1.4 USE CASE UC4: ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ (TOPLEVELTASKSTEST)

---

##### Test cases

<i>Description</i>	<i>ON</i>	<i>A loaded file</i>
	<i>RECEIVING</i>	<i>Request to return top level tasks</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>a set of top level tasks</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T4_V0_01	HappyDayScenario for getTopLevelTaskTest()
Pre-cond.		Load the sorted file
Output		A SimpleTableModel with the matching results
Method To test		appController.getTopLevelTasks()

##### Involved methods

MainControllerFactory()

---

2.1.5 USE CASE UC5: ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ (REPORTSTEST)

---

**Test cases**

<i>Description</i>	<i>ON</i>	<i>A loaded file</i>
	<i>RECEIVING</i>	<i>Request to extract a report</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A report in selected format</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T4_V0_01	HappyDayScenario for createReportTextTest()
Pre-cond.		Load the sorted file
Output		A text file
Method To test		appController.createReportText()

ID	T4_V0_02	HappyDayScenario for createReportHtmlTest()
Pre-cond.		Load the sorted file
Output		A html file
Method To test		appController.createReportHtml()

ID	T4_V0_03	HappyDayScenario for createReportMdTest()
Pre-cond.		Load the sorted file
Output		A markdown file
Method To test		appController.createReportMd()

## 2.2 TRACEABILITY MATRIX

Η αντιστοίχιση use cases σε id's φαίνεται στον Πίνακα 1:

UC1	ΦΟΡΤΩΣΕ
UC2	ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ
UC3	ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID
UC4	ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ
UC5	ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ

Πίνακας 1 Σύνοψη use cases και των id's τους

Ο Πίνακας 2 είναι ο traceability matrix για τους ελέγχους μας.

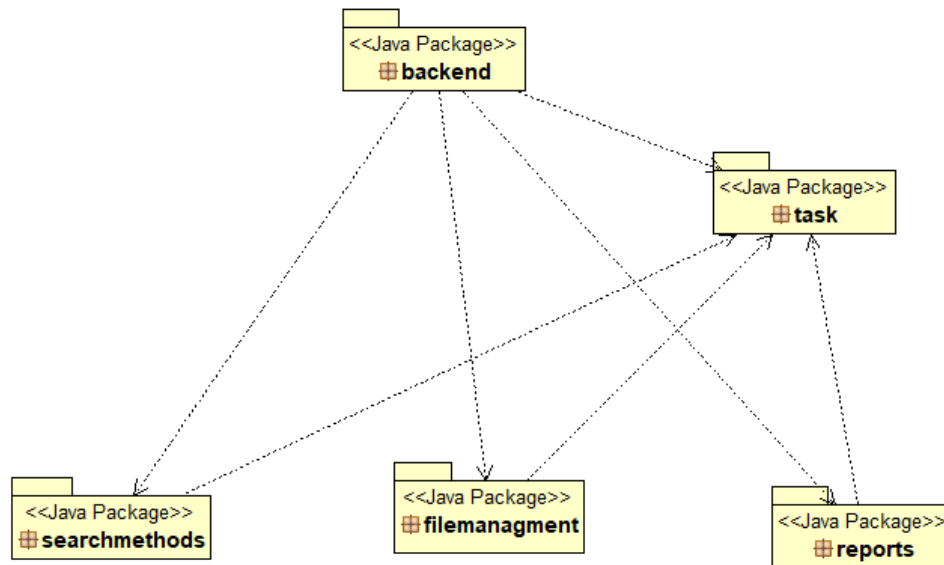
	UC1	UC2	UC3	UC4	UC5
T1_V0_01	X				
T1_V0_02	X				
T1_V0_03	X				
T1_V0_04	X				
T2_V0		X			
T3_V0			X		
T4_V0				X	
T5_V0_01					X
T5_V0_02					X
T5_V0_03					X

Πίνακας 2 Traceability matrix between use cases and tests

### 3 ΣΧΕΔΙΑΣΗ ΛΟΓΙΣΜΙΚΟΥ

#### 3.1 ΔΙΑΓΡΑΜΜΑΤΑ ΠΑΚΕΤΩΝ / ΥΠΟΣΥΣΤΗΜΑΤΩΝ

Το διάγραμμα των πακέτων του συστήματος ακολουθεί στο Σχ. 1.



Σχήμα 1. Διάγραμμα πακέτων

Ακολουθεί μια συνοπτική περιγραφή των λειτουργικών πακέτων του συστήματος.

(τα πακέτα που είναι υπεύθυνα για τα γραφικά δεν παρουσιάζονται εδώ)

#### ΠΑΚΕΤΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

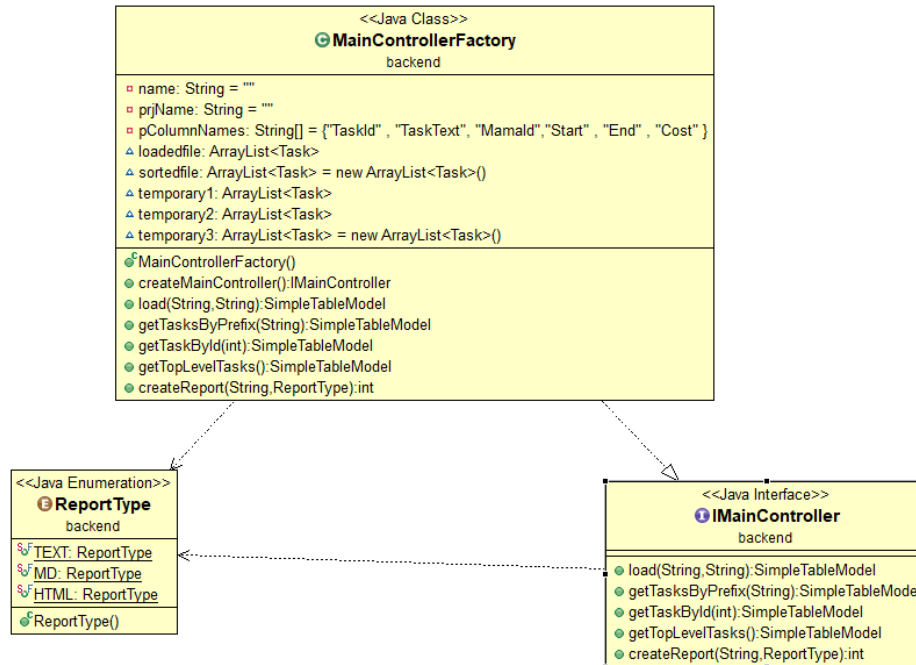
task	Περιέχει data base που είναι υπεύθυνη για την αλληλεπίδραση με το σύστημα
backend	Κεντρική business logic engine, along with the necessary interface to export to the boundary classes
searchmethods	Υποσύστημα αλληλεπίδρασης με τα αρχεία δεδομένων, για την ανάκτησή τους από το σύστημα
reports	Υποσύστημα παραγωγής αναφορών
filemanagment	Υποσύστημα αλληλεπίδρασης με τα αρχεία εισαγωγής, για την αποθήκευση τους από το σύστημα

Πίνακας 3. Συνοπτική περιγραφή πακέτων συστήματος

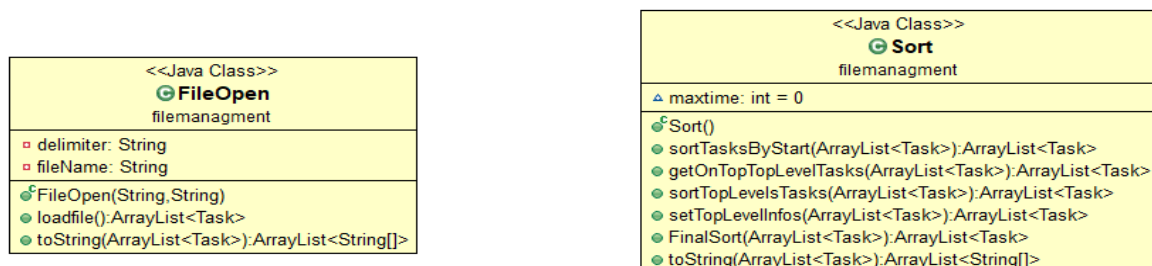
### 3.2 ΔΙΑΓΡΑΜΜΑΤΑ ΚΛΑΣΕΩΝ

Στην παρούσα υποενότητα, παρατίθενται τα διαγράμματα κλάσεων και ακολουθιών.

**Package** backend;



**Package** filemanagment;



**package** searchmethods;

<<Java Class>> <b>TaskById</b> searchmethods
▢ Id: int
⚙️ TaskById(int) ● getTasksById(ArrayList<Task>):ArrayList<Task> ● toString(ArrayList<Task>):ArrayList<String[]>

<<Java Class>> <b>TasksByPrefix</b> searchmethods
▢ prefix: String
⚙️ TasksByPrefix(String) ⚙️ convertStringToCharList(String):ArrayList<Character> ● getTasksByPrefix(ArrayList<Task>):ArrayList<Task> ● toString(ArrayList<Task>):ArrayList<String[]>

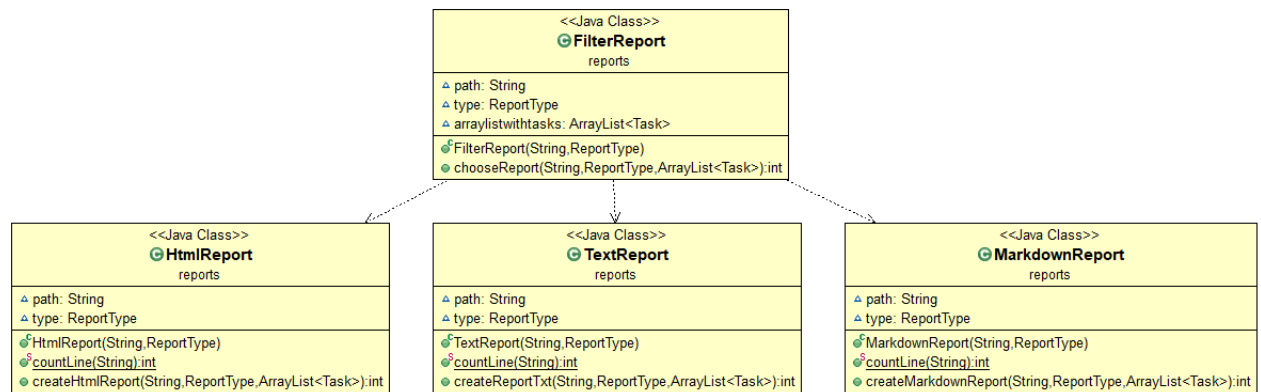
<<Java Class>> <b>TopLevelTasks</b> searchmethods
⚙️ TopLevelTasks() ● getTopLevelTasks(ArrayList<Task>):ArrayList<Task> ● toString(ArrayList<Task>):ArrayList<String[]>

**Package** tasks;

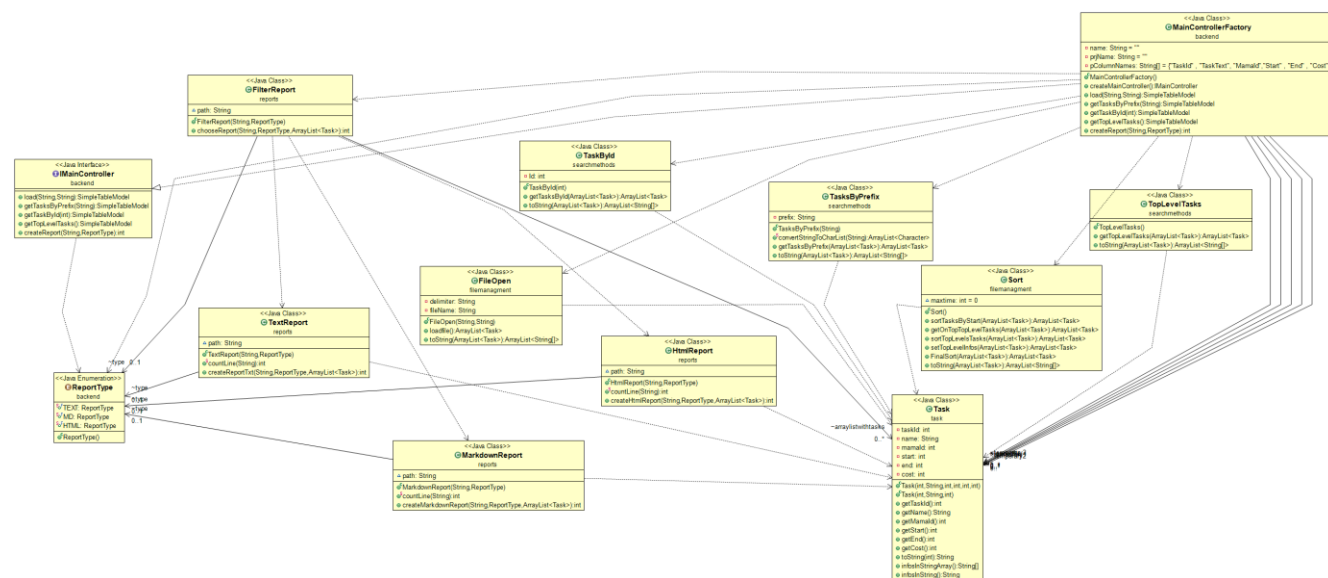
<<Java Class>> <b>Task</b> task
▢ taskId: int ▢ name: String ▢ mamald: int ▢ start: int ▢ end: int ▢ cost: int
⚙️ Task(int,String,int,int,int,int) ⚙️ Task(int,String,int) ● getTaskId():int ● getName():String ● getMamald():int ● getStart():int ● getEnd():int ● getCost():int ● toString(int):String ● infosInStringArray():String[] ● infosInString():String



## Package reports;



## OVERALL DIAGRAM



### 3.3 ΑΝΑΛΥΣΗ ΚΛΑΣΕΩΝ ΚΑΙ ΣΥΝΕΠΕΙΑ ΠΡΟΣ ΤΙΣ ΑΠΑΙΤΗΣΕΙΣ

Στην παρούσα ενότητα παραθέτουμε μια ανάλυση των κλάσεων και μια τεκμηρίωση της κάλυψης των βασικών use cases του συστήματος.

**Πρέπει ΥΠΟΧΡΕΩΤΙΚΑ να μου εξηγήσετε:**

**(α) Την ταξινόμηση των κλάσεων σε Domain/Business Logic/Boundary classes**

**(β) Τα interfaces between subsystems (emph., for Business Logic classes)**

#### 3.3.1 DOMAIN CLASSES

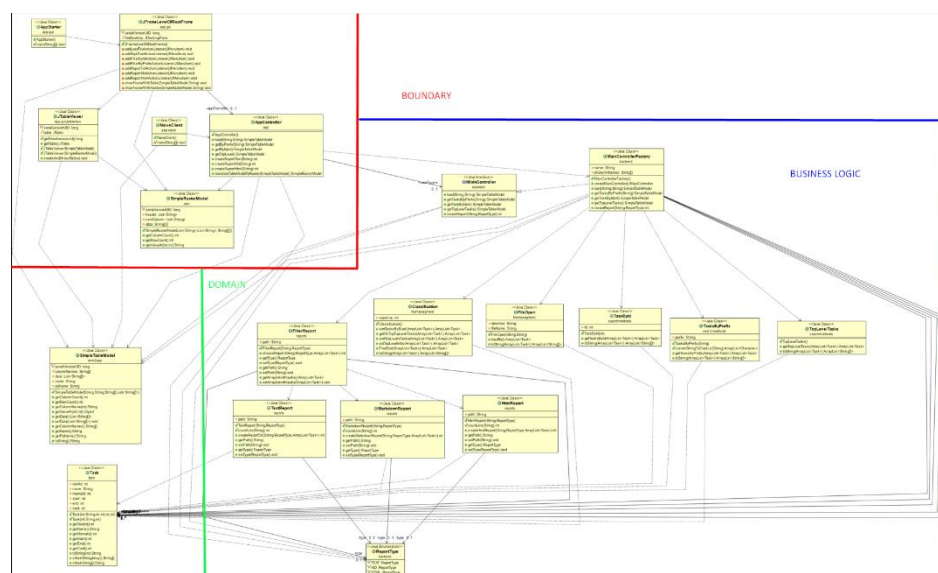
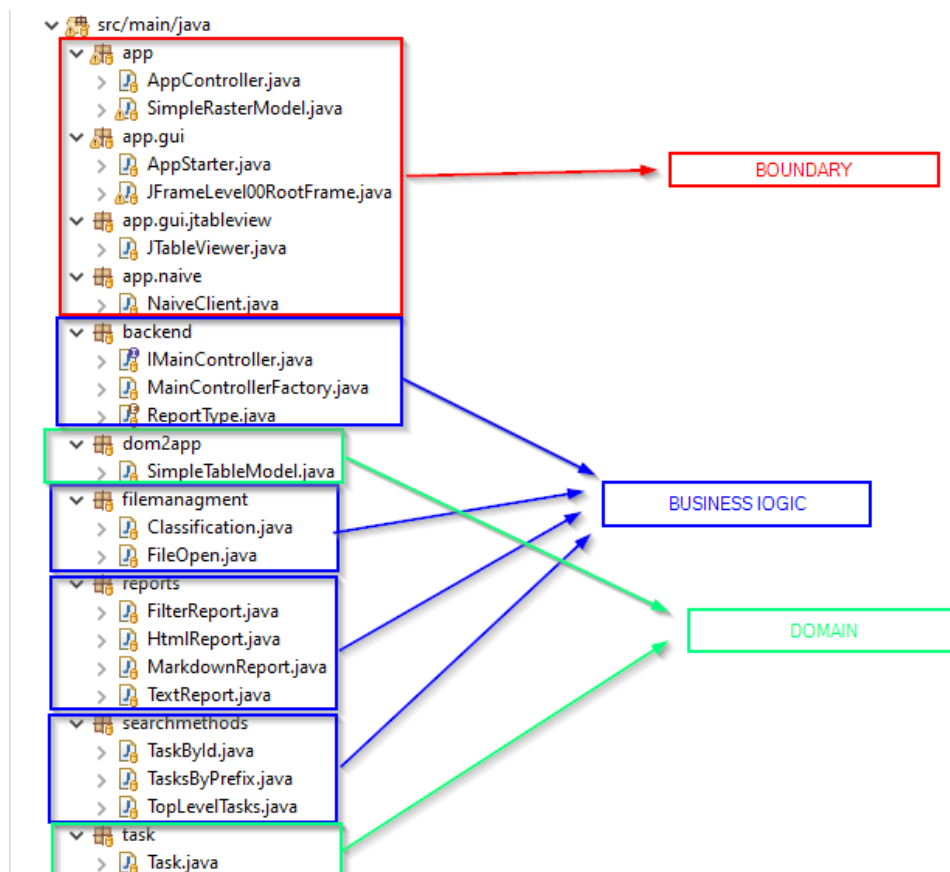
Package bookstoreAdvanced	CD, Book, μια αφηρημένη κλάση Item (A) για αυτές τα δύο, και ένα factory, το ItemFactory, για την κατασκευή στιγμιοτύπων. ShoppingCart.
Package task	Task με τα πεδία του
Package dom2app	SimpleTableModel extends AbstractTableModel και είναι η δομή υπεύθυνη για να απεικονίσει τα tasks σε γραφικά

#### 3.3.2 BUSINESS LOGIC CLASSES

Package backend	MainControllerFactory, IMainControllerFactory για την υλοποίηση όλων των use cases στο back-end.
Package filemanagment	Κλάσεις υπεύθυνες για την ταξινόμηση και τη φόρτωση των αρχείων
Package reports	Κλάσεις υπεύθυνες για τη δημιουργία reports
Package searchmethods	Κλάσεις υπεύθυνες για την αναζήτηση συγκεκριμένων tasks

#### 3.3.3 BOUNDARY CLASSES

Package app.naive	NaiveClient, μια client class για την αλληλεπίδραση με το χρήστη μέσω κονσόλας.
Package app.gui.jtableView	Πακέτο γραφικών Το πιο δουλειά είναι να αναπαραστήσει γραφικά J tables
Package App.gui	Πακέτο γραφικών
Package app	Πακέτο γραφικών



### 3.3.4 ΑΠΕΙΚΟΝΙΣΗ ΑΠΑΙΤΗΣΕΩΝ ΣΕ ΜΕΘΟΔΟΥΣ

Υπάρχουν 5 use cases για την απεικόνιση των διαγραμμάτων Gantt.

- ΦΟΡΤΩΣΕ
- ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ
- ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID
- ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ
- ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ

#### ΑΠΕΙΚΟΝΙΣΗ USE CASES ΣΕ ΜΕΘΟΔΟΥΣ

Use case	Back-end methods
• ΦΟΡΤΩΣΕ	<code>appController.load()</code>
• ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΠΟΥ ΠΛΗΡΟΥΝ ΤΟ ΚΡΙΤΗΡΙΟ ΑΝΑΖΗΤΗΣΗΣ	<code>appController.getByPrefix(Prefix)</code>
• ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΑ ΒΑΣΗ ID	<code>appController.getById(Id);</code>
• ΕΠΕΣΤΡΕΨΕ ΕΡΓΑΣΙΕΣ ΚΟΡΥΦΑΙΟΥ ΕΠΙΠΕΔΟΥ	<code>appController.getTopLevel();</code>
• ΔΗΜΙΟΥΡΓΗΣΕ ΑΝΑΦΟΡΑ	<code>appController.createReportText</code> <code>appController.createReportMd</code> <code>appController.createReportHtml</code>

Πίνακας 4 Επαλήθευση απεικόνισης use cases σε μεθόδους

## 4 ΛΟΙΠΑ ΣΧΟΛΙΑ

---

### 4.1 ΣΧΕΔΙΑΣΤΙΚΕΣ ΑΠΟΦΑΣΕΙΣ

---

Αποφασίσαμε να βάλουμε σε σχόλια ένα test (το οποίο υπάρχει μέσα στη FileManagement) σε σχόλια, διότι καλεί το `system.exit(0)` και αυτό έχει ως αποτέλεσμα να βγει από το JVM και κατά συνέπεια δε μας αφήνει να τρέξουμε τα υπόλοιπα test.

Ενδεικτική λύση θα ήταν να χρησιμοποιούσαμε τη βιβλιοθήκη `system lamda` ή `system rules`, αλλά χρειάζονται τουλάχιστον JUnit 5 και 4.9 αντίστοιχα. Αλλά αυτό που έχουμε είναι 4.14.

Όταν αποθηκεύουμε τα reports πρέπει να βάλουμε χειροκίνητα τα extensions (.html,.txt,.md)

### 4.2 ΣΗΜΕΙΑ ΚΙΝΔΥΝΟΥ

---

1. Αν στο searchbyid εισάγεις αριθμό κρασαρει το πρόγραμμα