

Ανάπτυξη εργαλείου διαχείρισης έργων λογισμικού

Φοιτητής : Κουκάρας Παρασκευάς

ΑΜ : 06/3059

Ίδρυμα/Τμήμα : Αλεξάνδρειο Τεχνολογικό Εκπαιδευτικό

Ίδρυμα/Πληροφορικής

Αντικείμενο Πτυχιακής(1/2)

Δημιουργία εργαλείου για διαχείριση έργων λογισμικού.

Με βάση τις περιπτώσεις χρήσης:

- Διαχειρίζεται τον χρόνο ανάπτυξης ενός έργου λογισμικού.
- Παρακολουθεί την ανάπτυξη νέων λειτουργιών, αλλαγών, σφαλμάτων, ανά περίπτωση χρήσης.
- Υλοποιεί τη λειτουργικότητα εκτίμησης προσπάθειας (effort estimation).
- Ενσωματώνει ένα διαθέσιμο εργαλείο μετρικών (metrics tool).

Αντικείμενο Πτυχιακής(2/2)

Η εργασία χωρίστηκε σε τρία (3) επιμέρους συστατικά κομμάτια κατά την σχεδίαση και συγγραφή της εφαρμογής:

- ✓ Interface της εφαρμογής
- ✓ Εισαγωγή effort estimation methods
- ✓ Ενσωμάτωση ενός εργαλείου metrics

Ανάπτυξη με τη λογική των components

Σημαντικός αρωγός το Visual Studio
(σχεδόν οτιδήποτε είναι ένα component).

Στόχοι χρήσης components στην εφαρμογή προς
ανάπτυξη:

- Ικανότερη επεκτασιμότητα
- Επαναχρησιμοποιησιμότητα
- Ταχύτητα στην ανάπτυξη

Διαθέσιμα εργαλεία στην αγορά

- ✓ Δεν εντοπίστηκε κάποιο που να ικανοποιεί συνολικά τις απαιτήσεις μας (τουλάχιστον freeware).
- ✓ Υπολογίζουν ή μόνο μετρικές ή εκτίμηση προσπάθειας ή δημιουργούν διαγράμματα Gantt

Στόχος μας:

- ❖ Ικανοποιητικό GUI.
- ❖ Δυνατότητα εξαγωγής effort estimation ανά περίπτωση χρήσης (μέθοδος Use Case Points).
- ❖ Μετρικές από κομμάτια κώδικα (Java).

Περιπτώσεις Χρήσης

- Δεν μας απασχόλησε η γραφική απεικόνιση των περιπτώσεων χρήσης , αλλά η αναλυτική τους περιγραφή και καταγραφή σε κάποιο έγγραφο.
- Χρησιμοποιούμε πιο απλούς τύπους καταγραφής ή πιο λεπτομερείς, ανάλογα με την κρίση μας και τις απαιτήσεις του έργου προς ανάπτυξη.

Πρόγραμμα που χρησιμοποιήθηκε για την ανάπτυξη του εργαλείου

Visual studio 2010.

- Περιβάλλον πιο γνώριμο στο χρήστη.
- Interface παραθύρων των windows.
- Ευκολότερη ομαδοποίηση των διαφόρων περιπτώσεων χρήσης σε βάση δεδομένων.

Λειτουργικότητα εκτίμησης προσπάθειας (effort estimation) 1/3

- Η εκτίμηση προσπάθειας που χρησιμοποιούμε είναι αυτή που προτάθηκε από τον Gustav Karner 1993.
- Μέθοδος υλοποίησης: σημεία περιπτώσεων χρήσης (Use Case Points).

Λειτουργικότητα εκτίμησης προσπάθειας (effort estimation) 2/3

Αναλύει:

- τους συμμετέχοντες (actors) της περίπτωσης χρήσης
- τα σενάρια
- ποικίλους τεχνικούς και εξωγενείς παράγοντες

Λειτουργικότητα εκτίμησης προσπάθειας (effort estimation) 3/3

1. Ο παράγοντας τεχνικής πολυπλοκότητας (TCF)
2. Ο παράγοντας εξωγενούς πολυπλοκότητας (ECF)
3. Σημεία αστάθμητου βάρους στις περιπτώσεις χρήσης (UUCP)
4. Ο παράγοντας παραγωγικότητας (PF)

Τελική προσπάθεια, η εξίσωση:

$$UCP = TCF * ECF * UUCP * PF$$

Παράγοντας τεχνικής πολυπλοκότητας (TCF) 1/2

- Υπάρχουν 13 τεχνικοί παράγοντες που υπολογίζουν τις επιπτώσεις στην παραγωγικότητα ενός έργου, τα διάφορα τεχνικά προβλήματα.
- Η ομάδα ανάπτυξης τους αποδίδει μία τιμή (perceived complexity) μεταξύ μηδέν και πέντε.

Παράγοντας τεχνικής πολυπλοκότητας (TCF) 2/2

- Η τιμή που αποδίδεται αφορά την πρόβλεψη πολυπλοκότητας του έργου.
- Τιμή “0” ελάχιστος σημασίας, τιμή “3” μέτριας και “5” ισχυρή επιρροή για το project.
- Πολλαπλασιάζεται με ένα βάρος (weight).

Εξίσωση:

$$TCF = 0.6 + (0.01 * \textit{Technical Total Factor})$$

Παράγοντας εξωγενούς πολυπλοκότητας (ECF) 1/2

- Παράγει μία εκτίμηση για την εμπειρία της ομάδας ανάπτυξης.
- Πιο έμπειρες ομάδες, παίζουν μεγαλύτερο ρόλο στη διαμόρφωση της UCP από ότι ομάδες με λιγότερη εμπειρία.
- Υπάρχουν 8 εξωγενείς παράγοντες που επηρεάζουν την επιτυχημένη ολοκλήρωση ενός έργου.

Παράγοντας εξωγενούς πολυπλοκότητας (ECF) 2/2

- Η ομάδα ανάπτυξης τους αποδίδει μία τιμή (perceived complexity) μεταξύ “0” και “5”.
- Τιμή “0” ελάχιστος σημασίας, τιμή “3” μέτριας και “5” ισχυρή επιρροή για το project.
- Πολλαπλασιάζεται με ένα βάρος (weight).

Εξίσωση:

$$ECF = 1.4 + (-0.03 * \textit{Environmental Total Factor})$$

Σημεία αστάθμητου βάρους (UUCP)

- The Unadjusted Use Case Weight (UUCW).
Συνολικός αριθμός δραστηριοτήτων (ή βημάτων) που εμπεριέχονται σε όλα τα σενάρια περιπτώσεων χρήσης.
- The Unadjusted Actor Weight (UAW).
Συνδυασμένη πολυπλοκότητα όλων των συμμετεχόντων (actors) σε όλες τις περιπτώσεις χρήσης.

Εξίσωση: $UUCP = UUCW + UAW$

Unadjusted Use Case Weight(UUCW)

- Προέρχεται από τον αριθμό περιπτώσεων χρήσης, χωρίζονται σε τρεις κατηγορίες: simple, average, complex.
- Κάθε περίπτωση χρήσης κατηγοριοποιείται με βάση τον αριθμό βημάτων που το σενάριό της περιέχει (συμπεριλαμβανομένων και των εναλλακτικών ροών).

Unadjusted Actor Weight(UAW)

- Οι τύποι των συμμετεχόντων ορίζονται ως simple, average, complex.
- Υπολογίζεται προσθέτοντας τον αριθμό των συμμετεχόντων σε κάθε κατηγορία.
- Πολλαπλασιάζουμε αυτό το σύνολο με τον προκαθορισμένο παράγοντα βαρύτητάς του.
- Τέλος προσθέτουμε τα παραγόμενα.

Παράγοντας παραγωγικότητας (PF)

- Είναι μια αναλογία ωρών εργασίας ανά περίπτωση χρήσης, βασιζόμενη σε παλιότερες περιπτώσεις.
- Αν δεν υπάρχουν παλιότερα στοιχεία, δίνεται μια τιμή ανάμεσα στο 15 και 30, συνήθως προτεινόμενη τιμή το 20.

Λειτουργικότητα εξαγωγής μετρικών κώδικα 1/2

- Οι μετρικές που χρησιμοποιήθηκαν είναι αυτές που έχουν προταθεί αρχικά από τους S.R Chidamber και C.F Kemerer το 1991.
- Χρησιμοποιούμε το πρόγραμμα ckjm ως το μέσο που θα παράγει τα αποτελέσματα από bytecode compiled Java αρχεία (.java). (Diomidis D. Spinellis)

Λειτουργικότητα εξαγωγής μετρικών κώδικα 2/2

Μετρικές που μας ενδιαφέρουν και υπολογίζουμε:

- WMC: Weighted methods per class
- DIT: Depth of Inheritance Tree
- NOC: Number of Children
- CBO: Coupling between object classes
- RFC: Response for a Class
- LCOM: Lack of cohesion in methods
- Ca: Afferent couplings
- NPM: Number of public methods

WMC: Weighted methods per class

- Υπολογίζει το σύνολο των πολυπλοκοτήτων των μεθόδων μιας κλάσης.
- Μέτρο πολυπλοκότητας μπορούμε να χρησιμοποιήσουμε την CC, ή ορίζουμε αυθαίρετα τιμή πολυπλοκότητας 1 για κάθε μέθοδο.
- Το `ckjm`, κάνει το παραπάνω, επομένως η WMC είναι ίση με τον αριθμό μεθόδων στην κλάση.

DIT - Depth of Inheritance Tree

- Προβλέπει για κάθε κλάση μία μέτρηση επιπέδων κληρονομικότητας από την κορυφή ιεραρχίας αντικειμένου.
- Στην Java όπου όλες οι κλάσεις κληρονομούν την Object η ελάχιστη αξία της DIT είναι 1.

NOC - Number of Children

Μετρά απλά τον αριθμό των άμεσων απογόνων μιας κλάσης.

CBO - Coupling between object classes

- Αντιπροσωπεύει τον αριθμό κλάσεων που συνδέονται με μια δεδομένη κλάση (efferent συζεύξεις, CE).
- Η σύζευξη μπορεί να εμφανιστεί μέσω κλήσεων μεθόδου, πρόσβασης σε πεδία, κληρονομικότητας, arguments, τύπων επιστροφής και εξαιρέσεων.

RFC - Response for a Class

Υπολογίζει τον αριθμό διαφορετικών μεθόδων που μπορούν να εκτελεσθούν όταν λαμβάνει ένα αντικείμενο της κλάσης, ένα μήνυμα.

LCOM - Lack of cohesion in methods

- Μετράει τα σετ μεθόδων σε μια κλάση που δεν συσχετίζονται μέσω του sharing, μερικών από τα πεδία της κλάσης.
- Ο αρχικός ορισμός της, (που είναι και αυτός που χρησιμοποιεί το `ckjm`) εξετάζει όλα τα ζεύγη των μεθόδων μιας κλάσης.

Ca - Afferent couplings

- Υπολογίζει πόσες άλλες κλάσεις χρησιμοποιούν τη συγκεκριμένη κλάση.
- Υπολογίζεται χρησιμοποιώντας τον ίδιο ορισμό με αυτόν για τον υπολογισμό της CBO (Ce).

NPM - Number of Public Methods

- Μετρά όλες τις μεθόδους σε μια κλάση που δηλώνονται ως δημόσιες.
- Μπορεί να χρησιμοποιηθεί για να μετρήσει το μέγεθος ενός API που παρέχεται από ένα πακέτο.

Καθορισμός αποδεκτών ορίων μετρικών 1/2

- Αποδεκτά όρια για τις βασικές “6” μετρικές.
- Προκύπτουν από συμπεράσματα του paper των Linda H. Rosenberg, Ruth Stapko και Albert Gallo.
- Στόχος : Εντοπισμός των πιο προβληματικών σημείων στον κώδικα, ώστε να είναι πιθανό να γίνουν βελτιώσεις στην απόδοσή του.

Καθορισμός αποδεκτών ορίων μετρικών 2/2

- Weighted Methods per Class (WMC) ≤ 25 ιδανικό, ≤ 40 αποδεκτό.
- Response for Class (RFC) ≤ 50 .
- Coupling Between Objects (CBO) ≤ 5 .
- Depth in Tree (DIT) > 5 .
- Number of Children (NOC) = πολύ μεταβλητή τιμή
- LCOM ≤ 2 (θεωρούμε), συνεκτική κλάση για τιμή “0”, τιμή $>$ του “0” υποδεικνύει πιθανή διάσπαση σε περισσότερες κλάσεις.

Συμπεράσματα

- Κύριος όγκος εργασίας, ήταν το προγραμματιστικό κομμάτι.
- Το αντικείμενο της πτυχιακής μπορεί να θεωρηθεί πολύ σημαντικό, καθώς το λογισμικό αναπτύσσεται ταχύτατα.
- Υπάρχει ανάγκη επίβλεψης κάθε προσπάθειας ανάπτυξης λογισμικού.

Future Work

- Προσθήκη διαγραμμάτων Gantt.
- Συμβατότητα με άλλους τύπους αρχείων όπως .crr, .cs.
- Επέκταση λίστας υπολογίσιμων μετρικών.

Ερωτήσεις



Demo

Software Tool

Metrics Effort Estimation Application Tools About Exit

+ X Save Undo

NumPX	NamePX	Priority	Steps	UseCaseType	Weight	Atomo	Omada	Duration	StartDate	FinishDate	DOCS	Effort	Metrikes	EffortTime
1	PX1	1	4	Average	10	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	14-Apr-13	15-Apr-13	Open	Calculate	Calculate	4.034
2	PX2	2	5	Simple	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	3	16-Apr-13	17-Apr-13	Open	Calculate	Calculate	2.1
3	PX3	3	2	Complex	15	<input checked="" type="checkbox"/>	<input type="checkbox"/>	4	20-Apr-13	25-Apr-13	Open	Calculate	Calculate	2.625
4	PX4	1	4	Simple	5	<input type="checkbox"/>	<input type="checkbox"/>		Apr-13		Open	Calculate	Calculate	2.288
5	px5	4	2	Simple	5	<input type="checkbox"/>	<input type="checkbox"/>		Jul-13		Open	Calculate	Calculate	10.5
6	px6	2	5	Simple	6	<input type="checkbox"/>	<input type="checkbox"/>		Jul-13		Open	Calculate	Calculate	12.48

Software Tool

Login

Username:

Password:

Windows Taskbar: 10:01 PM 27-Jun-13