

Σύστημα Δημόσιας Διάθεσης Δεδομένων

Θεμιστοκλεία Σιακαβάρα

Διπλωματική Εργασία

Επιβλέπων: Π. Βασιλειάδης

Ιωάννινα, Ιούνιος 2025



ΤΜΗΜΑ ΜΗΧ. Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
UNIVERSITY OF IOANNINA

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον επιβλέποντα καθηγητή μου, κύριο Π. Βασιλειάδη, για την πολύτιμη καθοδήγηση και τη διαρκή υποστήριξη του καθ' όλη τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου και τους φίλους μου για την αμέριστη στήριξη και ενθάρρυνση που μου προσέφεραν σε κάθε στάδιο της προσπάθειας μου.

Ιούνιος 2025

Θεμιστοκλεία Σιακαβάρα

Περίληψη στα ελληνικά

Σκοπός της παρούσας διπλωματικής εργασίας ήταν η ανάπτυξη ενός λογισμικού συστήματος, το οποίο αξιολογεί την ποιότητα συνόλων δεδομένων, σύμφωνα με τις αρχές Findable, Accessible, Interoperable, και Reusable (FAIR). Η αξιολόγηση πραγματοποιείται τόσο σε επίπεδο ολόκληρου αρχείου, όσο και σε επίπεδο επιμέρους πεδίων (στήλες).

Το σύστημα παρέχει δυνατότητες καταχώρησης συνόλων δεδομένων σε μορφή CSV, εκτέλεσης επιμέρους ή συνολικών ελέγχων σε αυτά τα δεδομένα, καθώς και παραγωγής αναφορών με τα αποτελέσματα της αξιολόγησης. Οι αναφορές είναι διαθέσιμες σε μορφές .txt και .md.

Η υλοποίηση πραγματοποιήθηκε σε γλώσσα Java, με χρήση του Maven ως εργαλείο διαχείρισης έργου και εξαρτήσεων. Το σύστημα έχει σχεδιαστεί με γνώμονα το modular design και την ευκολία επέκτασης.

Η εφαρμογή μπορεί να αποτελέσει χρήσιμο εργαλείο για την αυτόματη ή ημί-αυτόματη αξιολόγηση FAIR συνόλων δεδομένων στο πλαίσιο της επιστήμης δεδομένων.

Λέξεις Κλειδιά: FAIR, Ποιότητα δεδομένων, Έλεγχοι δεδομένων, Java, Maven.

Abstract

The aim of this thesis was the development of a software system for assessing the quality of datasets, based on the FAIR principles: Findable, Accessible, Interoperable and Reusable. The evaluation is performed both at the dataset level and at the level of individual fields (columns).

The system provides functionalities for registering CSV-formatted datasets, executing complete or targeted FAIR checks, and generating reports summarizing the results of the evaluation. The reports are available in .txt and .md formats.

The implementation was carried out in Java, using Maven for project and dependency management. The architecture of the system follows a modular and extensible design.

The application can serve as a useful tool for the automated or semi-automated FAIR evaluation of datasets, especially in the context of data science.

Keywords: FAIR, Data quality, Dataset evaluation, Java, Maven.

Πίνακας περιεχομένων

Κεφάλαιο 1. Εισαγωγή	1
1.1 Αντικείμενο της διπλωματικής	1
1.2 Οργάνωση του τόμου	2
Κεφάλαιο 2. Περιγραφή Θέματος	3
2.1 Στόχος της εργασίας	3
2.2 Αρχές FAIR	4
2.3 Οδηγίες ποιότητας δεδομένων της ΕΕ	5
2.3.1 Γενικές συστάσεις σχετικές με τις αρχές FAIR	6
2.3.2 Συστάσεις για συγκεκριμένες μορφές αρχείων	7
2.3.3 Συστάσεις για τη βελτίωση του επιπέδου διαφάνειας	11
2.4 Goldbook	12
2.4.1 Συλλογή δεδομένων	12
2.4.2 Προετοιμασία δεδομένων	13
2.4.3 Δημοσίευση δεδομένων	15
2.4.4 Διατήρηση δεδομένων	16
2.5 Ανάλυση απαιτήσεων	16
Κεφάλαιο 3. Σχεδίαση & Υλοποίηση	19
3.1 Use Cases	19
3.2 Ορισμός προβλήματος και αλγόριθμοι επίλυσης	22
3.2.1 Τυπικός ορισμός του προβλήματος	22
3.2.2 Περιγραφή των χρησιμοποιούμενων δομών δεδομένων	22
3.2.3 Αλγόριθμοι για την επίλυση του προβλήματος	23
3.3 Σχεδίαση και αρχιτεκτονική λογισμικού	25
3.3.1 Πακέτο app	26
3.3.2 Πακέτο config	26
3.3.3 Πακέτο engine	27
3.3.4 Πακέτο fairchecks.api	28

3.3.5	Πακέτο <i>fairchecks.globalChecks</i>	29
3.3.6	Πακέτο <i>fairchecks.columnChecks</i>	30
3.3.7	Πακέτο <i>fairchecks.factory</i>	31
3.3.8	Πακέτο <i>model</i>	31
3.3.9	Πακέτο <i>utils</i>	31
3.3.10	Πακέτο <i>report</i>	32
3.4	Σχεδίαση και αποτελέσματα ελέγχου του λογισμικού.....	32
3.5	Λεπτομέρειες εγκατάστασης και υλοποίησης.....	33
3.5.1	Οδηγίες εγκατάστασης και εκτέλεσης του <i>project</i> :.....	34
3.6	Επεκτασιμότητα του λογισμικού.....	36
Κεφάλαιο 4. Πειραματική Αξιολόγηση.....		37
4.1	Μεθοδολογία πειραματισμού.....	37
4.2	Αναλυτική παρουσίαση αποτελεσμάτων.....	37
4.2.1	<i>Dataset 1 – students_test.csv</i>	38
4.2.2	<i>Dataset 2 – students_test_wrong.csv</i>	38
4.2.3	<i>Dataset 3 – students_test_wrong2.csv</i>	39
4.2.4	<i>Dataset 4 – students_test_wrong3.csv</i>	39
4.2.5	<i>Dataset 5 – null_dataset.csv</i>	40
4.2.6	<i>Dataset 6 – no_access_test.csv</i>	40
4.2.7	<i>Dataset 7 – excel_test.xlsx and Dataset 8 – excel_test_wrong.xlsx</i>	40
4.2.8	<i>Dataset 9 – fruits_test.csv</i>	41
4.2.9	<i>Dataset 10 – fruits_test_wrong.csv</i>	42
4.2.10	<i>Dataset 11 – countries.csv</i>	43
Κεφάλαιο 5. Επίλογος.....		45
5.1	Σύνοψη και συμπεράσματα.....	45
5.2	Μελλοντικές επεκτάσεις.....	46
Κεφάλαιο 6. Βιβλιογραφία.....		47

Κεφάλαιο 1. Εισαγωγή

Το συγκεκριμένο κεφάλαιο αναφέρεται στην συνοπτική περιγραφή του αντικειμένου της διπλωματικής, και στην περιγραφή των επόμενων ενοτήτων.

1.1 Αντικείμενο της διπλωματικής

Η ραγδαία αύξηση της παραγωγής και δημοσίευσης συνόλων δεδομένων σε διάφορους επιστημονικούς και τεχνολογικούς τομείς έχει δημιουργήσει την ανάγκη για αξιολόγηση της ποιότητάς τους. Συνήθως, τα δεδομένα εκδίδονται χωρίς να γίνεται έλεγχος ποιότητας, γεγονός που οδηγεί σε προβλήματα όπως ελλιπή ή λανθασμένα δεδομένα, ή λάθος δομή σε σχέση με τα καθιερωμένα πρότυπα. Οι αρχές FAIR (Findable, Accessible, Interoperable, Reusable) αποτελούν θεμέλιο για την προώθηση την ανοικτής επιστήμης και της αξιοπιστίας της έρευνας, καθώς στοχεύουν στην ευκολία εντοπισμού, πρόσβασης, δια-λειτουργικότητας και επαναχρησιμοποίησης των ερευνητικών δεδομένων. Επομένως, η δυνατότητα συστηματικής και αυτοματοποιημένης αξιολόγησης βάσει των αρχών FAIR αποτελεί σημαντικό ζητούμενο στην ερευνητική κοινότητα.

Στόχος της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη ενός λογισμικού εργαλείου αξιολόγησης ποιότητας δεδομένων με ελέγχους βασισμένους στις αρχές FAIR. Το σύστημα επιτρέπει την καταχώρηση συνόλων δεδομένων σε μορφή CSV, την εκτέλεση συνολικών ή συγκεκριμένων FAIR ελέγχων, τόσο σε επίπεδο αρχείου όσο και σε επίπεδο πεδίων – στηλών, καθώς και την παραγωγή αναφορών με τα αποτελέσματα αξιολόγησης.

Το πρόβλημα της FAIR αξιολόγησης είναι δυσεπίλυτο και ιδιαίτερα απαιτητικό, καθώς οι ίδιες οι αρχές FAIR είναι διατυπωμένες με γενικό και συχνά περιγραφικό τρόπο, χωρίς να συνοδεύονται από αυστηρά μετρήσιμα κριτήρια. Η ανάπτυξη εργαλείων που μπορούν να αξιολογήσουν αυτόματα ή ημί-αυτόματα την ποιότητα δεδομένων σύμφωνα με αυτές τις αρχές απαιτεί συνδυασμό τεχνικών, ανάλυση δομής και περιεχομένου και δυνατότητα

προσαρμογής. Η παρούσα λύση διαφοροποιείται ως προς τον συνδυασμό modular σχεδίασης και πρακτικής ευελιξίας που προσφέρει στον χρήστη, όπως για παράδειγμα η εκτέλεση συγκεκριμένων ελέγχων σε επιλεγμένες στήλες.

Συνοπτικά, στην παρούσα εργασία αναπτύσσεται ένα εργαλείο που επιτρέπει την καταχώρηση και ανάγνωση συνόλων δεδομένων, την εκτέλεση ελέγχων FAIR τόσο σε επίπεδο συνόλου (global checks) όσο και σε επίπεδο συγκεκριμένων ελέγχων και στηλών (column checks) προς αξιολόγηση, καθώς και την παραγωγή αναφορών με τα αποτελέσματα της διαδικασίας.

Η αξιολόγηση της λειτουργικότητας του συστήματος πραγματοποιείται μέσω δοκιμών, με χρήση διαφορετικών dataset. Τα αποτελέσματα επιβεβαιώνουν την ορθή λειτουργία, ευελιξία και επεκτασιμότητα της εφαρμογής.

1.2 Οργάνωση του τόμου

Η συγκεκριμένη διπλωματική εργασία αποτελείται από 5 κεφάλαια τα οποία αναλύονται στις παρακάτω παραγράφους:

Στο κεφάλαιο 2 διατυπώνεται πιο αναλυτικά ο στόχος της συγκεκριμένης εργασίας, και παρατίθεται το θεωρητικό και τεχνολογικό υπόβαθρό της. Αναλύονται οι αρχές FAIR, και όχι μόνο, με βάση ερευνητικά άρθρα και γνωστούς οδηγούς για την αξιολόγηση της ποιότητας των δεδομένων. Επιπλέον, δίνεται η τελική ανάλυση των απαιτήσεων για την κατασκευή του λογισμικού.

Το κεφάλαιο 3 παρουσιάζει τη σχεδίαση και υλοποίηση του λογισμικού. Περιγράφονται τα use cases, η αρχιτεκτονική, οι αλγόριθμοι και τα διαγράμματα UML, καθώς και η διαδικασία ελέγχου και εγκατάστασης.

Το κεφάλαιο 4 περιλαμβάνει την αξιολόγηση της εφαρμογής μέσω δοκιμών σε πληθώρα συνόλων δεδομένων, καθώς και την ανάλυση των αποτελεσμάτων από την χρήση των ελέγχων.

Το κεφάλαιο 5 συνοψίζει τα συμπεράσματα της εργασίας και προτείνει πιθανές κατευθύνσεις για μελλοντική επέκταση ή βελτίωση του συστήματος.

Τέλος, γίνεται αναφορά στη βιβλιογραφία, με άρθρα που χρησιμοποιήθηκαν για τη διεκπεραίωση της διπλωματικής αυτής εργασίας.

Κεφάλαιο 2. Περιγραφή Θέματος

2.1 Στόχος της εργασίας

Ο στόχος αυτής της διπλωματικής εργασίας είναι η υλοποίηση ενός εργαλείου δημοσίευσης δεδομένων που θα διευκολύνει τη διασφάλιση ποιότητας δεδομένων και θα επιτρέπει τη δημοσίευση δομημένων μεταδεδομένων.

Το πρόβλημα που αντιμετωπίζει αυτό το εργαλείο είναι η έλλειψη μιας αποτελεσματικής διαδικασίας για τη διασφάλιση της ποιότητας των δεδομένων πριν τη δημοσίευσή τους, κάτι που είναι σημαντικό σε τομείς όπως η επιστήμη δεδομένων ή οι κρατικοί χώροι φύλαξης δεδομένων. Συνήθως, τα δεδομένα εκδίδονται χωρίς να γίνεται έλεγχος ποιότητας ή χωρίς τα κατάλληλα μεταδεδομένα, γεγονός που οδηγεί σε προβλήματα όπως ελλιπή ή λανθασμένα δεδομένα, ή λάθος δομή σε σχέση με τα καθιερωμένα πρότυπα.

Το εργαλείο θα λειτουργεί ως εξής:

- Ο χρήστης θα φορτώνει στο σύστημα ένα σύνολο δεδομένων.
- Το σύστημα θα εκτελεί ελέγχους ως προς την ποιότητα του συνόλου αυτού, κατά τους οποίους ο χρήστης θα μπορεί να εκτελεί και να αποθηκεύει σταδιακά τις δοκιμές του.
- Το σύστημα θα δημιουργεί μεταδεδομένα και λεπτομερείς αναφορές που θα εξηγούν τι πήγε καλά και ποια τα σφάλματα στους παραπάνω ελέγχους.

Ένα παράδειγμα κατά το οποίο φαίνεται η χρησιμότητα του εργαλείου είναι το εξής:

Ας υποθέσουμε ότι μια ομάδα ερευνητών προετοιμάζει ένα σύνολο δεδομένων για δημοσίευση. Χωρίς το κατάλληλο εργαλείο, η διαδικασία θα αποβεί χρονοβόρα, καθώς θα χρειαστεί να καθαρίσουν και να επαληθεύσουν χειροκίνητα το σύνολο δεδομένων, αλλά και να δημιουργήσουν τα μεταδεδομένα χειροκίνητα. Εκτός από χρονοβόρο, το αποτέλεσμα μπορεί να οδηγήσει και σε σφάλματα. Έτσι, η ομάδα μπορεί να καταλήξει να

δημοσιεύσει δεδομένα με μη εμφανή σφάλματα και τα μεταδεδομένα μπορεί να είναι ελλιπή ή ανακριβή. Το συγκεκριμένο, όμως, εργαλείο θα αυτοματοποιούσε μεγάλο μέρος αυτής της διαδικασίας, καθώς θα παρείχε ήδη τους κανόνες διασφάλισης της ποιότητας δεδομένων, και θα δημιουργούσε τα μεταδεδομένα αυτόματα.

2.2 Αρχές FAIR

Ένα σημαντικό πλαίσιο σχετικό με αυτή τη διπλωματική είναι οι κατευθυντήριες αρχές FAIR για τη διαχείριση δεδομένων. Σύμφωνα με το [WDA+16], τα δεδομένα πρέπει να είναι ευρέσιμα, προσβάσιμα, δια-λειτουργικά και επαναχρησιμοποιήσιμα (Findable, Accessible, Interoperable, and Reusable). Το FAIR είναι ένα σύνολο αρχών που ορίζουν βασικά χαρακτηριστικά και ιδιότητες που πρέπει να διαθέτουν τα δεδομένα, και κυρίως τα μεταδεδομένα τους. Πρωταρχικός σκοπός των αρχών FAIR είναι η ικανότητα χρήσης των δεδομένων από τις μηχανές με αυτόματο τρόπο, χωρίς δηλαδή να χρειάζεται η υποστήριξη από ανθρώπινο χέρι. Πιο συγκεκριμένα, σύμφωνα με τους συγγραφείς, ένα κατάλληλο λογισμικό πρέπει να μπορεί να βρει τον τύπο του ψηφιακού αντικειμένου, αν αυτό είναι χρήσιμο για την εκάστοτε εργασία, αν έχει περιορισμούς ή αν είναι προσβάσιμο, και να λάβει τα κατάλληλα μέτρα, όπως θα έκανε ένας άνθρωπος.

Οι αρχές FAIR είναι οι εξής [WDA+16]:

- Εύρεση
 - Τα (μετα)δεδομένα έχουν ένα καθολικά μοναδικό και μόνιμο αναγνωριστικό.
 - Τα δεδομένα περιγράφονται με πλούσια μεταδεδομένα.
 - Τα μεταδεδομένα περιλαμβάνουν καθολικά και ρητά το αναγνωριστικό των δεδομένων που περιγράφουν.
 - Τα (μετα)δεδομένα καταχωρούνται ή ευρετηριάζονται με δυνατότητα αναζήτησης.
- Πρόσβαση
 - Τα (μετα)δεδομένα μπορούν να ανακτηθούν μέσω του αναγνωριστικού τους με χρήση τυποποιημένου πρωτοκόλλου επικοινωνιών.
 - Το πρωτόκολλο είναι ανοιχτό, ελεύθερο και καθολικά εφαρμόσιμο.
 - Το πρωτόκολλο επιτρέπει μια διαδικασία ελέγχου ταυτότητας και εξουσιοδότησης, όπου αυτή είναι απαραίτητη.
 - Τα μεταδεδομένα είναι προσβάσιμα, ακόμη και όταν τα δεδομένα δεν είναι πλέον διαθέσιμα.

- Δια-λειτουργικότητα
 - Τα (μετα)δεδομένα χρησιμοποιούν μια επίσημη, προσβάσιμη, κοινή και ευρέως εφαρμόσιμη γλώσσα για την αναπαράσταση της γνώσης.
 - Τα (μετα)δεδομένα χρησιμοποιούν λεξιλόγια που ακολουθούν τις αρχές FAIR.
 - Τα (μετα)δεδομένα περιλαμβάνουν ειδικές αναφορές σε άλλα (μετα)δεδομένα.
- Επαναχρησιμοποίηση
 - Τα (μετα)δεδομένα περιγράφονται με πληθώρα χαρακτηριστικών, που είναι ακριβή και σχετικά.
 - Τα (μετα)δεδομένα δημοσιεύονται με σαφή και προσβάσιμη άδεια χρήσης.
 - Τα (μετα)δεδομένα σχετίζονται λεπτομερώς με την προέλευση,
 - Τα (μετα)δεδομένα πληρούν τα πρότυπα της κοινότητας που σχετίζονται με τον τομέα.

Στο [WDA+16] παρατίθενται και κάποια παραδείγματα εφαρμογής των αρχών FAIR, καθώς και το αποτέλεσμα που είχε η χρήση τους. Ένα από αυτά είναι το Dataverse, που είναι διαθέσιμο στο <https://www.dataverse.gr/>, και αποτελεί ένα λογισμικό αποθετήριο δεδομένων ανοιχτού κώδικα για την υποστήριξη δημόσιων αποθετηρίων κοινότητας ή αποθετηρίων δεδομένων θεσμικής έρευνας. Το Dataverse είναι εγκατεστημένο σε δεκάδες ιδρύματα παγκοσμίως, με το Harvard να αποτελεί το μεγαλύτερο από όλα και το οποίο είναι ανοιχτό σε όλους τους ερευνητές από όλα τα ερευνητικά πεδία. Το Dataverse τηρεί τις αρχές FAIR έχοντας δημόσιο το αναγνωριστικό του ψηφιακού αντικειμένου, παρέχοντας πρόσβαση σε (μετα)δεδομένα, άδειες χρήσης ή πληροφορίες έκδοσης αυτών και την ικανότητα αναζήτησης τους, περιλαμβάνοντας συμπληρωματικά αρχεία για την ανάλυση και κατανόηση δεδομένων, και, τέλος, παρέχοντας δημόσιες διεπαφές προσβάσιμες από μηχανήματα. Με παρόμοιο τρόπο, και οι άλλες πλατφόρμες που αναφέρονται από τους συγγραφείς (FAIRDOM, ISA, Open PHACTS, wwPDB, UniProt) χρησιμοποιούν τις παραπάνω αρχές.

2.3 Οδηγίες ποιότητας δεδομένων της ΕΕ

Το 2019 η Υπηρεσία Εκδόσεων της Ευρωπαϊκής Ένωσης δρομολόγησε το έργο «Κατευθυντήριες γραμμές για την ποιότητα των δεδομένων σχετικά με τη δημοσίευση συνόλων δεδομένων στην πύλη δημόσιων δεδομένων της ΕΕ [FF19]. Το έργο

αποτελούνταν από τρία μέρη: α) Δημιουργία προφίλ δεδομένων, β) δείκτες ποιότητας δεδομένων και μετρήσεις, γ) συστάσεις για την παροχή δεδομένων υψηλής ποιότητας.

Το τρίτο μέρος, δηλαδή τις συστάσεις για εξασφάλιση της ποιότητας των δεδομένων, τις παρέχει αναλυτικά ο οδηγός που δημιούργησε η Ευρωπαϊκή Ένωση το 2021 [POEU21].

2.3.1 Γενικές συστάσεις σχετικές με τις αρχές FAIR

Αρχικά, στον οδηγό αυτόν, αναφέρονται γενικές συστάσεις που αφορούν τη δημιουργία ενός σχεδίου διαχείρισης δεδομένων (Data Management Plan - DMP). Χρήσιμες για το σχέδιο αυτό είναι οι αρχές FAIR που αναφέρθηκαν παραπάνω, και που το γραφείο έκδοσης της ΕΕ αναλύει αναλυτικά και με παραδείγματα σε σχέση με τον τρόπο επίτευξής τους.

- Εύρεση
 - Τα δεδομένα περιγράφονται με μεταδεδομένα για βελτίωση εντοπισμού τους (τίτλος, περιγραφή, λέξεις-κλειδιά, κατηγορίες, χρονικές πληροφορίες, χωρικές πληροφορίες).
 - Τα δεδομένα με μηδενική (null) τιμή να επισημαίνονται ως τέτοια.
- Πρόσβαση
 - Τα δεδομένα να δημοσιεύονται χωρίς περιορισμούς πρόσβασης.
 - Τα δεδομένα να παρέχονται από μία προσβάσιμη διεύθυνση URL λήψης.
- Δια-λειτουργικότητα
 - Τα (μετα)δεδομένα να έχουν συγκεκριμένη μορφοποίηση ημερομηνίας και ώρας. Η μορφή πρέπει να είναι κωδικοποιημένη ως ISO 8601 (YYYY-MM-DD hh:mm:ss). Να χρησιμοποιείται η ζώνη ώρας και να προέρχεται από την συντονισμένη παγκόσμια ώρα (UTC).
 - Τα δεδομένα να έχουν συγκεκριμένη μορφοποίηση δεκαδικών αριθμών και αριθμών σε χιλιάδες.
 - Οι ακέραιοι αριθμοί διαχωρίζονται από τους δεκαδικούς με τελεία “.”
 - Οι αριθμοί σε χιλιάδες δεν πρέπει να χρησιμοποιούν διαχωριστές.
 - Τα δεδομένα να χρησιμοποιούν τυποποιημένη κωδικοποίηση χαρακτήρων.
 - Η κωδικοποίηση επιλογής στον ιστό είναι UTF-8, και πρέπει να ενεργοποιηθεί ρητά στο ‘Save As’.
 - Τα δεδομένα να μην αποτελούνται από ειδικούς χαρακτήρες.

- Επαναχρησιμοποίηση
 - Τα δεδομένα που δημοσιεύονται να είναι σε επαρκή ποσότητα και χρήσιμα.
 - Τα μεταδεδομένα να τηρούν τα πρότυπα της κοινότητας.
 - Τα δεδομένα να μην είναι διπλότυπα.
 - Τα δεδομένα να είναι ακριβή.
 - Τα μεταδεδομένα να έχουν πληροφορίες σχετικές με το μέγεθος τους σε byte.

2.3.2 Συστάσεις για συγκεκριμένες μορφές αρχείων

Οι συγγραφείς του [POEU21] περιλαμβάνουν επιπλέον στον οδηγό και συστάσεις για τις μορφές αρχείων CSV, XML, RDF, JSON και API. Οι πληροφορίες που μας παρέχουν αφορούν: α) τη μορφή δεδομένων, β) το λεξιλόγιο, και γ) τη δομή και τη σημασιολογία.

2.3.2.1 Μορφή δεδομένων

Αναλυτικά για κάθε είδος αρχείου:

- CSV
 - Τα δεδομένα να χρησιμοποιούν το ερωτηματικό, και όχι το κόμμα, ως διαχωριστή μεταξύ των τιμών, χωρίς κενά ή tabs σε καμία πλευρά.
 - Κάθε αρχείο δεδομένων να περιέχει έναν μόνο πίνακα.
 - Τα αρχεία δεδομένων να μην έχουν κενά διαστήματα και πρόσθετες πληροφορίες.
 - Τα δεδομένα να μην περιέχουν επεξηγήσεις, ημερομηνίες, τροποποιήσεις, τα ονόματα φύλλων κλπ. Αυτά θα τα περιέχουν τα μεταδεδομένα.
 - Ο τίτλος να αντιπροσωπεύεται μέσα στο όνομα της διανομής, και όχι να περιέχεται στο πραγματικό αρχείο.
 - Τα αρχεία να περιέχουν μόνο μία γραμμή κεφαλίδας.
 - Το κάθε αρχείο να περιέχει μόνο ένα φύλλο.
 - Τα αρχεία δεδομένων να περιέχουν κεφαλίδες στηλών, ως την πρώτη γραμμή τους.
 - Τα αρχεία δεδομένων να έχουν τον ίδιο αριθμό στηλών για όλες τις γραμμές.
 - Η μονάδα μιας τιμής δεδομένων να υποδεικνύεται με εύκολα επεξεργάσιμο τρόπο.

- Οι αριθμητικές τιμές να ακολουθούν τις γενικές συστάσεις από την υπό-ενότητα 2.3.1.
 - Η μονάδα μιας τιμής να δηλώνεται στη σχετική κεφαλίδα στήλης.
 - Τα (μετα)δεδομένα να αναφέρουν τη μονάδα μέτρησης που χρησιμοποιείται στα αντίστοιχα δεδομένα.
 - Εάν η μονάδα ποικίλλει, θα πρέπει να τα αρχεία να έχουν ειδική στήλη για αυτή.
- XML
 - Τα αρχεία να περιέχουν μία πλήρη δήλωση XML.
 - Τα αρχεία να περιέχουν μεταδεδομένα σχετικά με τη δομή του εγγράφου.
 - Τα αρχεία να περιέχουν πληροφορίες σχετικά με την έκδοση XML.
 - Τα αρχεία να περιέχουν την κωδικοποίηση χαρακτήρων.
 - Τα αρχεία να μην περιέχουν ειδικούς χαρακτήρες, αλλά να χρησιμοποιούνται οι αντίστοιχες οντότητες XML.
 - Τα αναγνωριστικά είναι μοναδικά, έχουν όνομα με νόημα, ίδια μορφή μεταξύ τους και δεν περιέχουν ειδικούς χαρακτήρες.
 - Τα δεδομένα να αντιπροσωπεύονται από στοιχεία και τα μεταδεδομένα να υλοποιούνται ως χαρακτηριστικά.
 - Τα αρχεία να μην εξαρτώνται από συγκεκριμένα προγράμματα ή εργαλεία που χρησιμοποιούνται για την επεξεργασία τους.
- RDF
 - Τα αναγνωριστικά πόρων πρέπει να είναι HTTP URIs.
 - Τα αρχεία να περιέχουν namespaces, όπου είναι δυνατόν, γραμμένα σε PascalCase.
 - Τα αρχεία να χρησιμοποιούν υπάρχοντα λεξιλόγια όποτε είναι δυνατόν.
- JSON
 - Τα δεδομένα να είναι κατάλληλου τύπου.
 - Τα δεδομένα με μηδενική τιμή να δηλώνονται με τη λέξη “null”.
 - Τα δεδομένα με δυαδική τιμή, να δηλώνονται είτε true είτε false.
 - Τα δεδομένα συμβολοσειρών δηλώνονται όπως σε ένα CSV αρχείο.
 - Τα δεδομένα που είναι αριθμοί ή απλές ακολουθίες, αποτελούνται από τα ψηφία 0-9, και προαιρετικά με σύμβολο πρόσημου ή/και υποδιαστολή.

- Τα δεδομένα που αποτελούν λίστες, ή αλλιώς πίνακες, περικλείονται σε αγκύλες “[]”, και τα στοιχεία που περιέχουν χωρίζονται με κόμμα “,”. Οι λίστες μπορεί να είναι και κενές.
- Τα δεδομένα που αποτελούν αντικείμενα, περικλείονται σε αγκύλες “{ }”, και περιέχουν οποιονδήποτε αριθμό ζευγών κλειδιού-τιμής διαχωρισμένα με κόμμα “,”.
- Τα δεδομένα να ομαδοποιούνται με χρήση ιεραρχιών.
- Τα δεδομένα να κωδικοποιούνται σε πίνακες μόνο εάν το μέγεθος της λίστας είναι δυναμικό.
- API
 - Το πρότυπο HTTP, εκτός από το πραγματικό ωφέλιμο φορτίο, να στέλνει και τον σωστό κωδικό κατάστασης για τα αιτήματα των πελατών.
 - Τα μεταδεδομένα να έχουν τις κατάλληλες κεφαλίδες. Τα κωδικοποιημένα δεδομένα πρέπει να είναι ακριβή και να ταιριάζουν με το ωφέλιμο φορτίο.
 - Τα δεδομένα που είναι πολλά να σελιδοποιούνται, με χρήση παραμέτρων μετατόπισης και ορίων.
 - Τα APIs να είναι τεκμηριωμένα, δηλαδή να περιλαμβάνουν διαθέσιμες διαδρομές, μορφές που επιστρέφονται, και κωδικούς κατάστασης.
 - Τα APIs θα πρέπει να έχουν το αναμενόμενο ωφέλιμο φορτίο, εάν επιτρέπουν τη μεταφόρτωση αρχείων.

2.3.2.2 Λεξιλόγιο

Στη συνέχεια, οι συγγραφείς παρατήρησαν ότι αυξάνεται η πληθώρα δεδομένων στο διαδίκτυο, και γι’ αυτό χρειάζεται να ληφθούν υπόψιν συστάσεις για την τυποποίηση των δεδομένων, με λεξιλόγια ελεγχόμενα από την ΕΕ, και για τον εμπλουτισμό των δεδομένων. Αυτό επιτυγχάνεται με τους εξής τρόπους, οι οποίοι ισχύουν εξίσου για όλα τα είδη αρχείων που αναφέρθηκαν παραπάνω:

- Στα δεδομένα να επαναχρησιμοποιούνται σαφές έννοιες από ελεγχόμενα λεξιλόγια RDF, όπως κατάλογοι αρχών, ταξινομίες, ταξινομήσεις ή ορολογίες. Η Υπηρεσία Εκδόσεων (Publication Office) διατηρεί ορισμένα λεξιλόγια τις ΕΕ και πίνακες αρχών που χρησιμοποιούνται στο <https://op.europa.eu/en/web/eu-vocabularies> για την τυποποίηση μεταδεδομένων.
- Οι ετικέτες των δεδομένων να αναφέρονται με μοναδικά αναγνωριστικά, δηλαδή URI, ώστε αν αυτές αλλάξουν, η αναφορά να μην χρειάζεται να προσαρμοστεί, επιτυγχάνοντας, έτσι, εναρμόνιση πινάκων.

- Μόλις οι ετικέτες υποδεικνύονται από τα μοναδικά αναγνωριστικά από τα ελεγχόμενα λεξιλόγια, οι λεπτομέρειες του URI να ανακτώνται και στα δεδομένα.
- Τα δεδομένα να αξιοποιούν τα URI και να δημοσιεύονται ως Διασυνδεδεμένα Δεδομένα (Linked Data) με τη μορφή σημασιολογικών γραμμάτων.

2.3.2.3 Δομή και σημασιολογία

Εκτός από την τυποποιημένη μορφή των δεδομένων, στο άρθρο [POEU21], τυποποιημένη θεώρησαν οι συγγραφείς ότι πρέπει να είναι και η μορφή των αρχείων τεκμηρίων (documentation) που επίσης θα δημοσιεύονται. Τα documentations είναι αρχεία που περιέχουν λεπτομερείς πληροφορίες σχετικές με τα δεδομένα, ώστε να γίνονται κατανοητά, και να χρησιμοποιούνται εύκολα από άλλους. Η τεκμηρίωση περιλαμβάνει: α) τα μεταδεδομένα, δηλαδή την πηγή, τη δομή, τους ορισμούς μεταβλητών και το περιεχόμενο των δεδομένων, β) πρότυπα και μορφές, δηλαδή τον τύπο του αρχείου των δεδομένων, γ) προέλευση δεδομένων, περιλαμβάνοντας και τυχόν αλλαγές που έχουν υποστεί, και δ) απαραίτητες οδηγίες για τη χρήση, την ερμηνεία και τη επαναχρησιμοποίηση των δεδομένων. Πιο αναλυτικά:

- Το documentation να δημοσιεύεται πάντα μαζί με τα δεδομένα, ιδανικά σε ξεχωριστή διανομή, και αργότερα να συνδέονται μεταξύ τους. Αυτό ισχύει για όλους τους τύπους αρχείων.
- Τα documentations να περιέχουν σχήματα για τον καθορισμό της δομής δεδομένων τους.
 - Τα αρχεία JSON χρησιμοποιούν τη γλώσσα σχήματος "JSON Schema".
 - Τα αρχεία XML χρησιμοποιούν πολλαπλές γλώσσες σχήματος, όπως RELAX NG, Schematron και XSD (XML Schema Definition Language), με την τελευταία να είναι η καλύτερη επιλογή.
 - Τα αρχεία CSV έχουν δύο επιλογές για τη δομή: α) Frictionless Data που χρησιμοποιεί JSON αρχεία, και β) UK Nation Archives, με χρήση CSV γλώσσα σχήματος.
 - Τα αρχεία RDF χρησιμοποιούν οντολογίες για γραφήματα ή αλλιώς σχήματα, χρησιμοποιώντας τη SHACL γλώσσα.
 - Τα APIs, τα οποία δεν είναι αρχεία δεδομένων αλλά δεδομένα στον ιστό προσβάσιμα μέσω URL, πρέπει να τεκμηριώνουν και τη δομή αλλά και τον τρόπο πρόσβασης σε αυτά στο διαδίκτυο. Οι προδιαγραφές γράφονται είτε σε JSON είτε σε YAML. Θα πρέπει να προσδιοριστούν οι ακόλουθες πτυχές:
 - URLs και τελικά σημεία.

- Πρωτόκολλα τελικών σημείων π.χ., HTTP, FTP.
- Μέθοδοι πρόσβασης, π.χ., Μέθοδοι HTTP, κωδικοί κατάστασης.
- Τρόποι τροποποίησης αποτελεσμάτων, π.χ., παράμετροι ερωτήματος, κεφαλίδες HTTP.
- Τα documentations να περιέχουν τη σημασιολογία δεδομένων, δηλαδή την περιγραφή επιμέρους ιδιοτήτων μιας δομής δεδομένων.
- Τα documentations να περιέχουν αλλαγές που έχουν υποστεί τα δεδομένα.
 - Ο αναλυτής να υιοθετεί μια συγκεκριμένη πολιτική δημοσίευσης του συνόλου δεδομένων.
 - Στα documentations να γράφεται εάν η νέα έκδοση των δεδομένων αποτελεί μεγάλη ή μικρή αλλαγή στο αρχικό σύνολο.
 - Τα documentations να περιέχουν τον αριθμό έκδοσης των δεδομένων. Σύμφωνα με τη σημασιολογική διαχείριση εκδόσεων πρέπει η έκδοση να αποτελείται από τρία ψηφία, χωρισμένα με τελείες, όπου το πρώτο θα αναφέρει μεγάλη αλλαγή, το δεύτερο μικρή αλλαγή και το τρίτο την ενημέρωση του κώδικα. Άλλη μέθοδος διαχείρισης εκδόσεων είναι η χρήση αναγνωριστικών ψηφιακών αντικειμένων (Digital Object Identifier - DOI), δηλαδή μία νέα DOI εκχωρείται για κάθε έκδοση ενός εγγράφου.
 - Τα documentations να περιέχουν μια αναλυτική περιγραφή με τις αλλαγές.
 - Για τα δεδομένα σε μορφή πίνακα, κάθε φύλλο θα πρέπει να δημοσιεύεται ως νέο σύνολο δεδομένων.
 - Παλιές εκδόσεις δεδομένων θα πρέπει να ορίζονται ως καταργημένες.
 - Νέες εκδόσεις δεδομένων θα πρέπει να συνδέονται με τις αντίστοιχες παλιές - αρχικές.

2.3.3 Συστάσεις για τη βελτίωση του επιπέδου διαφάνειας

Τέλος, στο [POEU21] αναλύονται και συστάσεις για τη βελτίωση του επιπέδου διαφάνειας (openness level), όσον αφορά τις μορφές αρχείων, με ιδιαίτερη έμφαση στη φάση δημοσίευσης της διαδικασίας προετοιμασίας δεδομένων.

Το μοντέλο πέντε αστέρων (Five-Star Model) [HK15], που αναπτύχθηκε το 2001 από τον Tim Berners-Lee, αποτελεί μια αξιοσημείωτη προσπάθεια κλιμάκωσης της διαφάνειας. Σύμφωνα με το μοντέλο αυτό, τα δεδομένα μπορούν να πετύχουν το πολύ πέντε αστέρια, υποδεικνύοντας το υψηλότερο επίπεδο διαφάνειας. Οι τάξεις είναι κλιμακωτές και κάθε

αρχείο καταφέρνει να αποκτήσει ένα αστέρι αν πληροί συγκεκριμένες προϋποθέσεις. Πιο συγκεκριμένα, για να απονεμηθεί κάθε αστέρι χρειάζονται τα εξής:

- Πρώτο αστέρι: Τα δεδομένα έχουν ανοιχτή άδειας χρήσης.
- Δεύτερο αστέρι: Η επιλεγμένη μορφή αρχείου πρέπει να είναι(ημί)δομημένη. Οι προτεινόμενες μορφές αρχείων είναι RDF, XML, JSON και CSV.
- Τρίτο αστέρι: Τα δεδομένα να μη χρησιμοποιούν ιδιόκτητες μορφές (proprietary formats).
- Τέταρτο αστέρι: Οι πόροι να χρησιμοποιούν URI ως αναγνωριστικά.
- Πέμπτο αστέρι: Τα δεδομένα να δημοσιεύονται ως Διασυνδεδεμένα Δεδομένα (Linked Data) με τη μορφή σημασιολογικών γραμμάτων.

2.4 Goldbook

Το 2018 δημοσιεύθηκε από την συμβουλευτική του Cargemini ένας χρυσός οδηγός (Gold book) που αφορά τα Open Data, δηλαδή τα δεδομένα που είναι προσβάσιμα σε όλους, μπορούν να τα επεξεργαστούν, να τα επαναχρησιμοποιήσουν και να αναδιανεμηθούν από οποιονδήποτε, για οποιονδήποτε σκοπό [CaCo18]. Στον οδηγό αυτό, αρχικά, προτείνεται η δημιουργία μιας στρατηγικής ανοικτών δεδομένων, η οποία θα οργανώσει τους στόχους και τον σκοπό δημοσίευσης των δεδομένων. Το επόμενο και σημαντικό βήμα, σύμφωνα με τους συγγραφείς, είναι η τεχνική προετοιμασία και εφαρμογή, που περιλαμβάνει τη συλλογή και την προετοιμασία, τη δημοσίευση και τη διατήρηση των δεδομένων.

2.4.1 Συλλογή δεδομένων

Πιο αναλυτικά, για τη συλλογή δεδομένων, οι συγγραφείς προτείνουν μια σειρά βημάτων.

- Ο αναλυτής θα πρέπει να δημιουργήσει μια επισκόπηση των δεδομένων που είναι ήδη διαθέσιμα.
- Ο αναλυτής θα πρέπει να ιεραρχήσει τα σύνολα δεδομένων με βάση την ετοιμότητα τους για δημοσίευση. Τα κριτήρια είναι τα εξής:
 - Τα δεδομένα να μπορούν να δημοσιευθούν, χωρίς να έχουν νομικούς, πολιτικούς ή οργανωτικούς περιορισμούς.
 - Τα δεδομένα να είναι σωστής ποιότητας, ώστε να μην χρειάζεται διεξοδικός χειρισμός πριν τη δημοσίευση.
 - Τα δεδομένα να είναι καθαρά, καλής ποιότητας, σωστής μορφής και τα προσωπικά δεδομένα να προστατεύονται.

- Τα δεδομένα να ανήκουν σε ένα από τα θέματα υψηλής αξίας, τα οποία είναι: α) Γεωχωρικά, β) παρατήρηση γης, γ) μετεωρολογικά, δ) στατιστική, ε) εταιρίες και ιδιοκτησία εταιριών, και στ) κινητικότητα. [EU24]
- Ο αναλυτής θα πρέπει να εξασκηθεί. Τα βήματα σύμφωνα με το εγχειρίδιο βέλτιστης πρακτικής της Ιρλανδίας (Irish Best Practice Handbook) είναι:
 - Ο αναλυτής να κάνει έλεγχο των δεδομένων που διαχειρίζεται επί του παρόντος.
 - Οι πληροφορίες για κάθε σύνολο δεδομένων θα πρέπει να περιλαμβάνουν:
 - Δυνατότητα δημοσίευσης ως ανοικτά δεδομένα.
 - Νομικές πληροφορίες.
 - Οργανωτικές πληροφορίες.
 - Τεχνικές πληροφορίες.
 - Εκτίμηση αξίας: α) Τα δεδομένα να χαρακτηρίζονται ως 'υψηλής αξίας', β) τα αποτελέσματα ελέγχου να είναι διαθέσιμα στην εθνική πύλη ανοικτών δεδομένων (Open Data Portal).
- Τα δεδομένα να τηρούν τον Χάρτη Ανοικτών Δεδομένων της G8 και τις κατευθυντήριες γραμμές της Ευρωπαϊκής Επιτροπής, οι οποίες αναλύθηκαν ήδη στο κεφάλαιο 2.3. Όσον αφορά τον χάρτη, χωρίζει τα δεδομένα σε 14 κατηγορίες: α) Εταιρίες, β) Έγκλημα και Δικαιοσύνη, γ) Παρατήρηση γης, δ) Εκπαίδευση, ε) Ενέργεια και Περιβάλλον, στ) Χρηματοδότηση και συμβάσεις, ζ) Γεωχωρικά, η) Παγκόσμια Ανάπτυξη, θ) Κυβερνητική Λογοδοσία και Δημοκρατία, ι) Υγεία, κ) Επιστήμη και Έρευνα, λ) Στατιστική, μ) Κοινωνική κινητικότητα και πρόνοια, και ν) Μεταφορές και Υποδομές.
- Ο αναλυτής δημοσιεύει τα δεδομένα σε συγκεκριμένη κατηγορία. Η κατηγορία μπορεί να είναι κάποια από τις παραπάνω.

2.4.2 Προετοιμασία δεδομένων

Δεδομένου ότι τα ανεπεξέργαστα δεδομένα δεν είναι χρήσιμα για δημοσίευση, οι συγγραφείς παραθέτουν ένα σύνολο ενεργειών για την προετοιμασία των δεδομένων. Η προετοιμασία αυτή αποτελείται από τέσσερεις κατηγορίες, για τις οποίες δίνουν συγκεκριμένες οδηγίες για την επίτευξή τους. Οι κατηγορίες αυτές είναι: α) Ποιότητα, β) Τεχνική διαφάνεια, γ) Μεταδεδομένα και δ) Νομική διαφάνεια.

Αρχικά, για την ποιότητα των δεδομένων το [CaCo18] δίνει τρεις πτυχές: α) Ποιότητα περιεχομένου, β) διαχρονικότητα, και γ) συνέπεια. Πιο αναλυτικά:

- Ποιότητα περιεχομένου
 - Το σύνολο δεδομένων να είναι πλήρες.
 - Τα δεδομένα να περιέχουν μια κεφαλίδα με μία μόνο περιγραφή του τι εμφανίζεται. Στα μεταδεδομένα πρέπει να περιγράφεται αυτή η κεφαλίδα.
 - Τα δεδομένα να έχουν ετικέτα με τον αριθμό έκδοσης τους.
 - Τα δεδομένα να περιέχουν πληροφορίες σχετικές με την προέλευσή τους, δηλαδή ποια είναι, από πού προέρχονται και γιατί έχουν δημοσιευτεί.
 - Τα δεδομένα να περιέχουν την κατάσταση στην οποία βρίσκονται, δηλαδή προσχέδιο, επικυρωμένο ή τελικό στάδιο.
 - Το σύνολο δεδομένων να είναι καθαρό.
 - Τα δεδομένα να μην έχουν κενά πεδία.
 - Τα δεδομένα να έχουν σωστές προεπιλεγμένες τιμές.
 - Τα δεδομένα να μην έχουν λάθος τιμές.
 - Τα δεδομένα να μην είναι διπλότυπα.
 - Τα δεδομένα να μην περιέχουν ευαίσθητες πληροφορίες απορρήτου.
 - Το σύνολο δεδομένων να είναι ακριβές.
 - Τα δεδομένα να είναι αρκετά ακριβή για τον πιθανό σκοπό τους.
 - Η ακρίβεια των δεδομένων να μην επηρεάζει την αξιοπιστία τους.
 - Τα δεδομένα να περιγράφουν τις επιλογές του χρονικού διαστήματος συλλογής των δεδομένων.
 - Τα δεδομένα να μη χρειάζονται συνάθροιση ή επιμερισμό.
- Διαχρονικότητα
 - Τα δεδομένα να περιέχουν μια έννοια της επικαιρότητάς τους ώστε να διατηρούνται ενημερωμένα μετά από μία διαδικασία ενημέρωσης.
- Συνέπεια
 - Η συνέπεια αφορά τον τρόπο που τα δεδομένα δημοσιεύονται σε μεταγενέστερες εκδόσεις με την πάροδο του χρόνου. Για παράδειγμα, εάν αλλάξουν τα ονόματα πεδίων των δεδομένων που συλλέγονται για μετρήσεις κάθε χρόνο, τα δεδομένα δεν μπορούν να συγκεντρωθούν από το ένα έτος στο άλλο.

Επόμενο στάδιο προετοιμασίας των δεδομένων είναι η τεχνική διαφάνεια (technical openness). Αυτό επιτυγχάνεται μέσω των συνδεδεμένων δεδομένων (Linked Data), τα οποία αυξάνουν τη δια-λειτουργικότητα και τη δυνατότητα εντοπισμού των συνόλων δεδομένων. Τα συνδεδεμένα δεδομένα είναι κομμάτια πληροφοριών που συνδέονται μέσω γραφήματος, όπως για παράδειγμα οντολογίες και χρησιμοποιούν διάφορες τεχνικές, μεταξύ των οποίων RDF, λεξιλόγια και URI.

Άλλη σημαντική πτυχή της προετοιμασίας δεδομένων, σύμφωνα με το [CaCo18], είναι τα μεταδεδομένα, τα οποία περιγράφουν τα δεδομένα, και σκοπός τους είναι η εύκολη ανακάλυψη των δεδομένων, η κατανόηση περιβάλλοντος και δομής δεδομένων, η εύρεση προέλευσης δεδομένων, και πώς μπορούν αυτά να χρησιμοποιηθούν. Οι βέλτιστες πρακτικές μεταδεδομένων είναι οι εξής:

- Τα μεταδεδομένα να δημοσιεύονται μαζί με τα δεδομένα.
- Τα μεταδεδομένα να γράφονται σε μορφή που να μπορεί να διαβάσει η μηχανή.
- Τα μεταδεδομένα να προσδιορίζονται από τυπικούς όρους.
- Τα μεταδεδομένα να περιγράφουν τα χαρακτηριστικά των δεδομένων.
- Τα μεταδεδομένα να παρέχουν πληροφορίες για τοπικές μεταβλητές, όπως ημερομηνία, ώρα, τύπους τιμών, γλώσσα.
- Τα μεταδεδομένα να παρέχουν πληροφορίες για άδειες, και πληροφορίες σχετικές με την προέλευση των δεδομένων.
- Τα μεταδεδομένα να έχουν πληροφορίες σχετικές με την ποιότητα των δεδομένων.

Όσον αφορά τη δημοσίευση των μεταδεδομένων, οι συγγραφείς προτείνουν το μοντέλο πέντε αστέρων που αναφέρθηκε και από την Ευρωπαϊκή Ένωση [POEU21] στην ενότητα 2.3.3.

Σύμφωνα με το άρθρο [CaCo18], τελευταίο βήμα για την ποιότητα δεδομένων είναι η νομική διαφάνεια. Εάν τα δεδομένα δεν είναι νόμιμα ανοιχτά, τότε δεν έχουν νόμιμο δικαίωμα να επαναχρησιμοποιηθούν.

2.4.3 Δημοσίευση δεδομένων

Μετά το στάδιο προετοιμασίας των δεδομένων, είναι το στάδιο δημοσίευσης τους, σύμφωνα με το [CaCo18]. Οι τρόποι δημοσίευσης των δεδομένων είναι τρεις.

- Δημοσίευση ως αρχεία στο διαδίκτυο.
- Μεταφόρτωση των δεδομένων σε μια πύλη. Αποτελεί τον πιο χρησιμοποιούμενο τρόπο για τη δημοσίευση ανοικτών δεδομένων.
- Δημοσίευση μέσω API.

2.4.4 Διατήρηση δεδομένων

Το τελικό στάδιο αποτελείται από τη διατήρηση δεδομένων και μεταδεδομένων, τον έλεγχο URI και διευθύνσεων URL, τον έλεγχο σχολίων των χρηστών και τη συνεχή βελτίωση των δεδομένων.

2.5 Ανάλυση απαιτήσεων

Σύμφωνα με τις ενότητες 2.2 – 2.4 και τα άρθρα που αφορούν, προκύπτουν κάποιες απαιτήσεις σχετικές με τα δεδομένα και τα μεταδεδομένα πριν τη δημοσίευσή τους. Οι απαιτήσεις αυτές στηρίζονται στις αρχές FAIR. Για μεγαλύτερη κατανόηση και διευκόλυνση, η ομαδοποίηση εδώ γίνεται ως IRAF, και όχι ως FAIR. Αυτό συμβαίνει γιατί η σειρά των βημάτων ενός αναλυτή για να δημοσιεύσει ένα σύνολο δεδομένων είναι η εξής:

- Προετοιμασία:
 - Δεδομένων. Αυτή η κατηγορία ανήκει στην ομάδα I, δηλαδή στην διαλειτουργικότητα, γι' αυτό και αποτελεί την πρώτη ομάδα.
 - Μεταδεδομένων. Το βήμα αυτό αφορά την R, δηλαδή στην επαναχρησιμοποίηση.
- Δημοσίευση και Συντήρηση. Τέλος, γι' αυτήν την κατηγορία έχουμε τις ομάδες A, δηλαδή προσβασιμότητα, και F, δηλαδή εύρεση.

Αφού έχουμε κατηγοριοποιήσει τις οδηγίες βάσει της λειτουργίας τους, τις ομαδοποιούμε επίσης ανάλογα με το αν σχετίζονται με την ποιότητα των δεδομένων ή των μεταδεδομένων.

Ακολουθούν τέσσερις πίνακες που αφορούν τις δύο προαναφερθείσες ομαδοποιήσεις, σύμφωνα με το άρθρο [POEU21], δηλαδή σύμφωνα με τις οδηγίες της Ευρωπαϊκής Ένωσης.

	Data Quality	Metadata Quality
I Interoperable	IEU1: Τα δεδομένα να έχουν συγκεκριμένη μορφοποίηση ημερομηνίας και ώρας. Η μορφή πρέπει να είναι κωδικοποιημένη ως ISO 8601 (YYYY-MM-DD hh:mm:ss). Να χρησιμοποιείται η ζώνη ώρας και να προέρχεται από την συντονισμένη παγκόσμια ώρα (UTC). IEU2: Τα δεδομένα να έχουν συγκεκριμένη μορφοποίηση δεκαδικών αριθμών και αριθμών σε χιλιάδες. IEU2.1: Οι ακέραιοι αριθμοί διαχωρίζονται από τους δεκαδικούς με τελεία ".". IEU2.2: Οι αριθμοί σε χιλιάδες δεν πρέπει να χρησιμοποιούν διαχωριστές. IEU3: Τα δεδομένα να χρησιμοποιούν τυποποιημένη κωδικοποίηση χαρακτήρων. IEU3.1: Η κωδικοποίηση επιλογής στον ιστό είναι UTF-8, και πρέπει να ενεργοποιηθεί ρητά στο 'Save As'. IEU3.2: Τα δεδομένα να μην αποτελούνται από ειδικούς χαρακτήρες. IEU4: Στα δεδομένα να επαναχρησιμοποιούνται σαφείς έννοιες από ελεγχόμενα λεξιλόγια RDF, όπως κατάλογοι αρχών, ταξινομίες, ταξινομήσεις ή ορολογίες. Η Υπηρεσία Εκδόσεων (Publication Office) διατηρεί ορισμένα λεξιλόγια της ΕΕ και πίνακες αρχών που χρησιμοποιούνται στο https://op.europa.eu/en/web/eu-vocabularies για την τυποποίηση μεταδεδομένων. IEU5: Οι ετικέτες των δεδομένων να αναφέρονται με μοναδικά αναγνωριστικά, δηλαδή URI, ώστε αν αυτές αλλάξουν, η αναφορά να μην χρειάζεται να προσαρμοστεί, επιτυγχάνοντας, έτσι, εναρμόνιση πινάκων. IEU6: Μόλις οι ετικέτες υποδεικνύονται από τα μοναδικά αναγνωριστικά από τα ελεγχόμενα λεξιλόγια, οι λεπτομέρειες του URI να ανακτώνται και στα δεδομένα. IEU7: Τα δεδομένα να αξιοποιούν τα URI και να δημοσιεύονται ως Διασυνδεδεμένα Δεδομένα (Linked Data) με τη μορφή σημασιολογικών γραμμμάτων. IEU9: CSV - Τα δεδομένα να χρησιμοποιούν το ερωτηματικό, και όχι το κόμμα, ως διαχωριστή μεταξύ των τιμών, χωρίς κενά ή tabs σε καμία πλευρά. IEU10: CSV - Κάθε αρχείο δεδομένων να περιέχει έναν μόνο πίνακα. IEU11: CSV - Τα αρχεία δεδομένων να μην έχουν κενά διαστήματα και πρόσθετες πληροφορίες. IEU11.1: CSV - Τα δεδομένα να μην περιέχουν επεξηγήσεις, ημερομηνίες, τροποποιήσεις, τα ονόματα φύλλων κλπ. Αυτά θα τα περιέχουν τα μεταδεδομένα. IEU11.2: CSV - Ο τίτλος να αντιπροσωπεύεται μέσα στο όνομα της διανομής, και όχι να περιέχεται στο πραγματικό αρχείο. IEU11.3: CSV - Τα αρχεία να περιέχουν μόνο μία γραμμή κεφαλίδας. IEU11.4: CSV - Το κάθε αρχείο να περιέχει μόνο ένα φύλλο. IEU15: CSV - Τα αρχεία δεδομένων να περιέχουν κεφαλίδες στηλών, ως την πρώτη γραμμή τους. IEU16: CSV - Τα αρχεία δεδομένων να έχουν τον ίδιο αριθμό στηλών για όλες τις γραμμές. IEU17: CSV - Η μονάδα μιας τιμής δεδομένων να υποδεικνύεται με εύκολα επεξεργάσιμο τρόπο. IEU17.1: CSV - Η μονάδα μιας τιμής να δηλώνεται στη σχετική κεφαλίδα στήλης. IEU17.3: CSV - Εάν η μονάδα ποικίλλει, θα πρέπει τα αρχεία να έχουν ειδική στήλη για αυτή. IEU18: XML - Τα δεδομένα να αντιπροσωπεύονται από στοιχεία και τα μεταδεδομένα να υλοποιούνται ως χαρακτηριστικά. IEU19: XML - Τα αρχεία να μην εξαρτώνται από συγκεκριμένα προγράμματα ή εργαλεία που χρησιμοποιούνται για την επεξεργασία τους. IEU22: JSON - Τα δεδομένα να είναι κατάλληλου τύπου. IEU22.1: JSON - Τα δεδομένα με μηδενική τιμή να δηλώνονται με τη λέξη "null". IEU22.2: JSON - Τα δεδομένα με δυαδική τιμή, να δηλώνονται είτε true είτε false. IEU22.3: JSON - Τα δεδομένα συμβολοσειρών δηλώνονται όπως σε ένα CSV αρχείο. IEU22.4: JSON - Τα δεδομένα που είναι αριθμοί ή απλές ακολουθίες, αποτελούνται από τα ψηφία 0-9, και προαιρετικά με σύμβολο πρόσημου ή/και υποδιαστολή. IEU22.5: JSON - Τα δεδομένα που αποτελούν λίστες, ή αλλιώς πίνακες, περιλαμβάνονται σε αγκύλες "[]", και τα στοιχεία που περιέχουν χωρίζονται με κόμμα ",". Οι λίστες μπορεί να είναι και κενές. IEU22.6: JSON - Τα δεδομένα που αποτελούν αντικείμενα, περιλαμβάνονται σε αγκύλες "{ }", και περιέχουν οποιονδήποτε αριθμό ζευγών κλειδιού-τιμής διαχωρισμένα με κόμμα ",". IEU23 : JSON - Τα δεδομένα να ομαδοποιούνται με χρήση ιεραρχιών. IEU24: JSON - Τα δεδομένα να κωδικοποιούνται σε πίνακες μόνο εάν το μέγεθος της λίστας είναι δυναμικό.	IEU1: Τα μεταδεδομένα να έχουν συγκεκριμένη μορφοποίηση ημερομηνίας και ώρας. Η μορφή πρέπει να είναι κωδικοποιημένη ως ISO 8601 (YYYY-MM-DD hh:mm:ss). Να χρησιμοποιείται η ζώνη ώρας και να προέρχεται από την συντονισμένη παγκόσμια ώρα (UTC). IEU8: Το documentation να δημοσιεύεται πάντα μαζί με τα δεδομένα, ιδανικά σε ξεχωριστή διανομή, και αργότερα να συνδέονται μεταξύ τους. Αυτό ισχύει για όλους τους τύπους αρχείων. IEU17.2: CSV - Τα μεταδεδομένα να αναφέρουν τη μονάδα μέτρησης που χρησιμοποιείται στα αντίστοιχα δεδομένα. IEU18: XML - Τα δεδομένα να αντιπροσωπεύονται από στοιχεία και τα μεταδεδομένα να υλοποιούνται ως χαρακτηριστικά. IEU20: RDF - Τα αναγνωριστικά πόρων πρέπει να είναι HTTP URIs. IEU21: RDF - Τα αρχεία να χρησιμοποιούν υπάρχοντα λεξιλόγια όποτε είναι δυνατόν.

Σχήμα 2.1 Πίνακας Interoperable για Data και Metadata Quality.

	Data Quality	Metadata Quality
R Reusable	REU1: Τα δεδομένα που δημοσιεύονται να είναι σε επαρκή ποσότητα και χρήσιμα. REU3: Τα δεδομένα να μην είναι διπλότυπα. REU4: Τα δεδομένα να είναι ακριβή. REU7.5: Για τα δεδομένα σε μορφή πίνακα, κάθε φύλλο θα πρέπει να δημοσιεύεται ως νέο σύνολο δεδομένων. REU9: XML - Τα αρχεία να μην περιέχουν ειδικούς χαρακτήρες, αλλά να χρησιμοποιούνται οι αντίστοιχες οντότητες XML. REU10: XML - Τα αναγνωριστικά είναι μοναδικά, έχουν όνομα με νόημα, ίδια μορφή μεταξύ τους και δεν περιέχουν ειδικούς χαρακτήρες. REU13: API - Τα δεδομένα που είναι πολλά να σελιδοποιούνται, με χρήση παραμέτρων μετατόπισης και ορίων.	REU2: Τα μεταδεδομένα να τηρούν τα πρότυπα της κοινότητας. REU5: Τα μεταδεδομένα να έχουν πληροφορίες σχετικές με το μέγεθος τους σε BYTE. REU6: Τα documentations να περιέχουν τη σημασιολογία δεδομένων, δηλαδή την περιγραφή επιμέρους ιδιοτήτων μιας δομής δεδομένων. REU7: Τα documentations να περιέχουν αλλαγές που έχουν υποστεί τα δεδομένα. REU7.1: Ο αναλυτής να υιοθετεί μια συγκεκριμένη πολιτική δημοσίευσης του συνόλου δεδομένων. REU7.2: Στα documentations να γράφεται εάν η νέα έκδοση των δεδομένων αποτελεί μεγάλη ή μικρή αλλαγή στο αρχικό σύνολο. REU7.3: Τα documentations να περιέχουν τον αριθμό έκδοσης των δεδομένων. Σύμφωνα με τη σημασιολογική διαχείριση εκδόσεων πρέπει η έκδοση να αποτελείται από τρία ψηφία, χωρισμένα με τελείες, όπου το πρώτο θα αναφέρει μεγάλη αλλαγή, το δεύτερο μικρή αλλαγή και το τρίτο την ενημέρωση του κώδικα. Άλλη μέθοδος διαχείρισης εκδόσεων είναι η χρήση αναγνωριστικών ψηφιακών αντικειμένων (Digital Object Identifier - DOI), δηλαδή μία νέα DOI εκχωρείται για κάθε έκδοση ενός εγγράφου. REU7.4: Τα documentations να περιέχουν μια αναλυτική περιγραφή με τις αλλαγές. REU7.6: Παλιές εκδόσεις δεδομένων θα πρέπει να ορίζονται ως καταργημένες. REU7.7: Νέες εκδόσεις δεδομένων θα πρέπει να συνδέονται με τις αντίστοιχες παλιές - αρχικές. REU8: XML - Τα αρχεία να περιέχουν μία πλήρη δήλωση XML. REU8.1: XML - Τα αρχεία να περιέχουν μεταδεδομένα σχετικά με τη δομή του εγγράφου. REU8.2: XML - Τα αρχεία να περιέχουν πληροφορίες σχετικές με την έκδοση XML. REU8.3: XML - Τα αρχεία να περιέχουν την κωδικοποίηση χαρακτήρων. REU10: XML - Τα αναγνωριστικά είναι μοναδικά, έχουν όνομα με νόημα, ίδια μορφή μεταξύ τους και δεν περιέχουν ειδικούς χαρακτήρες. REU11: RDF - Τα αρχεία να περιέχουν namespaces, όπου είναι δυνατόν, γραμμένα σε PascalCase. REU12: API - Τα μεταδεδομένα να έχουν τις κατάλληλες κεφαλίδες. Τα κωδικοποιημένα δεδομένα πρέπει να είναι ακριβή και να ταιριάζουν με το ωφέλιμο φορτίο. REU14: API - Τα APIs να είναι τεκμηριωμένα, δηλαδή να περιλαμβάνουν διαθέσιμες διαδρομές, μορφές που επιστρέφονται, και κωδικούς κατάστασης. REU14.1: API - Τα APIs θα πρέπει να έχουν το αναμενόμενο ωφέλιμο φορτίο, εάν επιτρέπουν τη μεταφόρτωση αρχείων. REU15: Τα documentations να περιέχουν σχήματα για τον καθορισμό της δομής δεδομένων τους. REU15.1: Τα αρχεία JSON χρησιμοποιούν τη γλώσσα σχήματος "JSON Schema". REU15.2: Τα αρχεία XML χρησιμοποιούν πολλαπλές γλώσσες σχήματος, όπως RELAX NG, Schematron και XSD (XML Schema Definition Language), με την τελευταία να είναι η καλύτερη επιλογή. REU15.3: Τα αρχεία CSV έχουν δύο επιλογές για τη δομή: α) Frictionless Data που χρησιμοποιεί JSON αρχεία, και β) UK Nation Archives, με χρήση CSV γλώσσα σχήματος. REU15.4: Τα αρχεία RDF χρησιμοποιούν οντολογίες για γραφήματα ή αλλιώς σχήματα, χρησιμοποιώντας τη SHACL γλώσσα. REU15.5: Τα APIs πρέπει να τεκμηριώνουν και τη δομή αλλά και τον τρόπο πρόσβασης σε αυτά στο διαδίκτυο. Οι προδιαγραφές γράφονται είτε σε JSON είτε σε YAML. Θα πρέπει να προσδιοριστούν οι ακόλουθες πτυχές: Α) URLs και τελικά σημεία, Β) Πρωτόκολλα τελικών σημείων π.χ., HTTP, FTP, Γ) Μέθοδοι πρόσβασης, π.χ., Μέθοδοι HTTP, κωδικοί κατάστασης, Δ) Τρόποι τροποποίησης αποτελεσμάτων, π.χ., παράμετροι ερωτήματος, κεφαλίδες HTTP.

Σχήμα 2.2 Πίνακας Reusable για Data και Metadata Quality.

	Data Quality	Metadata Quality
A Accessible	AEU1: Τα δεδομένα να δημοσιεύονται χωρίς περιορισμούς πρόσβασης. AEU2: Τα δεδομένα να παρέχονται από μία προσβάσιμη διεύθυνση URL λήψης.	AEU3: API - Το πρότυπο HTTP, εκτός από το πραγματικό ωφέλιμο φορτίο, να στέλνει και τον σωστό κωδικό κατάστασης για τα αιτήματα των πελατών.
F Findable	FEU2: Τα δεδομένα με μηδενική (null) τιμή να επισημαίνονται ως τέτοια.	FEU1: Τα δεδομένα περιγράφονται με μεταδεδομένα για βελτίωση εντοπισμού τους (τίτλος, περιγραφή, λέξεις-κλειδιά, κατηγορίες, χρονικές πληροφορίες, χωρικές πληροφορίες). FEU3: Το documentation να δημοσιεύεται πάντα μαζί με τα δεδομένα, ιδανικά σε ξεχωριστή διανομή, και αργότερα να συνδέονται μεταξύ τους. Αυτό ισχύει για όλους τους τύπους αρχείων.

Σχήμα 2.3 Πίνακας Accessible και Findable για Data και Metadata Quality.

Κεφάλαιο 3. Σχεδίαση & Υλοποίηση

3.1 Use Cases

Use Case 1: RegisterDataset

Περιγραφή και Στόχος: Το use case “RegisterDataset” επιτρέπει στον αναλυτή να φορτώσει ένα αρχείο δεδομένων στο σύστημα, με σκοπό την καταχώρηση του στο σύστημα για την αξιολόγηση της ποιότητάς του.

Ηθοποιός: Αναλυτής.

Προϋποθέσεις: Το αρχείο δεδομένων πρέπει να είναι διαθέσιμο σε αποδεκτή μορφή (csv).

Βασική Ροή Γεγονότων:

1. Το use case ξεκινάει όταν ο αναλυτής επιλέξει το σύνολο δεδομένων για ανάλυση.
2. Το σύστημα διαβάζει το dataset από το path που δόθηκε από τον αναλυτή, και το αποθηκεύει.

Ενναλακτική Ροή Γεγονότων: Το αρχείο δεδομένων δεν έχει σωστή μορφή, άρα το σύστημα δεν το φορτώνει και εμφανίζει μήνυμα λάθους. Η εφαρμογή τερματίζει.

Συνθήκες μετά την εκτέλεση: Το αρχείο δεδομένων έχει καταχωρηθεί στο σύστημα και είναι έτοιμο για αξιολόγηση.

Use Case 2: ExecuteGlobalChecks

Περιγραφή και Στόχος: Το use case “ExecuteGlobalChecks” εκτελεί σε ολόκληρο το αρχείο κάποιους ελέγχους βασισμένους στις αρχές FAIR (Findability, Accessibility, Interoperability, Reusability).

Ηθοποιός: Αναλυτής.

Προϋποθέσεις: Ο αναλυτής να έχει καταχωρήσει ένα αρχείο δεδομένων στο σύστημα.

Βασική Ροή Γεγονότων:

1. Ξεκινάει όταν ο χρήστης επιλέξει να γίνει η εκτέλεση των global checks με το alias του dataset που έχει καταχωρηθεί.
2. Το σύστημα εκτελεί τους προκαθορισμένους FAIR ελέγχους σε όλο το dataset.
3. Το σύστημα επιστρέφει τα αποτελέσματα των ελέγχων.

Ενναλακτική Ροή Γεγονότων:

- AF1: Ο χρήστης καθορίζει συγκεκριμένο global check που θέλει να ελέγξει.
 - Το σύστημα εκτελεί μόνο τον ζητούμενο έλεγχο.
 - Το σύστημα επιστρέφει τα αποτελέσματα του συγκεκριμένου ελέγχου.
- AF2: Το alias που έδωσε ο αναλυτής δεν αντιστοιχεί σε καταχωρημένο dataset.
 - Το σύστημα επιστρέφει σφάλμα.

Συνθήκες μετά την εκτέλεση: Το σύστημα έχει εκτελέσει τους ελέγχους που αφορούν όλο το αρχείο, και τα αποτελέσματά τους είναι διαθέσιμα.

Use Case 3: ExecuteColumnChecks

Περιγραφή και Στόχος: Το use case “ExecuteColumnChecks” εκτελεί ελέγχους για κάθε στήλη του αρχείου, βασισμένους στις αρχές FAIR (Findability, Accessibility, Interoperability, Reusability).

Ηθοποιός: Αναλυτής.

Προϋποθέσεις: Ο αναλυτής να έχει καταχωρήσει ένα αρχείο δεδομένων στο σύστημα.

Βασική Ροή Γεγονότων:

1. Το use case ξεκινάει όταν ο αναλυτής επιλέξει να γίνει η εκτέλεση των column checks με το alias του dataset που έχει καταχωρηθεί.
2. Το σύστημα εκτελεί τους προκαθορισμένους FAIR ελέγχους ανά στήλη.
3. Το σύστημα επιστρέφει τα αποτελέσματα των ελέγχων.

Ενναλακτική Ροή Γεγονότων:

- AF1: Ο χρήστης ζητάει έναν συγκεκριμένο έλεγχο σε όλες τις στήλες.
 - Το σύστημα εκτελεί τον συγκεκριμένο έλεγχο για κάθε στήλη.
 - Το σύστημα επιστρέφει το αποτέλεσμα.
- AF2: Ο χρήστης ζητά όλους τους ελέγχους σε μία μόνο στήλη.
 - Το σύστημα εκτελεί ελέγχους μόνο σε αυτή τη στήλη.
 - Το σύστημα επιστρέφει το αποτέλεσμα.
- AF3: Ο χρήστης ζητά έναν συγκεκριμένο έλεγχο σε μία μόνο στήλη.
 - Το σύστημα εκτελεί μόνο τον έλεγχο αυτό, για τη συγκεκριμένη στήλη.
 - Το σύστημα επιστρέφει το αποτέλεσμα.

- AF4: Το alias που έδωσε ο αναλυτής δεν αντιστοιχεί σε καταχωρημένο dataset.
- Το σύστημα επιστρέφει σφάλμα.

Συνθήκες μετά την εκτέλεση: Το σύστημα έχει εκτελέσει τους ελέγχους που αφορούν την κάθε στήλη του αρχείου ξεχωριστά, και τα αποτελέσματά τους είναι διαθέσιμα.

Use Case 4: GenerateGlobalReport

Περιγραφή και Στόχος: Το use case “GenerateGlobalReport” δημιουργεί μία αναφορά με τα αποτελέσματα ελέγχων ποιότητας που αφορούν ολόκληρο το dataset.

Ηθοποιός: Σύστημα.

Προϋποθέσεις: Το σύστημα πρέπει να έχει εκτελέσει τους ελέγχους.

Βασική Ροή Γεγονότων:

1. Το use case ξεκινάει όταν ο αναλυτής έχει επιλέξει την δημιουργία αναφοράς με τα αποτελέσματα ελέγχων ολόκληρου του αρχείου.
2. Ο αναλυτής επιλέγει το path που θα αποθηκευτεί η αναφορά, και τον τύπο του αρχείου.
3. Το σύστημα δημιουργεί την αναφορά και την αποθηκεύει στην επιθυμητή μορφή.

Εναλλακτική Ροή Γεγονότων: Δεν έχουν εκτελεστεί οι έλεγχοι ποιότητας, άρα δεν μπορεί να δημιουργηθεί η αναφορά.

Συνθήκες μετά την εκτέλεση: Η αναφορά είναι διαθέσιμη στον αναλυτή.

Use Case 5: GenerateColumnReport

Περιγραφή και Στόχος: Το use case “GenerateColumnReport” δημιουργεί μία αναφορά με τα αποτελέσματα ελέγχων ποιότητας που αφορούν την κάθε στήλη του dataset ξεχωριστά.

Ηθοποιός: Σύστημα.

Προϋποθέσεις: Το σύστημα πρέπει να έχει εκτελέσει τους ελέγχους ποιότητας και να έχει δημιουργήσει αναφορά με τα αποτελέσματα των global checks.

Βασική Ροή Γεγονότων:

1. Το use case ξεκινάει όταν ο αναλυτής έχει επιλέξει την δημιουργία αναφοράς με τα αποτελέσματα ελέγχων για κάθε στήλη, και τον τύπο αρχείου.
2. Το σύστημα συμπληρώνει την ήδη υπάρχουσα αναφορά με τους νέους ελέγχους και την αποθηκεύει στην επιθυμητή μορφή.

Ενναλακτική Ροή Γεγονότων: Δεν έχουν εκτελεστεί οι έλεγχοι ποιότητας, άρα δεν μπορεί να δημιουργηθεί η αναφορά.

Συνθήκες μετά την εκτέλεση: Η αναφορά είναι διαθέσιμη στον αναλυτή.

3.2 Ορισμός προβλήματος και αλγόριθμοι επίλυσης

3.2.1 Τυπικός ορισμός του προβλήματος.

Η υψηλή ποιότητα των δεδομένων είναι αναγκαία σε πολλούς τομείς, όπως για παράδειγμα η ανάλυση των δεδομένων και η επιστημονική έρευνα. Πολλά αρχεία δεδομένων, ωστόσο, δεν έχουν την κατάλληλη μορφή για δημοσίευση, δηλαδή μπορεί να περιέχουν προβλήματα όπως ελλείψεις τιμών, λάθος μορφοποιήσεις και ασυμβατότητες. Όλα τα παραπάνω δεν βοηθούν στη επεξεργασία και στην εξαγωγή συμπερασμάτων.

Το εργαλείο της παρούσας εργασίας προτείνει την λύση στο παραπάνω πρόβλημα. Μέσω του συγκεκριμένου συστήματος, γίνεται αξιολόγηση της ποιότητας των δεδομένων, βασισμένο σε συγκεκριμένες αρχές (Άρθρα [WDA+16], [POEU21], [CaCo18]), και δυνατότητα επεξεργασίας τους. Ο αναλυτής επιλέγει ένα αρχείο δεδομένων και το σύστημα του παρουσιάζει ένα σύνολο ερωτήσεων σχετικές με την ποιότητα του. Οι απαντήσεις αποθηκεύονται και αξιολογούνται. Το σύστημα παράγει αναφορά ποιότητας, ενώ υπάρχει και η επιλογή αποστολής δεδομένων σε εξωτερικά εργαλεία καθαρισμού, για τη βελτίωση των δεδομένων.

3.2.2 Περιγραφή των χρησιμοποιούμενων δομών δεδομένων.

Το σύστημα βασίζεται στις εξής δομές δεδομένων:

- **Dataset:** Η συγκεκριμένη δομή αναπαριστά το σύνολο δεδομένων που έχει καταχωρηθεί στο σύστημα.
- **GlobalCheck:** Η εξής δομή αποτελεί έναν έλεγχο που αφορά συνολικά ολόκληρο το αρχείο.
- **ColumnCheck:** Η συγκεκριμένη δομή αποτελεί έναν έλεγχο που αφορά κάθε στήλη του αρχείου δεδομένων, ξεχωριστά.
- **FairCheckResult:** Αυτή η δομή αναπαριστά το αποτέλεσμα ενός ελέγχου ποιότητας FAIR για μία στήλη ή για το συνολικό αρχείο.

- **Report:** Η εξής δομή αποτελεί την αναφορά, η οποία περιέχει τα αποτελέσματα των ελέγχων, είτε για το συνολικό αρχείο είτε για κάθε στήλη ξεχωριστά.

3.2.3 Αλγόριθμοι για την επίλυση του προβλήματος.

Οι αλγόριθμοι που υλοποιούνται στο σύστημα εξυπηρετούν την υλοποίηση των παραπάνω Use Cases. Κάθε Use Case αποτελείται από έναν ή περισσότερους εσωτερικούς αλγορίθμους που εκτελούν τα βήματα της λειτουργίας. Ακολουθεί η περιγραφή των αλγορίθμων αυτών:

3.2.3.1 Αλγόριθμος Καταχώρησης Συνόλου Δεδομένων

Είσοδος: Μονοπάτι αρχείου, όνομα alias, πληροφορία ύπαρξης header.

Διαδικασία:

- Το σύστημα διαβάζει ένα αρχείο CSV.
- Αν υπάρχει header, το αποθηκεύει ξεχωριστά.
- Κάνει parsing των σειρών σε λίστα από πίνακες (List<String[]>).
- Αποθηκεύει το τύπο κάθε στήλης, και το διαχωριστικό που χρησιμοποιείται για κάθε στήλη.
- Δημιουργεί αντικείμενο Dataset και το αποθηκεύει στη μνήμη.

Έξοδος: RegistrationResponse με επιτυχία, αποτυχία ή ήδη υπάρχον alias.

3.2.3.2 Αλγόριθμος Εκτέλεσης Όλων των Γενικών Ελέγχων FAIR (Global Checks)

Είσοδος: Dataset alias.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset με βάση το alias.
- Εκτελεί έναν προς έναν όλους τους γενικούς ελέγχους στο dataset.
- Για κάθε έλεγχο, δημιουργεί ένα Boolean αποτέλεσμα.

Έξοδος: Map με ονόματα ελέγχων και τα αντίστοιχα αποτελέσματα.

3.2.3.3 Αλγόριθμος Εκτέλεσης Συγκεκριμένου Γενικού Ελέγχου (Global Check with Check Id)

Είσοδος: Dataset alias, id συγκεκριμένου check.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset με βάση το alias.

- Εντοπίζει τον έλεγχο με το check id που δόθηκε, από το σύνολο όλων των ελέγχων.
- Εκτελεί τον συγκεκριμένο έλεγχο, και βρίσκει το αποτέλεσμα του.

Έξοδος: Map με το όνομα του ελέγχου και το αποτέλεσμα.

3.2.3.4 Αλγόριθμος Εκτέλεσης Όλων των Ελέγχων Στηλών για Όλες τις Στήλες (Column Checks)

Είσοδος: Dataset alias.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset με βάση το alias και αναλύονται οι στήλες.
- Για κάθε στήλη, το σύστημα ελέγχει πρώτα αν ο τύπος της είναι συμβατός με τον κάθε έλεγχο που θα εκτελεστεί. Αν είναι, τότε τον εκτελεί, αν όχι τότε πηγαίνει στον επόμενο έλεγχο. Οι έλεγχοι έχουν κατηγοριοποιηθεί από το σύστημα με βάση τις FAIR αρχές.
- Αποθηκεύει τα αποτελέσματα ως List<FairCheckResult> ανά στήλη.

Έξοδος: Nested Map με την στήλη, την κατηγορία FAIR και τα αποτελέσματα όλων των ελέγχων της κάθε κατηγορίας αντίστοιχα.

3.2.3.5 Αλγόριθμος Εκτέλεσης Ενός Συγκεκριμένου Ελέγχου Στήλης για Όλες τις Στήλες (Column Check with Check Id)

Είσοδος: Dataset alias, all που συμβολίζει το σύνολο των στηλών και το id του check που θα εκτελεστεί.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset βάσει του alias και αναλύονται οι στήλες.
- Εντοπίζει την κατηγορία FAIR που ανήκει ο έλεγχος.
- Για κάθε στήλη, το σύστημα ψάχνει τον συγκεκριμένο έλεγχο, μέσα από το σύνολο όλων των ελέγχων στήλης, και τον εκτελεί.
- Αποθηκεύει το αποτέλεσμα ως List<FairCheckResult> ανά στήλη.

Έξοδος: Nested Map με την στήλη, την κατηγορία FAIR που ανήκει το check, και το αποτέλεσμα του για κάθε στήλη αντίστοιχα.

3.2.3.6 Αλγόριθμος Εκτέλεσης Όλων των Ελέγχων Στηλών για Μία Συγκεκριμένη Στήλη (Column Checks with Column Name)

Είσοδος: Dataset alias, το όνομα της στήλης, και all για όλους τους ελέγχους.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset βάσει του alias και αναλύονται οι στήλες.
- Το σύστημα εντοπίζει την στήλη στην οποία θα εκτελεστούν οι έλεγχοι.
- Εκτελεί όλους τους ελέγχους στήλης, και αποθηκεύει τα αποτελέσματα ως List<FairCheckResult>.

Έξοδος: Nested Map με το όνομα της στήλης, τις κατηγορίες FAIR, και τα αποτελέσματα όλων των ελέγχων της κάθε κατηγορίας αντίστοιχα.

3.2.3.7 Αλγόριθμος Εκτέλεσης Ενός Συγκεκριμένου Ελέγχου Στήλης για Μία Συγκεκριμένη Στήλη (Column Check with Check Id and Column Name)

Είσοδος: Dataset alias, το όνομα της στήλης το id του ελέγχου που θα εκτελεστεί.

Διαδικασία:

- Το σύστημα εντοπίζει το dataset βάσει του alias και αναλύονται οι στήλες.
- Εντοπίζει την κατηγορία FAIR που ανήκει το check.
- Βρίσκει την στήλη μέσω του ονόματός της.
- Εντοπίζει τον έλεγχο βάσει του id του, τον εκτελεί για την συγκεκριμένη στήλη που βρέθηκε και αποθηκεύει το αποτέλεσμα.

Έξοδος: Nested Map με το όνομα της στήλης, την κατηγορία FAIR, και το αποτέλεσμα του ελέγχου.

3.2.3.8 Αλγόριθμος Δημιουργίας Αναφοράς (Report Generation)

Είσοδος: Dataset alias, αποτελέσματα ελέγχων, output path, τύπος αρχείου αναφοράς.

Διαδικασία:

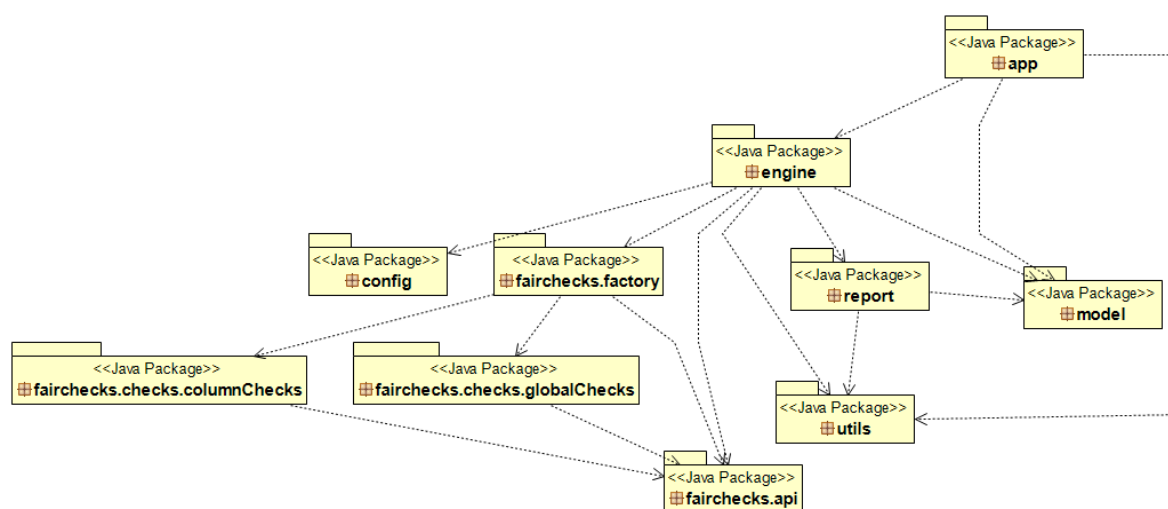
- Το σύστημα δημιουργεί το report με τον τύπο αρχείου από την είσοδο.
- Εισάγονται τα αποτελέσματα από τους ελέγχους που υλοποιήθηκαν, είτε global είτε column.
- Το σύστημα γράφει το τελικό αρχείο (TXT ή MD).

Έξοδος: Αρχείο αναφοράς με τα αποτελέσματα των ελέγχων αξιολόγησης FAIR.

3.3 Σχεδίαση και αρχιτεκτονική λογισμικού

Σε αυτό το κεφάλαιο παρουσιάζεται η περιγραφή σχεδίασης του συστήματος μέσω του διαγράμματος πακέτων και των διαγραμμάτων κάθε πακέτου, αλλά και κλάσης ξεχωριστά.

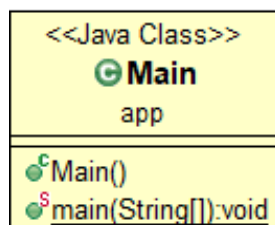
Η υλοποίηση του λογισμικού έγινε μέσω των κατάλληλων κλάσεων, οι οποίες χωρίζονται σε συνολικά 10 πακέτα. Στο παρακάτω διάγραμμα πακέτων UML παρουσιάζονται τα πακέτα αυτά, αλλά και η εξάρτηση μεταξύ τους:



Σχήμα 3.1 Διάγραμμα πακέτων

3.3.1 Πακέτο app

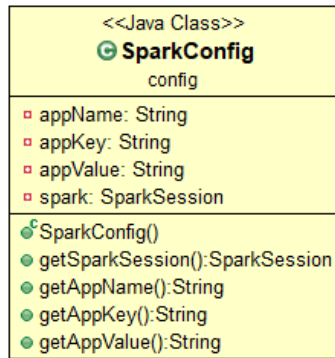
Το πακέτο app περιέχει μία μόνο κλάση, την Main. Στην κλάση αυτή υπάρχει μία main method, η οποία αποτελείται από ένα παράδειγμα για τον τρόπο που υλοποιείται το εργαλείο.



Σχήμα 3.2 Διάγραμμα UML κλάσεων του πακέτου app

3.3.2 Πακέτο config

Το πακέτο config αποτελείται, επίσης, από μία κλάση, την SparkConfig. Η συγκεκριμένη κλάση παρέχει πρόσβαση στο περιβάλλον Apache Spark που χρησιμοποιείται σε όλο το εργαλείο. Στην ουσία έχει σχεδιαστεί για την καταχώρηση και την επεξεργασία συνόλου δεδομένων. Την χρησιμοποιεί μόνο το πακέτο engine, εκεί που γίνεται η καταχώρηση του dataset.



Σχήμα 3.3 Διάγραμμα UML κλάσεων του πακέτου *config*

3.3.3 Πακέτο engine

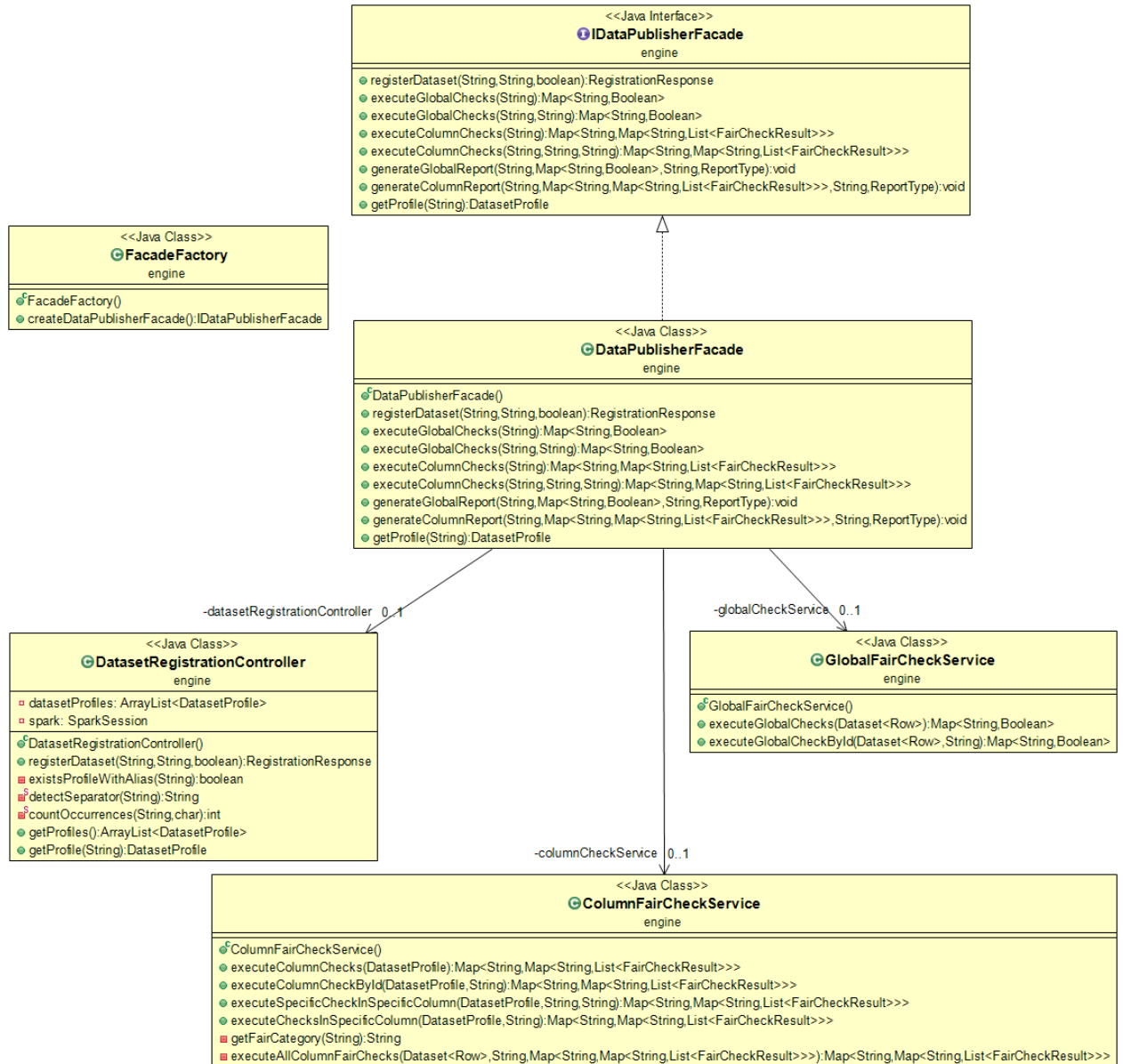
Το πακέτο engine περιέχει τις βασικές λειτουργικές μονάδες του συστήματος. Κύριες κλάσεις:

IDataPublisherFacade και DataPublisherFacade: Το interface και η κλάση που υλοποιεί τις μεθόδους του interface. Αποτελούν την κεντρική κλάση με όλα τα use cases.

Facade Factory: Δημιουργεί το IDataPublisherFacade, δηλαδή το εργαλείο.

DatasetRegistrationController: Υπεύθυνη για την διαχείριση καταχώρησης του dataset.

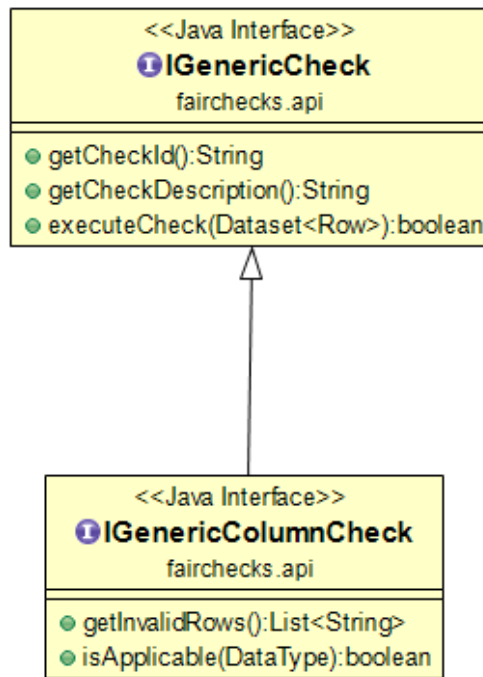
GlobalFairCheckService και ColumnFairCheckService: Εκτελούν τους ελέγχους FAIR.



Σχήμα 3.4 Διάγραμμα UML κλάσεων του πακέτου engine

3.3.4 Πακέτο fairchecks.api

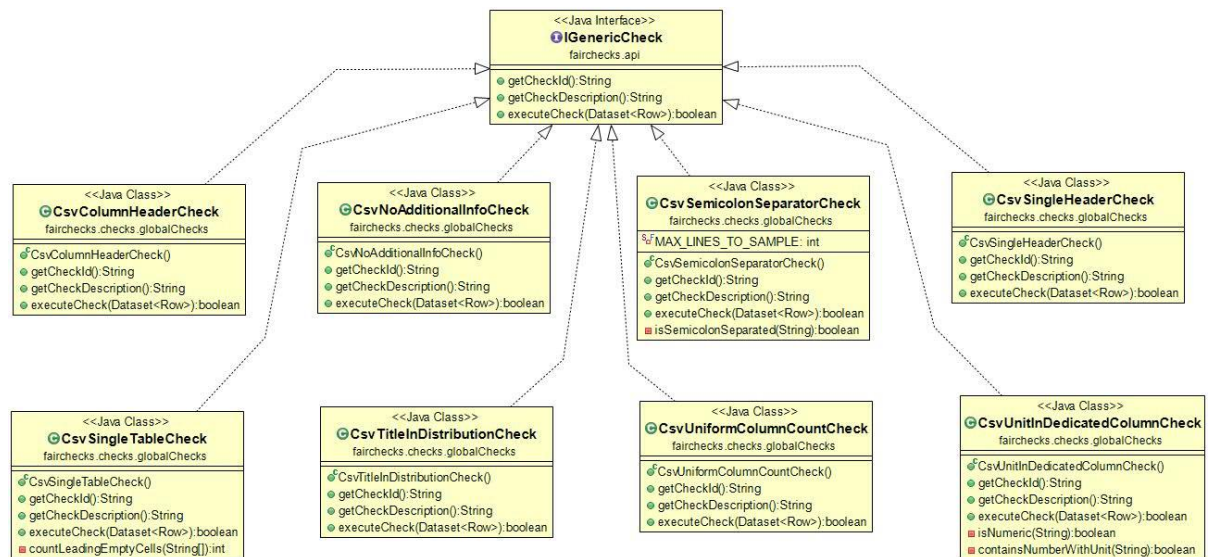
Το πακέτο fairchecks.api ορίζει τα interfaces για τους ελέγχους ποιότητας FAIR. Περιέχει το βασικό interface IGenericCheck με όλες τις μεθόδους για όλους τους ελέγχους, και το interface IGenericColumnCheck, το οποίο είναι επέκταση του πρώτου, περιέχοντας έξτρα δύο μεθόδους που αφορούν τους ελέγχους για κάθε στήλη μόνο.



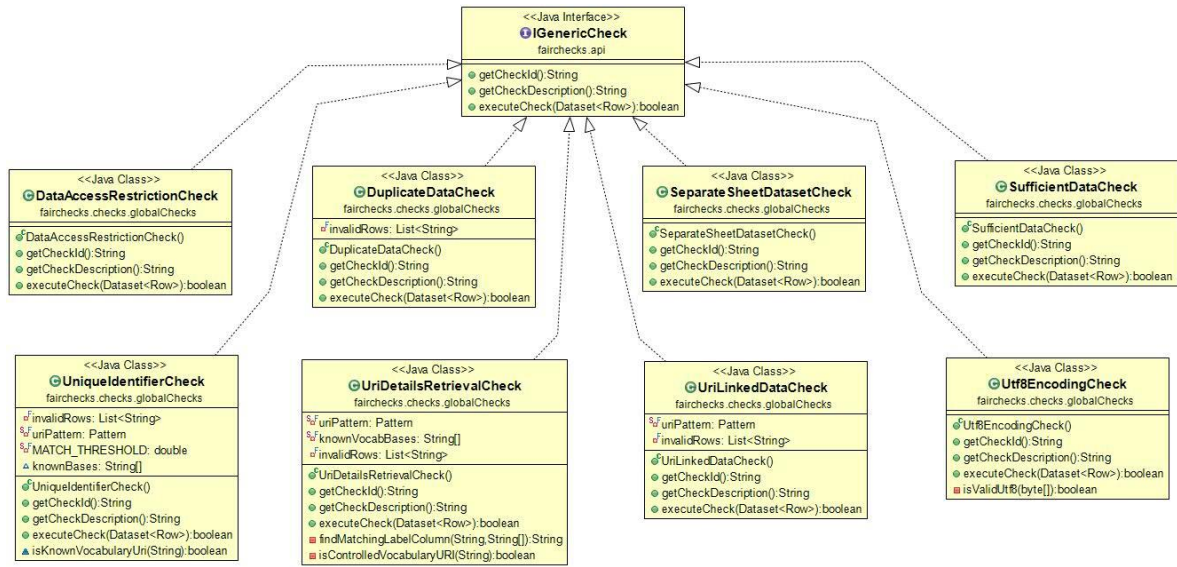
Σχήμα 3.5 Διάγραμμα UML κλάσεων του πακέτου *fairchecks.api*

3.3.5 Πακέτο *fairchecks.globalChecks*

Το πακέτο *fairchecks.globalChecks* περιέχει τις υλοποιήσεις των global ελέγχων. Κάθε κλάση αποτελεί έναν global έλεγχο, και υλοποιεί το *IGenericCheck*.



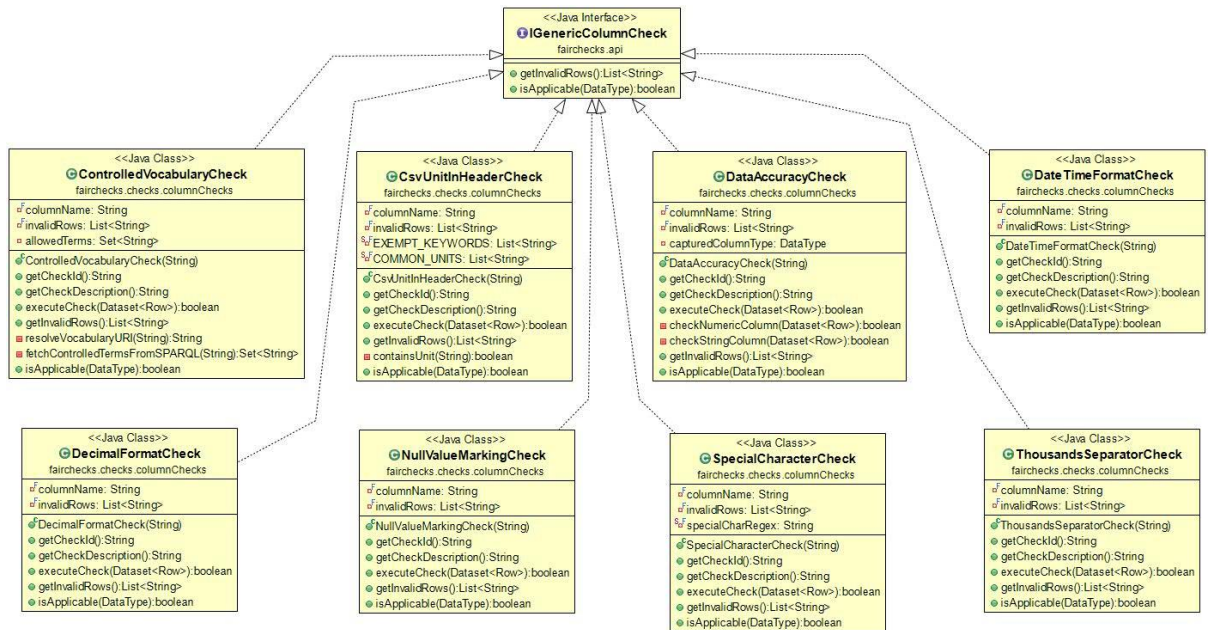
Σχήμα 3.6 Διάγραμμα UML κλάσεων του πακέτου *fairchecks.globalChecks*



Σχήμα 3.7 Διάγραμμα UML των υπόλοιπων κλάσεων του πακέτου *fairchecks.globalChecks*

3.3.6 Πακέτο *fairchecks.columnChecks*

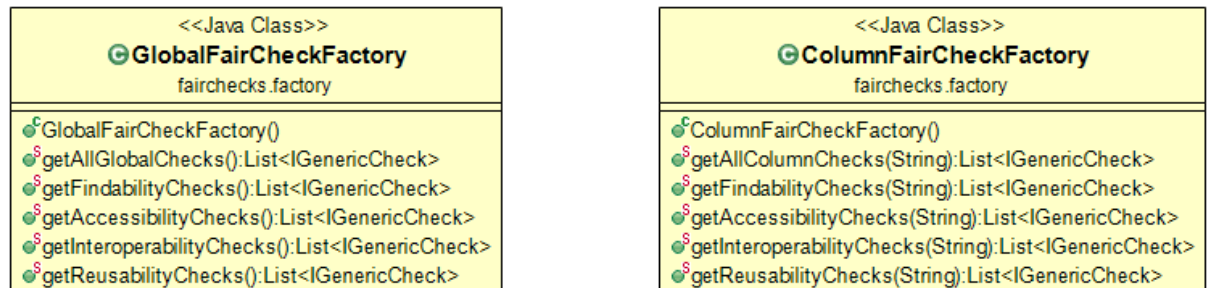
Το πακέτο *fairchecks.columnChecks* περιέχει τις υλοποιήσεις των column – level ελέγχων. Κάθε κλάση αποτελεί έναν έλεγχο, και υλοποιεί το *IGenericColumnCheck*.



Σχήμα 3.8 Διάγραμμα UML κλάσεων του πακέτου *fairchecks.columnChecks*

3.3.7 Πακέτο fairchecks.factory

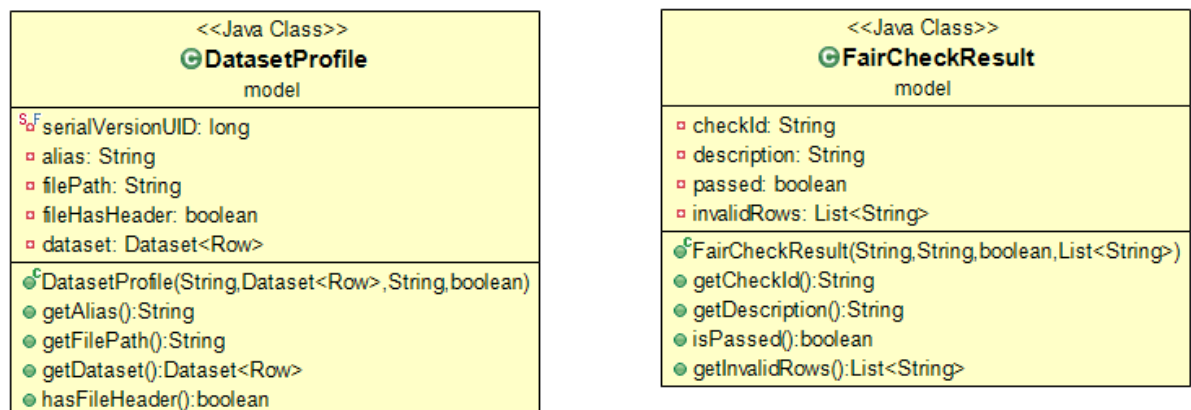
Το πακέτο fairchecks.factory περιλαμβάνει εργοστασιακές κλάσεις για τη δημιουργία ελέγχων. Περιέχει δύο κλάσεις, μία για τα global checks (GlobalFairCheckFactory), και μία για τα column checks (ColumnFairCheckFactory).



Σχήμα 3.9 Διάγραμμα UML κλάσεων του πακέτου fairchecks.factory

3.3.8 Πακέτο model

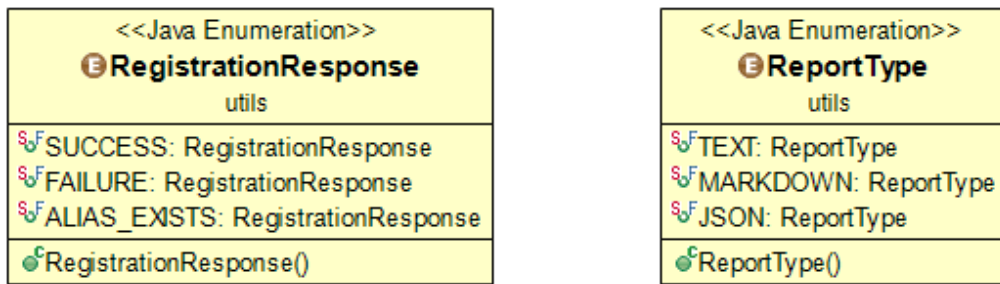
Το πακέτο model περιέχει δομές δεδομένων. Πιο συγκεκριμένα, αποτελείται από δύο κλάσεις. Η πρώτη είναι η DatasetProfile που αντιπροσωπεύει ένα dataset ενός csv αρχείου, καθώς και πληροφορίες για αυτό. Η δεύτερη είναι η FairCheckResult, η οποία αντιπροσωπεύει το αποτέλεσμα ενός FAIR ελέγχου που έγινε είτε σε όλο το σύνολο δεδομένων, είτε σε κάποια συγκεκριμένη στήλη.



Σχήμα 3.10 Διάγραμμα UML κλάσεων του πακέτου model

3.3.9 Πακέτο utils

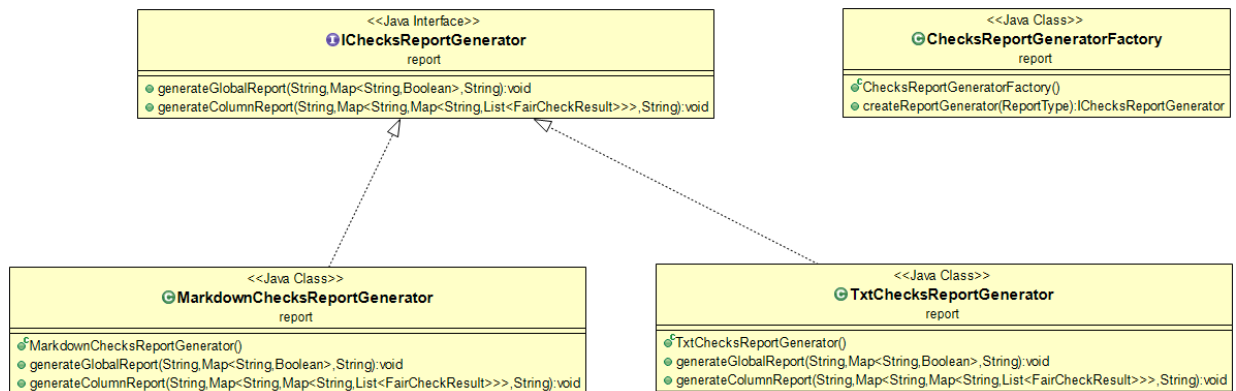
Το πακέτο utils περιλαμβάνει βοηθητικές δομές και τύπους. Μία από τις κλάσεις είναι η RegistrationResponse που χρησιμοποιείται όταν καταχωρείται ένα dataset, και η άλλη είναι η ReportType με τους διαφορετικούς τύπους αρχείων αναφοράς.



Σχήμα 3.11 Διάγραμμα UML κλάσεων του πακέτου *utils*

3.3.10 Πακέτο report

Το πακέτο report περιέχει τη λογική δημιουργίας αναφορών. Περιλαμβάνει την βασική κλάση interface `ICheckReportGenerator`, την οποία υλοποιούν οι κλάσεις `MarkdownChecksReportGenerator` και `TxtChecksReportGenerator`. Το αντικείμενο interface δημιουργείται από την κλάση `ChecksReportGeneratorFactory`.



Σχήμα 3.12 Διάγραμμα UML κλάσεων του πακέτου *report*

3.4 Σχεδίαση και αποτελέσματα ελέγχου του λογισμικού

Οι στόχοι ελέγχου λογισμικού περιβάλλοντος περιλαμβάνουν τα εξής:

- Την επιβεβαίωση ότι κάθε βασική λειτουργία (use case) αποδίδει σωστά αποτελέσματα.
- Την επιβεβαίωση ότι κάθε FAIR check αποδίδει σωστά αποτελέσματα.
- Την επιβεβαίωση ότι τα αποτελέσματα των ελέγχων γράφονται σωστά στις αναφορές.

Τα Unit Tests είναι γραμμένα με JUnit 4, και παίρνουν σαν είσοδο csv datasets φτιαγμένα ειδικά για testing όλων των ελέγχων FAIR, αλλά και συνολικά του εργαλείου. Ελέγχουν και σωστά αλλά και λανθασμένα datasets.

Η κλάση FacadeTests περιέχει unit tests για την λειτουργία του facade, δηλαδή τη σωστή υλοποίηση των use cases, βασικής και εναλλακτικής ροής γεγονότων. Επιπρόσθετα, υπάρχει μία κλάση για testing κάθε fair check ξεχωριστά, τόσο για τα global όσο και για τα column checks. Τα συγκεκριμένα tests περιέχουν και βοηθητικές κλάσεις για την καταχώρηση των datasets, ώστε να μην επαναλαμβάνεται σαν κώδικας στην κάθε κλάση. Ελέγχουν σωστά και λανθασμένα datasets, για αυτό τον λόγο η κάθε κλάση, δηλαδή το κάθε check, περιέχει ένα ή δύο tests. Επίσης, υπάρχουν tests για το πακέτο factory που δημιουργεί τα checks και τα κατηγοριοποιεί αναλόγως που ανήκουν στις FAIR αρχές. Τέλος, δημιουργήθηκαν και tests για την επιβεβαίωση της σωστής λειτουργίας του report, ελέγχοντας τη δημιουργία ενός report, και τη σωστή αποτύπωση των αποτελεσμάτων σε .md και .txt.

Συνολικά υπάρχουν 71 unit tests, τα οποία όλα τρέχουν επιτυχώς.

3.5 Λεπτομέρειες εγκατάστασης και υλοποίησης

Το συγκεκριμένο εργαλείο αναπτύχθηκε στην προγραμματιστική γλώσσα Java, και πιο συγκεκριμένα με Java 8, η οποία είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού με μεγάλη ευελιξία. Ως compiler για τον κώδικα της εφαρμογής χρησιμοποιήθηκε το JDK 1.8 από την OpenJDK. Το περιβάλλον ανάπτυξης στο οποίο έγινε η δημιουργία και συγγραφή του κώδικα του εργαλείου είναι το eclipse, και πιο συγκεκριμένα η έκδοση 2020-09. Το eclipse είναι ένα δωρεάν λογισμικό με δυνατότητες επεκτασιμότητας, πολύ διαδεδομένο στον χώρο της πληροφορικής. Το συγκεκριμένο λογισμικό δημιουργήθηκε ως Maven project, μέσα από το eclipse, ώστε να υπάρχει η ευελιξία και η δυνατότητα επεκτασιμότητας, μέσω της προσθήκης επιπλέον dependencies κάθε φορά που χρειάζεται. Το εργαλείο maven καθιστά εύκολη, και σχεδόν αυτόματη την εγκατάσταση και εφαρμογή κάποιας έξτρα βιβλιοθήκης. Άλλο ένα Plug-in που χρησιμοποιήθηκε μέσω του eclipse είναι το ObjectAid, με το οποίο δημιουργήθηκαν τα UML διαγράμματα κλάσεων και πακέτων, αξιοποιώντας τις πληροφορίες από τις κλάσεις που έχουν οριστεί στο project. Τέλος, για το testing του κώδικα, χρησιμοποιήθηκε το framework Junit 4, εύκολα εφαρμόσιμο από το eclipse.

3.5.1 Οδηγίες εγκατάστασης και εκτέλεσης του project:

Το project υπάρχει διαθέσιμο στο github και πιο συγκεκριμένα στην ιστοσελίδα:

<https://github.com/DAINTINESS-Group/DataPublisher>

Για την εγκατάστασή του θα χρειαστεί να γίνουν τα εξής βήματα:

- Κλωνοποίηση ή μεταφορά της εφαρμογής:
 - `git clone https://github.com/DAINTINESS-Group/DataPublisher.git`
`cd DataPublisher`
 - ή με απλή αντιγραφή του φακέλου
- Build της εφαρμογής:
 - `mvn clean install`
 - ή μέσω του eclipse ως εξής: File -> Import -> Existing Maven Projects
- Εκτέλεση της εφαρμογής:
 - μέσω Eclipse: Δεξί κλικ στην Main.java -> Run as Java Application.

Η εφαρμογή είναι διαχωρισμένη σε φακέλους. Πιο συγκεκριμένα στους εξής φακέλους:

- `/src/main/java` βρίσκεται ο κώδικας της εφαρμογής,
- `/src/test/java` βρίσκονται τα test cases,
- `/src/test/resources/datasets` υπάρχουν τα csv αρχεία δεδομένων που χρησιμοποιούνται για τον έλεγχο ορθότητας της εφαρμογής,
- `/src/test/resources/reports` αποθηκεύονται οι παραγόμενες αναφορές.

Ο κάθε χρήστης έχει τη δυνατότητα να χρησιμοποιήσει το dataset που θέλει, το οποίο μπορεί να είναι αποθηκευμένο σε οποιοδήποτε φάκελο του υπολογιστή του, αρκεί να γράψει το σωστό path στην μέθοδο καταχώρησης του αρχείου, και το αρχείο να είναι τύπου csv. Αυτό με την ευελιξία της επιλογής οποιουδήποτε φακέλου ισχύει και για το output path που θα αποθηκευτούν οι παραγόμενες αναφορές με τα αποτελέσματα. Όλα αυτά, είναι επεξεργάσιμα, και ορίζονται στην Main κλάση στον φάκελο `/src/main/java/app`.

Ένα παράδειγμα εκτέλεσης του project, με το dataset countries.csv, το οποίο περιέχει όλες τις πιθανές εκτελέσεις ελέγχων και παραγωγής αναφορών είναι το εξής:

```

public static void main(String[] args) {
    FacadeFactory factory = new FacadeFactory();
    IDataPublisherFacade facade = factory.createDataPublisherFacade();

    facade.registerDataset("src/test/resources/datasets/countries.csv",
        "myDataset", true);

    Map<String, Boolean> globalResults =
        facade.executeGlobalChecks("myDataset");
    Map<String, Map<String, List<FairCheckResult>>> columnResults =
        facade.executeColumnChecks("myDataset");

    String outputFolder = "src/test/resources/reports/";
    ReportType reportTypeMD = ReportType.MARKDOWN;
    facade.generateGlobalReport("myDataset", globalResults, outputFolder +
        "FAIR_Report.md", reportTypeMD);
    facade.generateColumnReport("myDataset", columnResults, outputFolder +
        "FAIR_Report.md", reportTypeMD);

    Map<String, Boolean> globalResultsForIEU16 =
        facade.executeGlobalChecks("myDataset", "IEU16");
    ReportType reportTypeTXT = ReportType.TEXT;

    facade.generateGlobalReport("myDataset", globalResultsForIEU16,
        outputFolder + "IEU16_Results_Report.txt", reportTypeTXT);

    Map<String, Map<String, List<FairCheckResult>>> columnResultsForFEU2 =
        facade.executeColumnChecks("myDataset", "all", "FEU2");
    facade.generateColumnReport("myDataset", columnResultsForFEU2,
        outputFolder + "FEU2_Results_Report.md", reportTypeMD);

    Map<String, Map<String, List<FairCheckResult>>> columnCapitalResults =
        facade.executeColumnChecks("myDataset", "Capital", "all");
    facade.generateColumnReport("myDataset", columnCapitalResults,
        outputFolder + "Capital_Column_Results_Report.md", reportTypeMD);

    Map<String, Map<String, List<FairCheckResult>>>
        columnCapitalResultsForFEU2 = facade.executeColumnChecks("myDataset",
        "Capital", "FEU2");
    facade.generateColumnReport("myDataset", columnCapitalResultsForFEU2,
        outputFolder + "Capital_Column_FEU2_Results_Report.txt",
        reportTypeTXT);
}

```

Σχήμα 3.13 Μέθοδος *main* που περιλαμβάνει όλες τις βασικές μεθόδους της εφαρμογής, με χρήση του dataset *countries.csv*

3.6 Επεκτασιμότητα του λογισμικού

Κατά την σχεδίαση του συστήματος λήφθηκε υπόψη η δυνατότητα μελλοντικής επέκτασης και συντήρησης. Το λογισμικό χρησιμοποιεί interfaces, factories, services και συλλογών αντικειμένων, ώστε να διευκολύνει την προσθήκη νέων λειτουργιών χωρίς μεγάλες τροποποιήσεις.

Κάποιες πιθανές μελλοντικές επεκτάσεις αφορούν τα εξής:

- Υποστήριξη επιπλέον τύπων αναφορών αποτελεσμάτων πέρα από TXT και Markdown.
- Υποστήριξη επιπλέον τύπων αρχείων δεδομένων για έλεγχο, εκτός από CSV. Αυτό είναι πιο απαιτητικό, καθώς θα πρέπει να υπάρξουν και κάποιες τροποποιήσεις σε υπάρχοντα FAIR checks, ή δημιουργία νέων.
- Προσθήκη νέων ελέγχων FAIR.
- Προσθήκη GUI διεπαφής.
- Παραγωγή βαθμολογίας, βάσει των αποτελεσμάτων των ελέγχων.

Η δομή του λογισμικού είναι κατάλληλη για επέκταση, χάρη στον σαφή διαχωρισμό ευθυνών και την χρήση interfaces. Επομένως, η προσθήκη νέων λειτουργιών καθίσταται δυνατή με ελάχιστη τροποποίηση υπάρχοντος κώδικα.

Κεφάλαιο 4. Πειραματική Αξιολόγηση

Η ενότητα αυτή παρουσιάζει αναλυτικά την πειραματική αξιολόγηση της μεθόδου μας.

4.1 Μεθοδολογία πειραματισμού

Σκοπός των πειραμάτων ήταν η εμπειρική αξιολόγηση της λειτουργικότητας και της αποτελεσματικότητας των FAIR ελέγχων που υλοποιήθηκαν στο πλαίσιο της εφαρμογής. Τα πειράματα επικεντρώθηκαν στην εφαρμογή των ελέγχων σε διαφορετικά σύνολα δεδομένων, με σκοπό να παρατηρηθεί αν το σύστημα εντοπίζει ασυνέπειες, και αν η παραγόμενη αναφορά αντικατοπτρίζει την πραγματική ποιότητα του dataset.

Για τον σκοπό αυτό, χρησιμοποιήθηκαν συνθετικά datasets που δημιουργήθηκαν χειροκίνητα ώστε να περιέχουν εσκεμμένα προβλήματα. Επιπλέον, δοκιμάστηκαν και πραγματικά datasets με μορφή CSV, που είναι διαθέσιμα μέσω ανοιχτών πηγών.

Όλα τα πειράματα πραγματοποιήθηκαν μέσω του Eclipse, με χρήση του εργαλείου Maven. Έγινε η εκτέλεση της εφαρμογής μέσω των tests και της κλάσης Main.java, αξιοποιώντας τα αρχεία εισόδου που βρέθηκαν στον φάκελο /src/test/resources/datasets. Οι αναφορές με τα αποτελέσματα των ελέγχων αποθηκεύονται στον φάκελο /src/test/resources/reports.

4.2 Αναλυτική παρουσίαση αποτελεσμάτων

Για την αξιολόγηση της λειτουργικότητας της εφαρμογής, δημιουργήθηκαν διαφορετικά συνθετικά datasets, καθένα με σκοπό να ενεργοποιήσει συγκεκριμένους ελέγχους ποιότητας. Οι έλεγχοι εκτελέστηκαν μέσω Junit tests, αξιοποιώντας την πλήρη ροή του συστήματος.

4.2.1 Dataset 1 – students_test.csv

Το συγκεκριμένο σύνολο δεδομένων κατασκευάστηκε για τον έλεγχο της σωστής λειτουργίας των global checks, δηλαδή των ελέγχων που αφορούν ολόκληρο το CSV αρχείο. Χρησιμοποιείται ως η σωστή περίπτωση στον έλεγχο των global checks μέσω Junit tests. Επίσης, χρησιμοποιείται και στον έλεγχο της γενικής μεθόδου ελέγχου όλων των global checks, αναμένοντας τα όλα τα αποτελέσματα να είναι σωστά, εκτός από το IEU9, γιατί έχει κόμμα αντί για ερωτηματικό ως διαχωριστή. Πιο συγκεκριμένα:

Global Check ID	Global Check Name	Result
IEU15	CsvColumnHeaderCheck	True
IEU11.1	CsvNoadditionalInfoCheck	True
IEU9	CsvSemicolonSeparatorCheck	False
IEU11.3	CsvSingleHeaderCheck	True
IEU10	CsvSingleTableCheck	True
IEU11.2	CsvTitleInDistributionCheck	True
IEU16	CsvUniformColumnCountCheck	True
IEU17.3	CsvUnitInDedicatedColumnCheck	True
AEU1	DataAccessRestrictionCheck	True
REU3	DuplicateDataCheck	True
REU1	SufficientDataCheck	True
IEU5	UniqueIdentifierCheck	True
IEU6	UriDetailsRetrievalCheck	True
IEU7	UriLinkedData	True
IEU3.1	Utf8EncodingCheck	True

Πίνακας 4.1 Αποτελέσματα των global FAIR checks για το dataset students_test.csv. Παρουσιάζεται το αναγνωριστικό κάθε ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

4.2.2 Dataset 2 – students_test_wrong.csv

Το dataset students_test_wrong.csv κατασκευάστηκε για τον έλεγχο συγκεκριμένων λανθασμένων περιπτώσεων στα global checks. Πιο συγκεκριμένα χρησιμοποιήθηκε στα εξής:

Global Check ID	Global Check Name	Result
IEU10	CsvSingleTableCheck	False
IEU5	UniqueIdentifierCheck	False
IEU3.1	Utf8EncodingCheck	False

Πίνακας 4.2 Αποτελέσματα κάποιων από τα global FAIR checks για το dataset *students_test.wrong2.csv*. Παρουσιάζεται το αναγνωριστικό κάθε ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

Στο συγκεκριμένο dataset, όπως φαίνεται και στον Πίνακα 4.2., υπάρχουν κάποια λάθη. Αρχικά, περιέχει δύο πίνακες αντί για έναν, κάτι που αποτελεί πρόβλημα. Επίσης, λείπουν URIs, μοναδικά αναγνωριστικά. Τέλος, η κωδικοποίηση του είναι ANSI αντί για UTF8.

4.2.3 Dataset 3 – students_test_wrong2.csv

Άλλο ένα σύνολο δεδομένων που δημιουργήθηκε για τον έλεγχο των γενικών ελέγχων (global checks), έχοντας λανθασμένες περιπτώσεις. Πιο συγκεκριμένα χρησιμοποιήθηκε στα Junit tests για τον έλεγχο των εξής:

Global Check ID	Global Check Name	Result
IEU11.1	CsvNoadditionalInfoCheck	False
IEU11.3	CsvSingleHeaderCheck	False
IEU16	CsvUniformColumnCountCheck	False
IEU6	UriDetailsRetrievalCheck	False
IEU7	UriLinkedData	False

Πίνακας 4.3 Αποτελέσματα κάποιων από τα global FAIR checks για το dataset *students_test.wrong2.csv*. Παρουσιάζεται το αναγνωριστικό κάθε ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

Όπως φαίνεται στον Πίνακα 4.3., το dataset κατασκευάστηκε με κάποια λανθασμένα χαρακτηριστικά, όπως το να περιέχει πληροφορίες για την τελευταία τροποποίηση του σαν σχόλιο μέσα στο dataset. Συμπληρωματικά, περιέχει την κεφαλίδα δύο φορές, ενώ ταυτόχρονα μία γραμμή έχει παραπάνω πληροφορίες – στήλες από αυτές της κεφαλίδας. Τέλος, δεν περιέχει URIs, άρα οι έλεγχοι που αφορούν τα μοναδικά αναγνωριστικά και την αξιοποίηση τους, έχουν ως αποτέλεσμα False.

4.2.4 Dataset 4 - students_test_wrong3.csv

Ακόμη ένα dataset που περιλαμβάνει λανθασμένες περιπτώσεις για τα global checks είναι το *students_test_wrong3.csv*. Αναλυτικότερα:

Global Check ID	Global Check Name	Result
IEU15	CsvColumnHeaderCheck	False
IEU11.2	CsvTitleInDistributionCheck	False

Πίνακας 4.4 Αποτελέσματα κάποιων από τα global FAIR checks για το dataset *students_test_wrong3.csv*. Παρουσιάζεται το αναγνωριστικό κάθε ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

Στο συγκεκριμένο σύνολο δεδομένων, σύμφωνα και με τον Πίνακα 4.4., η κεφαλίδα δεν βρίσκεται στην πρώτη γραμμή, όπως θα έπρεπε να είναι. Επιπλέον, μέσα στο dataset υπάρχει γραμμένος και ο τίτλος του, κάτι που είναι απαγορευτικό.

4.2.5 Dataset 5 – null_dataset.csv

Το συγκεκριμένο dataset δημιουργήθηκε ως μία ειδική περίπτωση λανθασμένου συνόλου δεδομένων. Περιέχει περισσότερες null τιμές σε σχέση με άλλες, χρήσιμες, τιμές. Για αυτό και χρησιμοποιείται ως μία λανθασμένη περίπτωση στον έλεγχο του:

Global Check ID	Global Check Name	Result
REU1	SufficientDataCheck	False

Πίνακας 4.5 Αποτέλεσμα του global FAIR check *SufficientDataCheck* για το *null_dataset.csv*. Παρουσιάζεται το αναγνωριστικό του ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

4.2.6 Dataset 6 – no_access_test.csv

Το dataset *no_access_test.csv* δημιουργήθηκε για τον έλεγχο προσβασιμότητας στα σύνολα δεδομένων. Στο συγκεκριμένο αρχείο έχουν περιοριστεί κάποια δικαιώματα πρόσβασης, κάτι που απαγορεύεται. Γι' αυτό και το αποτέλεσμα του Junit test είναι αρνητικό.

Global Check ID	Global Check Name	Result
AEU1	DataAccessRestrictionCheck	False

Πίνακας 4.6 Αποτέλεσμα του global FAIR check *DataAccessRestrictionCheck* για το *no_access_test.csv*. Παρουσιάζεται το αναγνωριστικό του ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

4.2.7 Dataset 7 – excel_test.xlsx and Dataset 8 - excel_test_wrong.xlsx

Τα δύο αυτά datasets κατασκευάστηκαν για την επιβεβαίωση λειτουργίας του ελέγχου *SeparateSheetDatasetCheck*. Στην ουσία ελέγχεται αν ένα σύνολο δεδομένων αποτελείται από ένα ή δύο φύλλα. Αυτό γίνεται στα αρχεία excel. Το *excel_test.xlsx* αποτελεί ένα dataset με ένα φύλλο, ενώ το *excel_test_wrong.xlsx* περιέχει δύο φύλλα.

Global Check ID	Global Check Name	Dataset	Result
REU7.5	SeparateSheetDatasetCheck	excel_test.xlsx	True
REU7.5	SeparateSheetDatasetCheck	excel_test_wrong.xlsx	False

Πίνακας 4.7 Αποτέλεσμα του global FAIR check *SeparateSheetDatasetCheck* για τα *excel_test.xlsx.csv* και *excel_test_wrong.xlsx*. Παρουσιάζεται το αναγνωριστικό του ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής, το όνομα του dataset και το αποτέλεσμα εκτέλεσης (True/False).

4.2.8 Dataset 9 – fruits_test.csv

Το σύνολο δεδομένων *fruits_test.csv* κατασκευάστηκε για την επιβεβαίωση σωστής λειτουργίας τόσο των global checks όσο και των column checks. Δεν χρησιμοποιείται σε όλα τα Junit tests για τον έλεγχο αυτό, αλλά σε όσα χρησιμοποιείται αποτελεί τη σωστή περίπτωση ενός dataset, εκτός μίας περίπτωσης. Την εξαίρεση αποτελούν οι στήλες *price* και *color*, οι οποίες περιέχουν κάποια αριθμητικά και ορθογραφικά λάθη, αντίστοιχα. Αναλυτικότερα, χρησιμοποιείται στα εξής:

Check Category	Column(s)	Check ID	Global Check Name	Result
Global	-	IEU9	CsvSemicolonSeparatorCheck	True
Global	-	IEU17.3	CsvUnitInDedicatedColumnCheck	True
Global	-	REU3	DuplicateDataCheck	True
Global	-	REU1	SufficientDataCheck	True
Column	currency	IEU4	ControlledVocabularyCheck	True
Column	weight (g)	IEU17.1	CsvUnitInHeaderCheck	True
Column	currency	REU4	DataAccuracyCheck	True
Column	price / color	REU4	DataAccuracyCheck	False
Column	purchase timestamp	IEU1	DateTimeFormatCheck	True
Column	price	IEU2.1	DecimalFormatCheck	True

Column	quantity	IEU2.2	ThousandsSeparatorCheck	True
---------------	----------	--------	-------------------------	------

Πίνακας 4.8 Αποτελέσματα κάποιων από τα *global FAIR checks* και κάποιων από τα *column FAIR checks* για το dataset *fruits_test.csv*. Παρουσιάζεται η κατηγορία κάθε ελέγχου (*Global/Column*), την στήλη (ή στήλες) που αφορά αν πρόκειται για *column check*, το αναγνωριστικό κάθε ελέγχου (*ID*), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (*True/False*).

4.2.9 Dataset 10 - fruits_test_wrong.csv

Αντίστοιχα, το dataset *fruits_test_wrong.csv* δημιουργήθηκε για την επιβεβαίωση της σωστής λειτουργίας ελέγχων, κυρίως για τις περιπτώσεις που το σύνολο δεδομένων περιλαμβάνει λανθασμένες τιμές. Σε κάποια tests χρησιμοποιείται και για τις σωστές τιμές. Χρησιμοποιείται τόσο σε *global checks*, όσο και σε *column checks*. Πιο συγκεκριμένα:

Check Category	Column(s)	Check ID	Global Check Name	Result
Global	-	IEU9	CsvSemicolonSeparatorCheck	False
Global	-	IEU17.3	CsvUnitInDedicatedColumnCheck	False
Global	-	REU3	DuplicateDataCheck	False
Global	-	REU1	SufficientDataCheck	False
Column	currency	IEU4	ControlledVocabularyCheck	False
Column	weight	IEU17.1	CsvUnitInHeaderCheck	False
Column	purchase timestamp	IEU1	DateTimeFormatCheck	False
Column	price	IEU2.1	DecimalFormatCheck	False
Column	Fruit / color	FEU2	NullValueMarkingCheck	True
Column	price	FEU2	NullValueMarkingCheck	False
Column	Fruit / color	IEU3.2	SpecialCharacterCheck	True

Column	price	IEU3.2	SpecialCharacterCheck	False
Column	quantity	IEU2.2	ThousandsSeparatorCheck	False

Πίνακας 4.9 Αποτελέσματα κάποιων από τα global FAIR checks και κάποιων από τα column FAIR checks για το dataset fruits_test_wrong.csv. Παρουσιάζεται η κατηγορία κάθε ελέγχου (Global/Column), την στήλη (ή στήλες) που αφορά αν πρόκειται για column check, το αναγνωριστικό κάθε ελέγχου (ID), το όνομα του ως κλάση της εφαρμογής και το αποτέλεσμα εκτέλεσης (True/False).

Σύμφωνα με τον Πίνακα 4.9., τα global checks που ελέγχονται δίνουν False ως αποτέλεσμα, γιατί το dataset έχει το κόμμα ως διαχωριστικό, δεν περιέχει στήλη με τις μονάδες μέτρησης που χρειάζονται, περιέχει διπλότυπες γραμμές, και περιέχει ανεπαρκείς πληροφορίες. Όσον αφορά τα column checks, πάλι ο έλεγχος με τις μονάδες μέτρησης είναι λάθος, καθώς η στήλη που ελέγχεται δεν περιλαμβάνει αυτή την πληροφορία στην κεφαλίδα της. Επιπρόσθετα, υπάρχουν λάθη στην διατύπωση τιμών σε ορισμένες στήλες, όπως για παράδειγμα η ημερομηνία να γράφει πρώτα την ημέρα αντί για τη χρονιά, οι δεκαδικοί να χωρίζονται με κόμμα, να υπάρχουν κενά πεδία αντί για τη λέξη "null", και οι αριθμοί σε χιλιάδες να έχουν κενό ή τελεία ως διαχωριστικό. Επιπλέον, σύμφωνα με κάποια ελεγχόμενα λεξιλόγια, η στήλη currency περιέχει τιμές που δεν είναι επίσημες. Τέλος, το dataset περιέχει και ειδικούς χαρακτήρες, όπως για παράδειγμα τον "\$".

4.2.10 Dataset 11 – countries.csv

Το τελευταίο dataset που χρησιμοποιήθηκε είναι το countries.csv, το οποίο είναι ένα έτοιμο σύνολο δεδομένων σε μορφή csv, αντλούμενο από ανοιχτή πηγή. Περιέχει περισσότερες στήλες και γραμμές από τα άλλα datasets, και χρησιμοποιήθηκε ως ένα καλό παράδειγμα για τον τρόπο χρήσης και λειτουργίας της εφαρμογής. Το dataset αυτό, χρησιμοποιείται στην κλάση Main.java, στην οποία τρέχουν όλες οι βασικές μέθοδοι του εργαλείου. Επομένως, το countries.csv είναι σημαντικό για την κατανόηση χρήσης της εφαρμογής. Εκτελούνται όλα τα global checks και όλα τα column checks, αλλά και συγκεκριμένα checks αναλόγως το αναγνωριστικό τους (ID) ή την στην στήλη που θέλουμε να ελέγξουμε. Τέλος, παράγονται και οι αντίστοιχες αναφορές.

Κεφάλαιο 5. Επίλογος

Η ενότητα αυτή συνοψίζει τη συνεισφορά και τα αποτελέσματα της εργασίας και παραθέτει προτάσεις για μελλοντικές επεκτάσεις της.

5.1 Σύνοψη και συμπεράσματα

Η παρούσα διπλωματική εργασία είχε ως αντικείμενο την ανάπτυξη ενός λογισμικού εργαλείου για την αξιολόγηση της ποιότητας δεδομένων με ελέγχους βασισμένους στις αρχές FAIR (Findable, Accessible, Interoperable, Reusable). Η αξιολόγηση υλοποιήθηκε τόσο σε επίπεδο συνόλου (global checks), όσο και σε επίπεδο πεδίου (column checks), επιτρέποντας έναν αναλυτικό και ευέλικτο έλεγχο της ποιότητας.

Το σύστημα που κατασκευάστηκε υποστηρίζει την καταχώρηση και ανάγνωση συνόλων δεδομένων, την εκτέλεση ελέγχων FAIR τόσο σε επίπεδο συνόλου (global checks) όσο και σε επίπεδο συγκεκριμένων ελέγχων και στηλών (column checks) προς αξιολόγηση, καθώς και την παραγωγή αναφορών, σε μορφή .txt και .md, με τα αποτελέσματα της διαδικασίας. Η υλοποίηση πραγματοποιήθηκε σε Java, με χρήση Maven για την καλύτερη οργάνωση και διαχείριση του project. Ιδιαίτερη έμφαση δόθηκε στην επεκτασιμότητα με χρήση κατάλληλων interfaces, για τον καθαρό διαχωρισμό ευθυνών ανά πακέτο.

Κατά την δημιουργία της εργασίας, και συγκεκριμένα του εργαλείου, διαπιστώθηκε ότι η αυτόματη αξιολόγηση δεδομένων, με την χρήση FAIR αρχών, είναι μια σύνθετη διαδικασία που απαιτεί ισορροπία μεταξύ τεχνικής ανάλυσης και εννοιολογικής ερμηνείας των αρχών. Η χρήση διακριτών ελέγχων ανά επίπεδο (global/ column) και η δυνατότητα εστιασμένου ελέγχου σε συγκεκριμένες στήλες, ενίσχυσαν τη χρηστικότητα του εργαλείου.

Η αξιολόγηση της λειτουργικότητας του συστήματος, μέσω δοκιμών όλων των πιθανών περιπτώσεων και σεναρίων, ανέδειξε την ορθή λειτουργία, τη σταθερότητα και τη

χρηστικότητα της εφαρμογής. Επιπλέον, επιβεβαιώθηκε η δυνατότητα επέκτασης του εργαλείου με νέα checks και formats αναφοράς με ελάχιστη ή και καμία τροποποίηση του υπάρχοντος κώδικα, κάτι που υποστηρίζει τη μακροχρόνια συντηρητικότητα του έργου. Η αξιολόγηση της λειτουργικότητας του συστήματος πραγματοποιείται μέσω δοκιμών, με χρήση διαφορετικών dataset. Τα αποτελέσματα επιβεβαιώνουν την ορθή λειτουργία, ευελιξία και επεκτασιμότητα της εφαρμογής.

5.2 Μελλοντικές επεκτάσεις

Η εργασία έθεσε τα θεμέλια για την αυτόματη αξιολόγηση της ποιότητας FAIR σε σύνολα δεδομένων. Ωστόσο, οι δυνατότητες περαιτέρω εξέλιξης του συστήματος, οι οποίες μπορούν να ενισχύσουν τη λειτουργικότητα και τη χρησιμότητά του, είναι πολυάριθμες. Αυτές οι προτάσεις παρατίθενται στις παρακάτω παραγράφους.

- Υποστήριξη επιπλέον τύπων αναφοράς αποτελεσμάτων. Προς το παρόν υπάρχει δυνατότητα δημιουργίας μιας αναφοράς σε .txt ή σε .md μορφή. Μπορεί να υπάρξει η δυνατότητα για νέους τύπους, όπως .json, .pdf ή .html μέσω της επέκτασης του interface στο πακέτο report.
- Υποστήριξη διαφορετικών μορφών δεδομένων, πέρα από CSV, όπως JSON, Excel, ή XML.
- Εμπλουτισμός ελέγχων FAIR με νέα global ή column checks που καλύπτουν επιπλέον αρχές ή που αφορούν διαφορετικούς τύπους αρχείων από το CSV.
- Προσθήκη διεπαφής χρήστη (GUI) για απλοποίηση της χρήσης.
- Αυτόματη εξαγωγή συνοπτικής βαθμολογίας FAIRness, βάσει των επιμέρους ελέγχων, ώστε να παρέχεται στο χρήστη μια συνολική εικόνα ποιότητας.
- Ενσωμάτωση με άλλα εργαλεία, όπως λογισμικό εκκαθάρισης λανθασμένων δεδομένων, για επέκταση του ρόλου του συστήματος.

Οι παραπάνω επεκτάσεις μπορούν να υλοποιηθούν σταδιακά, αξιοποιώντας την αρχιτεκτονική του υπάρχοντος συστήματος, και να καταστήσουν την εφαρμογή ένα ολοκληρωμένο εργαλείο FAIR αξιολόγησης για την ερευνητική κοινότητα.

Κεφάλαιο 6. Βιβλιογραφία

- [CaCo18] Capgemini Consulting, Open Data Goldbook for Data Managers and Data Holders, Practical guidebook for organizations wanting to publish Open Data, available at https://data.europa.eu/sites/default/files/european_data_portal_-_open_data_goldbook.pdf, last updated January 2018.
- [EU24] European Union, High-value datasets: Questions and Answers, available at <https://digital-strategy.ec.europa.eu/faqs/high-value-datasets-questions-and-answers>, 2024.
- [FF19] Fraunhofer FOKUS, Data quality guidelines for the publication of data sets in the EU Open Data Portal, available at <https://interoperable-europe.ec.europa.eu/sites/default/files/solution/documentation/2020-08/PRJ004%20-%20ISA2%20Fraunhofer%20Fokus%20-%20Data%20Quality%20Guidelines%20-%20Deliverable%20C3%20-%201.5.pdf>, 2019.
- [HK15] Michael Hausenblas, James G. Kim, 5-star deployment scheme for Open Data, available at <https://5stardata.info/en/>, last updated 201-08-31.
- [POEU21] Publications Office of the European Union, *Data.europa.eu data quality guidelines*, Publication Office, 2021, available at <https://data.europa.eu/doi/10.2830/79367>

- [WDA+16] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J.G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A.C. 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao & Barend Mons. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* 3, 160018 (2016). <https://doi.org/10.1038/sdata.2016.18>