



Politechnika Wroclawska

Modele i metryki w inżynierii oprogramowania

Lech Madeyski

Lech.Madeyski@pwr.wroc.pl

<http://madeyski.e-informatyka.pl/>

Politechnika Wroclawska

Instytut Informatyki

Zakład Inżynierii Oprogramowania



O mnie

- I-32/ZIO,
- IEEE/ACM/PTI/IR/...
- e-Informatica Software Engineering Journal - co-Editor-in-Chief
<http://www.e-informatyka.pl/e-Informatica>
- SEnS <http://www.e-informatyka.pl/sens/>
- Konkurs na najlepszą pracę magisterską z informatyki



O mnie

Zainteresowania badawcze:

- Empirical Studies in Software Engineering,
- Software Metrics and Quality Models,
- Software Quality, Software Process Improvement,
- Agile Software Development Methodologies and Practices (e.g. eXtreme Programming, TDD),
- Data mining in Software Engineering,
- Defect Prediction Models,
- Aspect-Oriented Programming,
- Web Applications Architectures, Design Patterns, Technologies.



O przedmiocie

- Tygodniowa liczba godzin zajęć:
 - 1W
 - 2P



Konsultacje

- Wtorek 11-13
- Czwartek 10-11, 17-18

Pokój 415 B-4



Wykład

Omówienie:

- podstawowych pojęć (metryka, pomiary, model)
- najbardziej interesujących publikacji dotyczących modeli na przykładzie predykcji błędów (w tym stosowanych metryk),
- planowanie i przeprowadzania badań empirycznych,
- metoda GQM,
- prawa w IO,
- pakiet statystyczny R (+ Sweavy + LaTeX).



Wykład

- 18X-13XII: chętni w grupach 1-4 os. mogą zgłaszać 10-45 min. prezentacje najciekawszych wyników i metod badawczych w zakresie modeli predykcji w IO i stosowanych metryk.
- [+0..10 procent punktów z kolokwium]-zapisy mailem (temat, czas i kompletna treść prezentacji).
- Decyduje atrakcyjność propozycji (mogą zostać odrzucone) i kolejność zgłoszeń.



Wykład - egzamin i wymagania wstępne

- 31I i 7II: pierwszy i drugi termin egzaminu
- Sala 30D1
- Wymagania wstępne zaliczenia wykładu:
 - Student **nie** musi zaliczyć zajęć towarzyszących, ale musi uczestniczyć w wykładach i stawić się na egzamin.
- Zasady dyskwalifikacji:
 - Student otrzymuje ocenę niedostateczną, jeżeli w czasie egzaminu zostanie przyłapany na korzystaniu ze ściąg lub pomocy kolegów.



Projekt (zespołowy 4 os.) - cel:

Zebranie danych na potrzeby modelu predykcji poprzez twórcze rozwinięcie znanych z literatury modeli predykcji oraz dostępnych zbiorów danych z projektów przemysłowych i open-source.

Punktem wyjścia mogą być publikacje i zbiory danych dostępne są na stronie <http://www.e-informatyka.pl/sens/Wiki.jsp?page=MMSE>

lub zaakceptowana przeze mnie inna propozycja.

Kiedy ocena? Harmonogram na następnych slajdach.

Co podlega ocenie? Walory praktyczne i badawcze (potencjalna publikowalność uzyskanych rezultatów) - zbiorów danych i raportów badawczych ->kolejne slajdy



Zadania w poszczególnych tygodniach

1. Podział na grupy 4-os. i wybór szefa zespołu, BHP.
2. Raport badawczy - propozycja badań rev.0 (propozycja badań, modelu, metryk znanych i NOWYCH [ma wpływ na wartość badawczą!], przegląd literatury [klasyczny lub lepiej systematyczny <http://www.e-informatyka.pl/sens/Wiki.jsp?page=SLR>])
3. Raport badawczy - propozycja badań rev.1 (pogłębiony przegląd literatury, precyzyjny opis znanych i NOWYCH metryk oraz wskazanie możliwości ich zebrania przez odpowiednie narzędzia - prezentacja zestawu narzędzi do zbierania metryk)



Zadania w poszczególnych tygodniach

4. Milestone 1: Raport badawczy [preferowany język ang. i LaTeX - klasa article] opisujący jakie projekty open source i przemysłowe [te najwyżej punktowane!] będą badane, za pomocą jakich narzędzi, jakie metryki produktu i procesu będą zbierane z wyraźnym zaznaczeniem co stanowi nowe ujęcie problemu, jakie nowe metryki planuje się zbierać, jakie nowe narzędzia planuje się stworzyć (niektóre z projektów, za zgodą prowadzącego, mogą skupić się na tym zadaniu).
5. Raport z postępów prac: uaktualniony raport badawczy (LaTeX + dane w pliku XLS i bazie danych)
6. J.w.



Zadania w poszczególnych tygodniach

7. J.w.

8. J.w.

9. Milestone 2: Raport z postępów prac: uaktualniony raport badawczy (LaTeX + dane w pliku XLS i bazie danych)

10. J.w. + udostępnienie zebranych danych do przeglądu

11. Raport pisemny z przeglądu cudzych projektów (omawianie błędów/obrona przed zarzutami) [PI,PM]

12. Raport z postępów prac (zawierający listę błędów wskazanych podczas przeglądu i podjętych czynności naprawczych): uaktualniony raport badawczy (LaTeX + dane w pliku XLS i bazie danych)



Zadania w poszczególnych tygodniach

13. J.w.

14. Milestone 3: Finalna postać raportu badawczego (LaTeX/j.ang.) a w nim też wstępna ocena możliwości predykcji za pomocą zebranych metryk (proste korelacje, regresja logistyczna/liniowa etc.) [na CD i papierze] + dane (XLS+baza danych) [na CD]

15. Wystawienie ocen



Projekt

OcenaZPrzeglądów=PM+PI

- PI - Waga i liczba znalezionych błędów u innej grupy podczas przeglądu 1 (np. błędy w wartościach metryk) [0..10pkt]
- PM - Waga i liczba błędów znalezionych u nas przez inną grupę podczas przeglądu 1 [0..5pkt]



Projekt - ocena finalna

Ocena=OcenaZPrzeglądów [0..15pkt]+

OcenaZaMilestone1[0..10pkt]+

OcenaZaMilestone2[0..10pkt]+

OcenaZaMilestone3[0..10pkt]+

WartośćDanych[0..30pkt]+

WartośćRaportu[0..25pkt]

WartośćDanych-Moja ocena wartości badawczej (możliwości publikacyjnych) i praktycznej zebranych danych [0..30pkt]

WartośćRaportu -zaawansowanie prac nad raportem badawczym / publikacją i jej wartość badawcza (publikacyjna) [0..25pkt]



Projekt - skala ocen i wymaganie wstępne do uzyskania zaliczenia :

Wymaganie wstępne: Student może opuścić co najwyżej 2 zajęcia projektowe.

Skala ocen: 5.5 95...100

- 5.0 85...94
- 4.5 80...84
- 4.0 70...79
- 3.5 65...69
- 3.0 55...64
- 2.0 ≤ 54 pkt
- Uwaga: Prowadzący zajęcia może ustalić dodatkowe wymagania i możliwość zdobycia dodatkowych punktów.



Projekt: zasady dyskwalifikacji

- Student otrzyma ocenę niedostateczną z projektu, jeżeli przedłoży rozwiązanie, którego nie jest autorem lub będące plagiatem lub kopią całości lub pewnej części innego rozwiązania, publikacji, raportu, zbioru danych etc.



Przykładowe tematy

Modele predykcji błędów:

- Predicting Top Crashes at an Early Stage to Prioritize Debugging Efforts
- Change Bursts as Defect Predictors
- Code Churn as Defect Predictors
- Mutation Score as Defect Predictors
- Organizational Structure as Defect Predictors
- Process Metrics as Defect Predictors
- Code Complexity as Defect Predictors
- Dependencies as Defect Predictors



Literatura

- Google
- Scholar Google
- Materiały konferencji PROMISE, MSR.
- Prestiżowe czasopisma naukowe z IO.
- Książki (na następnych slajdach)



Literatura - książki

Metryki i modele:

- S.H. Kan, Metrics and Models in Software Quality Engineering (2nd Edition), Addison-Wesley 2002

Książki z pogranicza IO i statystyki i książki ze statystyki:

- Introductory Statistics with R
- ...