

ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ ΓΙΑ ΤΟ ΑΚΑΔΗΜΑΪΚΟ
ΈΤΟΣ 2023-2024

ΟΜΑΔΑ 4937-5108-5169

ΑΛΕΞΑΝΔΡΟΣ ΚΩΣΤΟΠΟΥΛΟΣ, ΑΜ:4937

ΝΙΚΟΛΑΟΣ ΚΟΥΤΣΟΝΙΚΟΛΗΣ, ΑΜ:5108

ΓΕΩΡΓΙΟΣ ΝΤΖΕΛΒΕΣ, ΑΜ:5169

ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ

ΔΕΚΕΜΒΡΙΟΣ 2023

Ημερομηνία	Έκδοση	Περιγραφή	Συγγραφείς
2023/12/15	v.01	Οργάνωση απαιτήσεων σε use cases	4937,5108,5169
yyyy/mm/dd	v.02	Αρχική σχεδίαση κλάσεων και ελέγχων	4937,5108,5169
yyyy/mm/dd	...	Διορθώσεις στις uses cases, επεκτάσεις στη σχεδίαση κλάσεων και ελέγχων	4937,5108,5169
yyyy/mm/dd	...	ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ	4937,5108,5169

Στην τρέχουσα σελίδα, με γκρι, αχνά γράμματα παραθέτω οδηγίες, επεξηγήσεις και σχόλια. Στη δική σας αναφορά, αυτό το κομμάτι προφανώς θα πρέπει να το σβήσετε.

Στο υπόλοιπο κείμενο, για να μη περιπλέξω το στυλ περαιτέρω, οι οδηγίες παρατίθενται σε απλό Βασικό/Normal στυλ. Εσείς εκεί πρέπει να βάλετε το δικό σας κείμενο στη θέση του δικού μου.

ΟΔΗΓΙΕΣ

Αναμένεται να τηρήσετε το παρόν πρότυπο ΠΙΣΤΑ. Όχι επειδή είναι το απαύγασμα της καλαισθησίας ή της λειτουργικότητας, αλλά επειδή πρέπει να μάθετε να τηρείτε πρότυπα με πειθαρχία και συνέπεια. Για το λόγο αυτό, καλείσθε να ΜΕΙΝΕΤΕ ΠΙΣΤΑ ΣΤΟ ΣΤΥΛ ΤΟΥ ΚΕΙΜΕΝΟΥ! Είναι ευκαιρία επίσης, να μάθετε να χρησιμοποιείτε με ΕΥΧΕΡΕΙΑ επεξ. κειμένου.

Κατά τα λοιπά:

- Συμπληρώστε με τα σωστά στοιχεία το εξώφυλλο και το header, όπου υπάρχουν κόκκινα γράμματα. Αν δεν σας έχει δοθεί α/α ομάδα, είναι το concatenation των ΑΜ σας, με παύλες ανάμεσα, ταξινομημένα με αύξουσα σειρά.
- Ο παραπάνω πίνακας «Ιστορικό Εκδόσεων» συμπληρώνεται κάθε φορά που αλλάζετε κάτι στην αναφορά σας. Οι καταχωρήσεις που υπάρχουν ήδη είναι απλά ενδεικτικές.
- Στις Use Cases συμπληρώνετε ΟΛΕΣ τις Use Cases που έχετε εξάγει (εδώ παρατίθεται μόνο μία, ενδεικτικά). Το στυλ είναι από το υπόδειγμα για use cases που χρησιμοποιείται στο μάθημα.
- Δώστε ιδιαίτερη σημασία στους ελέγχους, καθώς είναι από τις λίγες φορές στην εκπαίδευσή σας που εξετάζεστε για την απάντηση που δώσατε στο ερώτημα «Πώς θα επιβεβαιώσω ότι ο κώδικάς μου κάνει αυτό που πρέπει?» (θυμηθείτε ότι στις συνεντεύξεις για δουλειά, ερωτήσεις επί του testing είναι από τις πιο κλασικές επιλογές των interviewers).
- Συμπληρώστε τα διαγράμματα με τις σχετικές εικόνες. Προσοχή: τα σχήματα πρέπει να είναι ΕΥΔΙΑΚΡΙΤΑ. Ο σκοπός των διαγραμμάτων είναι να μπορούν να μεταδώσουν ΣΤΟΥΣ ΑΛΛΟΥΣ τις σχεδιαστικές σας αποφάσεις.

• ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ – USE CASES

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

ΦΟΡΤΩΣΕ ΑΠΟ ΑΠΛΟΚΕΙΜΕΝΟ

ID: UC 1

DESCRIPTION AND GOAL

Η use case «ΦΟΡΤΩΣΕ ΑΠΟ ΑΠΛΟΚΕΙΜΕΝΟ» φορτώνει ένα αρχείο κειμένου.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Το αρχείο δεδομένων θα είναι ένα απλό delimited αρχείο κειμένου.

BASIC FLOW

- Το use case ξεκινάει όταν ο αναλυτής ζητήσει από το σύστημα να φορτώσει ένα αρχείο κειμένου.
- Το σύστημα θα ανοίξει και θα διαβάσει το αρχείο κειμένου.
- Το σύστημα θα δημιουργήσει objects για κάθε γραμμή του αρχείου.
- Το σύστημα θα φτιάξει μία συλλογή και θα αποθηκεύσει εκεί τα objects.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

ΑΝΑΚΤΗΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ

ID: UC 2

DESCRIPTION AND GOAL

Η use case «ΑΝΑΚΤΗΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ» επιστρέφει τα δεδομένα που ζητάει ο αναλυτής με βάση το αίτημα του.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει να έχει φορτωθεί το αρχείο που περιέχει τα δεδομένα.

BASIC FLOW

- Το use case ξεκινάει όταν ο αναλυτής ζητήσει από το σύστημα να του δώσει έναν συνδυασμό <χώρα, τύπο καταστροφής>.
- Ο αναλυτής θέτει το αίτημα με ένα όνομα στη μορφή "CountryName-DisasterType"
- Το σύστημα επιλέγει τα κατάλληλα δεδομένα από τη συλλογή.
- Το σύστημα επιστρέφει τα δεδομένα αυτά στον αναλυτή.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

Τα αιτήματα που κάνει ο αναλυτής κρατούνται από το σύστημα για μετέπειτα χρήση

ΠΑΡΟΥΣΙΑΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ

ID: UC 3

DESCRIPTION AND GOAL

Η use case «ΠΑΡΟΥΣΙΑΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ» εμφανίζει τα δεδομένα στον αναλυτή.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει να έχει φορτωθεί το αρχείο που περιέχει τα δεδομένα.

BASIC FLOW

- Όμοια με UC 2.
- Το σύστημα εμφανίζει τα δεδομένα στη κονσόλα.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

ΑΝΑΚΤΗΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ ΑΡΧΙΚΟ Τ ΕΛΙΚΟ ΕΤΟΣ

ID: UC 4

DESCRIPTION AND GOAL

Η use case «ΑΝΑΚΤΗΣΕ ΣΤΟΙΧΕΙΑ ΧΩΡΑΣ ΤΥΠΟΥ ΚΑΤΑΣΤΡΟΦΗΣ ΑΡΧΙΚΟ ΤΕΛΙΚΟ ΕΤΟΣ» επιστρέφει τα δεδομένα που ζητάει ο αναλυτής με βάση το αίτημα του.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει να έχει φορτωθεί το αρχείο που περιέχει τα δεδομένα.

BASIC FLOW

- Το use case ξεκινάει όταν ο αναλυτής ζητήσει από το σύστημα να του δώσει έναν συνδυασμό <χώρα, τύπο καταστροφής, αρχικό έτος, τελικό έτος>.
- Ο αναλυτής θέτει το αίτημα με ένα όνομα στη μορφή "CountryName-DisasterType-StartingYear-EndingYear"
- Το σύστημα ελέγχει το αρχικό έτος να είναι πριν από το τελικό έτος και να υπάρχουν μετρήσεις για αυτά τα έτη.
- Το σύστημα επιλέγει τα κατάλληλα δεδομένα από τη συλλογή.
- Το σύστημα επιστρέφει τα δεδομένα στον αναλυτή.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

Τα αιτήματα που κάνει ο αναλυτής κρατιούνται από το σύστημα για μετέπειτα χρήση.

ΠΑΡΟΥΣΙΑΣΕΣΤΟΙΧΕΙΑΧΩΡΑΣΤΥΠΟΥΚΑΤΑΣΤΡΟΦΗΣΑΡΧΙΚΟΤΕΛΙΚΟΕΤΟΣ

ID: UC 5

DESCRIPTION AND GOAL

Η use case «ΠΑΡΟΥΣΙΑΣΕΣΤΟΙΧΕΙΑΧΩΡΑΣΤΥΠΟΥΚΑΤΑΣΤΡΟΦΗΣΑΡΧΙΚΟΤΕΛΙΚΟΕΤΟΣ» εμφανίζει τα δεδομένα στον αναλυτή.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει να έχει φορτωθεί το αρχείο που περιέχει τα δεδομένα.

BASIC FLOW

- Όμοια με UC 4.
- Το σύστημα εμφανίζει τα δεδομένα στη κονσόλα.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

ΥΠΟΛΟΓΙΣΕ ΒΑΣΙΚΑ ΣΤΑΤΙΣΤΙΚΑ

ID: UC 6

DESCRIPTION AND GOAL

Η use case «Υπολόγισε Βασικά Στατιστικά» εμφανίζει τα δεδομένα στον αναλυτή.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει ο αναλυτής να έχει φτιάξει αιτήματα.

BASIC FLOW

- Το use case ξεκινάει μόλις ο αναλυτής δώσει το όνομα κάποιου αιτήματος.
- Το σύστημα υπολογίζει:
 1. μέγιστη τιμή.
 2. ελάχιστη τιμή.
 3. μέση τιμή.
 4. ενδιάμεση τιμή.
 5. συνολικό πλήθος συμβάντων.
 6. αριθμό χρόνων που είχαν συμβάν.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

ΥΠΟΛΟΓΙΣΜΟΣ REGRESSION

ID: UC 7

DESCRIPTION AND GOAL

Η use case «ΥΠΟΛΟΓΙΣΜΟΣ REGRESSION» υπολογίζει την γραμμική παλινδρόμηση.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Πρέπει να υπάρχει το/τα ανάλογο/γα αίτημα/ματα

BASIC FLOW

- Συνέχεια του **UC 6**.
- Το use case ξεκινάει μόλις ο αναλυτής δώσει το όνομα κάποιου αιτήματος της μορφής <CountryName-DisasterType-StartingYear-EndingYear>.
- Το σύστημα υπολογίζει την γραμμική παλινδρόμηση.
- Το σύστημα εμφανίζει το αποτέλεσμα.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

ΑΠΟΘΗΚΕΥΣΕΣΕΑΡΧΕΙΟΚΕΙΜΕΝΟΥ

ID: UC 8

DESCRIPTION AND GOAL

Η use case «ΑΠΟΘΗΚΕΥΣΕΣΕΑΡΧΕΙΟΚΕΙΜΕΝΟΥ» εξάγει τα αποτελέσματα σε ένα αρχείο κειμένου.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

Να υπάρχουν αιτήματα.

BASIC FLOW

- Το use case ξεκινάει μόλις ο αναλυτής επιλέξει κάποιο από τα αιτήματα για αποθήκευση σε μορφή κειμένου(text file).
- Το σύστημα καταγράφει:
 - α) το όνομα του αιτήματος.
 - β) το όνομα της χώρας.
 - γ) τον τύπο της καταστροφής.
 - δ) τη λίστα των μετρήσεων.
 - ε) τα αποτελέσματα των στατιστικών.
 - στ) το αποτέλεσμα του regression.

EXTENSIONS / VARIATIONS

- ΑΠΟΘΗΚΕΥΣΕΣΕΑΡΧΕΙΟMARKDOWN

BASIC FLOW

- Το use case ξεκινάει μόλις ο αναλυτής επιλέξει κάποιο από τα αιτήματα για αποθήκευση σε μορφή markdown(md file).
- Το σύστημα καταγράφει τα ίδια όπως και στο text file και επιπλέον:
 - α) το όνομα αιτήματος σε **bold**
 - β) την χώρα και τον τύπο καταστροφής σε *italics*
 - γ) τα στοιχεία των μετρήσεων σε πίνακάκι

- ΑΠΟΘΗΚΕΥΣΕΣΕΑΡΧΕΙΟHTML

BASIC FLOW

- Το use case ξεκινάει μόλις ο αναλυτής επιλέξει κάποιο από τα αιτήματα για αποθήκευση σε μορφή html(html file).
- Παρόμοια με το Markdown

POST CONDITIONS

-

ΠΡΑΓΜΑΤΟΠΟΙΗΣΕΞΕΟΔΟΑΠΟΤΟΠΡΟΓΡΑΜΜΑ

ID: UC 9

DESCRIPTION AND GOAL

Η use case «ΠΡΑΓΜΑΤΟΠΟΙΗΣΕΞΕΟΔΟΑΠΟΤΟΠΡΟΓΡΑΜΜΑ» πραγματοποιεί έξοδο από το πρόγραμμα.

ACTORS (ESP. PRIMARY ACTOR)

Ο αναλυτής.

PRECONDITIONS

-

BASIC FLOW

- Το use case ξεκινάει όταν ο αναλυτής ζητήσει να βγει από το πρόγραμμα.
- Το σύστημα ζητάει από τον αναλυτή να επιβεβαιώσει την έξοδο.
 1. Αν ο αναλυτής την επιβεβαιώσει τότε πραγματοποιείται έξοδος από το πρόγραμμα.
 2. Αν δεν την επιβεβαιώσει, δεν γίνεται τίποτα.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

- ΣΧΕΔΙΑΣΗ ΕΛΕΓΧΩΝ

Οι έλεγχοι που σχεδιάστηκαν και εντάχθηκαν στην υλοποίηση περιγράφονται παρακάτω. Εδώ, ως υπόδειγμα: το project με την διάσπαση χρονοσειράς σε φάσεις.

- ΕΛΕΓΧΟΣ USE CASES VIA SYSTEM TESTS

Στην αρχική σχεδιαστική φάση, αρκεί να συμπληρώσετε την λεκτική περιγραφή με τις OREOS προδιαγραφές. Στην τελική φάση, συμπληρώστε και τις λεπτομέρειες σε σχέση με τις εμπλεκόμενες μεθόδους και το setup input, output, pre-post conditions, ...

- USE CASE UC1: LOAD DATA

Test cases

<i>Description</i>	<i>ON</i>	<i>any context</i>
	<i>RECEIVING</i>	<i>Request to parse a specific txt file with a valid timeline</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A timeline with the correct size and no offending <time,value>pairs</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T1_V0_01	HappyDayScenario for SimpleTextParser.parse()
Pre-cond.		No specific precondition constructed
Input		input_test.txt, a small file with less than 10 entries, all valid
Output		a timeline with the same #entries as the contexts of input_test.txt and no offending values
Post-cond.		No state properties tested
Method To test		SimpleTextParser.parse(filename)

ID	T1_V0_02	HappyDayScenario for MainEngine.setTimeLine()
		...identical setup with T1_V0_01
Method To test		MainEngine.setTimeLine(filename)

Involved methods

MainEngine.setTimeLine(),

IParser.parse --> SimpleTextParser.parse(filename)

Not designed yet: T1_V1: missing file, T1_V2: invalid values in input file

- **USE CASE UC2: ANALYZE TIMELINE (AND PRODUCE PHASES)**

Test cases

<i>Description</i>	<i>ON</i>	<i>A time line having being loaded</i>
	<i>RECEIVING</i>	<i>Request to analyze a valid timeline into phases</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>a set of phases</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T2_V0_01	HappyDayScenario for NaiveAnalyser.producePhases()
Pre-cond.		Load input_test.txt, a small file with less than 10 entries, all valid, for o(5) phases, and produce timeline
Input		the abovementioned timeline
Output		A correct #phases, with the correct points inside
Post-cond.		No state properties tested
Method To test		NaiveAnalyser.producePhasesFromTimeLine (TimeLine)

ID	T2_V0_02	HappyDayScenario for MainEngine.producePhases()
		...identical setup with T2_V0_01
Method To test		MainEngine.producePhases()

Involved methods

MainEngine.producePhases()

AnalyserFactory.createAnalyzer()

IAnalyzer --> NaiveAnalyser.producePhasesFromTimeLine (TimeLine)

Not designed yet: T2_V1: null time line, T2_V2: timeline with only one phase

- **USE CASE UC3: VISUALIZE TIMELINE**

Test cases

<i>Description</i>	<i>ON</i>	<i>A time line having being loaded</i>
	<i>RECEIVING</i>	<i>Request to visualize a valid timeline via a specific visualizer</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>An appropriate visualization</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T3_V0_01	HappyDayScenario for MainEngine.visualize()
Pre-cond.		Load input_test.txt, a small file with less than 10 entries, all valid, for o(5) phases, and produce timeline
Input		the abovementioned timeline, " <code>HtmlVisualizer</code> " as the tested visualizer
Output		A correct visualization, expressed as a 2D raster of chars
Post-cond.		No state properties tested
Method To test		MainEngine. <code>setVisualizer(String)</code> MainEngine.visualize()

ID	T3_V0_02	HappyDayScenario for MainEngine.visualize()
		...identical setup with T2_V0_01
Input		" <code>ConsoleVisualizer</code> " as the tested visualizer

Involved methods

MainEngine.visualize()

- USE CASE UC4: VISUALIZE PHASES (TO CONSOLE)

Test cases

<i>Description</i>	<i>ON</i>	<i>A time line having being loaded</i>
	<i>RECEIVING</i>	<i>Request to visualize the phases of the timeline</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>An appropriate visualization for the phases</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T4_V0	HappyDayScenario for MainEngine.reportPhases()
Pre-cond.		Load input_test.txt, a small file with less than 10 entries, all valid, for o(5) phases, produce timeline and analyze it to phases
Input		the abovementioned timeline and its phases
Output		A correct visualization (approximation: the size of the descriptions produced is the same with the number of phases)
Post-cond.		No state properties tested
Method To test		MainEngine. reportPhases()

Involved methods

MainEngine.reportPhases()

NaiveAnalyser.reportToConsole()

Phase.consoleVerticalReport()

• TRACEABILITY MATRIX

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

UC1	Load Data
UC2	Analyze TimeLine
UC3	Present TimeLine
UC4	Present Phases

Πίνακας 1 Σύνοψη use cases και των id's τους

Ο Πίνακας 2 είναι ο traceability matrix για τους ελέγχους μας. Στη συνέχεια, οι έλεγχοι επεξηγούνται πιο αναλυτικά.

	UC1	UC2	UC3	UC4
T1_V0_01	X			
T1_V0_02	X			
T2_V0_01		X		
T2_V0_02		X		

T3_V0_01			X	
T3_V0_02			X	
T4_V0				X

Πίνακας 2 Traceability matrix between use cases and tests

- **ΕΚΚΡΕΜΟΤΗΤΕΣ (ToDo)**

Εκκρεμούν μη υλοποιημένοι έλεγχοι ως ακολούθως (αν υπάρχουν εκκρεμότητες, παραθέστε την *TODO* λίστα ελέγχων που πρέπει να ετοιμαστούν)

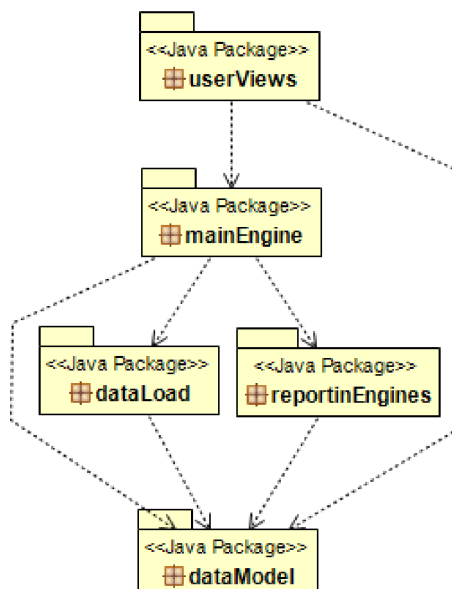
- Unit tests are missing for several classes, both at the model and at the business logic level, specifically, class XXX, YYY, ZZZ

- **ΣΧΕΔΙΑΣΗ ΛΟΓΙΣΜΙΚΟΥ**

- **ΔΙΑΓΡΑΜΜΑΤΑ ΠΑΚΕΤΩΝ / ΥΠΟΣΥΣΤΗΜΑΤΩΝ**

Η ανάλυση του κώδικα σε υποσυστήματα και πακέτα έχει νόημα μόνο όταν το μέγεθος και η πολυπλοκότητα του κώδικα επιτάσσουν την εν λόγω διαίρεση.

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



Σχήμα 1. Διάγραμμα πακέτων (εδώ: από την αξιολόγηση εστιατορίου)

Ακολουθεί μια συνοπτική περιγραφή των πακέτων του συστήματος.

ΠΑΚΕΤΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

userViews	Περιέχει τις boundary classes που είναι υπεύθυνες για την αλληλεπίδραση με το χρήστη
mainEngine	Κεντρική business logic engine, along with the necessary interface to export to the boundary classes
dataLoad	Υποσύστημα αλληλεπίδρασης με τα αρχεία δεδομένων, για την ανάκτησή τους από το σύστημα
reportEngines	Υποσύστημα παραγωγής αναφορών
dataModel	Domain classes of the system

Πίνακας 3. Συνοπτική περιγραφή πακέτων συστήματος (εδώ: από την αξιολόγηση εστιατορίου)

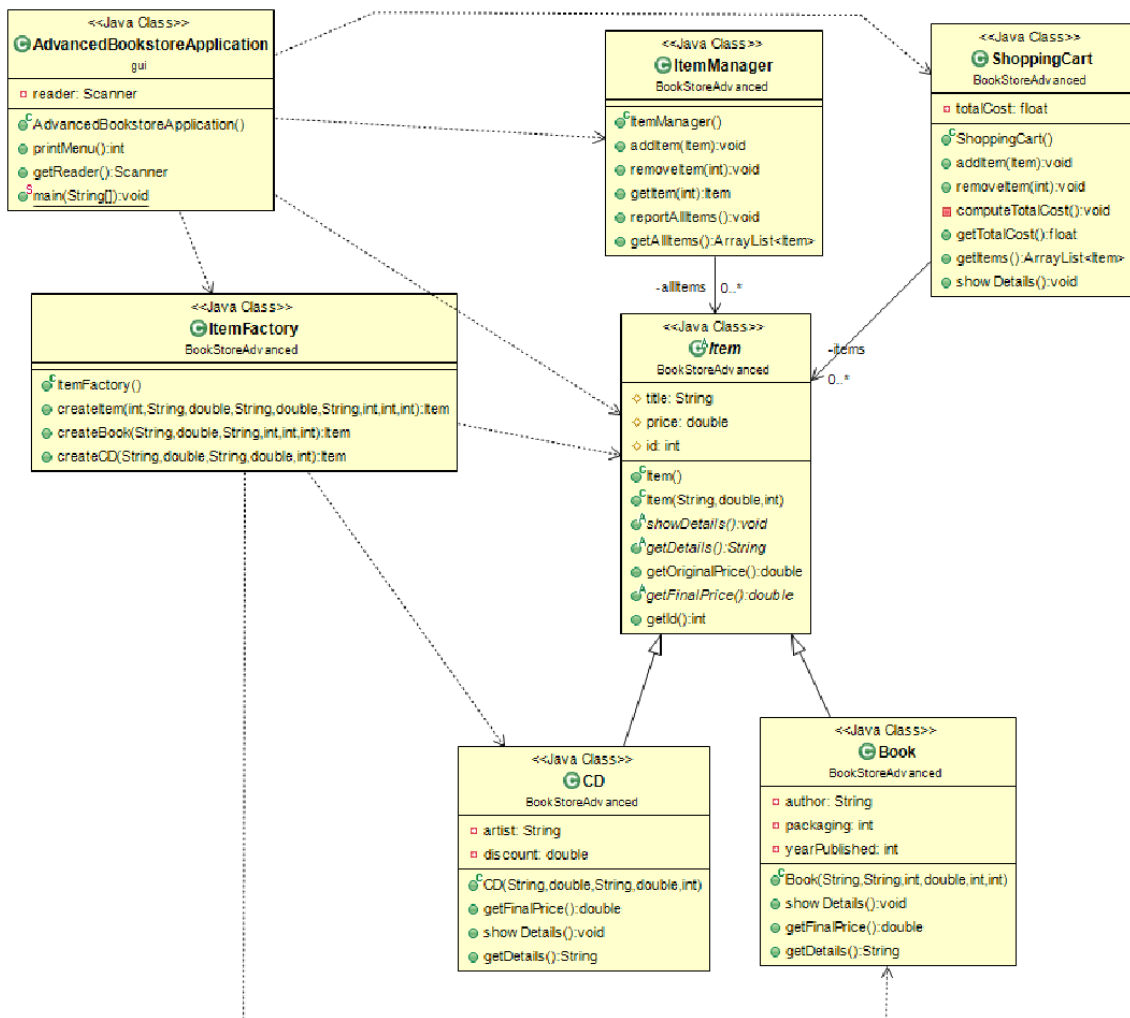
- ΔΙΑΓΡΑΜΜΑΤΑ ΚΛΑΣΕΩΝ

Στην παρούσα υποενότητα, παρατίθενται τα διαγράμματα κλάσεων και ακολουθιών.

Π.χ., για το παράδειγμα με τον έλεγχο του βιβλιοπωλείου έχουμε περισσότερα του ενός διαγράμματα.

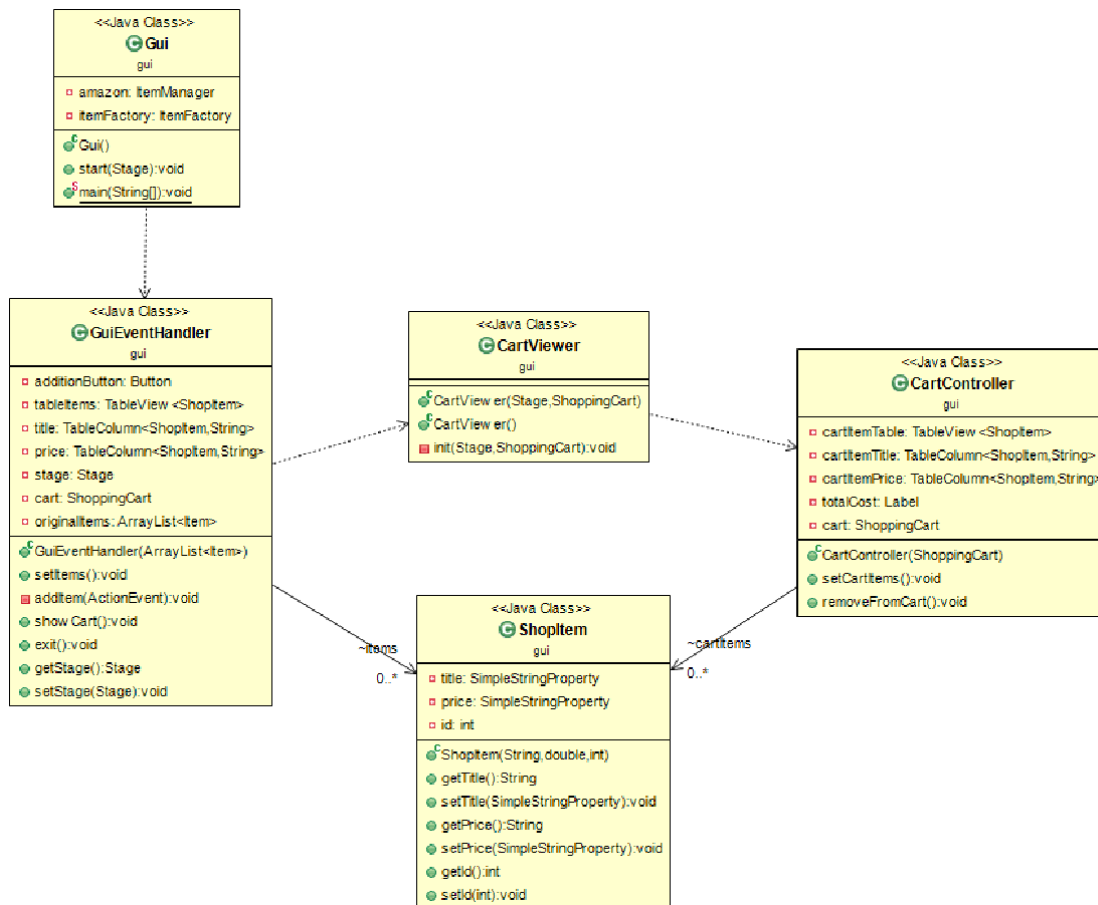
(ΠΙΠΡΟΣΟΧΗ: μόλις αλλάξαμε project απ' όπου δανειζόμαστε παραδείγματα!)

package bookstoreAdvanced;



Σχήμα 2. Διάγραμμα κλάσεων για το πακέτο bookstoreAdvanced

package gui;

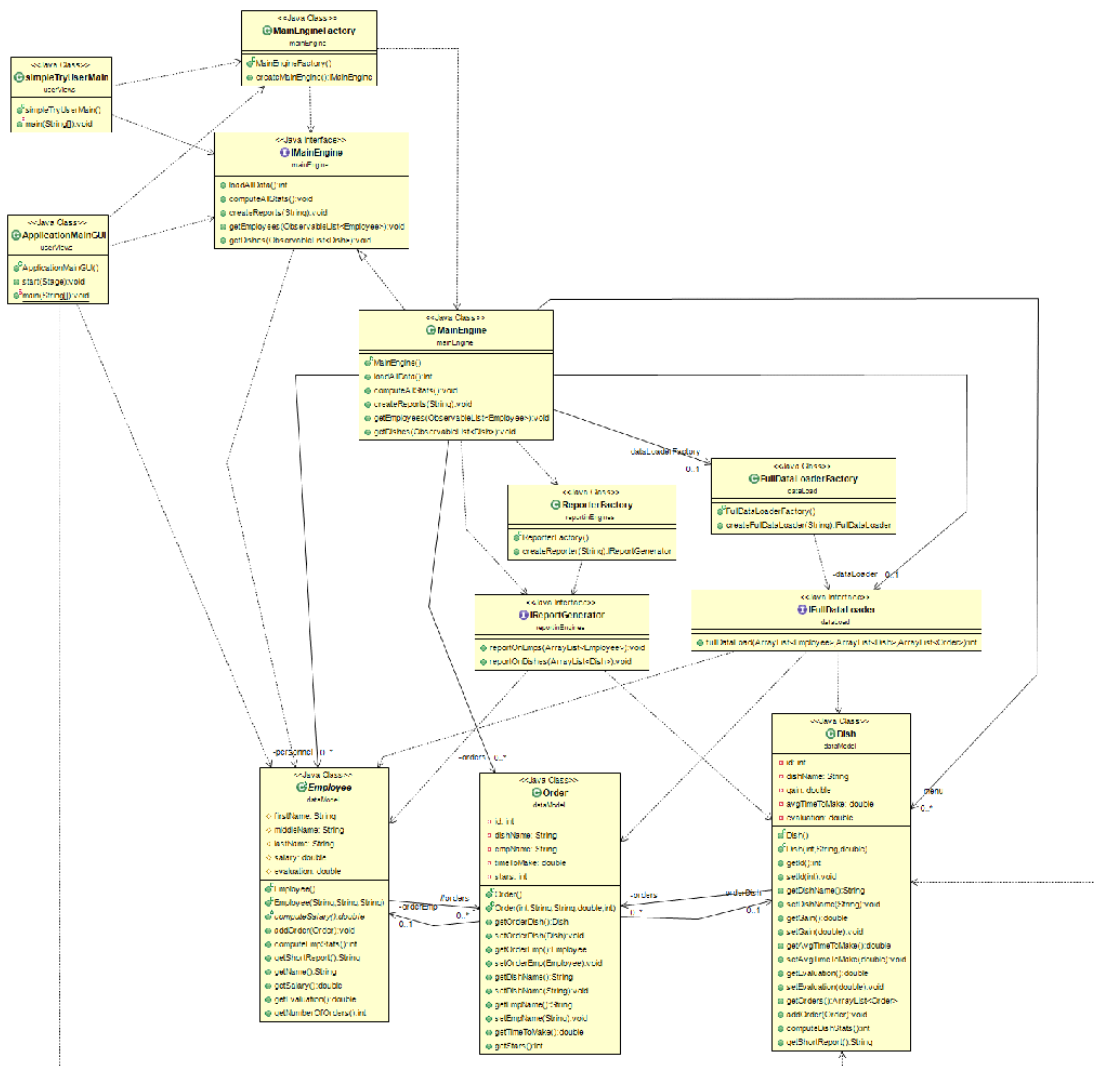


Σχήμα 3. Διάγραμμα κλάσεων για το πακέτο gui

Μπορείτε, επίσης, στα διαγράμματα που δίνετε, να βάζετε και συνεργαζόμενες κλάσεις από άλλα πακέτα. Το διάγραμμα κλάσεων του Σχ. 4 παρουσιάζει τις κεντρικές κλάσεις της εφαρμογής αξιολόγησης εστιατορίου, μαζί με τις συνεργαζόμενες κλάσεις.

- Πλεονεκτήματα: συνολική εποπτεία του συστήματος σε μία (1) απεικόνιση.
- Μειονεκτήματα: η απεικόνιση είναι στα όρια του οπτικά διακριτού. Από ένα σημείο κι έπειτα, είναι αδύνατο να διακρίνει κανείς τι είναι στο διάγραμμα (ακόμα κι αν έχει ακολουθήσει μια προσεγμένη διάταξη στο χώρο), οπότε το διάγραμμα γίνεται πρακτικά άχρηστο. (θυμηθείτε: το διάγραμμά σας, εγώ θα το δω σε χαρτί – θα έχουμε μπροστά μας μόνο ό,τι τυπώσετε)

Παρατηρήστε επίσης, ότι αν τυχόν χρειάζεται να τεκμηριώσουμε / εξηγήσουμε / ανακτήσουμε την συνεργασία ενός υποσυνόλου κλάσεων, είναι απολύτως νόμιμο να κατασκευάζουμε διαγράμματα με κλάσεις από διαφορετικά πακέτα -- δεν μας περιορίζουν τα όρια των πακέτων, δλδ. Τα διαγράμματα πρέπει να εξυπηρετούν μια ομάδα ανάπτυξης λογισμικού και όχι οι άνθρωποι τα διαγράμματα. (Εννοείται πως αυτό δεν είναι δικαιολογία για να τα ισοπεδώσουμε όλα...)



Σχήμα 4. Διάγραμμα κλάσεων επεξήγησης ενός πακέτου με συνεργαζόμενες κλάσεις

• 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)

(γ) Την απεικόνιση των use cases σε μεθόδους (όχι σε κλάσεις, σε μεθόδους)

Αυτού του είδους η τεκμηρίωση δεν θα υπήρχε σε μια επαγγελματική αναφορά – όμως, επαληθεύει την οργάνωση και την πληρότητα της σχεδίασής σας.

• DOMAIN CLASSES

Package bookstoreAdvanced	CD, Book, μια αφηρημένη κλάση Item(A) για αυτές τα δύο, και ένα factory, το ItemFactory, για την κατασκευή στιγμιότυπων. ShoppingCart.
---------------------------	--

- BUSINESS LOGIC CLASSES

Package bookstoreAdvanced	<p>ItemManager, για την υλοποίηση όλων των use cases στο back-end. ItemManager:</p> <ul style="list-style-type: none"> • Interfaces with domain classes via Item+ Factory. • Has no interface to boundary classes (!!!)
---------------------------	--

- BOUNDARY CLASSES

Package gui	<p>AdvancedBookstoreApplication, μια client class για την αλληλεπίδραση με το χρήστη μέσω κονσόλας.</p> <p>Gui, GuiEventHandler, CartViewer, CartController, ShoppingItem, για το graphical user interface με το χρήστη (οι handler/controller classes χειρίζονται τα events από τη γραφική διαπροσωπεία).</p>
-------------	--

- ΑΠΕΙΚΟΝΙΣΗ ΑΠΑΙΤΗΣΕΩΝ ΣΕ ΜΕΘΟΔΟΥΣ

Υπάρχουν 4 use cases για το ηλ. βιβλιοπωλείο: εμφάνισε προϊόντα καταστήματος, πρόσθεσε προϊόν στο καλάθι, διέγραψε προϊόν από το καλάθι, εμφάνισε προϊόντα του καλαθιού.

ΑΠΕΙΚΟΝΙΣΗ USE CASES ΣΕ ΜΕΘΟΔΟΥΣ

Use case	Back-end methods	Front-end methods
εμφάνισε προϊόντα καταστήματος	ItemManager.reportAllItems()	Gui.start() GuiEventHandler.setItems()
πρόσθεσε προϊόν στο καλάθι	ShoppingCart.addItem()	GuiEventHandler.addItem()
διέγραψε προϊόν από το καλάθι	ShoppingCart.removeItem()	CartController.removeFromCart()
εμφάνισε προϊόντα του καλαθιού	ShoppingCart.showDetails()	GuiEventHandler.showCart()

Πίνακας 4 Επαλήθευση απεικόνισης use cases σε μεθόδους

- ΔΙΑΓΡΑΜΜΑΤΑ ΑΚΟΛΟΥΘΙΩΝ

Αν ζητούνται / υπάρχουν

- ΛΟΙΠΑ ΣΧΟΛΙΑ

Εδώ προστίθενται όποια σχόλια μπορεί να υπάρχουν (αν υπάρχουν) για σχεδιαστικές υποθέσεις, αποφάσεις, ελλείψεις και σημεία κινδύνου, ή για οτιδήποτε άλλο κρίνεται σημαντικό να καταγραφεί για τη μελλοντική συντήρηση του κώδικα.

- ΣΧΕΔΙΑΣΤΙΚΕΣ ΑΠΟΦΑΣΕΙΣ

Αν υπάρχει λόγος να καταγραφούν εναλλακτικές σχεδιάσεις και γιατί αποφασίσθηκε να προκριθεί κάποια από αυτές.

- ΣΗΜΕΙΑ ΚΙΝΔΥΝΟΥ

Αν υπάρχουν.

- ΕΚΚΡΕΜΟΤΗΤΕΣ (ToDo)

Αν υπάρχουν. Π.χ., εδώ:

- Missing interface(s) between `ItemManager` and boundary classes