



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
Π.Μ.Σ. ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΤΕΥΘΥΝΣΗ ΣΥΣΤΗΜΑΤΑ ΥΠΟΛΟΓΙΣΤΩΝ**

Σύγκριση Μοντέλων Προγραμματισμού

ΒΑΣΙΛΕΙΟΣ ΚΟΥΤΣΙΑΝΟΣ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

*Επιβλέπων Καθηγητής: ΑΛΕΞΑΝΔΡΟΣ ΧΑΤΖΗΓΕΩΡΓΙΟΥ
Εξεταστές: ΧΡΗΣΤΟΥ – ΒΑΡΣΑΚΕΛΗΣ ΔΗΜΗΤΡΙΟΣ*

Τμήμα Εφαρμοσμένης Πληροφορικής

Πανεπιστήμιο Μακεδονίας
Θεσσαλονίκη

ΣΕΠΤΕΜΒΡΙΟΣ 2009

Copyright © ΒΑΣΙΛΕΙΟΣ ΚΟΥΤΣΙΑΝΟΣ, 2009
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Η έγκριση της μεταπτυχιακής εργασίας από το Τμήμα Εφαρμοσμένης Πληροφορικής του Πανεπιστημίου Μακεδονίας δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

ΠΕΡΙΛΗΨΗ

Η πραγματικότητα αποδεικνύει ότι διαφορετικές φιλοσοφίες προγραμματισμού καταλήγουν σε τελείως διαφορετικής ποιότητας κώδικα. Για παράδειγμα, ο κώδικας που αναπτύσσεται από βιομηχανίες, ο κώδικας που αναπτύσσεται σε κοινότητες ανοικτού λογισμικού (open-source), το λογισμικό που αναπτύσσεται υπό μορφή βιβλιοθηκών (APIs) αλλά και το λογισμικό που υλοποιούν φοιτητές, διαφέρουν ριζικά στη δομή και ποιότητα μεταξύ τους. Στόχος της παρούσας μελέτης είναι η ποιοτική σύγκριση των συστημάτων ανοικτού λογισμικού και των βιβλιοθηκών. Τα ποιοτικά χαρακτηριστικά και κριτήρια που χρησιμοποιήθηκαν για την σύγκριση είναι οι μετρικές, οι έλεγχοι (audits) και τα πρότυπα σχεδίασης. Η μέθοδος της συγκριτικής ανάλυσης είναι η στατιστικά σημαντική διαφορά των παραπάνω κριτηρίων.

ABSTRACT

The reality proves that different philosophies of planning lead in perfectly different quality of code. As an example, the code that is developed by industries, the code that is developed in communities of open software (open-source), the software that are developed under form of libraries (APIs) but also the software that they materialise students, differ radically in the structure and quality from each other. Objective of present study is the qualitative comparison of systems open computational and libraries. The qualitative characteristics and criteria that were used for the comparison are metric, the controls (audits) and the models of designing. The method of comparative analysis is the statistically important difference of above criteria.

ΠΕΡΙΕΧΟΜΕΝΑ

1 Εισαγωγή - Σκοπός της εργασίας.....	6
2. Βιβλιοθήκες Λογισμικού.....	7
3. Αξιολόγηση Λογισμικού.....	11
3.1 Περιγραφή Μετρικών – Στατιστικά Σημαντικών.....	11
3.2 Περιγραφή Ελέγχων – Design Flaws.....	19
3.3 Περιγραφή Προτύπων Σχεδίασης – Design Patterns.....	24
4. Πειραματική Μελέτη.....	31
4.1 Περιγραφή Πειραμάτων.....	31
4.2 Στατιστική Ανάλυση Μετρικών.....	36
4.3 Στατιστική Ανάλυση Ελέγχων – Design Flaws.....	64
4.4 Στατιστική Ανάλυση Προτύπων Σχεδίασης – Design Patterns.....	72
5. Αποτελέσματα και Συζήτηση.....	73
6. Συμπεράσματα.....	76
6. Βιβλιογραφία.....	77

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή – Σκοπός της Εργασίας

Στόχος της εργασίας είναι η θεωρητική μελέτη και πειραματική καταγραφή αυτών των διαφορών εστιάζοντας στις διαφορές μεταξύ λογισμικού βιβλιοθηκών (API's) και λογισμικού έργων ανοικτού κώδικα. Οι βιβλιοθήκες εν γένει αποτελούν παραδείγματα προσεγμένης σχεδίασης όπου εφαρμόζονται οι ευρέως αποδεκτές αρχές σχεδίασης, υλοποιούνται πρότυπα σχεδίασης και το λογισμικό συμμορφώνεται με τους περισσότερους ευρετικούς κανόνες. Τα έργα ανοικτού κώδικα, αν και υπάρχουν πολύ μεγάλες διαφορές μεταξύ τους, εν γένει δεν χαρακτηρίζονται από υψηλής ποιότητας αρχιτεκτονική σχεδίαση.

Ως όχημα για την πειραματική σύγκριση των δύο προσεγγίσεων θα επιλεγούν διάφορα έργα από κάθε κατηγορία, υλοποιημένα σε κάποια αντικειμενοστρεφή γλώσσα προγραμματισμού (κατά προτίμηση Java). Η καταγραφή μπορεί να αφορά όχι μόνο επιλεγμένες γενιές των προϊόντων λογισμικού που θα επιλεγούν αλλά και την εξέλιξη που παρουσιάζουν τα έργα εξετάζοντας πολλαπλές γενιές. Τα πειραματικά αποτελέσματα θα αποτελέσουν τη βάση για συστηματική στατιστική μελέτη των τυχόν διαφορών. Η μελέτη πολλαπλών γενεών λογισμικού ενδέχεται να αποκαλύψει διαφορές σχετικά με την εξέλιξη της ποιότητας του λογισμικού σε βιβλιοθήκες και έργα ανοικτού κώδικα.

Στο Κεφάλαιο 2 αναλύουμε και εξετάζουμε γιατί αναμένουμε οι βιβλιοθήκες να είναι καλύτερα και περισσότερο ποιοτικά στοιχεία σε σύγκριση με τα συστήματα ανοικτού κώδικα. Στο Κεφάλαιο 3 πραγματοποιείται μια αξιολόγηση λογισμικού, στην οποία περιγράφουμε τις στατιστικά σημαντικές μετρικές, τους ελέγχους (audits) και τα πρότυπα σχεδίασης (Design Patterns) και συγκεκριμένα την κατηγορία Design Flaws. Στο 4^ο Κεφάλαιο καταγράφεται η πειραματική μελέτη, γίνεται μια περιγραφή των βιβλιοθηκών και των συστημάτων ανοικτού κώδικα που συμμετείχαν στην εργασία, τα κριτήρια που χρησιμοποιήθηκαν για την επιλογή τους, τα πακέτα λογισμικού που χρησιμοποιήθηκαν. Επιπροσθέτως γίνεται στατιστική ανάλυση των μετρικών, των ελέγχων και των προτύπων σχεδίασης.

Τα αποτελέσματα των στατιστικών ελέγχων και η σχετική συζήτηση σχετικά με τους τρόπους στατιστικής ανάλυσης και σύγκρισης των βιβλιοθηκών έναντι των συστημάτων ανοικτού κώδικα, αναφέρονται στο 5^ο Κεφάλαιο. Τέλος στο 6^ο Κεφάλαιο γίνεται αναφορά στα συμπεράσματα της έρευνας και γίνονται προτάσεις για μελλοντική έρευνα.

ΚΕΦΑΛΑΙΟ 2

ΒΙΒΛΙΟΘΗΚΕΣ ΛΟΓΙΣΜΙΚΟΥ

Σχεδίαση Βιβλιοθηκών

Μεταξύ των ανθρώπων που γράφουν τις βιβλιοθήκες στην java, υπάρχει μια κοινή παρερμηνεία ότι οι βιβλιοθήκες αποτελούνται από τίποτα περισσότερο από τις κλάσεις και τις μεθόδους.

Άλλη μια φορά, θυμηθείτε γιατί ενδιαφερόμαστε για την ανάπτυξη βιβλιοθηκών: θέλουμε να συγκεντρώσουμε ενδεικτικά τις αιτήσεις μας από τις μεγάλες δομικές μονάδες, όπως οι κοινές βιβλιοθήκες, τα πλαίσια, οι σκελετοί εφαρμογής, καθώς επίσης και οι συνδυασμοί αυτών. Πιστεύουμε ότι εάν οι άνθρωποι που σχεδιάζουν αυτές τις εφαρμογές κάνουν καλά την εργασία τους, γράφουν δηλαδή κατάλληλα, έπειτα η δημιουργία της βιβλιοθήκης θα είναι μια απλή λειτουργία και δεν θα πρέπει να σπαταλήσουμε χρόνο για την διόρθωσή της, την ανάγνωση του κώδικα πηγής, την επιδιόρθωση του, και γενικά την προσπάθεια να καταλάβουμε τι έκαναν άλλοι.

Μια βιβλιοθήκη που σχεδιάζεται από αυτήν την άποψη είναι πολύ πλουσιότερη και αρτιότερη από ένα σύστημα ανοικτού κώδικα. Παραθέτουμε κάποια πλεονεκτήματα τα οποία σύμφωνα με τον Tulach (1) αποδεικνύουν την ανωτερότητα των βιβλιοθηκών έναντι των συστημάτων ανοικτού κώδικα:

Είναι μικρή σε μέγεθος : Μια μικρή βιβλιοθήκη είναι εκείνη που έχει όσο το δυνατόν λιγότερα δημόσια μέλη ανά κλάση, καθώς και όσο το δυνατόν ελάχιστες κλάσεις. Αυτό διευκολύνει την κατανόηση, την αποσφαλμάτωση, και την αλλαγή του API. Μια βιβλιοθήκη θα πρέπει να **κάνει ένα πράγμα και να το κάνει καλά:** Η λειτουργικότητα της θα πρέπει να είναι εύκολο να εξηγηθεί. Επίσης θα πρέπει να είναι **όσο το δυνατόν μικρότερη:** θα πρέπει να πληροί τις απαιτήσεις της.

Είναι πλήρης: Μια πλήρης βιβλιοθήκη επιτελεί την αναμενόμενη λειτουργία. Επίσης, εάν μια συνάρτηση μέλος είναι σε λάθος κλάση, πολλοί δυνητικοί χρήστες της λειτουργίας, δεν θα την εντοπίσουν.

Έχουν σαφή και απλή σημασιολογία: Όπως και με άλλες εργασίες σχεδιασμού, θα πρέπει να εφαρμόζουν την αρχή της λιγότερης έκπληξης. Κάνουν κοινές εργασίες εύκολα. Σπάνιες εργασίες πρέπει να είναι δυνατόν να γίνουν, αλλά δεν είναι το ζητούμενο.

Είναι διαισθητικές: Όπως και με οτιδήποτε άλλο σε έναν υπολογιστή, μια βιβλιοθήκη θα πρέπει να είναι διαισθητική. Διαφορετική εμπειρία και ιστορικό οδηγεί σε διαφορετικές αντιλήψεις για το τι είναι διαισθητικό και τι δεν είναι. Μια βιβλιοθήκη είναι διαισθητική, εάν ένας άπειρος χρήστης την χρησιμοποιεί χωρίς να

χρειάζεται η ανάγνωση των εγγράφων, και εάν ένας προγραμματιστής που δεν γνωρίζει την βιβλιοθήκη μπορεί να καταλάβει τον κώδικα της.

Είναι εύκολο να απομνημονευτεί: Να καταστεί η βιβλιοθήκη εύκολο να την θυμόμαστε, να επιλέξετε μια συνεπή και συγκεκριμένη ονομασία. Χρησιμοποιήστε αναγνωρίσιμα σχήματα και έννοιες, και να αποφεύγονται οι συντομογραφίες.

Ο Κώδικας να είναι αναγνώσιμος : Ο κώδικας γράφεται μία φορά, αλλά διαβάζεται (αποσφλαματώνεται και αλλάζει) πολλές φορές. Ο αναγνώσιμος κώδικας μπορεί μερικές φορές να απαιτεί περισσότερο χρόνο για να γραφεί, αλλά εξοικονομεί χρόνο σε όλο τον κύκλο ζωής του προϊόντος.

Η Boolean Παράμετρος : Boolean παράμετροι συχνά οδηγούν σε δυσανάγνωστο κώδικα. Πιο συγκεκριμένα, είναι σχεδόν πάντα λάθος να προσθέσετε μια bool παράμετρο σε μια υπάρχουσα λειτουργία. Ο προβληματισμός έγκειται στο γεγονός ότι η παράμετρος bool αποθηκεύει μια λειτουργία, βοηθώντας έτσι τη μείωση του μεγέθους της λειτουργίας.

Ονοματοδοσία στις λειτουργίες και στις παραμέτρους: Το νούμερο ένα κανόνας της ονοματοδοσίας στις λειτουργίες είναι ότι θα πρέπει να είναι σαφές από το όνομα εάν η λειτουργία έχει παρενέργειες ή μη. Τα ονόματα των παραμέτρων αποτελούν σημαντική πηγή πληροφοριών για τον προγραμματιστή, ακόμη και αν δεν εμφανίζονται στον κώδικα που χρησιμοποιεί η βιβλιοθήκη.

Γράψτε στην βιβλιοθήκη σας νωρίς και συχνά: Ο κώδικας υπάρχει για πάντα και γίνεται κατανοητός με παραδείγματα και δοκιμές μονάδων.

Η εφαρμογή δεν πρέπει να προσκρούσει στην βιβλιοθήκη: Οι λεπτομέρειες της εκτέλεσης συγχέουν τους χρήστες και εμποδίζουν την ελευθερία να αλλάξει η εφαρμογή.

Θέματα τεκμηρίωσης: Μια βιβλιοθήκη επειδή διαβάζεται (αρκετές φορές), είναι κατανοητή από άλλους και έτσι είναι σύντομη, ακριβής και καλύπτει κάθε ενιαία κλάση/λειτουργία

Απόδοση σχεδίασης βιβλιοθήκης: Οι κακές αποφάσεις μπορούν να περιορίσουν την απόδοση (Παρέχει κατασκευαστή αντί της στατικής μεθόδου. Χρησιμοποιεί τον τύπο εφαρμογής αντί της διεπαφής).

Οι άνθρωποι τείνουν να συνδέσουν την ακρίβεια με την κομψότητα. Με παρόμοιο τρόπο, όσον αφορά την συζήτηση των ιδιοτήτων μιας καλής βιβλιοθήκης, πολλοί θα πουν ότι μια καλή βιβλιοθήκη πρέπει να είναι όμορφη. Πράγματι, το να είναι οτιδήποτε όμορφο είναι ένα πλεονέκτημα. Μια καλή εντύπωση δημιουργείται πάντα μετά την πρώτη επαφή και εάν κάτι είναι συμπαθητικό, έχει πολύ μεγαλύτερη πιθανότητα της αποδοχής. Γι' αυτό είναι αποδεκτό να προσπαθήσει να σχεδιάσει κανείς μια όμορφη βιβλιοθήκη, αλλά η ομορφιά δεν μπορεί να είναι το μόνο μέτρο μιας καλής βιβλιοθήκης. Ο αρχικός στόχος της εφαρμοσμένης μηχανικής είναι να παραχθούν τα συστήματα εργασίας. Το γεγονός ότι βαθιά στα μυαλά μας έχουμε την παλαιά ελληνική κληρονομιά, το συναίσθημα ότι εάν κάτι είναι σωστό, αυτό πρέπει

επίσης να είναι όμορφο, δεν πρέπει να μας αποσπάσει από το κύριο σκοπό: σχεδιάζοντας βιβλιοθήκες που είναι εύχρηστες, ευρέως χρησιμοποιήσιμες, και παραγωγικές.

Μια προσέγγιση εφαρμοσμένης μηχανικής χρειάζεται έναν αντικειμενικό τρόπο να μετρήσει την ποιότητα των προϊόντων του. Στα εξής τμήματα, θα μελετήσουμε τις διάφορες πτυχές ορατές στο χρήστη μιας βιβλιοθήκης, και θα αναλύσουμε τη σημασία και τους τρόπους να προσαρμοστούν σε αυτούς.

Δυνατότητα κατανόησης

Εκείνοι που πρέπει να χρησιμοποιήσουν μια βιβλιοθήκη πρέπει να είναι σε θέση να την καταλάβουν. Αυτό είναι μια ασαφής δήλωση, αλλά και το σημαντικότερο. Μια βιβλιοθήκη είναι για την επικοινωνία μεταξύ του προγραμματιστή που γράφει την βιβλιοθήκη και τον προγραμματιστή που γράφει μια εφαρμογή χρησιμοποιώντας εκείνη την βιβλιοθήκη.

Οι έννοιες που παρέχονται από μια καλή βιβλιοθήκη πρέπει να βρεθούν μέσα στον ορίζοντα του χρήστη, διαφορετικά δεν θα γίνουν κατανοητοί. Ο σχεδιαστής μιας βιβλιοθήκης πρέπει να καταλάβει αυτό που είναι κοινή λογική για τους χρήστες και να χρησιμοποιήσει αυτήν την γνώση για την σχεδίαση της βιβλιοθήκης.

Όταν κωδικοποίηση μιας βιβλιοθήκης είναι άγνωστη, υπάρχει μια απλή λύση βασισμένη στην παρατήρηση ότι οι περισσότεροι χρήστες αναζητούν μεταξύ των υπάρχουσών εφαρμογών, μια εφαρμογή που κάνει κάτι παρόμοιο με το στόχο τους, αντιγράφουν την εφαρμογή, και την τροποποιούν για να ανταποκριθεί στις ανάγκες τους. Γι' αυτό δεν έχει σημασία πόσο διαφορετική μπορεί να είναι μια βιβλιοθήκη. Έχοντας πολλά παραδείγματα για το πώς να την χρησιμοποιήσουν οι ενδιαφερόμενοι χρήστες, αυξάνει την πιθανότητα ότι οι υπεύθυνοι για την ανάπτυξη μπορούν να βρουν κάτι κοντά στις ανάγκες τους.

Συνέπεια

Μια άλλη σημαντική πτυχή μιας βιβλιοθήκης που μπορεί να ελεγχθεί εύκολα είναι εάν είναι συνεπής. Εάν οι χρήστες μιας βιβλιοθήκης πρέπει να επενδύσουν το χρόνο να μάθουν μια έννοια, είναι σημαντικό ότι η έννοια εφαρμόζεται με συνέπεια σε ολόκληρη την βιβλιοθήκη.

Είναι προτιμητέο να επιτραπεί η ίδια κλωστή πρότυπο να χρησιμοποιηθεί από ολόκληρη την βιβλιοθήκη ή την κατηγορία, παρά η τεκμηρίωση ότι ορισμένες μέθοδοι πρέπει να κληθούν μόνο από ένα ορισμένο νήμα. Παρά το γεγονός ότι οι περισσότερες εφαρμογές είναι καλά σχεδιασμένες, η συμβατότητα ορισμένων κομματιών πάει να υπάρχει και ο τρόπος που οι υπεύθυνοι για την ανάπτυξη ενημερώνονται για αυτό είναι ότι δεν μπορούν πλέον να συντάξουν τον κώδικα που λειτουργεί στις προηγούμενες εκδόσεις. Αυτή η εξέλιξη, εκτός αν έχει σαφώς συμφωνηθεί, είναι το χειρότερο είδος προδοσίας της εμπιστοσύνης.

Δυνατότητα χρήσης

Ακόμη και η ομορφότερη βιβλιοθήκη είναι άχρηστη εάν εκείνοι που πρόκειται να την χρησιμοποιήσουν δεν μπορούν να την βρουν ή να καταλάβουν εύκολα πώς να την

χρησιμοποιήσουν. Λίγες καταστάσεις είναι χειρότερες από εκείνη στην οποία βρίσκοντας ένα πρόγραμμα που υποστηρίζει ότι παρέχει μια βιβλιοθήκη για να βοηθήσει να λυθεί το πρόβλημά σας, αλλά σας παραπέμπει σε μια εφαρμογή με 5 πακέτα και 30 κλάσεις χωρίς καμιά καθοδήγηση. Στις περισσότερες περιπτώσεις, το σύνολο των κλάσεων δεν ενδιαφέρει τους περισσότερους χρήστες βιβλιοθηκών. Ενδιαφέρονται απλά να κάνουν την εργασία τους. Για αυτούς τους λόγους είναι σημαντικότερο να δουν τα παραδείγματα της χρήσης βιβλιοθηκών, που επιτρέπουν την επιλογή της έκδοσης που είναι πιο κοντά σε αυτό που θέλουν να κάνουν. Ανεξάρτητα από ποιο τύπο της βιβλιοθήκης παρέχετε, αυτό που είναι σημαντικό είναι να δημιουργηθεί μια ενιαία θέση που μπορεί να χρησιμεύσει ως μια αφετηρία και μπορεί να στείλει τους ανθρώπους στις κατευθύνσεις που λύνουν τα προβλήματά τους. Δεδομένου ότι οι άνθρωποι δεν σκέφτονται από την άποψη των κλάσεων, είναι σημαντικό να οργανωθεί αυτό το σημείο εισόδων κατά τρόπο βέλτιστο, βασισμένο στους πραγματικούς ή τουλάχιστον αναμενόμενους στόχους.

Οι απλοί στόχοι πρέπει να είναι εύκολοι

Μερικές φορές μια βιβλιοθήκη υποστηρίζει πολλαπλών στόχων ακροατήρια. Παραδείγματος χάριν, μια βιβλιοθήκη επιτρέπει στους ανθρώπους να ανακαλύψουν ένα αντικείμενο από το όνομά του και επίσης για να συνδέσει τα αντικείμενα βάσει των ονομάτων τους. Αυτές οι δύο ενέργειες δημιουργούνται για δύο απολύτως διαφορετικές ομάδες χρηστών της βιβλιοθήκης.

Αυτό το είδος του διαχωρισμού είναι σημαντικότερο από την τεκμηρίωση της βιβλιοθήκης δεδομένου ότι παρέχει ευελιξία: οι διαφορετικές ομάδες χρηστών της βιβλιοθήκης μπορούν εύκολα να στρέψουν το ενδιαφέρον στην μελέτη της χωρίς να πρέπει να υποβληθεί η ερώτηση, «είναι αυτό κάτι το οποίο πρέπει να γνωρίζω;» για κάθε κλάση που βλέπουν στα περιεχόμενα τεκμηρίωσης.

Κάνοντας αυτό το είδος του διαχωρισμού λανθασμένα στοχεύοντας σε λάθος ομάδα χρηστών, μειώνεται χαρακτηριστικά η χρησιμότητα της βιβλιοθήκης.

ΚΕΦΑΛΑΙΟ 3

ΑΞΙΟΛΟΓΗΣΗ ΛΟΓΙΣΜΙΚΟΥ

3.1 Περιγραφή Μετρικών – Στατιστικά Σημαντικών

Τι είναι μετρική (6);

Οι μετρικές χρησιμοποιούνται για να προσδιορίσουν τα προβλήματα στη βάση κώδικα. Τα αποτελέσματα από τις μετρικές μπορούν να βοηθήσουν να στραφεί η προσοχή της ομάδας στα πιο σύνθετα τμήματα του κώδικα. Οι κύριες κατηγορίες μετρικών είναι οι βασικές, συνοχής, πολυπλοκότητας, σύζευξης, κληρονομικότητας, κληρονομικότητας – βασισμένης στην σύζευξη, μέγιστες, πολυμορφισμού, αναλογίας. Η επιλογή των συγκεκριμένων κατηγοριών μετρικών έγινε με κριτήριο που διερευνούν την ποιότητα του κώδικα του συστήματος που εξετάζουμε.

Attribute Complexity (AC)

Ορίζεται ως το άθροισμα των ιδιοτήτων σε κάθε κλάση. Μπορείτε να οργανώσετε την βαρύτητα για τους τύπους, τον τύπο enum και τον τύπο σειράς χωριστά.

Χρησιμοποίησε «*» για να καθοριστούν οι τύπους ενός πακέτου με όλα τα υπο - πακέτα του. Παραδείγματος χάριν, `java.lang.*` σημαίνει ότι η σειρά καθορίζει όλες τις κατηγορίες του πακέτου `java.lang` και του υπό - πακέτου της. Για να επεξεργαστείτε όλους τους τύπους που δεν απαριθμούνται στον πίνακα, διευκρινίστε την τελευταία σειρά με «*».

Η σειρά είναι σημαντική, επειδή ο έλεγχος των ιδιοτήτων πηγαίνει από την κορυφή του πίνακα προς τα κάτω. (Οι επαναλήψεις ενός τύπου δεν μετρούνται, έτσι εάν ένας συγκεκριμένος τύπος ακολουθεί έναν γενικότερο τύπο που συμπεριλαμβάνεται ήδη, ο συγκεκριμένος τύπος δεν μετρείται. Παραδείγματος χάριν, ο τύπος `java.lang.*` δεν θα μετρηθεί εάν ακολουθείται από τον τύπο `java.*`.)

Attribute Inheritance Factor (AIF)

Η συγκεκριμένη μετρική προέρχεται από την ακολουθία MOOD (μετρικές για την αντικειμενοστρεφή ανάπτυξη). Υπολογίζεται ως κλάσμα. Ο αριθμητής είναι το ποσό των κληρονομημένων ιδιοτήτων σε όλες τις κλάσεις στο πρόγραμμα. Ο παρονομαστής είναι ο συνολικός αριθμός διαθέσιμων ιδιοτήτων (που καθορίζονται τοπικά ως κληρονομημένες) για όλες τις κατηγορίες.

Average Inheritance Usage Ratio (AIUR)

Καθορίζουμε την μετρική AIUR για μια παραγόμενη κλάση, ως την μέση τιμή της μετρικής IUR που υπολογίζεται μεταξύ εκείνης της κλάσης και όλων των προγόνων

της. Για παράδειγμα, για την κλάση Γ, θεωρούμε το σύνολο των προγόνων της να είναι {AI}, (i=1, NOA):

$$AIUR = \frac{\sum_{i=1}^{NOA} IUR(C, A_i)}{NOA}$$

Όπου NOA είναι ο αριθμός των προγόνων κλάσεων της κλάσης Γ.

Access Of Foreign Data (AOFD)

Η Μετρική AOFD αντιπροσωπεύει τον αριθμό εξωτερικών κλάσεων από τις οποίες μια δεδομένη κλάση έχει πρόσβαση στις ιδιότητες των κλάσεων, άμεσα ή μέσω των μεθόδων accessor. Όσο υψηλότερη η αξία AOFD για μια κατηγορία, τόσο υψηλότερη η πιθανότητα ότι η κατηγορία είναι έτοιμη ή πρόκειται να γίνει μια unfocused-κλάση. Οι εσωτερικές κλάσεις και υπέρ - κλάσεις δεν μετρούνται.

Average Use of Interface (AUF)

Η μετρική AUF ορίζεται ως ο μέσος αριθμός των μερών διεπαφής μιας κλάσης που χρησιμοποιούνται από μια άλλη κλάση. Η μετρική AUF υπολογίζεται συνολικά ως ο αριθμός των χρησιμοποιημένων μελών για κάθε μια από τις πελάτες-κλάσεις και τη διαίρεση του με τον συνολικό αριθμό κλάσεων πελατών (COC).

Coupling Between Objects (CBO)

Αντιπροσωπεύει τον αριθμό άλλων κλάσεων με τις οποίες μια κλάση συνδέεται . Μετρά τον αριθμό τύπων αναφοράς που χρησιμοποιούνται στις δηλώσεις ιδιοτήτων, επίσημες παράμετροι, τύποι επιστροφής , τοπικές μεταβλητές. Οι πρωτότυπο τύποι, τύποι από το πακέτο java.lang και supertypes δεν μετρούνται.

Η υπερβολική σύζευξη μεταξύ των αντικειμένων είναι καταστρεπτική στο σχέδιο διαμόρφωσης και αποτρέπει την επαναχρησιμοποίηση. Όσο πιο ανεξάρτητη είναι μια κλάση, τόσο πιο εύκολα μπορεί να επαναχρησιμοποιηθεί σε μια άλλη εφαρμογή. Προκειμένου να βελτιωθεί η διαμόρφωση και να προωθηθεί η ενθυλάκωση, η σύζευξη κλάσης διά-αντικειμένου πρέπει να περιοριστεί στο ελάχιστο. Όσο μεγαλύτερος ο αριθμός σύζευξης, τόσο ψηλότερη η ευαισθησία στις αλλαγές σε άλλα μέρη του σχεδίου, και επομένως η συντήρηση είναι δυσκολότερη. Όσο υψηλότερη η σύζευξη κλάσης διά-αντικειμένου, τόσο αυστηρότερες είναι οι ανάγκες ελέγχου.

Class Locality (CL)

Η μετρική CL υπολογίζεται ως ο σχετικός αριθμός των εξαρτήσεων που μια κλάση έχει πακέτο της. Προκειμένου να υπολογιστεί η μετρική θα διαιρέσουμε την αξία CBO με το συνολικό αριθμό κλάσεων από τον οποίο εξαρτάται η υπό μέτρηση κλάση. Οι εσωτερικές κλάσεις δεν πρέπει να μετρηθούν.

Clients Of Class (COC)

Η μετρική COC ορίζεται ως ο αριθμός κλάσεων που χρησιμοποιούν τη διεπαφή της υπό - μέτρηση κλάσης. Οι εσωτερικές κλάσεις δεν μετρούνται. Στα πλαίσια αυτής της μετρικής, μια κλάση A χρησιμοποιεί την διεπαφή μιας κλάσης Γ εάν (τουλάχιστον) καλεί μια δημόσια μέθοδο ή έχει πρόσβαση σε μια δημόσια ιδιότητα εκείνης της κλάσης.

Comment Ratio (CR)

Μετρά την αναλογία των σχολίων τεκμηρίωσης ή/και εφαρμογής των συνολικών γραμμών κώδικα (τα σχόλια συμπεριλαμβάνονται στην αρίθμηση κώδικα).

Μπορείτε επίσης να διευκρινίσετε ποιος τύπος σχολίων να χρησιμοποιήσετε για την αναλογία.

- Τα σχόλια τεκμηρίωσης είναι σχόλια Javadoc.
- Τα σχόλια εφαρμογής είναι οποιοσδήποτε άλλος τύπος σχολίων.

Data Abstraction Coupling (DAC)

Η μετρική DAC μετρά τον αριθμό τύπων αναφοράς που χρησιμοποιούνται στις δηλώσεις ιδιοτήτων. Οι πρωτότυποι τύποι, τύποι από το πακέτο `java.lang` και `superTypes` δεν μετρούνται.

Depth Of Inheritance Hierarchy (DOIH)

Το μήκος της αλυσίδας κληρονομιάς από τη ρίζα του δέντρου κληρονομιάς στην υπό - μέτρηση κλάση είναι η μετρική DOIH για την κλάση.

FanOut (FO)

Η μετρική FO μετρά τον αριθμό τύπων αναφοράς που χρησιμοποιούνται στις δηλώσεις ιδιοτήτων, επίσημες παράμετροι, τύποι επιστροφής, ρίχνουν τις δηλώσεις και τις τοπικές μεταβλητές. Οι απλοί τύποι και supertypes δεν μετρούνται.

Inheritance Usage Ratio (IUR)

Η μετρική IUR είναι μια μετρική που καθορίζεται μεταξύ μιας υποκλάσης και μιας από τις προγονές κλάσεις της. Είναι ο σχετικός αριθμός των κληρονομημένων μελών από την κλάση προγόνων που χρησιμοποιείται στην παραγόμενη κλάση.

Ένα μέλος μιας κλάσης προγόνων είναι κληρονομημένο - συγκεκριμένο μέλος εάν η χρήση της συσχετίζεται με την κληρονομιά. Προσδιορίζουμε τα ακόλουθα κληρονομημένα - συγκεκριμένα μέλη:

- προστατευμένα μέλη και μέθοδοι στοιχείων
- μη-ιδιωτικές εικονικές μέθοδοι.

Το IUR υπολογίζεται με τον υπολογισμό του αριθμού κληρονομημένων - συγκεκριμένων μελών της κλάσης προγόνων που χρησιμοποιούνται στην υποκλάση, και το διαιρούν έπειτα με τους συνολικούς αριθμούς κληρονομημένων - συγκεκριμένων μελών από την κλάση πρόγονο. Οι μόνες χρήσεις που μετρούνται είναι: η πρόσβαση των προστατευμένων μελών στοιχείων, η κλήση των προστατευμένων μεθόδων και ο επαναπροσδιορισμός μιας εικονικής μεθόδου.

Method Inheritance Factor (MIF)

Αυτή η μετρική είναι από την ακολουθία MOOD (μετρικές για την αντικειμενοστρεφή ανάπτυξη). Υπολογίζεται ως μέρος. Ο αριθμητής είναι το ποσό των κληρονομημένων μεθόδων σε όλες τις κλάσεις στο πρόγραμμα. Ο παρονομαστής είναι ο συνολικός αριθμός διαθέσιμων μεθόδων (που καθορίζονται τοπικά ως κληρονομημένες) για όλες τις κλάσεις.

Maximum Number Of Levels (MNOL)

Μετρά το μέγιστο βάθος των επαναλήψεων εάν, για και ενώ στο σώμα των μεθόδων. Λογικές μονάδες με έναν υψηλό αριθμό ενθυλακωμένων επιπέδων χρήζουν απλοποίηση της διαδικασίας εφαρμογής, επειδή οι ομάδες που περιέχουν περισσότερα από επτά κομμάτια πληροφοριών είναι όλο και περισσότερο σκληρότερες στην επίλυση προβλημάτων.

Number of Client Classes (NCC)

NOCP είναι ο αριθμός άλλων πακέτων που χρησιμοποιούν το υπό - μέτρηση πακέτο. Ένα πακέτο χρησιμοποιεί ένα άλλο πακέτο εάν τουλάχιστον μια από τις κλάσεις του χρησιμοποιούν εκείνο το πακέτο (δηλ. καλεί τις μεθόδους, έχει πρόσβαση στις ιδιότητες ή επεκτείνει μια κλάση που δηλώνεται σε εκείνη το πακέτο).

Number Of External Dependencies (NOED)

NOED είναι ο αριθμός κλάσεων από άλλα πακέτα από τα οποία η υπό - μέτρηση κλάση εξαρτάται . Μια κλάση A εξαρτάται από μια άλλη κλάση Ψ, εάν η κλάση A καλεί τις μεθόδους ή/και έχει πρόσβαση στις ιδιότητες ή/και επεκτείνει την κλάση B.

Number Of Remote Methods (NORM)

Επεξεργάζεται όλες τις μεθόδους και τους κατασκευαστές και μετρά τον αριθμό διάφορων αποκαλούμενων μακρινών μεθόδων. Μια μακρινή μέθοδος ορίζεται ως μια μέθοδος που δεν δηλώνεται η ίδια στην κλάση ή στους προγόνους της.

Package Usage Ratio (PUR)

PUR μετρικό ορίζεται ως ο σχετικός αριθμός κλάσεων από το υπό - μέτρηση πακέτο που χρησιμοποιούνται από τη εξωτερική όψη . Ο αριθμός των κλάσεων που χρησιμοποιούνται θα διαιρεθεί με το συνολικό αριθμό κλάσεων στο πακέτο, από το οποίο αποκλείουμε τις εσωτερικές κλάσεις, δηλ. CP. Κατά συνέπεια:

$$PUR = \frac{PIS}{PS}$$

Tight Class Cohesion (TCC)

Η μετρική TCC ορίζεται ως ο σχετικός αριθμός άμεσα συνδεδεμένων μεθόδων. Δύο μέθοδοι συνδέονται άμεσα εάν έχουν πρόσβαση σε μια κοινή μεταβλητή περίπτωσης της κλάσης.

True Comment Ratio (TCR)

Μετρά την αναλογία των σχολίων τεκμηρίωσης ή/και εφαρμογής των συνολικών γραμμών κώδικα (όλα τα σχόλια αποκλείονται από την αρίθμηση κώδικα).

Μπορείτε επίσης να διευκρινίσετε ποιος τύπος σχολίων να χρησιμοποιήσετε για την αναλογία.

- Τα σχόλια τεκμηρίωσης είναι σχόλια Javadoc.
- Τα σχόλια εφαρμογής είναι οποιοσδήποτε άλλος τύπος σχολίων.

Violations of Demeter's Law (VOD)

Ο νόμος Demeter

Καθορισμός 1 (πελάτης) μέθοδος M είναι πελάτης της μεθόδου φ που συνδέεται με την κλάση Γ, εάν στο εσωτερικό της M το μήνυμα φ στέλνεται σε ένα αντικείμενο της κλάσης Γ, ή στην κλάση Γ. Εάν το φ είναι εξειδικευμένο σε μια ή περισσότερες υποκλάσεις, κατόπιν το M είναι μόνο πελάτης του φ που συνδέεται με την υψηλότερη κλάση στην ιεραρχία. Η μέθοδος M είναι πελάτης κάποιας μεθόδου που συνδέεται με την Γ.

Καθορισμός 2 (προμηθευτής) Εάν η M είναι πελάτης της κλάσης Γ έπειτα η Γ είναι προμηθευτής στο M. Με άλλα λόγια, μια κλάση προμηθευτών σε μια μέθοδο είναι μια κλάση οι μέθοδοι της οποίας καλούνται στη μέθοδο.

Καθορισμός 3 (κατηγορία εξοικείωσης) Μια κλάση C1 είναι μια κλάση εξοικείωσης μεθόδου M που συνδέεται με την κλάση C2, εάν C1 είναι προμηθευτής στο M και C1 δεν είναι ένα από τα ακόλουθα:

1. όπως C2
2. μια κλάση που χρησιμοποιείται στη δήλωση ενός επιχειρήματος της M
3. μια κλάση που χρησιμοποιείται στη δήλωση μιας μεταβλητής περίπτωσης C2

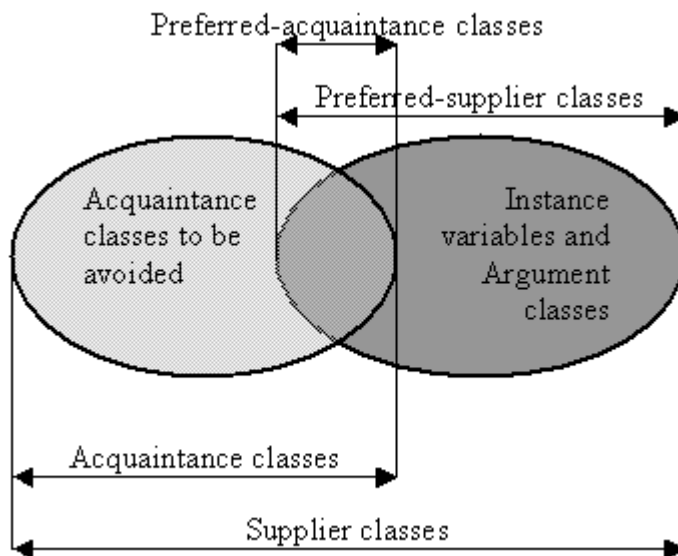
Καθορισμός 4 (κατηγορία προτιμώ-εξοικείωσης) Μια κατηγορία προτιμώ-εξοικείωσης μεθόδου M είναι είτε:

1. μια κλάση αντικειμένων που δημιουργήθηκαν άμεσα στην M, ή
2. μια κλάση που χρησιμοποιείται στη δήλωση μιας καθολικής μεταβλητής που χρησιμοποιείται στην M.

Σημείωση πραγματοποίησης. Η άμεση δημιουργία σημαίνει ότι ένα δεδομένο αντικείμενο δημιουργείται μέσω του χειριστή νέο.

Καθορισμός 5 (κατηγορία προτιμώ-προμηθευτών) Η κλάση Ψ καλείται προτιμώ-προμηθευτή στη μέθοδο Μ (που συνδέεται με την κλάση Γ) εάν το Β είναι προμηθευτής στην Μ και ικανοποιείται ένας από τους ακόλουθους όρους :

1. Το Β χρησιμοποιείται στη δήλωση μιας μεταβλητής περίπτωσης του Γ, 2. Το Β χρησιμοποιείται στη δήλωση ενός επιχειρήματος της Μ, συμπεριλαμβανομένου του Γ και των υπέρ - κλάσεις του,
3. Το Β είναι μια προτιμημένη κλάση εξοικείωσης της Μ.



Η σχέση μεταξύ των διάφορων τύπων των κλάσεων προμηθευτών

Η μορφή κλάσης του νόμου Demeters έχει δύο εκδόσεις: μια ακριβής έκδοση και μια έκδοση ελαχιστοποίησης. Η ακριβής μορφή του νόμου δηλώνει ότι κάθε κλάση προμηθευτών μιας μεθόδου πρέπει να είναι προτιμημένος προμηθευτής.

Η μορφή ελαχιστοποίησης είναι πιο ελαστική από την πρώτη έκδοση και απαιτεί μόνο να ελαχιστοποιήσει τον αριθμό κατηγοριών εξοικείωσης κάθε μεθόδου.

Παρατηρήσεις.

1. Το κίνητρο πίσω από το νόμο Demeter είναι να εξασφαλιστεί ότι το λογισμικό είναι όσο το δυνατόν πιο μορφοματικό. Ο νόμος μειώνει αποτελεσματικά τα περιστατικά εμφωλευμένου μηνύματος στέλνει και απλοποιεί τις μεθόδους.

2. Ο καθορισμός του νόμου κάνει μια διαφορά μεταξύ των κλάσεων που συνδέονται με τη δήλωση της μεθόδου και των κλάσεων που χρησιμοποιούνται στο σώμα της μεθόδου, δηλ. οι κλάσεις που συνδέονται με την εφαρμογή του. Το πρώτο περιλαμβάνει την κλάση όπου η μέθοδος είναι συνημμένη, τα υπέρ - κλάσεις του, οι κλάσεις που χρησιμοποιούνται στις δηλώσεις των μεταβλητών περίπτωσης και οι

κλάσεις που χρησιμοποιούνται για να δηλώσουν τα επιχειρήματα της μεθόδου. Υπό κάποια έννοια, υπάρχουν «αυτόματες» συνέπειες της δήλωσης μεθόδου. Μπορούν να προέλθουν εύκολα από τον κώδικα και να παρουσιαστούν από μια μηχανή αναζήτησης. Όλες οι άλλες κλάσεις προμηθευτών στις μεθόδους εισάγονται στο σώμα της λειτουργίας, η οποία σημαίνει ότι αυτά τα ζεύγη δημιουργήθηκαν συγκεκριμένα για να εφαρμόσουν τη μέθοδο. Μπορούν μόνο να καθοριστούν από μια προσεκτική ανάγνωση της εφαρμογής.

Violations of Demeters Law - VOD

Ο καθορισμός αυτής της μετρικής είναι βασισμένος στη μορφή ελαχιστοποίησης του νόμου Demeter. Με βάση τις έννοιες που καθορίζονται εκεί, και λαμβάνοντας υπόψη ότι η μορφή ελαχιστοποίησης του νόμου Demeters απαιτεί ότι ο αριθμός κλάσεων εξοικείωσης πρέπει να κρατηθεί χαμηλός, καθορίζουμε την μετρική VOD:

Καθορισμός 6 (VOD μετρική) Σε μια κλάση Γ και Α του συνόλου όλων των κλάσεων εξοικείωσής του, $VOD(\Gamma) = |A|$

Ανεπίσημα, VOD είναι ο αριθμός κλάσεων εξοικείωσης μιας δεδομένης κλάσεων.

Κρατώντας την τιμή VOD για μια κλάση χαμηλή, προσφέρει διάφορα οφέλη, που απαριθμούνται κατωτέρω:

1. Έλεγχος συζεύξεων. Ένα πρόγραμμα με μια χαμηλή τιμή VOD είναι η ένδειξη της ελάχιστης σύζευξης «χρήσης» μεταξύ των αφαιρέσεων. Αυτό σημαίνει ότι ένας μειωμένος αριθμός μεθόδων μπορεί να επικαλεστεί. Αυτό καθιστά τις μεθόδους πιο επαναχρησιμοποιήσιμες.

2. Κρύψιμο δομών. Η μείωση VOD αντιπροσωπεύει στην πραγματικότητα τη μείωση της άμεσης ανάκτησης των υποτομέων «μέρος-» της ιεραρχίας. Με άλλα λόγια, τα δημόσια μέλη πρέπει να χρησιμοποιηθούν με έναν περιορισμένο τρόπο.

3. Εντοπισμός των πληροφοριών. Μια χαμηλή τιμή VOD επίσης σημαίνει ότι οι πληροφορίες κλάσης είναι εντοπισμένες. Αυτό μειώνει την πολυπλοκότητα προγραμματισμού.

Πηγή: Ing. Radu Marinescu. Μια αντικειμενοστρεφής ακολουθία μετρικών στη σύζευξη. Universitatea «Politehnica» Timisoara, Facultatea de Automatica Si Calculatoare, Departamentul de Calculatoare λογισμικό Si Inginerie. Σεπτεμβρίου 1998.

3.2 Περιγραφή Ελέγχων – Design Flaws

Τι είναι έλεγχος (audit);

Χρησιμοποιούμε τους λογιστικούς ελέγχους για να βρούμε και να καθορίσουμε τα προβλήματα πριν από την αναθεώρηση κώδικα. Μ' αυτόν τον τρόπο οι υπεύθυνοι για την ανάπτυξη μπορούν να εστιάσουν στη λογική του κώδικα παρά στην προσαρμογή στα εταιρικά πρότυπα. Οι λογιστικοί έλεγχοι αποτελούν και εκείνοι κριτήριο που αποδεικνύει την σωστή σχεδίαση λογισμικού.

Data Class (DATACLASS)

Οι κλάσεις δεδομένων είναι κάτοχοι στοιχείων και σχεδόν πάντα άλλες κλάσεις στηρίζονται πάνω τους. Η έλλειψη λειτουργικών μεθόδων μπορεί να δείξει ότι τα στοιχεία και η συμπεριφορά τους δεν κρατούνται σε μια θέση, αυτό είναι ένα σημάδι ενός μη αντικειμενοστρεφούς σχεδίου. Οι κλάσεις στοιχείων μειώνουν τη συντηρησιμότητα, τη δυνατότητα δοκιμής και την κατανόηση του συστήματος.

Data Clump (DCL)

Ο έλεγχος DCL ανιχνεύει τις κλάσεις που περιέχουν διάφορες ανεξάρτητες ομάδες πεδίων και μεθόδων. Μια τέτοια κλάση μπορεί να διαιρεθεί σε διάφορες μικρότερες κλάσεις, κάθε μια εκτελεί το στόχο της. Οι μικρότερες κλάσεις με τα καθορισμένα με σαφήνεια σύνολα ευθυνών είναι ευκολότερο να γίνουν κατανοητές και να τροποποιηθούν.

Feature Envy (FE)

Τα αντικείμενα είναι ένας μηχανισμός για να τοποθετηθούν μαζί τα δεδομένα και οι διαδικασίες που επεξεργάζονται τα δεδομένα. Αυτός ο έλεγχος αναφέρεται σε εκείνες τις μεθόδους που φαίνονται να αναφέρονται στα δεδομένα μιας άλλης κλάσης από αυτή που είναι πραγματικά. Αυτές οι μέθοδοι έχουν πρόσβαση άμεσα ή μέσω των μεθόδων accessor σε πολλά δεδομένα από μια άλλη κλάση. Αυτό μπορεί να είναι μια ένδειξη ότι η μέθοδος ήταν τοποθετημένη σε λάθος μέρος, και ότι πρέπει να μετακινηθεί προς μια άλλη κλάση.

Field is Used as Temporary Variable (FUTV)

Το FUTV ανιχνεύει τα πεδία που χρησιμοποιούνται ως προσωρινές μεταβλητές. Αυτό σημαίνει ότι μια τιμή που ορίζεται σε έναν τέτοιο πεδίο δεν χρησιμοποιείται με

οποιαδήποτε μέθοδο εκτός από τη μέθοδο που περιέχει την ανάθεση. Ένα προσωρινό πεδίο πρέπει να αντικατασταθεί από μια τοπική μεταβλητή. Το FUTV εξετάζει μόνο τα πεδία με την ιδιωτική πρόσβαση προεπιλογής, δεδομένου ότι τα δημόσια και προστατευμένα πεδία μπορούν να χρησιμοποιηθούν έξω από το αναλυθέν πακέτο.

God Class (GC)

Σε ένα καλό αντικειμενοστρεφές σχέδιο η νοημοσύνη ενός συστήματος διανέμεται ομοιόμορφα μεταξύ των υψηλότερων επιπέδων κλάσεων. Αυτός ο έλεγχος αναφέρει εκείνες τις κλάσεις που τείνουν να συγκεντρώσουν τη νοημοσύνη του συστήματος. Μια περίπτωση μιας God – κλάσης εκτελεί το μεγαλύτερο μέρος της εργασίας, που εξουσιοδοτεί μόνο τις δευτερεύουσες λεπτομέρειες σε ένα σύνολο τετριμμένων κλάσεων και που χρησιμοποιεί τα δεδομένα από άλλες κλάσεις. Αυτό ασκεί αρνητική επίδραση στην ικανότητα επαναχρησιμοποίησης και κατανόησης εκείνου του μέρους του συστήματος. Αυτό το πρόβλημα σχεδίου μπορεί να επιλυθεί μερικώς με την κακή-μυρωδιά «μεγάλη κλάση».

God Method (GM)

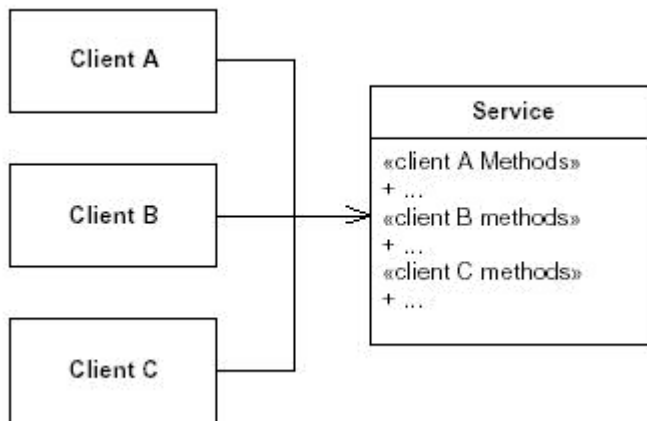
Με τον ίδιο τρόπο που οι God - κλάσεις συγκεντρώνουν τη λειτουργία ενός υποσυστήματος (ή μερικές φορές ολόκληρου του συστήματος), οι God - μέθοδοι τείνουν να κάνουν το ίδιο πράγμα στο επίπεδο κλάσης. Μερικές φορές μια μέθοδος αρχίζει ως «κανονική» μέθοδος, αλλά έπειτα όλο και περισσότερες λειτουργίες προστίθενται σε αυτήν, έως ότου παίρνει τίθεται εκτός ελέγχου και καθίσταται αδύνατο να διατηρηθεί ή να γίνει κατανοητή.

God Package (GP)

Εφαρμόζοντας αυστηρά την κοινή αρχή περάτωσης (CCP), τα πακέτα τείνουν να γίνουν πολύ μεγάλα και μη - συνεκτικά. Η κοινή αρχή περάτωσης (CCP) είναι άλλη μια αρχή σχεδίασης πακέτων λογισμικού που προσπαθεί να εξετάσει τη δυνατότητα χρησιμοποίησης και συντηρησιμότητας τους. Η αρχή συμβουλεύει ποιες κλάσεις πρέπει να ενοποιηθούν μαζί από άποψη αλλαγής και διανομής. Ένα άλλο σύμπτωμα αυτού του προτύπου σχεδίασης είναι ο μεγάλος αριθμός πελατών του πακέτου (δηλ. κλάσεις από άλλα πακέτα) που κάνουν υπερβολική χρήση του συγκεκριμένου πακέτου. Η συνέπεια είναι ότι οι πελάτες αυτού του πακέτου πρέπει να κάνουν εξονυχιστικό έλεγχο ακόμα κι αν η αλλαγή στο πακέτο δεν έχει επιπτώσεις σε αυτούς.

ISP Violation (ISPV)

Η αρχή διαχωρισμού διεπαφών δηλώνει ότι «πολλές συγκεκριμένες διεπαφές πελατών είναι καλύτερες από μια διεπαφή γενικού σκοπού». Το σχήμα παρουσιάζει μια κλάση με πολλούς πελάτες, και μια μεγάλη διεπαφή για να τους εξυπηρετήσει όλους. Σημειώστε ότι όποτε μια αλλαγή γίνεται σε μια από τις μεθόδους, σημαίνει ότι οι κλήσεις ClientA, ClientB και ClientC μπορούν να επηρεαστούν. Μπορεί να είναι απαραίτητο να ανασυντάξουμε και να τους ανακατανεύουμε. Αυτό είναι ανεπιτυχές.



Long Message Chain (LMC)

Ο έλεγχος LMC ψάχνει μια αλυσίδα εξουσιοδότησης των μεθόδων. Μια μέθοδος εξουσιοδότησης δεν εκτελεί οποιαδήποτε δράση αλλά καλεί τη μέθοδο εκπροσώπων. Το LMC παράγει ένα μήνυμα προειδοποίησης εάν το μήκος μιας τέτοιας αλυσίδας μηνυμάτων είναι υψηλότερο από την τιμή που ορίζεται από την ανώτατη ιδιοκτησία μήκους αλυσίδων κλήσης (2, εξ ορισμού).

Misplaced Class (MC)

Στο God – πακέτο , συχνά συμβαίνει να χρειάζεται μια κλάση τις κλάσεις από άλλα πακέτα περισσότερο από εκείνες που ανήκουν στο πακέτο της. Σε αυτήν την περίπτωση, ειδικά εάν η κλάση χρησιμοποιεί κυρίως ένα άλλο πακέτο, προσπαθήστε να μετακινήσετε την κλάση προς εκείνο το πακέτο.

Refused Bequest (RB)

Μια υπό κλάση κληρονομεί μερικές από τις μεθόδους και τα δεδομένα των γονέων της. Αυτός ο έλεγχος αναφέρεται στην περίπτωση που δεν χρησιμοποιούν οι υποκλάσεις τα μέλη των προγόνων τους που είχαν σχεδιαστεί συγκεκριμένα να επαναχρησιμοποιηθούν από τους απογόνους. Μπορεί να είναι μια ένδειξη ότι η ιεραρχία κάνει λάθος, από την άλλη αν η κλάση παιδιών αρνείται μόνο την κληρονομημένη εφαρμογή, αυτό δεν είναι τόσο κακό αλλά εάν ο απόγονος αρνείται τη διεπαφή έπειτα η ιεραρχία κάνει λάθος σίγουρα.

Προσδιορίζουμε τρεις τρόπους χρήσης ενός κληρονομημένου μέλους:

- κλήση μιας μεθόδου που καθορίζεται στις υπέρ - κλάσεις
- πρόσβαση μιας ιδιότητας που καθορίζεται στις υπέρ - κλάσεις
- αγνοήστε μια εικονική μέθοδο που δηλώνεται/που καθορίζεται στις υπέρ - κλάσεις

Return Zero-Length Array (RZLA)

Για να βελτιώσετε την αξιοπιστία του κώδικά σας, χρησιμοποιήστε πάντα τους πίνακες μηδενικού μήκους αντί των κενών αναφορών. Αυτό θα επιτρέψει σε σας για να αποφύγετε τους πρόσθετους ελέγχους για τις μηδενικές τιμές και να αποφύγετε τις μηδενικές εξαιρέσεις αναφοράς . Ο έλεγχος RZLA εξετάζει ότι οι μέθοδοι που επιστρέφουν τους τύπους πινάκων δεν επιστρέφουν ποτέ τις μηδενικές τιμές.

Shotgun Surgery (SS)

Αυτός ο έλεγχος αφορά μια αλλαγή που υπονοεί πολλές (μικρές) αλλαγές σε πολλές διαφορετικές κλάσεις. Οι αλλαγές που είναι διασκορπισμένες σε πολλές θέσεις είναι δύσκολο να βρεθούν. Κατά συνέπεια, η γρήγορη εύρεση των αλλαγών επηρεάζει τη συντηρησιμότητα του κώδικα. Με άλλα λόγια, αυτό η το έλεγχος αντιμετωπίζει το ζήτημα της ισχυρής σύζευξης στην εφαρμογή. Αλλά θεωρεί όχι μόνο τη δύναμη, αλλά και τη διασπορά της σύζευξης.

Subclasses Differ only in Methods returning Constant values (SDMC)

Ο έλεγχος SDMC ανιχνεύει εάν όλες οι υποκλάσεις μιας κλάσης ποικίλλουν μόνο στις μεθόδους που επιστρέφουν τα σταθερά στοιχεία. Σε τέτοιες περιπτώσεις, προτείνεται να αλλαχτούν οι μέθοδοι στα πεδία υπέρ - κλάσης και να διαγραφούν οι υποκλάσεις.

Subclasses have the Same Member (SSM)

Ο έλεγχος SSM ανιχνεύει είτε δύο είτε περισσότερες άμεσες υποκλάσεις μιας κλάσης ή την διεπαφή δηλώνει πεδία με το ίδιο όνομα και τύπο ή τις μεθόδους με το ίδιο όνομα και υπογραφή. Ο λογιστικός έλεγχος, εξετάζει το ποσοστό επί τοις εκατό των κλάσεων όπου το ίδιο μέλος δηλώνεται, διευκρινίζοντας το ελάχιστο ποσοστό των υποκλάσεων με το ίδιο μέλος. Εξ ορισμού, είναι ίσο με 0, και το SSM παράγει ένα μήνυμα όταν δηλώνεται το ίδιο μέλος από δύο ή περισσότερες υποκλάσεις. Εάν η επιλογή τίθεται 100, κατόπιν ένα μήνυμα θα παραχθεί μόνο όταν δηλώνεται το ίδιο μέλος από όλες τις υποκλάσεις.

Ο SSM λογιστικός έλεγχος παράγει το μήνυμα ακόμα κι αν το ίδιο πεδίο καθορίζεται σε διάφορες κλάσεις χρησιμοποιώντας την ίδια διεπαφή. Αν και ένα πεδίο δεν μπορεί να μεταφερθεί προς τη διεπαφή, ο προγραμματιστής μπορεί να εισαγάγει μια πρόσθετη κλάση εφαρμόζοντας αυτήν την διεπαφή και να μετακινήσει το πεδίο εκεί.

Suspicious switch statement Usage (SSU)

Ο έλεγχος SSU ανιχνεύει εάν μια μέθοδος περιέχει έναν τομέα “switch on” σε μια κλάση. Ένα πεδίο πρέπει να είναι ενός ακέραίου τύπου, μπορεί να προσεγγιστεί είτε άμεσα είτε μέσω μιας μεθόδου accessor. Τέτοιες εντολές SWITCH πρέπει συχνά να αντικατασταθούν με τη χρήση διάφορων υποκλάσεων.

Use Interface Instead of Implementation (UII)

Για να αυξήσετε την αφαίρεση στον κώδικά σας, πρέπει να επικαλεστείτε τις μεθόδους χρησιμοποιώντας τις διεπαφές και όχι τις κλάσεις που εφαρμόζονται από αυτές τις διεπαφές. Επομένως, αντί των αλλαγών στις κλάσεις εφαρμογής, είναι καλύτερο να κάνετε αλλαγές στην διεπαφή καθορίζοντας αυτήν την μέθοδο.

Use Singleton Pattern (USP)

Ο έλεγχος USP ανιχνεύει τις μη-τελικές κλάσεις που έχουν μόνο τα στατικά μέλη. Τέτοιες κλάσεις δεν μπορούν να προσαρμοστούν, δηλ., δεν είναι δυνατόν να αντλήσουν μια υποκλάση και να επαναπροσδιορίσουν μια ή περισσότερες μεθόδους, επομένως το έλεγχος Singleton πρέπει να χρησιμοποιηθεί αντ' αυτού. Σύμφωνα με αυτό το πρότυπο, όλες οι στατικές μεταβλητές και οι μέθοδοι πρέπει να αντικατασταθούν με τα μέλη περίπτωσης και πρέπει να υπάρξει μια στατική μέθοδος getInstance () που επιστρέφει την απλή περίπτωση της κλάσης.

Το Singletons πρέπει να δηλώσει τους ιδιωτικούς κατασκευαστές για να δείξει ότι δεν προορίζονται να είναι άμεσα.

Wide Subsystem Interface (Lack Of Facade) (WSI)

Αυτός ο έλεγχος εμπνέεται από το κίνητρο για το σχέδιο προσώπων. Παρόμοια με τις κλάσεις, μπορούμε επίσης να μιλήσουμε για τη διεπαφή ενός υποσυστήματος. Αυτή η διεπαφή αποτελείται από τις κλάσεις που προσεγγίζονται ή, ακριβέστερα, είναι προσιτές από τη εξωτερική όψη του πακέτου. Ο έλεγχος αναφέρεται στην κατάσταση όπου αυτή η διεπαφή είναι πολύ ευρεία, το οποίο προκαλεί μια πολύ σφιχτή σύζευξη μεταξύ του πακέτου και του υπόλοιπου κόσμου, η οποία είναι ανεπιθύμητη. Στις περισσότερες από τις περιπτώσεις η διεπαφή μπορεί να μειωθεί με την εισαγωγή μιας κλάσης προσώπων. Εάν αυτό δεν βοηθά, αποτελεί μια ένδειξη ότι εμπεριέχεται το έλεγχος «God - Package». Σε αυτήν την περίπτωση η πιθανότητα επανατοποθέτησης μερικών από τις κλάσεις πρέπει να εξεταστεί.

3.3 ΠΕΡΙΓΡΑΦΗ ΠΡΟΤΥΠΩΝ ΣΧΕΔΙΑΣΗΣ – DESIGN PATTERNS

Τι είναι ένα πρότυπο σχεδίασης (4);

Εάν ένα πρόβλημα εμφανίζεται επανειλημμένως, μια λύση σε εκείνο το πρόβλημα έχει χρησιμοποιηθεί αποτελεσματικά. Εκείνη η λύση περιγράφεται ως πρότυπο σχεδίασης. Τα πρότυπα σχεδίασης είναι στρατηγικές που δεν εξαρτώνται από την γλώσσα προγραμματισμού που έχει χρησιμοποιηθεί, για τα κοινά αντικειμενοστρεφή προβλήματα σχεδίασης. Όταν σχεδιάζετε ένα σύστημα, πρέπει να ξέρετε τα ονόματα μερικών κοινών λύσεων. Η εκμάθηση των προτύπων σχεδίασης είναι καλή για τους ανθρώπους για να επικοινωνήσουν μεταξύ τους αποτελεσματικά. Στην πραγματικότητα, μπορεί να είχατε εξοικειωθεί με μερικά πρότυπα σχεδίασης, δεν μπορείτε να χρησιμοποιήσετε τα γνωστά ονόματα για να τα περιγράψετε. Η εταιρεία SUN προτείνει GOF (συμμορία των τεσσάρων--τέσσερις τύποι πρωτοπόρων που έγραψαν ένα βιβλίο που ονομάστηκε τα «πρότυπα σχεδίασης» - στοιχεία του επαναχρησιμοποιήσιμου αντικειμενοστρεφούς λογισμικού), έτσι χρησιμοποιούμε εκείνο το βιβλίο ως οδηγό μας για να περιγράψουμε τις λύσεις.

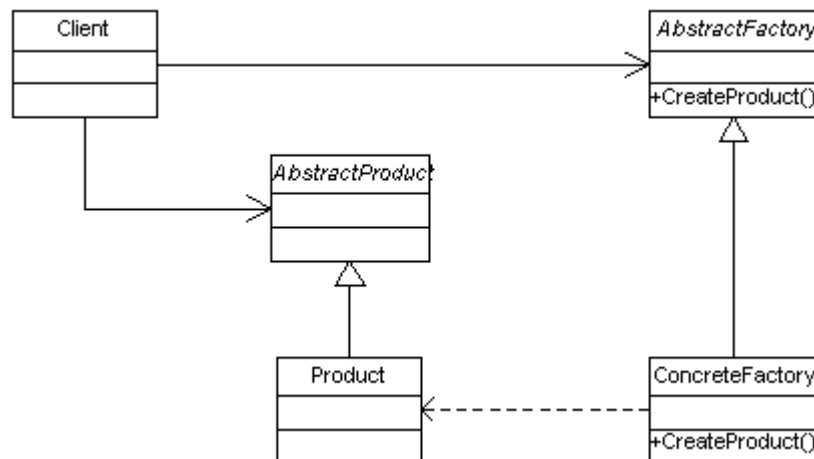
FACTORY METHOD

Παρέχει μια αφαίρεση ή μια διεπαφή και αφήνει την υποκατηγορία ή εφαρμογή των κατηγοριών να αποφασίσει ποια κατηγορία ή μέθοδος πρέπει να κληθεί, βασισμένη στους όρους ή τις παραμέτρους που δίνονται.

Χρήση - Πλεονεκτήματα

- Συνδέει τις παράλληλες ιεραρχίες κλάσεων.
- Μια κλάση χρειάζεται τις υποκλάσεις της για να προσδιορίσει το αντικείμενο.

- Μια κλάση δεν μπορεί να προλαμβάνει τις υποκλάσεις της, οι οποίες πρέπει να δημιουργηθούν.
- Μια οικογένεια των αντικειμένων πρέπει να χωριστεί με τη χρησιμοποίηση της κοινής διεπαφής.
- Ο κώδικας πρέπει να εξετάσει τη διεπαφή, όχι να θέσει σε εφαρμογή τις κλάσεις.
- Αποκρύψτε τις καθορισμένες κλάσεις από τον πελάτη.
- Οι μέθοδοι Factory μπορούν να παραμετροποιηθούν.
- Το επιστρεφόμενο αντικείμενο μπορεί να είναι είτε αφηρημένο είτε συγκεκριμένο αντικείμενο.



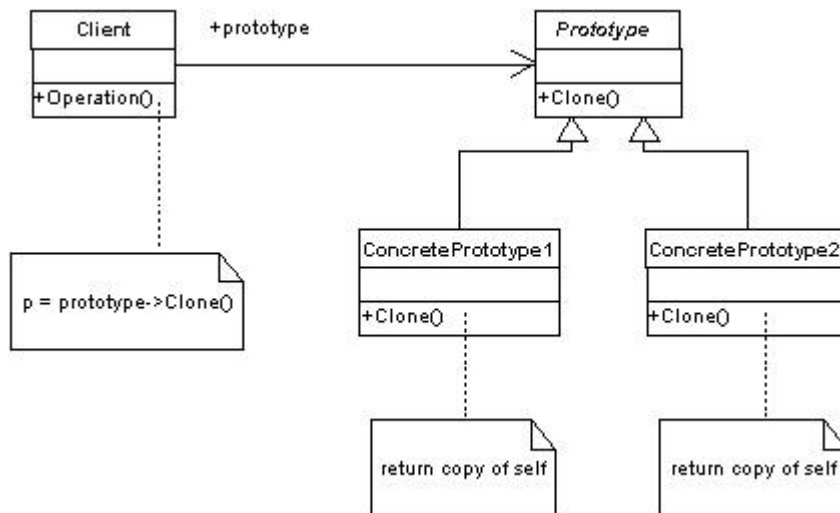
(6). Factory method UML diagram

PROTOTYPE

Περιγράφει την κλωνοποίηση ενός αντικειμένου με τη μείωση του κόστους της δημιουργίας

Χρήση - Πλεονεκτήματα

- Όταν υπάρχουν πολλές υποκλάσεις που διαφέρουν μόνο στο είδος αντικειμένων,
- Ένα σύστημα πρέπει να είναι ανεξάρτητο σχετικά με το πώς τα αντικείμενά του δημιουργούνται, συντίθενται, και αντιπροσωπεύονται.
- Δυναμική σύνδεση ή φόρτωση μια μεθόδου.
- Προσθέστε και αφαιρέστε τα αντικείμενα στο χρόνο εκτέλεσης.
- Προσδιορίστε τα νέα αντικείμενα με την αλλαγή της δομής του.
- Διαμορφώστε μια εφαρμογή με τις κλάσεις δυναμικά.



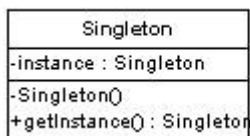
(7). Prototype UML diagram

SINGLETON

Περιγράφει μια περίπτωση μιας κλάσης ή μιας τιμής καθολικά προσπελάσιμης σε μια εφαρμογή.

Χρήση - Πλεονεκτήματα

- Εξασφαλίστε μοναδική περίπτωση με τον καθορισμό μιας κλάσης ως τελικής για να αποτρέψετε την κλωνοποίηση.
- Κάνετε μια μέθοδο ή μεταβλητή έναν δημόσια or/and στατική.
- Πρόσβαση στην περίπτωση από τον τρόπο που παρείχατε.
- Καλός έλεγχος περίπτωσης μιας κλάσης.
- Καθορίστε μια τιμή κοινή σε όλες τις περιπτώσεις κάνοντάς την στατική.



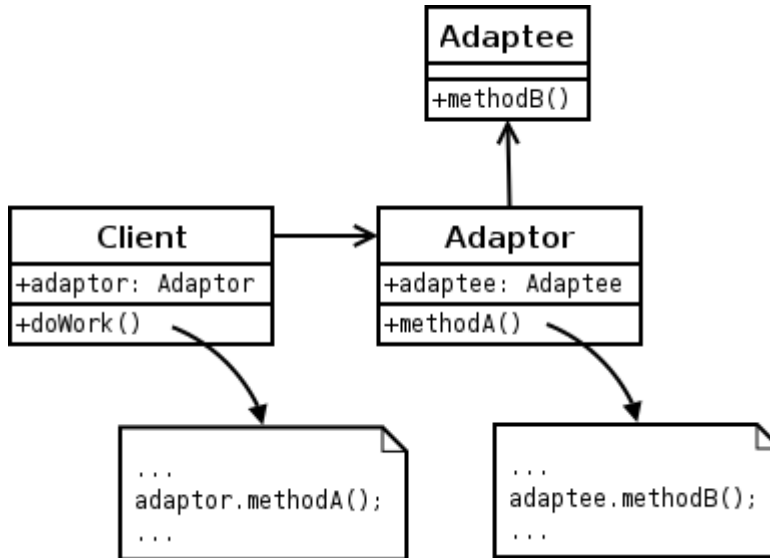
(8). Singleton UML diagram

(OBJECT) ADAPTER COMMAND

Μετατρέψτε τις υπάρχουσες διεπαφές σε μια νέα διεπαφή για να επιτύχετε τη συμβατότητα και την ικανότητα επαναχρησιμοποίησης των ανεξάρτητων κλάσεων σε μια εφαρμογή. Επίσης γνωστό ως πρότυπο σχεδίασης Wrapper.

Χρήση – Πλεονεκτήματα

- Προσπαθήστε να συνδυάσετε αρμονικά μια διεπαφή (WindowAdapter, κ.λπ.)
- Κάνετε τις ανεξάρτητες κλάσεις να δουλεύουν από κοινού.
- Πολλαπλασιάστε την συμβατότητα.
- Αυξήστε την διαφάνεια των κλάσεων.
- Επιτύχετε το στόχο από την κληρονομικότητα ή από την σύνθεση



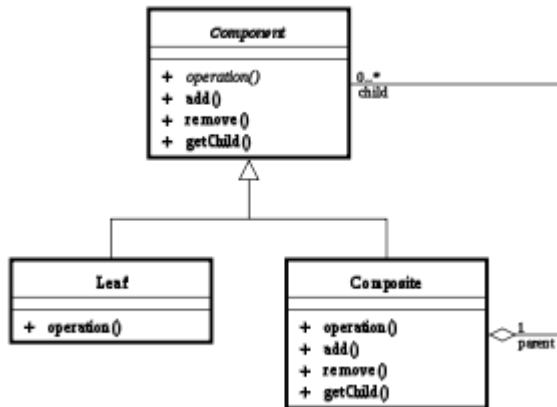
(9). Adapter Command UML diagram

COMPOSITE

Δημιουργήστε ένα σύνθετο αντικείμενο από τα στοιχειώδη αντικείμενα του ίδιου όπως μια δομή δέντρων.

Χρήση – Πλεονεκτήματα

- Θελήστε να αντιπροσωπεύσετε μια μερική - ολόκληρη σχέση όπως το σύστημα φακέλων δέντρων
- Ομαδοποιήστε τα συστατικά για να διαμορφώσετε τα μεγαλύτερα συστατικά, τα οποία μπορούν στη συνέχεια να ομαδοποιηθούν για να διαμορφώσουν τα ακόμα μεγαλύτερα συστατικά.



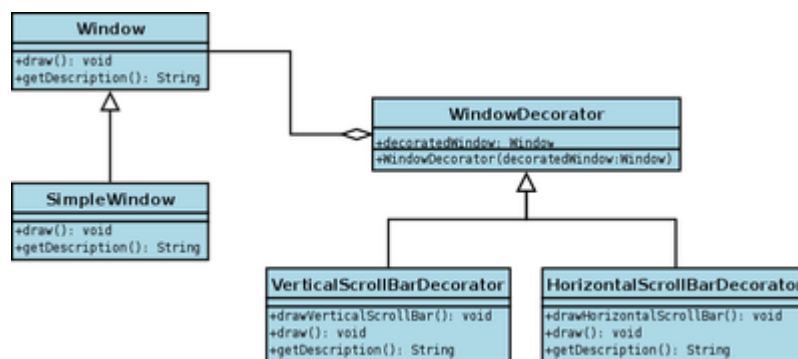
(10). Composite UML diagram

DECORATOR

Συνδέστε τις πρόσθετες ενέργειες ή τις λειτουργίες με ένα αντικείμενο δυναμικά ή στατικά.

Χρήση – Πλεονεκτήματα

- Παρέχετε μια εναλλακτική λύση.
- Προσθέστε τη νέα λειτουργία σε ένα αντικείμενο χωρίς επιρροή σε άλλα αντικείμενα.
- Καταστήστε μια λειτουργία εύκολα προστιθέμενη και αφαιρούμενη δυναμικά.
- Περισσότερη ευελιξία από τη στατική κληρονομικότητα.
- Διαφανής στο αντικείμενο.

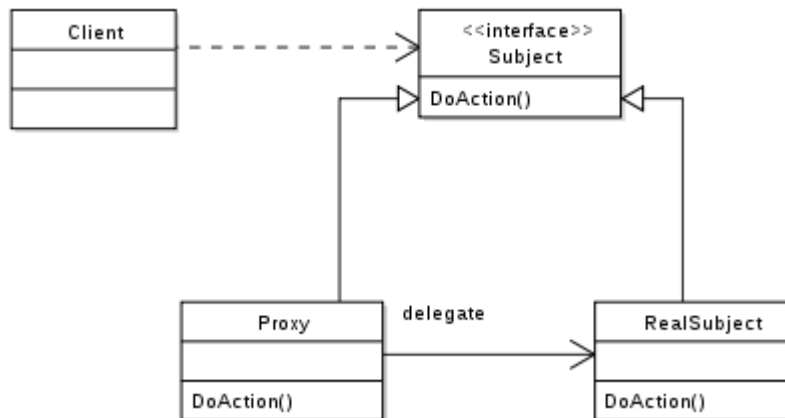


(11). Decorator UML diagram

PROXY

Χρησιμοποιήστε ένα απλό αντικείμενο για να αντιπροσωπεύσετε ένα σύνθετο ή να παρέχετε placeholder για ένα άλλο αντικείμενο για να ελέγξει την πρόσβαση σε το.

- Εάν δημιουργώντας ένα αντικείμενο είναι πάρα πολύ ακριβή έγκαιρο ή μνήμη.
- Αναβάλετε τη δημιουργία έως ότου χρειάζεστε το πραγματικό αντικείμενο.
- Φορτώστε μια μεγάλη εικόνα (χρονοβόρα).
- Φορτώστε ένα μακρινό αντικείμενο πέρα από το δίκτυο κατά τη διάρκεια των μέγιστων περιόδων.
- Το δικαίωμα πρόσβασης απαιτείται σε ένα σύνθετο σύστημα.



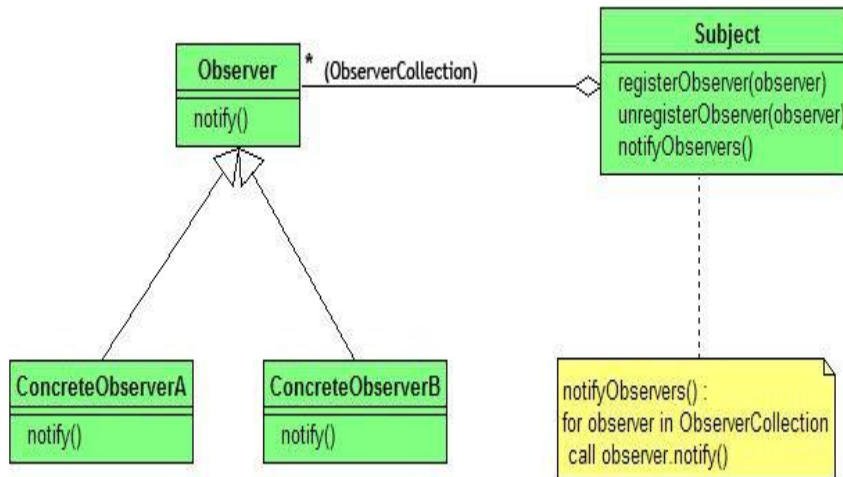
(12). Proxy UML diagram

OBSERVER

Όταν η κατάσταση ενός αντικειμένου αλλάζει, αλλάζουν τα πάντα που εξαρτώνται από αυτό.

Χρήση - Πλεονεκτήματα

- Μια αλλαγή έχει επιπτώσεις σε ένα ή πολλά αντικείμενα.
- Πολλοί αντιτίθενται ότι η συμπεριφορά εξαρτάται από μια κατάσταση αντικειμένου.
- ΑΚΑ «δημοσιεύω-προσυπογράφω».
- Διατηρήστε τη συνέπεια μεταξύ των αντικειμένων.

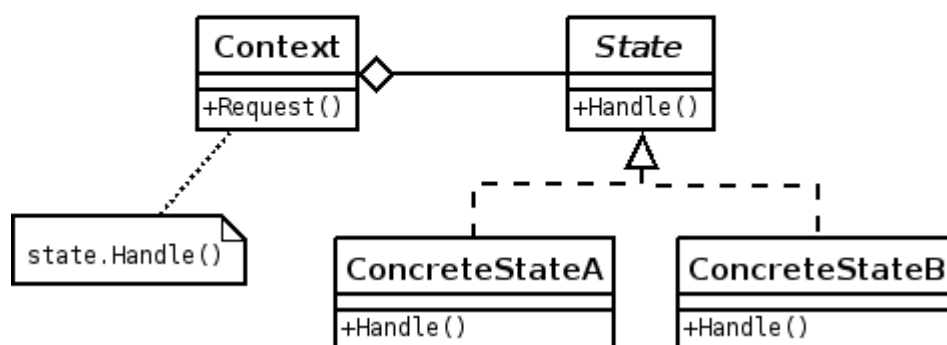


8. Observer UML diagram

STATE - STRATEGY

Η αλλαγή συμπεριφοράς ενός αντικειμένου αντιπροσωπεύεται από τις κλάσεις μέλη του, οι οποίες μοιράζονται την ίδια υπέρ - κλάση.

- Ανάγκη να ελεγχθούν πολλές καταστάσεις χωρίς να χρησιμοποιηθούν εντολές If – else ή εντολές SWITCH.
- Χρησιμοποιήστε μια κλάση για να αντιπροσωπεύσετε μια κατάσταση, όχι μια σταθερά ή κάτι άλλο.
- Κάθε κατάσταση πρέπει να είναι μια υποκλάση της ίδιας υπέρ - κλάσης.
- Απλοποιήστε και διευκρινίστε το πρόγραμμα.



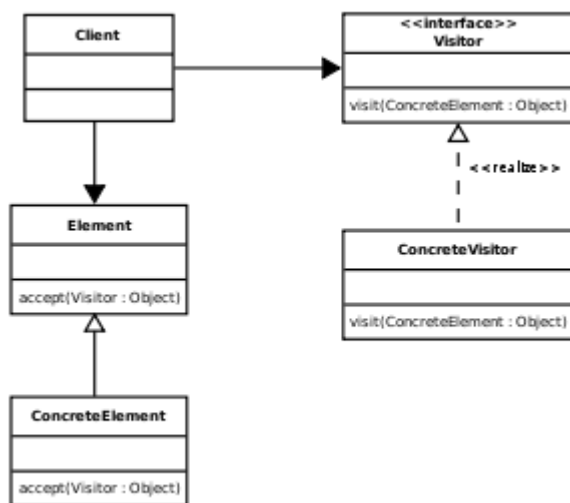
9. State - Strategy UML diagram

VISITOR

Καθορίστε μια νέα λειτουργία για να εξετάσετε τις κλάσεις των στοιχείων χωρίς αλλαγή των δομών τους.

Χρήση – Πλεονεκτήματα

- Προσθέστε διαδικασίες σε μια δέσμη κλάσεων που έχουν διαφορετικές διεπαφές.
- Εύκολο να προσθέσετε νέες διαδικασίες.
- Το πέρασμα των ιεραρχιών κλάσεων μπορεί να σπάσει την ενθυλάκωση.



10. Visitor UML diagram

ΚΕΦΑΛΑΙΟ 4

ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ

4.1 Περιγραφή Πειραμάτων

Τα κριτήρια για την επιλογή των βιβλιοθηκών και των συστημάτων που συμμετείχαν στην έρευνα είναι τα ακόλουθα:

- Να είναι συστήματα ανοικτού κώδικα.
- Να είναι μεσαίου μεγέθους.
- Κριτήριο μεγέθους να είναι ο αριθμός κλάσεων και γραμμών κώδικα.
- Τα πολύ μεγάλα συστήματα λογισμικού δεν παρήγαγαν αποτελέσματα.

Οι βιβλιοθήκες και τα συστήματα που χρησιμοποιήθηκαν είναι τα εξής (5):

ΒΙΒΛΙΟΘΗΚΕΣ – API

Libjava 4.0.7	Είναι ένα σύνολο συνδέσεων της java για το widget κουτί εργαλείων GTK.
JDom 1.0.4	Είναι μια java -κεντρική και java-βελτιστοποιημένη βιβλιοθήκη. Χρησιμοποιεί τις συλλογές της java, είναι απολύτως φυσικό API για τους τρέχοντες υπεύθυνους για την ανάπτυξη της java, και παρέχει ένα χαμηλού κόστους σημείο εισόδων για τη χρησιμοποίηση XML.
Math 5.0.9	Η βιβλιοθήκη μαθηματικών προτύπων είναι μια γενική βιβλιοθήκη math με την υποστήριξη για τους μεγάλους αριθμούς, που ενεργούν όπως οι πρωτότυποι τύπους, με πολλές μαθηματικές λειτουργίες (όπως τριγωνομετρικές, λογάριθμους, κ.λπ.).
JavaBeans 7.0.6	Είναι μια φορητή, πλατφόρμα-ανεξάρτητο συστατικό πρότυπο που γράφεται στη γλώσσα προγραμματισμού της java. Η αρχιτεκτονική JavaBeans χτίστηκε μέσω μιας συνεργάσιμης προσπάθειας βιομηχανίας και επιτρέπει στους υπεύθυνους για την ανάπτυξη να γράφουν επαναχρησιμοποιήσιμα συστατικά στη γλώσσα προγραμματισμού της java.
Awt 1.2.3	Περιέχει όλες τις κλάσεις για τη δημιουργία των διεπαφών με τον χρήστη ,για τις γραφικές παραστάσεις και τις εικόνες ζωγραφικής. Ένα αντικείμενο διεπαφών με τον χρήστη όπως ένα κουμπί ή ένας scrollbar καλείται, στην ορολογία AWT, ένα συστατικό. Η συστατική κλάση είναι η ρίζα όλων των συστατικών AWT. Δείτε το συστατικό για μια λεπτομερή περιγραφή των ιδιοτήτων που όλα τα συστατικά AWT μοιράζονται.
Sconman 0.4	Είναι ακριβώς μια βιβλιοθήκη της java χωρίς γραφικό περιβάλλον. Αν και όχι στο αρχικό πεδίο, έγινε σύντομα σαφές ότι μια γραφική διεπαφή με τον χρήστη θα ήταν πολύ

	χρήσιμη.
NIO 4.4	Η νέα εισόδου-εξόδου βιβλιοθήκη (NIO) εισήχθη με JDK 1.4. Παρέχει ταχύτητα, προσανατολισμένη στην I/O λειτουργία στον τυποποιημένο κώδικα της java. Καθορίζει τις κλάσεις για την διαχείριση των δεδομένων, και εκμεταλλεύεται τις χαμηλού επιπέδου βελτιστοποιήσεις με τέτοιο τρόπο έτσι ώστε το αρχικό πακέτο I/O, χωρίς χρησιμοποίηση του εγγενούς κώδικα.
Algstudy 9.0	Παρέχει τις εφαρμογές των αλγορίθμων (ταξινόμηση, έρευνα, κ.λπ.) και των δομών δεδομένων (κατάλογοι, δέντρα, κ.λπ.).
IO 4.5	Οι περισσότερες εφαρμογές πρέπει να επεξεργαστούν κάποια εισαγωγή και να παραγάγουν κάποια παραγωγή βασισμένη σε εκείνη την εισαγωγή. Ο σκοπός του πακέτου της java IO (java.io) είναι να καταστεί αυτό πιθανό στην java.
Applet 6.0.8	Ένα applet είναι ένα πρόσθετο είδος προγράμματος της java που επιτρέπει σε μια μηχανή αναζήτησης με την τεχνολογία της java να την μεταφορτώσει από το Διαδίκτυο και να την εκτελέσει. Ένα applet ενσωματώνεται χαρακτηριστικά μέσα σε μια ιστό-σελίδα και τρέχει στα πλαίσια της μηχανής αναζήτησης.
Sax 2r3	Είναι ένα κοινό front-end για τους καταμητές XML, όπως το JDBC για την πρόσβαση βάσεων δεδομένων. Χρησιμοποιείται ευρέως από τα προγράμματα ανοικτού - κώδικα όπως Apache και από τους εταιρικούς χρήστες όπως την SUN, την IBM, την Oracle και την Microsoft.
Opentermerv 2.5.6	Παρέχει πλήρως διανεμηθέντα τον κεντρικό υπολογιστή ορολογίας που επιτρέπει σε έναν πελάτη να πλοηγήσει τις διάφορες ταξινομήσεις μέσω μιας γενικής ταξινόμησης πρότυπο API (GAPI).
Adwords 0.11	Αφήνει τους υπεύθυνους για την ανάπτυξη να χτίσουν τις εφαρμογές που αλληλεπιδρούν άμεσα με την πλατφόρμα AdWords. Με αυτές τις εφαρμογές, οι διαφημιστές και οι τρίτοι μπορούν αποτελεσματικότερα και δημιουργικά να διαχειριστούν τους μεγάλους ή σύνθετους απολογισμούς AdWords και τις εκστρατείες τους.
Jwma 0.9.8	Είναι μια εφαρμογή webmail μετά από την πρότυπη ιδέα 2 αρχιτεκτονικής JSP. Είναι βασισμένο στην τυποποιημένη java API και στηρίζεται σε μερικές διαθέσιμες ανοικτές τεχνολογίες πηγής (Xerces, κάστορας, Τζακάρτα-ORO).
Net 8.0	Παρέχει τις κλάσεις για την λειτουργία των εφαρμογών δικτύωσης. Χρησιμοποιώντας τις κλάσεις υποδοχών, μπορείτε να επικοινωνήσετε με οποιοδήποτε κεντρικό υπολογιστή στο διαδίκτυο ή να θέσετε σε εφαρμογή τον κεντρικό υπολογιστή Διαδικτύου σας.

ΣΥΣΤΗΜΑΤΑ ΛΟΓΙΣΜΙΚΟΥ – APPLICATIONS

Jat 6.2	Το Jat είναι ένα εύκολο να επεκταθεί πλαίσιο ανοικτού κώδικα της java. Παρέχει τη μορφοματική και
---------	---

	εύκαμπτη βασική λειτουργία για την ανάπτυξη εφαρμογών της java, βελτιώνοντας το χρόνο ξεκινήματος της εφαρμογής. Η πύλη Jat είναι μια ενισχυμένη έκδοση για την οικοδόμηση μιας πλήρους εφαρμογής Ιστού.
ImageTagExtractor 1.2	Αυτό το σχέδιο προγράμματος άρχισε αρχικά για να φιλοξενεί το Mp3ImageTagExtractor. Έχει επεκταθεί τώρα για να περιλάβει το ShoutcastPlsDownloader και ενδεχομένως άλλα μικρά προγράμματα ανοικτού κώδικα στην java .
CaféSip 0.8.4	Το πρωτόκολλο συνόδου (SIP) χρησιμοποιείται ευρέως για τις τηλεφωνικές υπηρεσίες στο Διαδίκτυο. Το CafeSip παρέχει μια ακολουθία εργαλείων και εφαρμογών ανοικτού κώδικα για τη δημιουργία των προσαρμοσμένων υπηρεσιών SIP και των εφαρμογών χρησιμοποιώντας την java.
JasperReports 3.5.0	Είναι το δημοφιλέστερο σύστημα ανοικτού κώδικα επιχειρησιακής νοημοσύνη και υποβολής εκθέσεων της μηχανής και JasperServer
JUnit 3.9.7	Το JUnit είναι ένα εξεταστικό πλαίσιο μονάδων για τη γλώσσα προγραμματισμού της java.
JavaCalendar 1.2.1	Το JavaCalendar είναι ένα πρόγραμμα ημερολογίου που γράφεται στην java. Ο σκοπός του προγράμματος είναι να παρασχεθεί ένα ημερολογιακό τμήμα που μπορεί να χρησιμοποιηθεί στην java, βασισμένο σε προγράμματα για την επιλογή/την επίδειξη της ημερομηνίας και του χρόνου με φιλικό προς το χρήστη τρόπο.
Gelatin 1.5	Είναι μια πρωτοβουλία ανάπτυξης ανοικτού κώδικα που στοχεύει στην παροχή πλούσιων πακέτων εργαλείων API για τη γλώσσα προγραμματισμού της java. Το πρόγραμμα αποτελείται από δύο κύρια προγράμματα: Το κουτί εργαλείων και το ObjectBase ερώτησης Abstract.
Core 1.2.3	Είναι μια ελάχιστη διανομή του λειτουργικού συστήματος GNU/Linux με σκοπό να είναι η βάση για ένα πλήρες σύστημα που κατασκευάζεται από τον τελικό - χρήστη.
JBUILDER 0.18	Το JBuilder, τώρα με τις επαναστατικές ικανότητες επαναχρησιμοποίησης κώδικα, παρέχει σε σας την ανάλυση και τη σκιαγράφηση των εργαλείων που πρέπει να βελτιώσετε την ποιότητα κώδικα για να βελτιώσετε την απόδοση της εφαρμογής
JEdit 1.3.4	jEdit είναι ο συντάκτης κειμένων ενός ώριμου προγραμματιστή με τις εκατοντάδες (που μετρούν το χρόνο που αναπτύσσεται plugins) των ανθρωποετών της ανάπτυξης εφαρμογών.
JMol 4.8	Το Jmol είναι παρατηρητής μορίων πηγής για τους σπουδαστές, τους εκπαιδευτικούς, και τους ερευνητές στη χημεία και τη βιοχημεία. Είναι ανεξάρτητη -

	πλατφόρμα, που τρέχει στα Windows, MAC OS X, και συστήματα Linux/Unix.
JSqslide 0.7	Είναι προγράμματα ανοικτού κώδικα που περιλαμβάνουν: jsqslide - ένα επεκτάσιμο βασισμένο σε java περιβάλλον ανάπτυξης , ένας συντάκτης ιδιοκτησίας για τα αντικείμενα της java UI.
Barbet 0.0.3	Το Barbet γράφεται στην java και χρησιμοποιεί το JDBC API για να διαχειριστεί τις βάσεις δεδομένων από διαφορετικούς προμηθευτές. Αξιοποιεί πλήρως διάφορα νέα χαρακτηριστικά γνωρίσματα διαθέσιμα σε J2SE στην έκδοση 5.
Javadjvu 0.8	Παρέχει ένα ανοικτού κώδικα applet και τις εικονικές μηχανές της java υπολογιστών γραφείου.
Deputy 2.0.4	Υποστηρίζει τη δημιουργία και τη συντήρηση των σύνθετων 1.x προγραμμάτων Maven που συγκεντρώνονται από άλλα προγράμματα Maven.

Για την διενέργεια των στατιστικών ελέγχων θα κάνουμε χρήση του Independent – Samples T test του στατιστικού πακέτου SPSS, ενέργεια που χρησιμοποιείται για να ελέγξει αν οι μέσες τιμές 2 ανεξάρτητων δειγμάτων πληθυσμών παρουσιάζουν στατιστικά σημαντική διαφορά (2).

Ικανή και αναγκαία συνθήκη για την χρήση του Independent – Samples T test είναι να εξετάσουμε αν οι κατανομές των τιμών των ανεξάρτητων δειγμάτων είναι κανονικές. Ο συγκεκριμένος έλεγχος (3) επιτυγχάνεται είτε χρησιμοποιώντας τον έλεγχο Kolmogorov – Smirnov, ο οποίος εξετάζει αν μια κατανομή μεταξύ 2 δειγμάτων είναι κανονική και παράγει μια απάντηση «ναι» ή «όχι», είτε χρησιμοποιώντας γραφήματα κανονικών πιθανοτήτων όπως είναι τα γραφήματα PP και QQ.

Οι κανονικές κατανομές πιθανότητας (κατανομές PP) παρουσιάζουν αναμενόμενες στατιστικές από ένα δείγμα που λαμβάνεται από την κανονική διανομή με το μέσο όρο 0 και τη διαφορά 1 ενάντια στις τιμές δεδομένων που διατάσσονται από μικρότερες σε μεγαλύτερες. Εάν τα δεδομένα που εξετάζονται έχουν μια κανονική κατανομή συχνότητας, μια κατανομή των δεδομένων ενάντια στις αναμενόμενες στατιστικές πρέπει να παραγάγει μια ευθεία γραμμή. Το SPSS παράγει επίσης μια παραλλαγή της κατανομής πιθανότητας αποκαλούμενης κατανομή q-q, ή την κατανομή quantile-quantile στις οποίες τα εκατοστημόρια της τυποποιημένης κανονικής κατανομής σχεδιάζονται ενάντια στα εκατοστημόρια των δεδομένων. Η πλοκή q-q είναι πιο ευαίσθητη στις παρεκκλίσεις από την κανονικότητα στις ουρές της κατανομής, ενώ η κανονική κατανομή πιθανότητας είναι πιο ευαίσθητη στις παρεκκλίσεις κοντά στο μέσο όρο της κατανομής. Εάν η παρατηρηθείσα καμπύλη σημείων είναι επάνω από ή κάτω από την κανονική γραμμή κατανομών, αυτό δείχνει ότι η κύρτωση απέχει από μια κανονική κατανομή, ενώ εάν η παρατηρηθείσα πλοκή είναι S-shaped, αυτό δεικνύει ότι τα δεδομένα δεν είναι ορθά.

Στην παρούσα έρευνα οι κατανομές των μέσων τιμών των μετρικών, των ελέγχων και των προτύπων σχεδίασης των ανεξάρτητων δειγμάτων βιβλιοθηκών και συστημάτων ανοικτού κώδικα είναι κανονικές.

Ο στατιστικός έλεγχος διενεργείται ως εξής:

Σύγκριση μέσων τιμών 2 ανεξάρτητων δειγμάτων (3):

□ Για μια ποσοτική μεταβλητή σε δύο ανεξάρτητους πληθυσμούς μας ενδιαφέρει για τις (άγνωστες) μέσες τιμές τους:

□ Έλεγχος υπόθεσης

H0: $\mu_1 = \mu_2$

H1: $\mu_1 \neq \mu_2$

□ Εύρεση δ.ε. για τη διαφορά $\mu_1 - \mu_2$ (90%, 95%, 99%).

Τα πακέτα λογισμικού που χρησιμοποιήθηκαν για τον στατιστικό έλεγχο, για την εύρεση των μετρικών και των ελέγχων και για την εύρεση των προτύπων σχεδίασης είναι τα SPSS Statistics, Eclipse και Design Pattern Detection.

4.2 Στατιστική Ανάλυση – Μετρικών

Μετρική AC

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.026 < 0.05 Οι διασπορές έχουν διαφορά (θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.008 < 0.05 Οι μέσες τιμές έχουν διαφορά (θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-28,45, -4,88).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής AC μεταξύ API (13,26) και APPLICATIONS (29,93) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική AID

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.014 < 0.05 Οι διασπορές έχουν διαφορά (θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.11 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά (θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-27,8, 3,18).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής AID μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική AIF

1^ο βήμα: Είναι οι ίσες;

sig. = 0.026 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.015 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-38,98, -4,621).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής AIF μεταξύ API (22,13) και APPLICATIONS (43,93) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική AIUR

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.683 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.001 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-13,39, 47,41).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής AIUR μεταξύ API (73,46) και APPLICATIONS (43,06) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική ALD

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.003 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.155 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-226,75, 39,95).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής ALD μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική AOFD

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.178 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.009 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,31, -0,35).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής AOFD μεταξύ API (0,46) και APPLICATIONS (1,8) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική AUF

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.446 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.043 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-16,52, -0,28).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής AUF μεταξύ API (14,66) και APPLICATIONS (23,06) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική CBO

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.016 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-5,91, -1,95).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής CBO μεταξύ API (4,4) και APPLICATIONS (8,33) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική CC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.004 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.188 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-347,22, 74,69).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής CC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική CF

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.22 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.68 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-19,32, 29,18).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής CF μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική CHC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.219 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.07 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,67, 0,071).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής CHC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική CIW

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.433 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.596 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2035,98, 1190,91).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής CIW μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική CL

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.044 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (9,52, 28,75).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής CL μεταξύ API (64,93) και APPLICATIONS (45,8) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική CM

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.968 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.609 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,64, 1,58).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής CM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική COC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.018 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.002 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,37, -0,37).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής COC μεταξύ API (0,73) και APPLICATIONS (1,6) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική CR

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.380 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (13,14, 29,78).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής CR μεταξύ API (54,4) και APPLICATIONS (32,93) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική DAC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.006 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.01 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,84, -0,29).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής DAC μεταξύ API (0,66) και APPLICATIONS (1,73) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική DD

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 1.00 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,48, -1,12).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής DD μεταξύ API (2,33) και APPLICATIONS (4,13) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική DOIΗ

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.271 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.019 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,58, -1,55).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής DOIΗ μεταξύ API (0,93) και APPLICATIONS (1,8) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική FO

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.073 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.009 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-3,47, -0,53).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής FO μεταξύ API (3,53) και APPLICATIONS (5,53) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική IUR

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.631 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (15,93, 44,6).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής IUR μεταξύ API (80,6) και APPLICATIONS (50,33) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική LCOM1

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.22 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.435 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-68,94,156,00).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής LCOM1 μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική LCOM2

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.006 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.137 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-11,7, 11,57).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής LCOM2 μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική LCOM3

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.119 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.059 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-23,65, 0,45).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής LCOM3 μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική LOC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.525 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.83 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-35795,41, 28943,55).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής LOC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική MIC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.402 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.925 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-5,46, 5,99).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής MIC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική MIF

1^ο βήμα: Είναι οι διασπορές των πληθυσμών ίσες;

sig. = 0.152 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-45,86, 15,6).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής MIF μεταξύ API (52,8) και APPLICATIONS (85,3) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική MNOB

1^ο βήμα: Είναι οι διασπορές των πληθυσμών ίσες;

sig. = 0.009 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.195 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-112,29, 24,96).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής MNOB μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική MNOL

1^ο βήμα: Είναι οι διασπορές των πληθυσμών ίσες;

sig. = 0.218 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.049 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-1,58, -0,002).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής MIF μεταξύ API (1,26) και APPLICATIONS (2,06) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική MNOP

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.83 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.176 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$:(-0,16, 0,83).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής MNOP μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική MPC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.005 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.157 > 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$:(-644,52, 115,05).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής MPC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική MSOO

1ο βήμα: Είναι οι διασπορές των πληθυσμών ίσες;

sig. = 0.991 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.533 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-3,11, 1,64).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής MSOO μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NAM

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.599 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.589 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-248,96, 144,16).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NAM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NCC

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.024 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.027 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-15,5, -1,02).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NCC μεταξύ API (3,13) και APPLICATIONS (11,4) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική NOA

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.061 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.175 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-809,97, 154,37).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOA μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOAM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.15 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.762 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-6,47, 8,73).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOAM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.29 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.491 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-368,57, 181,11).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOCC

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.761 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.523 > 0.05 Οι μέσες δεν τιμές έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-0,72, 1,39).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOCC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOCON

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.917 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.971 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-156,52, 162,25).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOCON μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOCP

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.408 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.004 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,21, -0,46).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOCP μεταξύ API (0,6) και APPLICATIONS (1,93) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική NOED

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.002 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-7,16, -2,97).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NOED μεταξύ API (3,4) και APPLICATIONS (8,47) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική NOIS

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.078 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.182 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1911,78, 380,58).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής μεταξύ NOIS API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOLV

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.008 > 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.19 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-208,66, 45,33).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NOLV μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.260 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.414 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2940,74, 1247,002).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOO

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.337 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.577 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2074,69, 1177,75).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής NOO μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOOM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.425 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.777 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-0,82, 1,09).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NOOM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOP

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.004 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.189 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-188,33, 40,86).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NOP μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NOPA

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.009 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.058 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-55,59, 1,05).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NOPA μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική NORM

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.103 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.002 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-14,64, -3,49).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής NORM μεταξύ API (8,8) και APPLICATIONS (17,87) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική PC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.881 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.125 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-59,26, 7,66).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής PC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική PF

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.116 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.509 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-10,59, 20,85).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής PF μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PIS

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.053 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.096 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-154,43, 13,23).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής PIS μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PPkgM

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.621 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.926 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-6,07, 5,53).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής PPkgM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PPriivM

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.323 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.738 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-8,001, 5,73).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής PPriivM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PProtM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.048 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας άνισες διασπορές.
sig. = 0.683 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-5,16, 7,7).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής PProtM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PPubM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.065 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.99 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-10,7, 10,84).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής PPubM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PS

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.640 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω των τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.638 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-216,18, 134,7).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής PS μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική PUR

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.009 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-40,005, -15,06).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής PUR μεταξύ API (10,26) και APPLICATIONS (37,8) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική RFC

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.663 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.547 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-27,007, 14,6).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής RFC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική TCC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.775 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2° βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.027 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3° βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-6,01, -0,38).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές της μετρικής TCC μεταξύ API (5,46) και APPLICATIONS (8,66) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική TCR

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.124 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2° βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3° βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (53,29, 154,84).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής TCR μεταξύ API (83,3) και APPLICATIONS (78,6) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική TRAp

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.116 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2° βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.719 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,54, 2,2).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής TRAp μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική TRAu

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.549 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.753 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-7,63, 10,43).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής TRAu μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική TRDp

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.085 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.304 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,8, 8,68).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής TRDp μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική TRDu

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.333 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2° βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.444 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3° βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-5,56, 12,36).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής TRDu μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική VOD

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.537 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2° βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.00 < 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3° βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-2,86, -1,007).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής VOD μεταξύ API (1,66) και APPLICATIONS (3,6) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Μετρική WCM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.565 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.108 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-7,003, 0,73).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής WCM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική WMPC1

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.299 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.588 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-16,58, 28,7).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής WMPC1 μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική WMPC2

1^ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.388 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.520 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-7,15, 13,82).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής WMPC2 μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Μετρική WOC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.13 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.417 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-64,1, 150,5).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές της μετρικής WOC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

4.3 Στατιστική Ανάλυση – Ελέγχων

Audit – DATACLASS

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.02 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσω τιμών υποθέτοντας άνισες διασπορές.

sig. = 0.05 = 0.05 Οι μέσες τιμές έχουν διαφορά(θεωρούνται άνισες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,34, -0,002).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές του audit DATACLASS μεταξύ API (0,067) και APPLICATIONS (1,27) παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 \neq \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο δεν περιέχει το 0.

Audit – DCL

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.192 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.408 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-24,19, 57,79).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit DCL μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – FE

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.036 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.

sig. = 0.179 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-25,76, 57,79).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές του audit FE μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – FUTV

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.776 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.878 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-12,25, 14,25).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit FUTV μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – GC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.072 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.180 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-11,62, 2,29).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit GC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – GM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.081 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.277 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-28,66, 8,53).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit GM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – GP

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.075 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.278 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,9, 0,56).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit GP μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – ISPV

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.109 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.211 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-22,72, 5,25).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit ISPV μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – LMC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.091 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.570 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-9,58, 17,05).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit LMC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – MC

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.421 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.506 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-2,43, 1,23).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit MC μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – RB

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.033 > 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.1 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-49,12, 4,59).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές του audit RB μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – RZLA

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.799 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.890 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-2,09, 1,82).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit RZLA μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – SS

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.157 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.301 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-17,08, 5,48).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit SS μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – SSM

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.929 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.693 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-7,7, 11,44).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit SSM μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1=\mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – SSU

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.02 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές. sig. = 0.142 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-2,53, 0,4).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές του audit SSU μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1=\mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – UII

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.264 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές. sig. = 0.562 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1-\mu_2$:(-1,80, 0,1).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit UII μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1=\mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – USP

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.057 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.

sig. = 0.211 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-5,72, 1,32).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν ίσες. Οι μέσες τιμές του audit USP μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

Audit – WSI

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.012 < 0.05 Οι διασπορές έχουν διαφορά(θεωρούνται άνισες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας άνισες διασπορές.

sig. = 0.165 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας άνισες

διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-1,97, 0,37).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές του audit WSI μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

4.3 Στατιστική Ανάλυση Προτύπων Σχεδίασης – Design Patterns

Στην συγκεκριμένη ανάλυση, εξετάζουμε συνολικά τα πρότυπα σχεδίασης για όλες τις βιβλιοθήκες και τα συστήματα. Καταλήξαμε στον προκείμενο τρόπο στατιστικής ανάλυσης, γιατί ορισμένα πρότυπα σχεδίασης είχαν μηδενικές τιμές, οπότε κρίθηκε απαραίτητο να μην συμπεριληφθούν στην ανάλυση.

1ο βήμα: Είναι οι μέσες τιμές ίσες;

sig. = 0.942 > 0.05 Οι διασπορές δεν έχουν διαφορά(θεωρούνται ίσες).

2^ο βήμα: Θεωρούμε το t-test για έλεγχο μέσων τιμών υποθέτοντας ίσες διασπορές.
sig. = 0.438 > 0.05 Οι μέσες τιμές δεν έχουν διαφορά(θεωρούνται ίσες).

3^ο βήμα: Θεωρούμε το δ.ε. της διαφοράς των μέσων τιμών υποθέτοντας ίσες διασπορές 95% δ.ε. για τη διαφορά $\mu_1 - \mu_2$: (-11,29, 5,02).

Συμπεράσματα

Εδώ οι διασπορές μπορούν να θεωρηθούν άνισες. Οι μέσες τιμές των προτύπων σχεδίασης μεταξύ API και APPLICATIONS δεν παρουσιάζουν στατιστικά σημαντική διαφορά ($\mu_1 = \mu_2$).

Το παραπάνω συμπέρασμα προκύπτει και από το 95% δ.ε. το οποίο περιέχει το 0.

ΚΕΦΑΛΑΙΟ 5

Αποτελέσματα και Συζήτηση

Στην στατιστική ανάλυση των μετρικών, των ελέγχων και των προτύπων σχεδίασης, αναμέναμε σύμφωνα με τον Tulach(1) οι βιβλιοθήκες είναι καλύτερα στοιχεία από τα συστήματα λογισμικού. Οι μετρικές που προέκυψε ότι είναι στατιστικά σημαντικές και αποτελούν ένδειξη της ανωτερότητας των βιβλιοθηκών έναντι των συστημάτων αφορούν την σχεδίαση διεπαφών ,την απλότητα στην σχεδίαση, την ορατότητα των κλάσεων και των συστατικών, την δημιουργία πακέτων που εμπεριέχουν τις κλάσεις του προγράμματος, την σχεδίαση με γνώμονα την κληρονομικότητα και την αποφυγή λεπτομερειών που αφορούν την εφαρμογή και οι οποίες προκαλούν σύγχυση στον χρήστη.

Οι στατιστικά σημαντικά μετρικές AUF, COC αποδεικνύουν την ανωτερότητα των βιβλιοθηκών έναντι των συστημάτων ανοικτού κώδικα, αφορούν την δημιουργία διεπαφών που σύμφωνα με τον Tulach (1) είναι κριτήριο καλής σχεδίασης μιας βιβλιοθήκης. Υπάρχουν πολλοί λόγοι για τους οποίους ισχύει το παραπάνω κριτήριο:

Επανακλήσεις: Εάν η διεπαφή προορίζεται να εφαρμοστεί από τον κώδικα χρηστών, κατόπιν είναι συχνά πιο λειτουργική από μια αφηρημένη κλάση. Αυτό ισχύει συνήθως για τις διεπαφές με μόνο μια μέθοδο: Μόλις διαπιστωθεί ότι υπάρχουν διάφορες μέθοδοι , και μια κλάση εφαρμογής πρέπει μόνο να κάνει χρήση σε μια από αυτές. Επιπλέον εάν μια διεπαφή έχει τρεις μεθόδους και έπειτα θελήσουμε να έχουμε τέσσερις το οποίο δεν είναι συνήθως δυνατό καθώς είδαμε. Μια αφηρημένη κλάση μπορεί να αποφύγει αυτά τα προβλήματα.

Πολλαπλάσια κληρονομικότητα: Είναι περιστασιακά χρήσιμο να είναι σε θέση να εφαρμόσει μια διεπαφή βαθιά την ιεραρχία κληρονομιάς. Είναι συνήθως κακή πρακτική να εφαρμοστεί κάποια τυχαία διεπαφή σε μια κλάση της οποίας αρχικός σκοπός είναι κάτι άλλο. Η εφαρμογή της διεπαφής σε μια ιδιωτική εσωτερική κλάση είναι συνήθως πιο λειτουργική, και έπειτα φυσικά θα μπορούσε εξ ίσου καλά να λειτουργήσει μια αφηρημένη κλάση.

Δυναμικά πληρεξούσια: Η κλάση πληρεξούσιου επιτρέπει σε σας για να κάνει μια εφαρμογή οποιασδήποτε διεπαφής να εκτελεστεί στο χρόνο εκτέλεσης, καλώντας οποιασδήποτε από τις μεθόδους της διεπαφής με μια κλήση.

Η μετρική CBO αναφέρεται σ'εκείνο το κριτήριο της ποιότητας των βιβλιοθηκών που αφορά την μινιμαλιστική σχεδίασή της.

Λόγω της απαίτησης συμβατότητας, είναι πολύ ευκολότερο να προστεθούν πράγματα από να αφαιρεθούν. Έτσι μην προσθέστε τίποτα στην βιβλιοθήκη αν δεν είστε βέβαιοι ότι το χρειάζεστε.

Υπάρχει μια προσέγγιση στην σχεδίαση βιβλιοθηκών που παρατηρείται συχνά. Σκεφτείτε πάνω απ' όλα τι θα θελήσει ενδεχομένως να κάνει ένας χρήστης με την βιβλιοθήκη και προσθέστε μια μέθοδο για να το επιτύχετε. Η περισσότερη ουσία είναι εκεί στην βιβλιοθήκη. Ποιες κλάσεις και μέθοδοι είναι οι σημαντικές;

Όσο μεγαλύτερη η βιβλιοθήκη, τόσα περισσότερα πράγματα μπορούν να πάνε στραβά. Η εφαρμογή δεν πρόκειται να είναι τέλεια, αλλά η ίδια επένδυση στην κωδικοποίηση και τη δοκιμή θα παραγάγει τα καλύτερα αποτελέσματα για μια μικρότερη βιβλιοθήκη.

Εάν η βιβλιοθήκη έχει περισσότερες μεθόδους από ότι χρειάζεται, κατόπιν καταλαμβάνει περισσότερο χώρο από ότι χρειάζεται.

Η σωστή προσέγγιση είναι να βασιστεί η βιβλιοθήκη στον κώδικα παραδείγματος. Σκεφτείτε ποια προβλήματα μια ομάδα χρηστών θέλει να επιλύσει με την βιβλιοθήκη. Προσθέστε ακριβώς αρκετές κλάσεις και μεθόδους για να λύσετε εκείνα τα προβλήματα. Κωδικοποιήστε τις λύσεις. Αυτό επιτρέπει σε σας να ελέγξετε ότι η βιβλιοθήκη είναι χρήσιμη. Σαν ευτυχή παρενέργεια, σας δίνει μερικές βασικές δοκιμές. Και μπορείτε (και πρέπει) να μοιραστείτε τα παραδείγματα με τους χρήστες σας.

Η μετρική CL αφορά την ελαχιστοποίηση της ορατότητας των κλάσεων, των μεθόδων. Σε μια άρτια σχεδιασμένη βιβλιοθήκη, τα μόνα πεδία που πρέπει να είναι ορατά είναι στατικά και τελικά. Ο κώδικας της βιβλιοθήκης επίκειται σε συνεχείς διορθώσεις εν αντιθέσει με τον κώδικα των συστημάτων ανοικτού κώδικα και έτσι είναι πιο λειτουργικό οι αλλαγές να πραγματοποιούνται σε όσο το δυνατόν λιγότερα πεδία.

Οι μετρικές CR, TCR έχουν σχέση με την συγγραφή εγχειριδίου χρήσης, που για τις βιβλιοθήκες κρίνεται απαραίτητο να υπάρχει επειδή μια βιβλιοθήκη μελετάται πολλές φορές για να γίνει κατανοητή από τους χρήστες. Έτσι το εγχειρίδιο πρέπει να είναι ακριβές και να καλύπτει όλες τις κλάσεις – μεθόδους που χρησιμοποιούνται.

Στην σχεδίαση βιβλιοθηκών πρέπει να υπάρχει προσοχή με τα πακέτα που την αποτελούν. Η γλώσσα της java έχει αρκετά περιορισμένους τρόπους για την διαφάνεια των κλάσεων και των μεθόδων. Συγκεκριμένα, εάν μια κλάση ή μια μέθοδος είναι ορατή σε άλλο πακέτο, κατόπιν είναι ορατή σε όλο τον κώδικα σε όλα τα πακέτα. Αυτό σημαίνει ότι εάν χωρίζουμε την βιβλιοθήκη σε διάφορα πακέτα, πρέπει να αποφύγουμε να καταστήσουμε δεδομένα δημόσια ακριβώς έτσι ώστε ο κώδικας σε άλλα πακέτα στην βιβλιοθήκη να μην μπορεί να έχει πρόσβαση σε αυτά.

Η απλούστερη λύση για να αποφευχθεί αυτό είναι να τεθεί ολόκληρη η βιβλιοθήκη σ' ένα πακέτο. Για μια βιβλιοθήκη με λιγότερες από περίπου 30 δημόσιες κλάσεις αυτό είναι συνήθως η καλύτερη προσέγγιση.

Επιθυμητή παράμετρος στην σχεδίαση βιβλιοθηκών είναι είτε σχεδιάζουμε για κληρονομικότητα είτε δεν την επιτρέπουμε καθόλου. Οι μετρικές που έχουν σχέση με την κληρονομικότητα είναι οι DOIH και MIF.

Η ανάλυση των λεπτομερειών που σχετίζονται με την εφαρμογή της βιβλιοθήκης, δεν πρέπει να αναφέρονται γιατί δημιουργούν σύγχυση στους χρήστες. Η μετρική που έχει άμεση σχέση με το πιο πάνω κριτήριο καλής σχεδίασης μιας βιβλιοθήκης είναι η μετρική MNOL.

Οι μετρικές που είναι στατιστικά σημαντικές και αποδεικνύουν ανωτερότητα των συστημάτων έναντι των βιβλιοθηκών αφορούν τις ιδιότητες των κλάσεων, την κληρονομικότητα των ιδιοτήτων και την επεξεργασία των μεθόδων και των κατασκευαστών των κλάσεων. Πραγματοποιώντας μια βαθύτερη ανάλυση οι μετρικές AC και NORM αφορούν την πολυπλοκότητα του συστήματος, ένα σύστημα ανοικτού κώδικα έχει μεγαλύτερο βαθμό πολυπλοκότητας από μια βιβλιοθήκη, οι μετρικές AOFD, DAC, FO και VOD αναφέρονται στην ένωση των μονάδων του συστήματος και συγκεκριμένα στην πρόσβαση των δεδομένων που αποτελούν το σύστημα η οποία είναι υψηλότερη σ' ένα σύστημα ανοικτού κώδικα εν συγκρίσει με μιας βιβλιοθήκης λόγω της φιλοσοφίας προγραμματισμού του συστήματος.

Όσον αφορά τους ελέγχους που προκύπτει ότι είναι στατιστικά σημαντικοί και αποδεικνύουν την ανωτερότητα των βιβλιοθηκών, πρόκειται για τον έλεγχο που αφορά τις κλάσεις δεδομένων και την έλλειψη λειτουργικών μεθόδων μιας εφαρμογής. Τα συγκεκριμένα μειονεκτήματα αποτελούν ένδειξη κακής αντικειμενοστραφούς σχεδίασης, γεγονός που ερμηνεύεται πως ένα σύστημα ανοικτού κώδικα έχει μικρότερο βαθμό συντηρισιμότητας, δυνατότητα δοκιμής και μικρότερο βαθμό κατανόησης σε σχέση με μια βιβλιοθήκη.

Για τα πρότυπα σχεδίασης προέκυψε ότι δεν υπάρχει στατιστικά σημαντική διαφορά μεταξύ των βιβλιοθηκών και των συστημάτων. Ενδεχομένως το μέγεθος των συστημάτων που εξετάσαμε να είναι τέτοιο που δεν κατέστη δυνατή ή απαραίτητη η εφαρμογή προτύπων σχεδίασης.

ΚΕΦΑΛΑΙΟ 6

Συμπεράσματα

Η επιλογή των βιβλιοθηκών και των συστημάτων ανοικτού κώδικα που χρησιμοποιήθηκαν στην έρευνα έγινε αυθαίρετα. Ενδεχομένως χρησιμοποιώντας διαφορετικά συστήματα και βιβλιοθήκες να εξαγάγαμε διαφορετικά συμπεράσματα. Επιπροσθέτως το μέγεθος των συστημάτων και των βιβλιοθηκών δεν επέτρεψε την εξαγωγή ασφαλών συμπερασμάτων. Η συμπεριφορά των συγκεκριμένων προϊόντων λογισμικού ενδέχεται να ήταν διαφορετική, όσον αφορά τα κριτήρια της ποιότητας , μετρικές , έλεγχοι , πρότυπα σχεδίασης.

Η έλλειψη ορισμένων προτύπων σχεδίασης ή η εμφάνιση μηδενικών τιμών σε κάποια συστήματα ανοικτού κώδικα και σε βιβλιοθήκες, μπορεί να οφείλεται στο μέγεθος τους. Μεγαλύτερα συστήματα ανοικτού κώδικα και βιβλιοθηκών πιθανόν να συμπεριλαμβάνουν , στα πλαίσια των κανόνων της αντικειμενοστρώσης σχεδίασης , πρότυπα σχεδίασης. Επιπλέον από τις κατηγορίες των μετρικών, κρίναμε απαραίτητο να μετρηθούν οι κατηγορίες που αφορούν την ενθυλάκωση , την κάλυψη δοκιμής και από τις κατηγορίες ελέγχων επιλέξαμε να μετρηθεί η κατηγορία «αδυναμίες σχεδίασης».

Η επιλογή των κατηγοριών των μετρικών και των ελέγχων έγινε με κριτήριο την σημαντικότητα των μετρικών και των ελέγχων που ανήκουν στις συγκεκριμένες κατηγορίες αλλά και για λόγους περιορισμού των μετρήσεων σε μεγέθη μετρήσιμα.

Η περαιτέρω έρευνα σχετικά με τις διαφορές των βιβλιοθηκών και των συστημάτων, θα επικεντρωθεί σε μεγαλύτερα σε μέγεθος προγράμματα που στάθηκε αδύνατον να εξεταστούν χρησιμοποιώντας τα τρέχοντα προϊόντα λογισμικού. Ίσως καταστεί δυνατή και η ανάπτυξη ενός εργαλείου το οποίο να επεμβαίνει στον κώδικα της εφαρμογής και να διορθώνει τα σφάλματα που αφορούν τους ελέγχους – audits της εφαρμογής.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Jaroslav, T. (2008). *Practical API Design*, Apress.
2. Wohlin, C. & Runeson, P & Martin, H & Magnus, O & Bjorn, R & Anders, W. (2000). *Experimentation in Software Engineering*, Lund University, Sweden, Kluwer Academic Publishers .
3. *The Normal Frequency Distribution*. Ανάκτηση 12/07/2009 από World Wide Web: <http://www.microbiologybytes.com/maths/spss2.html>
4. *About Design Pattern*. Ανάκτηση 20/08/2009 από World Wide Web: <http://www.javacamp.org/designPattern/>
5. *Find and Develop Open Source Software*. Ανάκτηση 06/05/2009 από World Wide Web: <http://sourceforge.net/>
6. *Factory Method Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: <http://www.apwebco.com/gofpatterns/creational/FactoryMethod.html>
7. *Protoype Method Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: <http://www.apwebco.com/gofpatterns/creational/Prototype.html>
8. *Singleton Method Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: <http://www.apwebco.com/gofpatterns/creational/Prototype.html>
9. *Object Adapter Command Method Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: <http://en.wikipedia.org/wiki/File:ObjectAdapter.png>
10. *Composite Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: http://en.wikipedia.org/wiki/Composite_pattern
11. *Decorator Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: http://en.wikipedia.org/wiki/Decorator_pattern

12. *Proxy Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: http://en.wikipedia.org/wiki/Proxy_pattern

13. *Observer Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: http://en.wikipedia.org/wiki/Observer_pattern

14. *State - Strategy Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web:
http://en.wikipedia.org/wiki/File:State_Design_Pattern_UML_Class_Diagram.png

15. *Visitor Design Pattern, UML diagram*. Ανάκτηση 18/07/2009 από World Wide Web: http://en.wikipedia.org/wiki/Visitor_pattern