

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΤΜΗΜΑ ΜΗΧ. Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ

ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ ΓΙΑ ΤΟ
ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2023-2024

ΟΜΑΔΑ 4920-4977

ΠΑΠΑΔΟΠΟΥΛΟΥ ΜΑΡΙΑ, ΑΜ:4920

ΑΒΑΝΙΔΗ ΙΩΑΝΝΑ-ΜΑΡΙΑ, ΑΜ:4977

ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ

ΔΕΚΕΜΒΡΙΟΣ 2023

ΙΣΤΟΡΙΚΟ ΕΚΔΟΣΕΩΝ ΤΗΣ ΠΑΡΟΥΣΑΣ ΑΝΑΦΟΡΑΣ

Ημερομηνία	Έκδοση	Περιγραφή	Συγγραφείς
2023/11/20	v.01	Οργάνωση απαιτήσεων σε use cases	4920, 4977
2023/11/24	v.02	Αρχική σχεδίαση κλάσεων και ελέγχων	4920, 4977
2023/12/1	V.03	Διορθώσεις στις uses cases, επεκτάσεις στη σχεδίαση κλάσεων και ελέγχων	4920, 4977
2023/12/18	V.04	ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ	4920, 4977

1 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ – USE CASES

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

LOADDATA

ID: UC 1

DESCRIPTION AND GOAL

Το use case «loadData» φορτώνει τα δεδομένα στο σύστημα.

ACTORS (ESP. PRIMARY ACTOR)

Ο Αναλυτής

PRECONDITIONS

Υπάρχει ένα αρχείο με δεδομένα

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής φορτώσει το αρχείο κειμένου με τις μετρήσεις στο σύστημα.
2. Το σύστημα αναπαριστά κατάλληλα κάθε γραμμή του αρχείου.

EXTENSIONS / VARIATIONS

1. Στην περίπτωση κατά την οποία δεν υπάρχουν δεδομένα στο σύστημα εμφανίζεται ένα μήνυμα που ενημερώνει ότι δεν υπάρχουν διαθέσιμα δεδομένα.

POST CONDITIONS

Τα δεδομένα έχουν φορτωθεί με επιτυχία.

FINDSINGLECOUNTRYINDICATOR

ID: UC 2

DESCRIPTION AND GOAL

Το use case «displayCountryDisasterData» κάνει ανάκτηση και παρουσίαση των στοιχείων του αιτήματος και αποθηκεύει το όνομά του.

ACTORS (ESP. PRIMARY ACTOR)

Ο Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα.

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής κάνει αίτημα της μορφής:
 - 1.1 "Χώρα-Τύπος Καταστροφής".
2. Το Σύστημα αναλύει τα δεδομένα που δόθηκαν.
3. Το Σύστημα παρουσιάζει τα δεδομένα που ανέλυσε.

EXTENSIONS / VARIATIONS

Στη περίπτωση που το αίτημα έχει λάθος μορφή εμφανίζει ένα μήνυμα που ενημερώνει τον αναλυτή ότι το αίτημα έχει λάθος μορφή.

POST CONDITIONS

1. Το Σύστημα επιστρέφει τα δεδομένα που ζητούνται.
2. Δημιουργείται όνομα του αιτήματος.

FINDSINGLECOUNTRYINDICATORYEARRANGE

ID: UC 3

DESCRIPTION AND GOAL

Το use case «displayCountryDisasterData» κάνει ανάκτηση και παρουσίαση των στοιχείων του αιτήματος και αποθηκεύει το όνομά του.

ACTORS (ESP. PRIMARY ACTOR)

Ο Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα.

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής κάνει αίτημα της μορφής:
 - 1.1 "Χώρα-Τύπος Καταστροφής_[Αρχικό έτος Τελικό έτος]"
2. Το Σύστημα αναλύει τα δεδομένα που δόθηκαν.
3. Το Σύστημα παρουσιάζει τα δεδομένα που ανέλυσε.

EXTENSIONS / VARIATIONS

Στη περίπτωση που το αίτημα έχει λάθος μορφή εμφανίζει ένα μήνυμα που ενημερώνει τον αναλυτή ότι το αίτημα έχει λάθος μορφή.

POST CONDITIONS

1. Το Σύστημα επιστρέφει τα δεδομένα που ζητούνται.
2. Δημιουργείται όνομα του αιτήματος.

GETDESCRIPTIVESTATS

ID: UC 4

DESCRIPTION AND GOAL

Το use case «calculateStatistics» υπολογίζει βασικά στατιστικά του αιτήματος.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα και υπάρχουν ονόματα αιτημάτων.

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής δώσει το όνομα ενός αιτήματος.
2. Το Σύστημα υπολογίζει τη μέγιστη τιμή.
3. Το Σύστημα υπολογίζει την ελάχιστη τιμή.
4. Το Σύστημα υπολογίζει τον αριθμό των χρονιών που είχαν κάποιο συμβάν.
5. Το Σύστημα υπολογίζει την μέση τιμή.
6. Το Σύστημα υπολογίζει την ενδιάμεση τιμή.
7. Το Σύστημα υπολογίζει το συνολικό πλήθος συμβάντων.

EXTENSIONS / VARIATIONS

Στη περίπτωση που δεν έχει αποθηκευτεί το συγκεκριμένο όνομα αιτήματος εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι δεν υπάρχει το όνομα αιτήματος.

POST CONDITIONS

Το σύστημα έχει υπολογίσει βασικά περιγραφικά χαρακτηριστικά.

GETREGRESSION

ID: UC 5

DESCRIPTION AND GOAL

To use case «calculateRegression» υπολογίζει το Regression.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα και έχει προστεθεί η σχετική βιβλιοθήκη.

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής δώσει το όνομα ενός αιτήματος.
2. Το σύστημα υπολογίζει το Regression.

EXTENSIONS / VARIATIONS

1. Στη περίπτωση που δεν έχει προστεθεί η σχετική βιβλιοθήκη εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι δεν έχει είναι εφικτός ο υπολογισμός του Regression.
2. Στη περίπτωση που δεν έχει αποθηκευτεί το συγκεκριμένο όνομα αιτήματος εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι δεν υπάρχει το όνομα αιτήματος.

POST CONDITIONS

Το σύστημα έχει υπολογίσει το Regression.

CREATEREPORTTXT

ID: UC 6

DESCRIPTION AND GOAL

Το use case «createReportTxt» αποθηκεύει μια αναφορά για ένα αίτημα, σε συγκεκριμένο τύπο αρχείου και με συγκεκριμένη μορφολογία.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα και υπάρχουν ονόματα αιτημάτων.

BASIC FLOW

1. Το use case αρχίζει όταν αναλυτής επιλεγεί ως τρόπο αποθήκευσης το txt.
2. Ο Αναλυτής επιλέγει όνομα αιτήματος για να αναλύσει.
 - 2.1 Το Σύστημα καταγράφει το όνομα του αιτήματος.
 - 2.1 Το Σύστημα καταγράφει το όνομα της χώρας.
 - 2.1 Το Σύστημα καταγράφει τον τύπο καταστροφής.
 - 2.1 Το Σύστημα καταγράφει τη λίστα μετρήσεων.
 - 2.1 Το Σύστημα καταγράφει τα βασικά στατιστικά.
 - 2.1 Το Σύστημα καταγράφει το αποτέλεσμα του Regression.

EXTENSIONS / VARIATIONS

Στη περίπτωση που ο τρόπος αποθήκευσης που επιλέγει ο αναλυτής δεν είναι ένας από τους διαθέσιμους εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι ο τρόπος αποθήκευσης που επέλεξε δεν είναι διαθέσιμος.

POST CONDITIONS

Το σύστημα αποθηκεύει μια αναφορά των αιτημάτων.

CREATEREPORTMD

ID: UC 7

DESCRIPTION AND GOAL

Το use case «createReportMd» αποθηκεύει μια αναφορά για ένα αίτημα, σε συγκεκριμένο τύπο αρχείου και με συγκεκριμένη μορφολογία.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα και υπάρχουν ονόματα αιτημάτων.

BASIC FLOW

1. Το use case αρχίζει όταν αναλυτής επιλεγεί ως τρόπο αποθήκευσης το markdown κείμενο.
2. Ο Αναλυτής επιλέγει όνομα αιτήματος για να αναλύσει.
 - 2.1 Το Σύστημα καταγράφει το όνομα του αιτήματος με **bold**.
 - 2.2 Το Σύστημα καταγράφει το όνομα της χώρας με *italics*.
 - 2.3 Το Σύστημα καταγράφει του τύπου καταστροφών με *italics*.
 - 2.4 Το Σύστημα καταγράφει τη λίστα μετρήσεων σε πίνακάκι.
 - 2.5 Το Σύστημα καταγράφει τα βασικά στατιστικά.
 - 2.6 Το Σύστημα καταγράφει το αποτέλεσμα του Regression.

EXTENSIONS / VARIATIONS

Στη περίπτωση που ο τρόπος αποθήκευσης που επιλέγει ο αναλυτής δεν είναι ένας από τους διαθέσιμους εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι ο τρόπος αποθήκευσης που επέλεξε δεν είναι διαθέσιμος.

POST CONDITIONS

Το σύστημα αποθηκεύει μια αναφορά των αιτημάτων.

CREATEREPORTTOHTML

ID: UC 8

DESCRIPTION AND GOAL

Το use case «createReportToHtml» αποθηκεύει μια αναφορά για ένα αίτημα, σε συγκεκριμένο τύπο αρχείου και με συγκεκριμένη μορφολογία.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Υπάρχουν δεδομένα στο σύστημα και υπάρχουν ονόματα αιτημάτων.

BASIC FLOW

1. Το use case αρχίζει όταν αναλυτής επιλεγεί ως τρόπο αποθήκευσης το html.
- 2 Ο Αναλυτής επιλέγει όνομα αιτήματος για να αναλύσει.
 - 2.1 Το Σύστημα καταγράφει το όνομα του αιτήματος με bold.
 - 2.2 Το Σύστημα καταγράφει το όνομα της χώρας με italics.
 - 2.3 Το Σύστημα καταγράφει του τύπου καταστροφών με italics.
 - 2.4 Το Σύστημα καταγράφει τη λίστα μετρήσεων σε πινακάκι.
 - 2.5 Το Σύστημα καταγράφει τα βασικά στατιστικά.
 - 2.6 Το Σύστημα καταγράφει το αποτέλεσμα του Regression.

EXTENSIONS / VARIATIONS

Στη περίπτωση που ο τρόπος αποθήκευσης που επιλέγει ο αναλυτής δεν είναι ένας από τους διαθέσιμους εμφανίζεται μήνυμα που ενημερώνει τον αναλυτή ότι ο τρόπος αποθήκευσης που επέλεξε δεν είναι διαθέσιμος.

POST CONDITIONS

Το σύστημα αποθηκεύει μια αναφορά των αιτημάτων.

EXITPROGRAM

ID: UC 9

DESCRIPTION AND GOAL

To use case «exitProgram» τερματίζει το πρόγραμμα.

ACTORS (ESP. PRIMARY ACTOR)

Αναλυτής

PRECONDITIONS

Ο χρήστης είναι εντός του προγράμματος.

BASIC FLOW

1. Το use case αρχίζει όταν ο αναλυτής επιλέξει έξοδο.
2. Το Σύστημα ρώτα τον αναλυτή αν είναι σίγουρος ότι θέλει να αποχωρήσει από το πρόγραμμα.
 - 2.1 ο αναλυτής απαντάει ναι.
 - 2.1.1. Το σύστημα τερματίζει το πρόγραμμα.

EXTENSIONS / VARIATIONS

-

POST CONDITIONS

-

2 ΣΧΕΔΙΑΣΗ ΕΛΕΓΧΩΝ

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

2.1 ΕΛΕΓΧΟΣ USE CASES VIA SYSTEM TESTS

2.1.1 USE CASE UC1: LOAD

Test cases

<i>Description</i>	<i>ON</i>	<i>any context</i>
	<i>RECEIVING</i>	<i>Request to load and parse a specific tsv file</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A list with disasters</i>
	<i>SUCH THAT</i>	<i>Collect the disasters with their details in a list</i>

ID	T1_V0_01	<i>Successful load</i>
Pre-cond.		<i>TSV file exists</i>
Input		<i>The TSV File</i>
Output		<i>Return the condition of loading</i>
Post-cond.		<i>No state to test</i>
Method To test		<i>testManagerLoadClimateSuccess();</i>

ID	T1_V0_02	<i>Empty List</i>
Pre-cond.		<i>TSV file exists</i>
Input		<i>The TSV File</i>
Output		<i>Return a list with disasters</i>
Post-cond.		<i>No state to test</i>
Method To test		<i>testManagerLoadClimateFull()</i>

Involved methods

Manager.load();

2.1.2 USE CASE UC2: FILTERSINGLECOUNTRYINDICATOR

Test cases

<i>Description</i>	<i>ON</i>	Any state
	<i>RECEIVING</i>	A request Name, a country name, and a disaster type
	<i>ENSURE</i>	That the System
	<i>OUTPUTS</i>	A request
	<i>SUCH THAT</i>	state is intact

ID	T2_V0_01	Check correct country name
Pre-cond.		Tsv file is loaded
Input		An existing country Name, a request name and a disaster type
Output		A correct request
Post-cond.		No state properties tested
Method To test		testManagerFindSingleCountryIndicator();

ID	T2_V0_02	Check wrong country name
Pre-cond.		Tsv file is loaded
Input		An non existing country Name, a request name and a disaster type
Output		Enter Correct request
Post-cond.		No state properties tested
Method To test		testManagerFindSingleCountryIndicatoNoCountry();

Involved methods

Manager.findSingleCountryIndicator(String requestName, String countryName, String indicatorString)

2.1.3 USE CASE UC3: FILTERSINGLECOUNTRYINDICATORYEARRANGE

Test cases

<i>Description</i>	<i>ON</i>	<i>Any State</i>
	<i>RECEIVING</i>	<i>A request Name, a country name, a disaster type and a specific year range to analyze</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A request</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T3_V0_01	<i>Correct Year Range</i>
Pre-cond.		<i>Tsv file is loaded</i>
Input		<i>A request Name, a country name, a disaster type and a specific existing year range to analyze (startYear<endYear)</i>
Output		<i>A correct Request</i>
Post-cond.		<i>No state properties tested</i>
Method To test		<i>testManagerFindSingleCountryIndicatorYearRange();</i>

ID	T3_V0_02	<i>Wrong Year Range</i>
Pre-cond.		<i>Tsv file is loaded</i>
Input		<i>A request Name, a country name, a disaster type and a specific non existing year range to analyze (startYear>endYear)</i>
Output		<i>Enter Correct Request</i>
Post-cond.		<i>No state properties tested</i>
Method To test		<i>testFindSingleCountryIndicatorYearRangeGreaterEndYear();</i>

Involved methods

Manager. findSingleCountryIndicatorYearRange(String requestName, String countryName, String indicatorString, int startYear, int endYear)

2.1.4 USE CASE UC4: GETDESCRIPTIVESTATS

Test cases

<i>Description</i>	<i>ON</i>	<i>Any State</i>
	<i>RECEIVING</i>	<i>A request name</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A string with the descriptive stats</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T4_V0_01	<i>Correct descriptive stats</i>
Pre-cond.		<i>A TSV file is loaded and a correct request is made</i>
Input		<i>A request name</i>
Output		<i>A string with the correct descriptive stats</i>
Post-cond.		<i>No state properties tested</i>
Method To test		<i>testManagerGetDescriptiveStats();</i>

Involved methods

Manager.getDescriptiveStats(String requestName);

2.1.5 USE CASE UC5: GETREGRESSION

Test cases

<i>Description</i>	<i>ON</i>	<i>Any State</i>
	<i>RECEIVING</i>	<i>A request name</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>A string with the regression</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T5_V0_01	Correct regression
Pre-cond.		A TSV file is loaded and a correct request is made
Input		A request name
Output		A string with the correct descriptive stats
Post-cond.		No state properties tested
Method To test		testManagerGetDescriptiveStats();

Involved methods

Manager.getRegression(String requestName);

2.1.6 USE CASE UC6: REPORTToTXT

Test cases

<i>Description</i>	<i>ON</i>	<i>Any State</i>
	<i>RECEIVING</i>	<i>A request Name</i>
	<i>ENSURE</i>	<i>That the System</i>
	<i>OUTPUTS</i>	<i>Number of lines</i>
	<i>SUCH THAT</i>	<i>state is intact</i>

ID	T6_V0_01	Correct txt format
Pre-cond.		A TSV file is loaded, a correct request is made, descriptive stats and regression have been calculated
Input		A request name
Output		Number of recorded lines of the file
Post-cond.		No state properties tested
Method To test		testReportFirstTxt()

Involved methods

Report.reportTxt (String outputFileName, String requestName);

2.1.7 USE CASE UC7: REPORTToMD

Test cases

<i>Description</i>	<i>ON Any State</i>
	<i>RECEIVING A request Name</i>
	<i>ENSURE That the System</i>
	<i>OUTPUTS Number of lines</i>
	<i>SUCH THAT state is intact</i>

ID	T7_V0_01	Correct markdown format
Pre-cond.		A TSV file is loaded, a correct request is made, descriptive stats and regression have been calculated
Input		A request name
Output		Number of recorded lines of the file
Post-cond.		No state properties tested
Method To test		testReportMD()

Involved methods

Report.reportMd (String outputFileName, String requestName);

2.1.8 USE CASE UC8: REPORTToHTML

Test cases

<i>Description</i>	<i>ON Any State</i>
	<i>RECEIVING A request Name</i>
	<i>ENSURE That the System</i>
	<i>OUTPUTS Number of lines</i>
	<i>SUCH THAT state is intact</i>

ID	T8_V0_01	Correct html format
Pre-cond.		A TSV file is loaded, a correct request is made, descriptive stats and regression have been calculated
Input		A request name
Output		Number of recorded lines of the file
Post-cond.		No state properties tested
Method To test		testReportHTML()

Involved methods

Report.reportHtml (String outputFileName, String requestName);

2.2 TRACEABILITY MATRIX

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

UC1	LOAD
UC2	filterSingleCountryIndicator
UC3	filterSingleCountryIndicatorYeaRange
UC4	getDescriptiveStats
UC5	getRegression
UC6	reportToTxt
UC7	reportToMd
UC8	reportToHtml

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	UC5	UC6	UC7	UC8
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_01				X				
T5_V0_01					X			
T6_V0_01						X		
T7_V0_01							X	
T8_V0_01								X

Πίνακας 2 Traceability matrix between use cases and tests

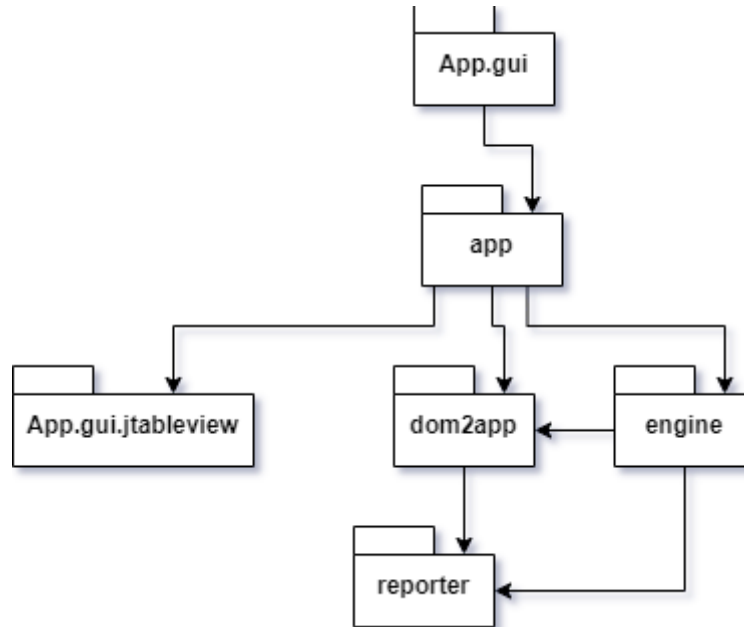
2.3 ΕΚΚΡΕΜΟΤΗΤΕΣ (ToDo)

3 ΣΧΕΔΙΑΣΗ ΛΟΓΙΣΜΙΚΟΥ

3.1 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.

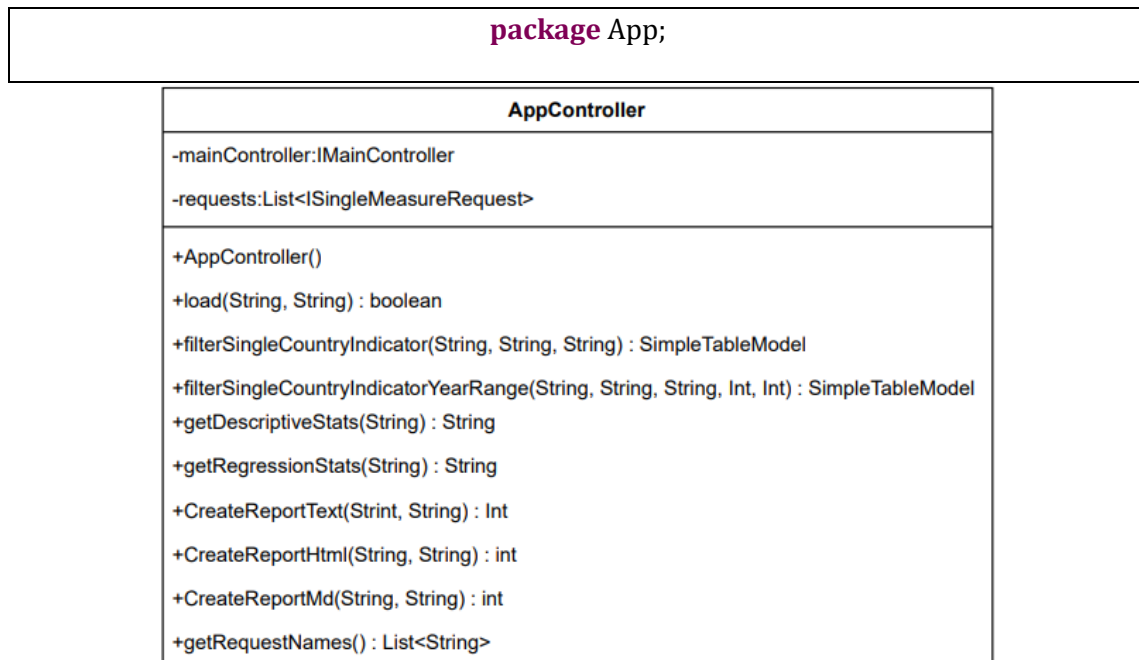
ΠΑΚΕΤΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

App	Υποσύστημα αλληλεπίδρασης με τα αρχεία δεδομένων, για την ανάκτησή τους από το σύστημα
App.gui	Κεντρικές business logic classes
App.gui.jtableview	Περιέχει τα domain classes που είναι υπεύθυνα για τη παρουσίαση των δεδομένων στον αναλυτή
Dom2app	Περιέχει τις boundary classes and interfaces του συστήματος που είναι υπεύθυνα για τη δημιουργία αιτημάτων
engine	Περιέχει τις boundary classes and interface που είναι υπεύθυνες για την αλληλεπίδραση με το χρήστη
reporter	Υποσύστημα παραγωγής αναφορών

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

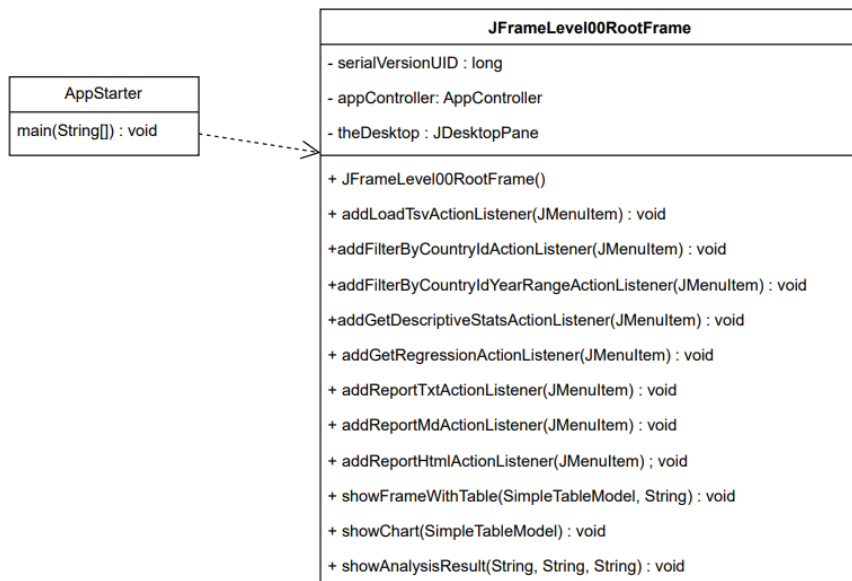
3.2 ΔΙΑΓΡΑΜΜΑΤΑ ΚΛΑΣΕΩΝ

Στην παρούσα υποενότητα, παρατίθενται τα διαγράμματα κλάσεων και ακολουθιών.



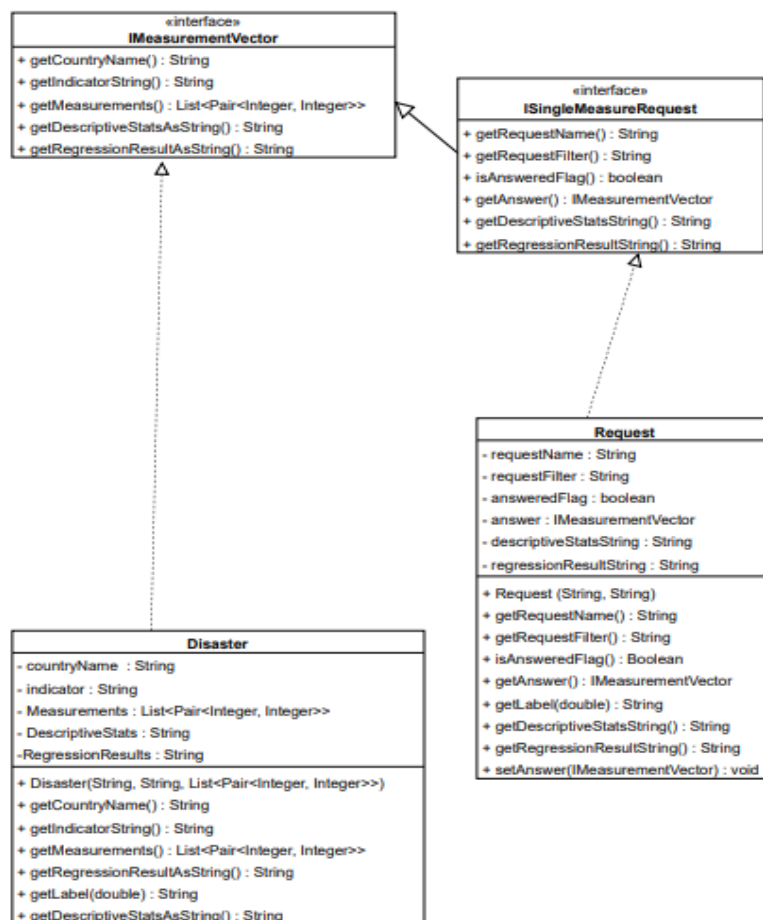
Σχήμα 1. Διάγραμμα κλάσεων για το πακέτο App

package App.gui;

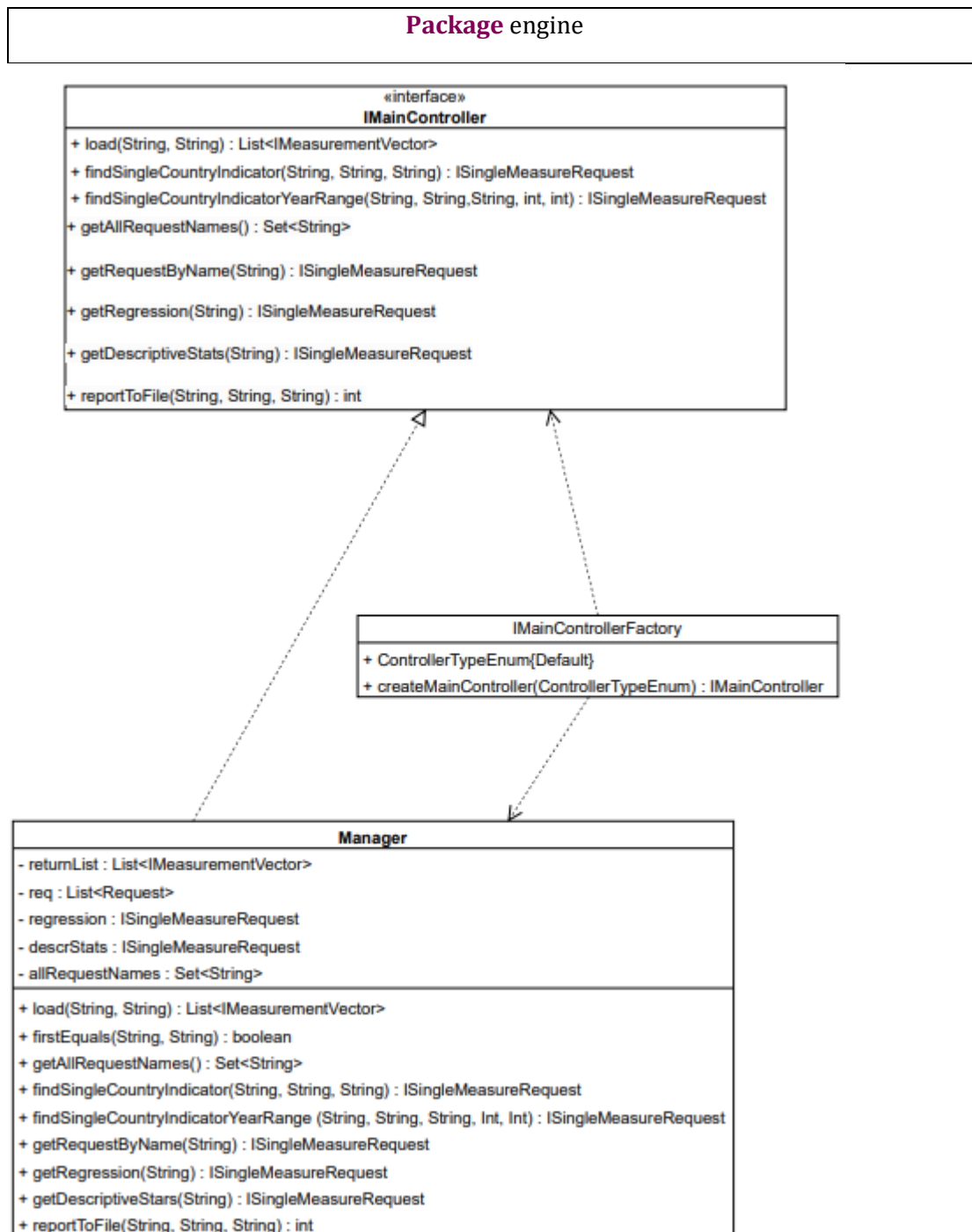


Σχήμα 2. Διάγραμμα κλάσεων για το πακέτο App.gui

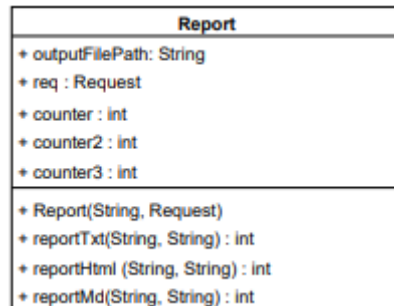
Package dom2app



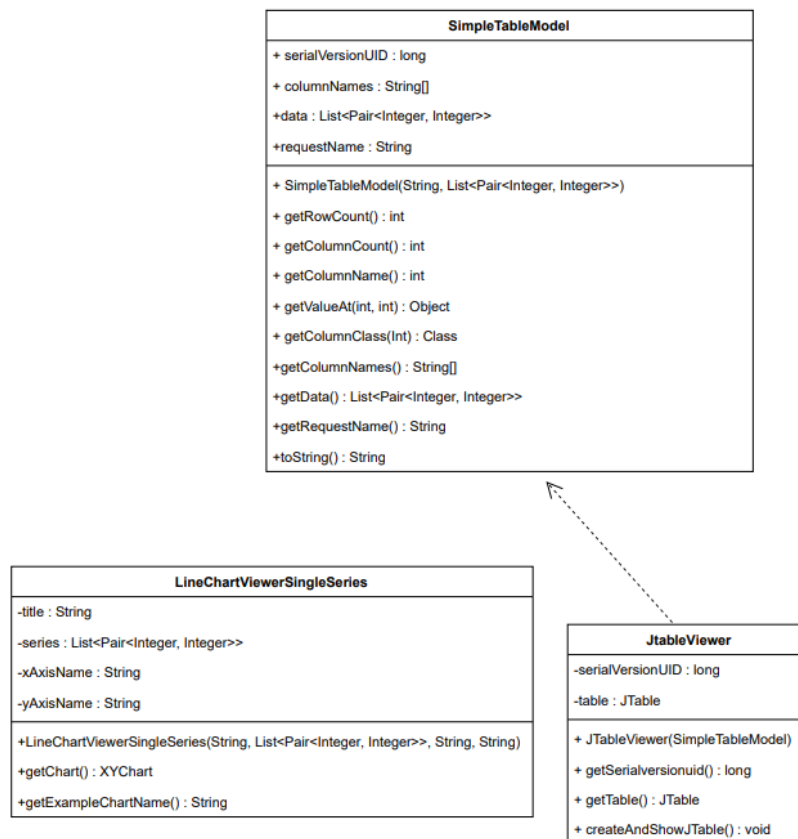
Σχήμα 3. Διάγραμμα κλάσεων για το πακέτο dom2app



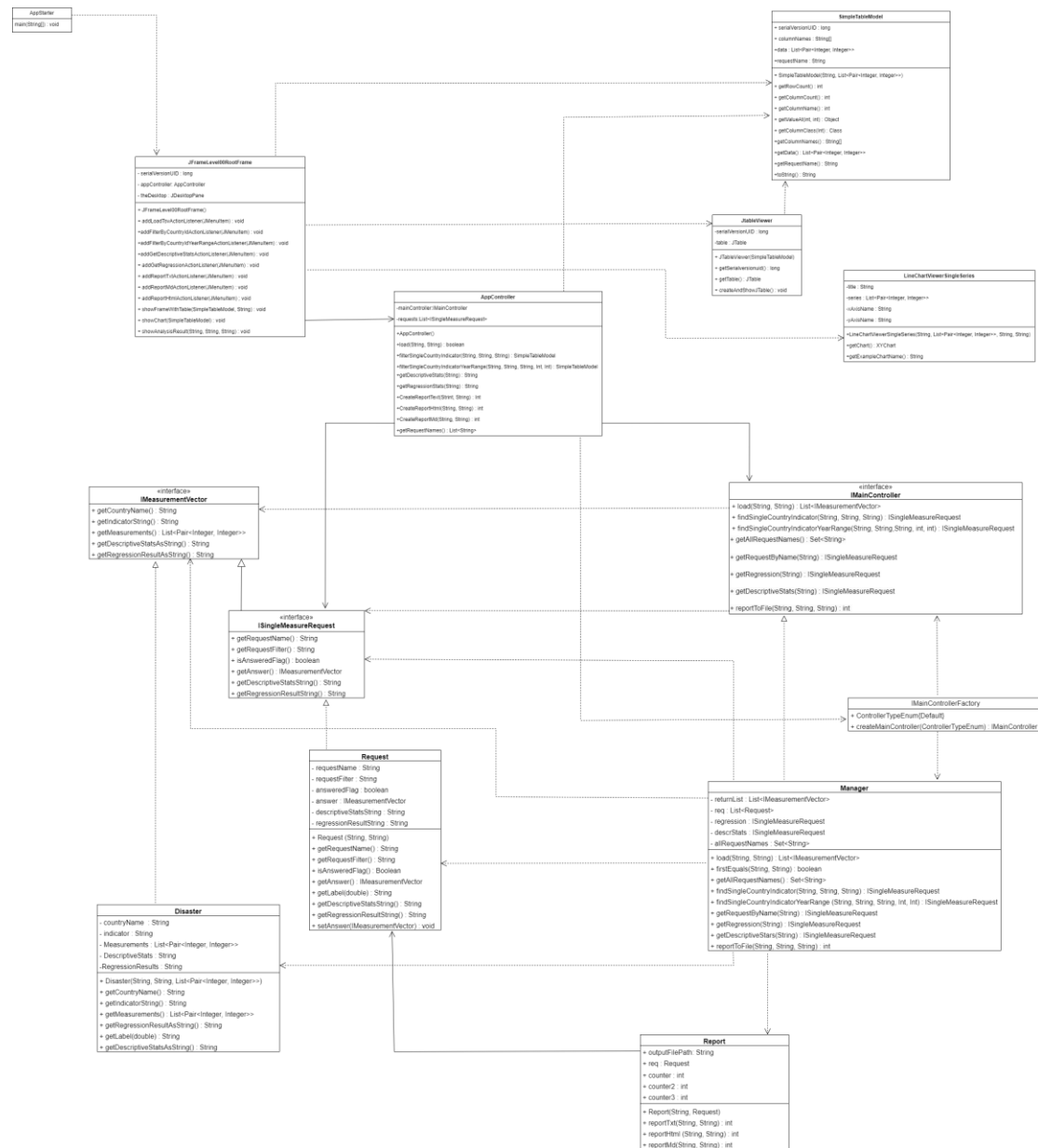
Σχήμα 4. Διάγραμμα κλάσεων για το πακέτο engine

Package Reporter

Σχήμα 5. Διάγραμμα κλάσεων για το πακέτο reporter

Package App.gui.jtableview

Συνολικό UML



Σχήμα 6. Διάγραμμα κλάσεων επεξήγησης ενός πακέτου με συνεργαζόμενες κλάσεις

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.

3.3.1 DOMAIN CLASSES

Package app.gui.jtableview	SimpleTableModel, κλάση υπεύθυνη για τη διατήρηση των δεδομένων σε πίνακα LineChartViewerSingleSeries, JtableViewer κλάσεις υπεύθυνες για τη δημιουργία του πίνακα και των γραφημάτων
Package reporter	Έξοδος δεδομένων σε αρχεία Html, Md, Text

3.3.2 BUSINESS LOGIC CLASSES

Package app	Φόρτωση στοιχείων από το frontend στο backend
Package app.gui	Κλάσεις για την εμφάνιση του γραφικού περιβάλλοντος και τη σύνδεση του με το backend

3.3.3 BOUNDARY CLASSES

Package dom2app	Δημιουργία αιτημάτων
Package engine	Κύριο μέρος του backend, εκεί επιτελούνται όλες οι λειτουργίες του προγράμματος

3.3.4 ΑΠΕΙΚΟΝΙΣΗ ΑΠΑΙΤΗΣΕΩΝ ΣΕ ΜΕΘΟΔΟΥΣ

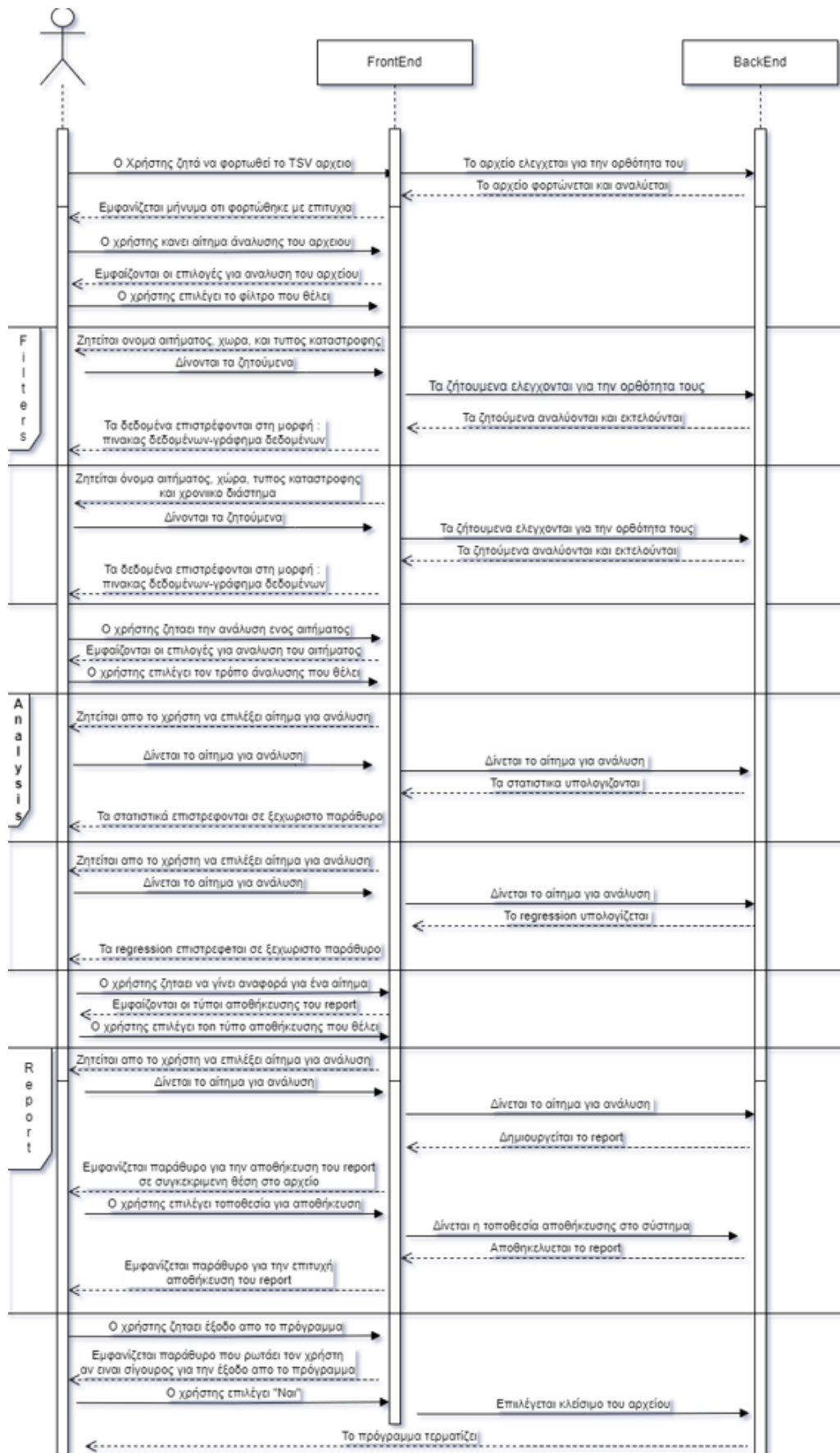
ΑΠΕΙΚΟΝΙΣΗ USE CASES ΣΕ ΜΕΘΟΔΟΥΣ

Use case	Back-end methods	Front-end methods
loadData	Manager.load()	JFrameLevel00RootFrame.addLoadTs vActionListener() AppController.load()
displayCountryDisa sterData	Manager.findSingleCountryIndi cator Manager.findSingleCountryIndi catorYearRange()	JFrameLevel00RootFrame.addFilter ByCountryIdActionListener() JFrameLevel00RootFrame.addFilterB yCountryIdYearRangeActionListener () AppController.filterSingleCountryIndi cator() AppController.filterSingleCountryIndi catorYearRange()
calculateStatistics	Disaster.getDescriptiveStatsA sStrin() Request.	JFrameLevel00RootFrame.addGetDes criptiveStatsActionListener() AppController.getDescriptiveStat

	getDescriptiveStatsString()	s()
calculateRegression	Disaster.getRegressionAsString() Request.getRegressionString()	JFrameLevel100RootFrame.addGetRegressionActionListener() AppController.getRegrssion()
saveReport	Report.reportTxt() Report.reportHtml() Report.reportMd()	JFrameLevel100RootFrame. addReportTxtActionListener() .addReportMdActionListener() .addReportHtmlActionListener AppController.createReportText() .createReportMd() <u>.createReportHtml()</u>
exitProgram	JFrameLevel100RootFrame()	JFrameLevel100RootFrame()

Πίνακας 4 Επαλήθευση απεικόνισης use cases σε μεθόδους

3.4 ΔΙΑΓΡΑΜΜΑΤΑ ΑΚΟΛΟΥΘΙΩΝ



4 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.

4.1 ΣΧΕΔΙΑΣΤΙΚΕΣ ΑΠΟΦΑΣΕΙΣ

-

4.2 ΣΗΜΕΙΑ ΚΙΝΔΥΝΟΥ

Η ψυχική μας υγεία!!

4.3 ΕΚΚΡΕΜΟΤΗΤΕΣ (ToDo)-BONUS

<https://www.youtube.com/watch?v=MQbJVtviQs>



(Εμείς όταν τελειώσαμε επιτέλους την εργασία)