

Podpora malých tímov pri modelovaní softvéru

Bakalárska práca

Jakub Ondik



PeWe@FIIT
personalized web group

- Synchronizácia a konzistencia modelov
- Určenie autorstva jednotlivých častí modelov
- Detekcia a korekcia chýb v modeloch
 - Včasné upozornenie na chyby v prípade začínajúcich študentov

- Analyzovaných 20 prác študentov Princípov softvérového inžinierstva
- Najčastejšie sa vyskytujúce chyby v diagramoch študentov:

Typ diagramu	Chyby
Diagram činností	Chýbajúce spájacie uzly
	Nesprávne pomenované počiatočné alebo koncové uzly
Diagram prípadov použitia	Nesprávna určené prípady použitia
	Nesprávne určené alebo chýbajúce vzťahy include
Diagram tried (entitno-relačný)	Nesprávne určené alebo chýbajúce kardinality
	Chýbajúce triedy
Sekvenčný diagram	Nesprávne určené správy
	Zle umiestnené alebo označené fragmenty
Stavový diagram	Nesprávne nazvané prechody (udalosti, podmienky)
	Nesprávne nazvané stavy

- ACME
- Minerva a Hydra
- UMLint
- Spoločné problémy
 - Nedostatočné pokrytie typov chýb
 - Detekcia chýb až po vyžiadaní
 - Absencia automatickej opravy

- Založená na pravidlách
- Lahko rozšíriteľná o nové pravidlá
- Podmienená akciou používateľa
- Sledovanie akcií a chýb používateľa
- Zložená z 3 častí
 - Načítanie pravidiel vrátane aktualizácie
 - Automatická detekcia a korekcia chýb na vyžiadanie
 - Sledovanie akcií a chýb – tvorba datasetu

- Kontrola aktualizácie pri štarte a na vyžiadanie
- JSON štruktúra
- Vytvorenie inštancie a postupné spracovanie

```
{
  name: 'Flow between decision node and activity should contain
    guard',
  element: {
    type: 'ControlFlow',
    stereotype: '*'
  },
  attribute: {
    type: 'TransitionGuard',
    stereotype: '-'
  },
  content: {
    defectMsg: 'Chybajuci GUARD pre prechod, ktory smeruje
      do aktivity "{0}".',
    valid: 'getTransitionGuard',
    correct: 'setTransitionGuard(podmienka)',
    cond: 'Connector.getSrcType(Decision) &
      Connector.getTargetType(Activity)'
  }
}
```

- Automatická detekcia chýb založená na akciách
 - Sledovači udalostí
- Upozornenie na chybu v čase jej vzniku
- Možnosť zvýrazniť prvok súvisiaci s chybou
- Korekcia detegovanej chyby na vyžiadanie
- Ignorovanie chyby

- Sledovanie a uchovávanie vykonaných akcií
- Ukladanie informácií o vzniknutých chybách
 - Ignorovanie detegovaných chýb
- Tvorba datasetu
 - 42 624 akcií študentov
 - Predikcia chýb

- Klient – server architektúra
- Klient v podobe rozšírenia pre Enterprise Architect
- Server v podobe REST webovej služby
- Nasadenie na predmete Princípy softvérového inžinierstva
 - 25 študentov, 18 prác - skupina C
 - Porovnanie so študentmi, ktorí nepoužívali rozšírenie
 - Skupina A - 20 minuloročných prác, B - 20 tohtoročných prác

Typ chyby	Sk. A	Sk. B	Sk. C
Nesprávne určenie strážcovia	9	59	0
Chýbajúce body rozšírenia	11	37	0
Chýbajúce názvy asociácií	7	50	0
Chýbajúce kardinality	47	8	0
Nesprávne nazvané triedy	12	12	1

- Vytvorenie pravidiel pre nové typy chýb
- Doplnenie pravidiel pre sekvenčný diagram
- Predikcia chýb pomocou strojového učenia
- Automatické hodnotenie častí modelov
- Explicitná spätná väzba od pedagóga priamo v modeli

- Identifikované chyby v diagramoch študentov
- Návrh metódy pre detekciu a korekciu chýb
- Implementácia navrhnutej metódy a vytvorenie pravidiel
- Nasadenie a zistenie nových typov chýb
- Príspevok na medzinárodnú konferenciu FSE 2016