SVEUČILIŠTE U RIJECI

**TEHNIČKI FAKULTET**

Preddiplomski studij računarstva

Seminarski rad

**Defect classification and analysis**

(Klasifikacija i analiza pogrešaka)

<https://s3.amazonaws.com/academia.edu.documents/33650399/IJCA_Publication.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1556839235&Signature=vpns4I%2FeKkYPAaavz5SSCygzZoE%3D&response-content-disposition=inline%3B%20filename%3DDefect_Analysis_and_Prevention_for_Softw.pdf>

<http://www.professionalqa.com/types-of-defects-in-software-testing>

<https://melsatar.blog/2018/06/26/the-software-process-improvement-spi-reward-or-risk/>

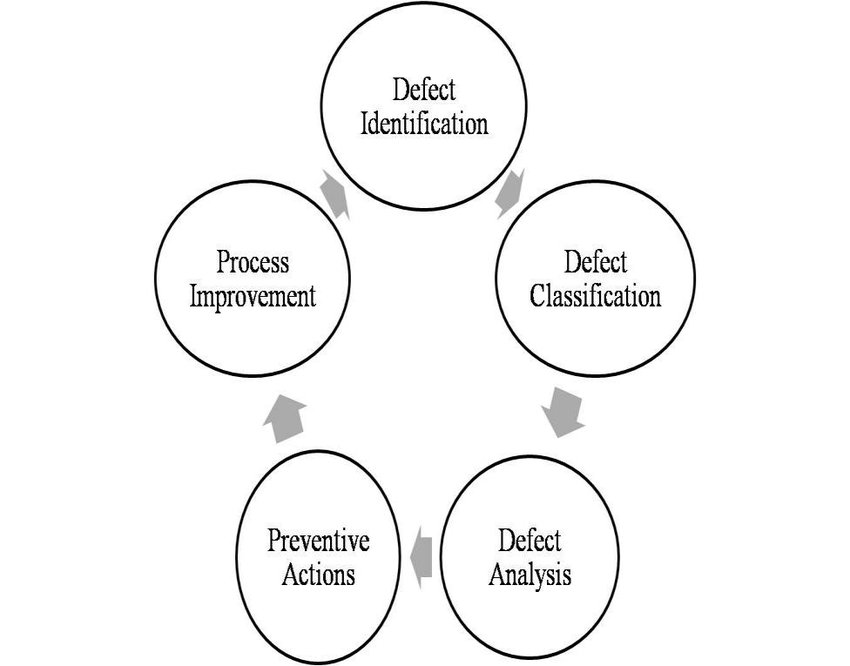
<https://www.softwaretestinghelp.com/defect-prevention-methods/>

1. UVOD

Softverske pogreške mogu biti definirane kao nepravilnosti u razvoju programskog proizvoda zbog kojih proizvod ne zadovoljava dana očekivanja. Pogreške se događaju tijekom cijelog procesa razvoja, te je njihova prevencija postala bitna stavka u procesu poboljšanja kvalitete proizvoda. Svrha prevencije pogrešaka je identificiranje istih, te sprječavanje njihovog ponovnog pojavljivanja.

1. Ciklus procesa poboljšanja

Na Slici 1 prikazan je ciklus procesa poboljšanja programskog proizvoda. Pogreške se prvo identificiraju, te se zatim klasificiraju i analiziraju. Pronalazeći glavni uzrok, one se eliminiraju i uspostavlja se preventivni mehanizam da bi se spriječilo njihovo ponavljanje. Time se poboljšava programski proizvod.



Slika 1

* 1. Identifikacija pogrešaka

Pogreške se pronalaze planiranim aktivnostima čiji je cilj isključivo razotkrivanje istih. Generalno, pogreške se identificiraju tijekom raznih stadija života programskog proizvoda pomoću aktivnosti kao što su *pregled dizajna, pregled GUI-a*, *pregled koda*… Jednom kada su identificirane, pogreške se klasificiraju jednom od metoda.

* 1. Klasifikacija pogrešaka
     1. Klasifikacija na temelju ozbiljnosti pogreške

Ozbiljnost pogreške se definira po stupnju utjecaja iste na programski proizvod tj. koliko intenzivno će ona utjecati na proizvod. Može biti kategorizirana u jednu od četiri moguće kategorije:

1. Kritična pogreška – ove pogreške zahtijevaju neposrednu pažnju. Kritične pogreške utječu na esencijalne funkcionalnosti koje mobu rezultirati padom cijelog sustava.
2. Velika pogreška – pogreške koje utječu na osnovne funkcionalnosti proizvoda. Pogreška nije rezultirala padom cijelog sustava ali su bitne funkcionalnosti narušene.
3. Mala pogreška – pogreške koje nemaju značajan utjecaj na rad programskog proizvoda. One ne spriječavaju korisnika u izvršavanju zadataka.
4. Trivijalna pogreška – pogreške koje nemaju nikakav utjecaj na rad programskog proizvoda. Ponekad se čak i ignoriraju, kao npr. gramatičke greške
   * 1. Klasifikacija na temelju vjerojatnosti

Pogreške se klasificiraju ovisno o tome kolika je vjerojatnost da će se pogreška pojaviti te da će korisnik naići na nju.

1. Visoka vjerojatnost – postoji velika vjerojatnost da će većina korisnika doći do pogreške
2. Srednja vjerojatnost – polovica korisnika je u mogućnosti naići na pogrešku
3. Niska vjerojatnost – pogreške su detektirane kod nekoliko korisnika ili čak niti jednog
   * 1. Klasifikacija po prioritetu

Pogreške se mogu promatrati i iz poslovne perspektive. Neke pogreške treba ispraviti prije, dok se druge mogu ispraviti u kasnijim stadijima razvoja, ovisno o trenutnim potrebama i zahtjevima poslovnog sustava.

1. Visoki prioritet – označava najistaknutije potrebe sustava i pogreška se mora ispraviti što je prije moguće
2. Srednji prioritet – pogreške je moguće ispraviti u bilo kojoj slijedećoj verziji produkta
3. Nizak prioritet – pogreške se ne moraju zasebno ispravljati, već zajedno sa drugim pogreškama koje trebaju ispravak
   1. Analiza pogreške

Pitanja koja se postavljaju prilikom analize pogrešaka su:

Što? – identifikacija i klasifikacija

Gdje? – u kojem koraku

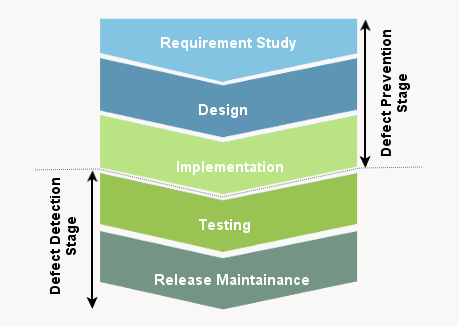
Kada? – kada se greška pojavila

Kako / Zašto?

* 1. Preventivne akcije

Na Slici 2 se može vidjeti u kojim fazama se pogreške mogu preventirati a u kojima detektirati. Postoje nekoliko metoda pomoću kojih se preventiraju pogreške. Neke od njih su:

* Pregled i inspekcija – ova metoda uključuje članove tima koji pregledavaju sve krajnje produkte.
* Uspoređivanje sa prototipovima
* Evidentiranje pogrešaka i dokumentacija – ova metoda pruža informacije i parametre pomoću kojih se pogreške analiziraju
* Analiza uzroka – o tome ćemo više kasnije



Slika 2

* 1. Unaprjeđenje proizvoda

Zadnja faza u ciklusu je unaprjeđenje proizvoda. Provođenjem svih stadija u ciklusu programski proizvod postaje sve efikasnije i kvalitetniji te se tako unaprjeđuje.