
	METODIKA	Verzia :	1.0
	Prehliadky zdrojového kódu	Strana :	1
		Číslo :	M2015-5
		Dátum vydania:	7/12/2015

Metodika č. 2015 – 4

Prehliadky zdrojového kódu

	VYPRACOVAL
FUNKCIA	Manažér prehliadok kódu
MENO, TITUL	Martin Žalondek, Bc.
DÁTUM	06/12/2015
PODPIS	


	METODIKA	Verzia :	1.0
	Prehliadky zdrojového kódu	Strana :	2
		Číslo :	M2015-5
		Dátum vydania:	7/12/2015

1. Účel a dedikácia metodiky

Prehliadka kódu je systematická analýza zdrojového kódu. Jej účelom je odstránenie nedostatkov, ktoré si autor pri písaní kódu nevšimol. Táto metodika je určená pre osoby zodpovedné za prehliadky kódu, manažment verzií a rovnako pre všetkých, ktorí vytvárajú zdrojový kód.


a. Slovník pojmov

Inšpektor – osoba, ktorá má za úlohu vykonať prehliadku zdrojového kódu

	METODIKA	Verzia :	1.0
	Prehliadky zdrojového kódu	Strana :	3
		Číslo :	M2015-5
		Dátum vydania:	7/12/2015

1. Párové programovanie

V projekte využívame prístup párového programovania. Jedná sa o techniku agilného vývoja, kedy súčasne dvaja vývojári pracujú na rovnakej časti kódu ako pár. Jeden píše kód, zatiaľ čo druhý reviduje každý riadok napísaného kódu a premýšľa nad novými nápismi a nepriaznivými dôsledkami, ktoré sa môžu vyskytnúť v budúcnosti. Role sa pravidelne striedajú.

	METODIKA	Verzia :	1.0
	Prehliadky zdrojového kódu	Strana :	4
		Číslo :	M2015-5
		Dátum vydania:	7/12/2015

2. Prehliadka kódu použitím verziovacieho nástroja

Verziovací nástroj [Bitbucket](#) (pozri Metodika verziovania zdrojového kódu – M2015-3), poskytuje prostredie na zobrazovanie zmien v zdrojovom kóde. Ak člen tímu odošle do repozitára novú verziu kódu, požiada niekoho z ostatných členov o revíziu. Ten následne spíše zoznam pripomienok, ktoré je potrebné zapracovať a odovzdá ich autorovi kódu. Nie je odporúčané inšpektorovi zasahovať do kódu bez informovania autora. V budúcnosti to môže spôsobiť komplikácie.

Inšpektor ma za úlohu okrem chýb v kóde sledovať aj dodržiavanie zaužívaných konvencií ako napr. spôsob pomenovania metód a premenných. Autor kódu je povinný inšpektorovi vysvetliť jednotlivé časti kódu a to písaním komentárov pri jednotlivých sekciách kódu. Prehliadky sa musia robiť priebežne na malých množstvách kódu (maximálne 100 – 200 riadkov) a nie nárazovo, aby sa zaistila dôkladnosť prehliadky.

```

openseience/searchDataset/views.py
def get(self, request, *args, **kwargs):
    searchForm = SearchForm()
    number_of_results = 0

    if 'search_bar' in kwargs:
        query=kwargs['search_bar']
        results = []
        try:
            index = search.Index("dataset_index")
            search_results = index.search(query)

            returned_count = len(search_results.results)

            for doc in search_results:
                title = doc.field('title').value
                author = doc.field('author').value
                uploaded = doc.field('uploaded').value
                description = doc.field('description').value
                results.append(
                    [title,author,uploaded,descr

except search.Error:

```

```

def get(self, request, *args, **kwargs):
    searchForm = SearchForm()
    number_of_results = 0
    results = []

    if request.GET.get('search_bar') is not None:
        print("Search bar je: "+request.GET.get('search_bar'))
        query=request.GET.get('search_bar')
        results = []
        try:
            index = search.Index("dataset_index")
            search_results = index.search(query)

            number_of_results = len(search_results.results)

            for doc in search_results:
                title = doc.field('title').value
                author = doc.field('author').value
                uploaded = doc.field('uploaded').value
                description = doc.field('description').value
                results.append(
                    [title,author,uploaded,descr

```

Obr. 1 Porovnanie rozdielov v kóde v nástroji Bitbucket