

SPIS TREŚCI

Scenariusze Testowe	2
Wstęp.....	2
Stan przeprowadzonych testów w kolejnych wersjach.....	3
Bezpieczeństwo aplikacji.....	4
Łączenie z aplikacją poprzez https na serwerze tomcat.....	4
Testy wydajnościowe	4
Wstęp.....	4
Przygotowany scenariusz testowy.....	4
Wyniki	5
Zakończenie implementacji – statystyki	9
Continuous integration.....	9
Github	11

SCENARIUSZE TESTOWE

WSTĘP

Scenariusze testowe zostały stworzone na podstawie wymagań funkcjonalnych aplikacji dostępnych w dokumencie z II fazy systemu, można je znaleźć w dokumencie z III fazy. Pokrywają one 100% wymagań funkcjonalnych, realizując każde co najmniej raz.

Wersja 0.0.2 aplikacji opisana w poprzednim dokumencie zawierała 89% przewidzianych funkcjonalności. Brakującymi funkcjonalnościami było dodawanie nowego członka do istniejącej grupy oraz opuszczanie grupy.

W obecnej wersji 0.0.3 wymienione wyżej funkcjonalności zostały dodane, przez co stan realizacji wymagań funkcjonalnych wynosi 100%.

Poniżej znajduje się tabelka z obecnym statusem zaliczonych testów funkcjonalnych. Obrazuje ona ilość zrealizowanych funkcjonalności w systemie.

STAN PRZEPROWADZONYCH TESTÓW W KOLEJNYCH WERSJACH

Wersja aplikacji	1.1	1.2	1.3	1.4	1.5	1.6	2.1	2.2	2.3	2.4	2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	Passed	Failed	Percent of realization
0.0.1																				9	10	47 %
0.0.2																				17	2	89 %
0.0.3																				19	0	100 %

W wersji 0.0.3 wszystkie testy zostały przeprowadzone z wynikiem pomyślnym.

Oznacza to, że wszystkie funkcjonalności założone w systemie zostały zrealizowane.

BEZPIECZEŃSTWO APLIKACJI

ŁĄCZENIE Z APLIKACJĄ POPRZECZ HTTPS NA SERWERZE TOMCAT

W celu zwiększenia poziomu bezpieczeństwa aplikacji, serwer tomcat został skonfigurowany pod kątem łączenia się z aplikacją cyberwej przez https.

Instrukcja konfiguracji SSL dla tomcata znajduje się na Wiki projektowym pod adresem <https://github.com/pitaatip/Cyberwej/wiki/SSL>

Serwer uruchamia się i pod adresem <https://localhost:8443/cyberwej-web/pages/loggingPage.jsf> można połączyć się z aplikacją. Poprawność działania zostanie zaprezentowana w czasie spotkania.

Ciekawym problemem jest fakt, że przy uruchomieniu aplikacji po https psuje się czcionka ustawiona w nagłówku szablonu strony aplikacji. Pomimo kilku prób nie udało się zidentyfikować i naprawić problemu, zmieniona czcionka pozostaje tajemnicą.

TESTY WYDAJNOŚCIOWE

WSTĘP

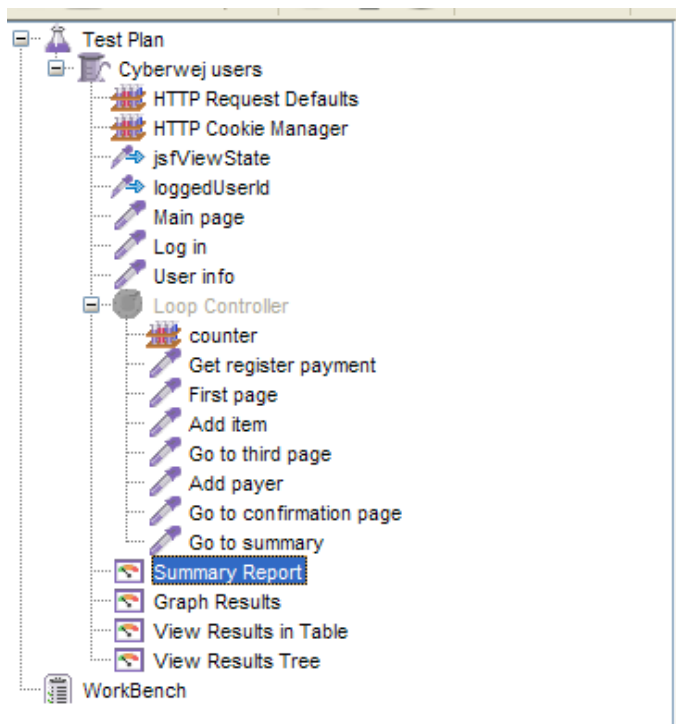
Testy wydajnościowe zostały przygotowane przy użyciu narzędzia JMeter. Za ich pomocą chcieliśmy w sposób automatyczny sprawdzić działanie naszego systemu przy większej ilości użytkowników oraz jego wydajność podczas pracy.

Ponadto, przeprowadzone zostało porównanie wydajności aplikacji (czas jej odpowiedzi na zapytanie) przy pustej i wypełnionej bazie danych.

PRZYGOTOWANY SCENARIUSZ TESTOWY

Scenariusz testowy zakłada zalogowanie się do aplikacji, przejście na ekran główny a następnie na ekran informacji o użytkowniku. Następnie w pętli dodawane są nowe płatności, poprzez automatyczne uzupełnianie i przechodzenie przez 3 formularze procesu dodawania nowej płatności.

Scenariusz taki został przygotowany w JMeterze, jego zapis widoczny jest na załączonym screenie.



Scenariusz ten został następnie uruchomiony w dziesięciu wątkach (startowanych po kolei co 3 sekundy) tak, aby symulował dziesięciu użytkowników obsługujących aplikację.

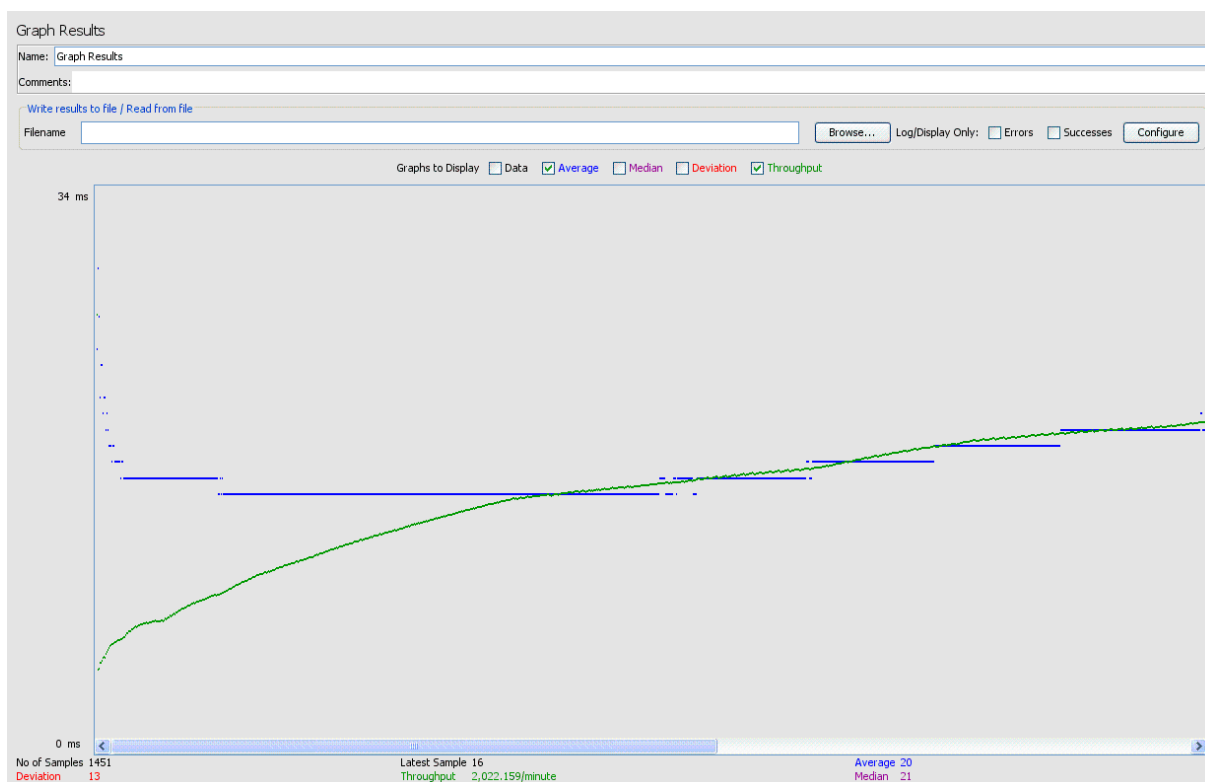
Sprawdzany był czas odpowiedzi aplikacji:

- przy pustej bazie danych, z wyłączoną częścią dodawania nowych płatności
- podczas wypełniania bazy - przy dodawaniu nowych płatności
- przy wypełnionej w ten sposób bazie danych, z wyłączoną częścią dodawania nowych płatności

WYNIKI

Poniżej przedstawione są wyniki przedstawiające czas odpowiedzi aplikacji w różnych momentach jej użytkowania. Do przetestowania aplikacji przy pustej i pełnej bazie danych, wykorzystywany był początek skryptu opisanego w powyższym akapicie.

PUSTA BAZA



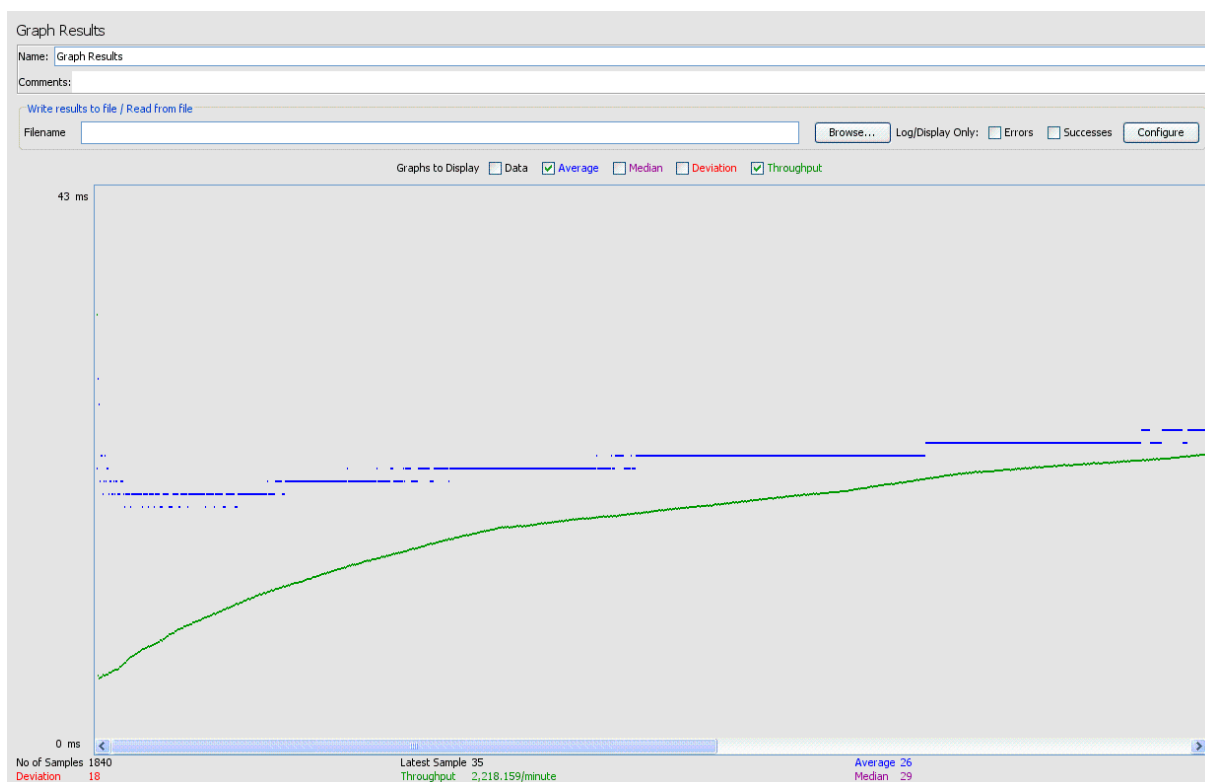
	Ilość wywołań	Avg [ms]	Min [ms]	Max [ms]	% błędów	stron / s	KB/s	średni rozmiar [B]
Main page	487	5	2	32	0	11.33	30.876	2790.525667
Log in	484	33	19	61	0	11.326	115.57	10448.13223
User info	480	23	13	62	0	11.262	88.64	8059.95
Total	1451	20	2	62	0	33.703	233.28	7087.979325

PODCZAS WYKONYWANIA SKRYPTU



	Ilość wywołań	Avg [ms]	Min [ms]	Max [ms]	% błędów	stron / s	KB/s	średni rozmiar [B]
Main page	10	5	2	18	0	0.3702	1.0557	2919.8
Log in	10	30	26	40	0	0.3684	3.969	11031.6
User info	10	20	15	33	0	0.366	2.8808	8060.3
Get register payment	250	19	11	81	0	5.0262	41.28	8410.016
First page	248	41	23	317	0.004032	4.9715	41.504	8548.596774
Add item	245	48	22	250	0	4.9193	44.481	9259.2
Go to third page	244	40	21	255	0	4.9277	50.827	10562.04098
Add payer	244	61	29	284	0	4.9038	52.602	10984.22541
Go to confirmation page	242	25	14	190	0	4.8901	45.576	9543.747934
Go to summary	242	71	18	509	0.247934	4.8687	54.541	11471.19835
Total	1745	43	2	509	0.034957	34.818	332.3	9773.113467

WYPEŁNIONA BAZA



	Ilość wywołań	Avg [ms]	Min [ms]	Max [ms]	% błędów	stron / s	KB/s	średni rozmiar [B]
Main page	617	4	2	59	0	12.421	33.853	2790.896272
Log in	615	39	23	80	0	12.42	134.65	11101.08943
User info	609	33	18	77	0	12.332	97.106	8063.211823
Total	1841	26	2	80	0	36.952	263.83	7311.053232

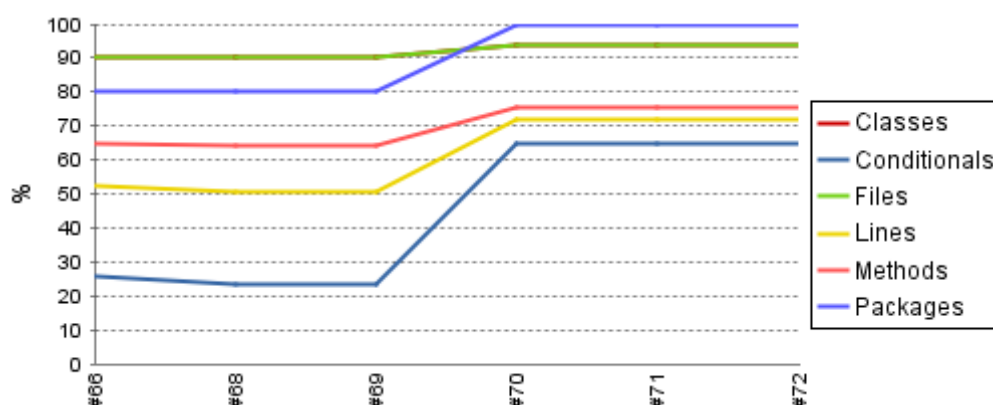
ZAKOŃCZENIE IMPLEMENTACJI – STATYSTYKI

Ponieważ IV faza kończy projekt cyberweji, poniżej załączam statystyki w momencie zakończenia implementacji.

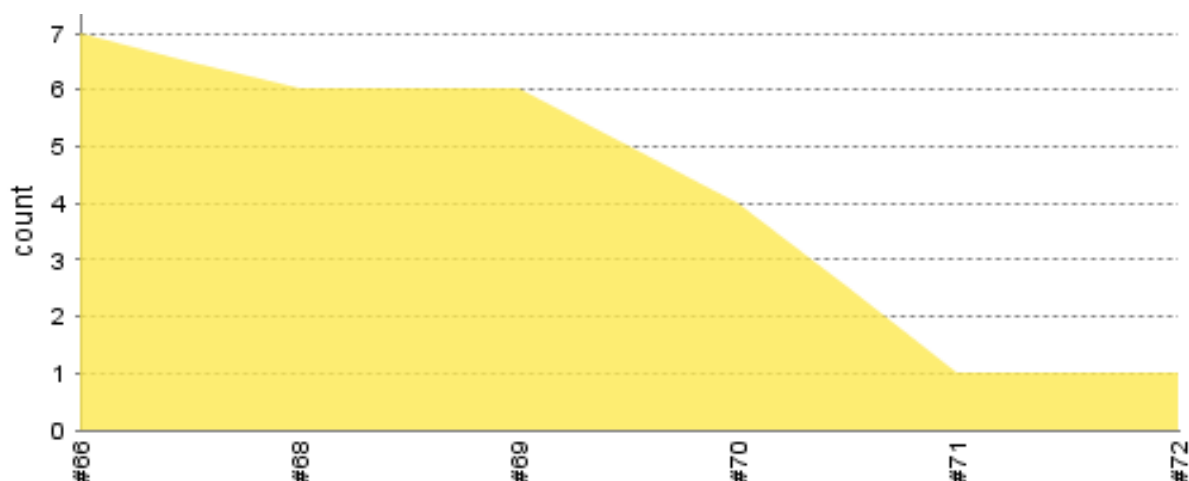
CONTINUOUS INTEGRATION

Code Coverage

Classes 94% **Conditionals** 65% **Files** 94% **Lines** 72% **Methods** 75%
Packages 100%

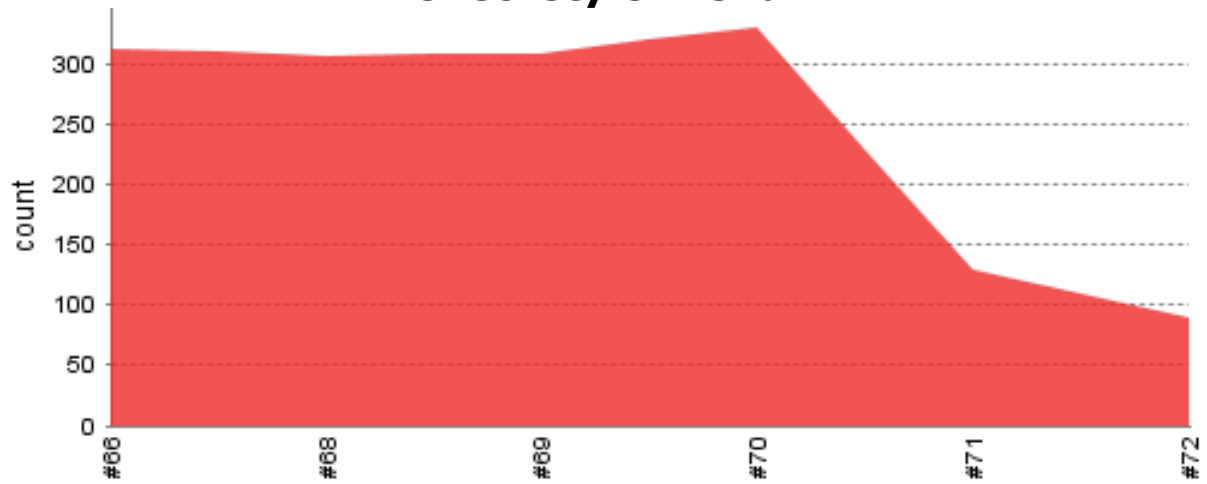


PMD Trend



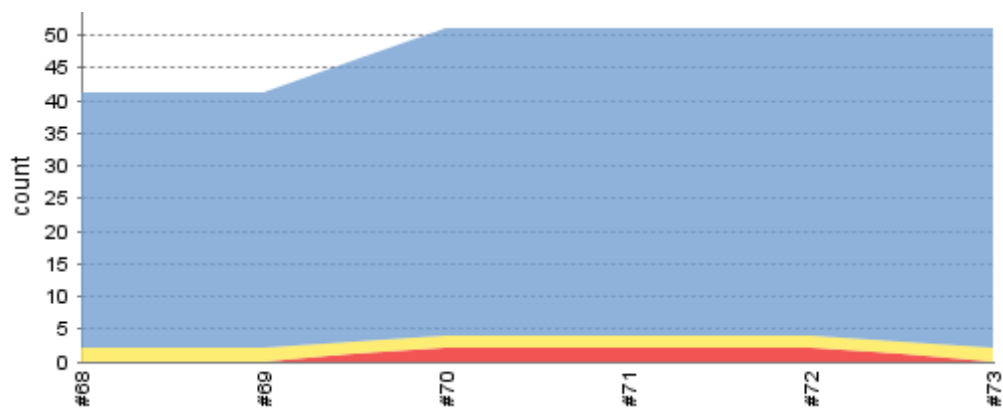
Zakończony na etapie 1 warningu.

Checkstyle Trend



Zakończony na 89 warningach

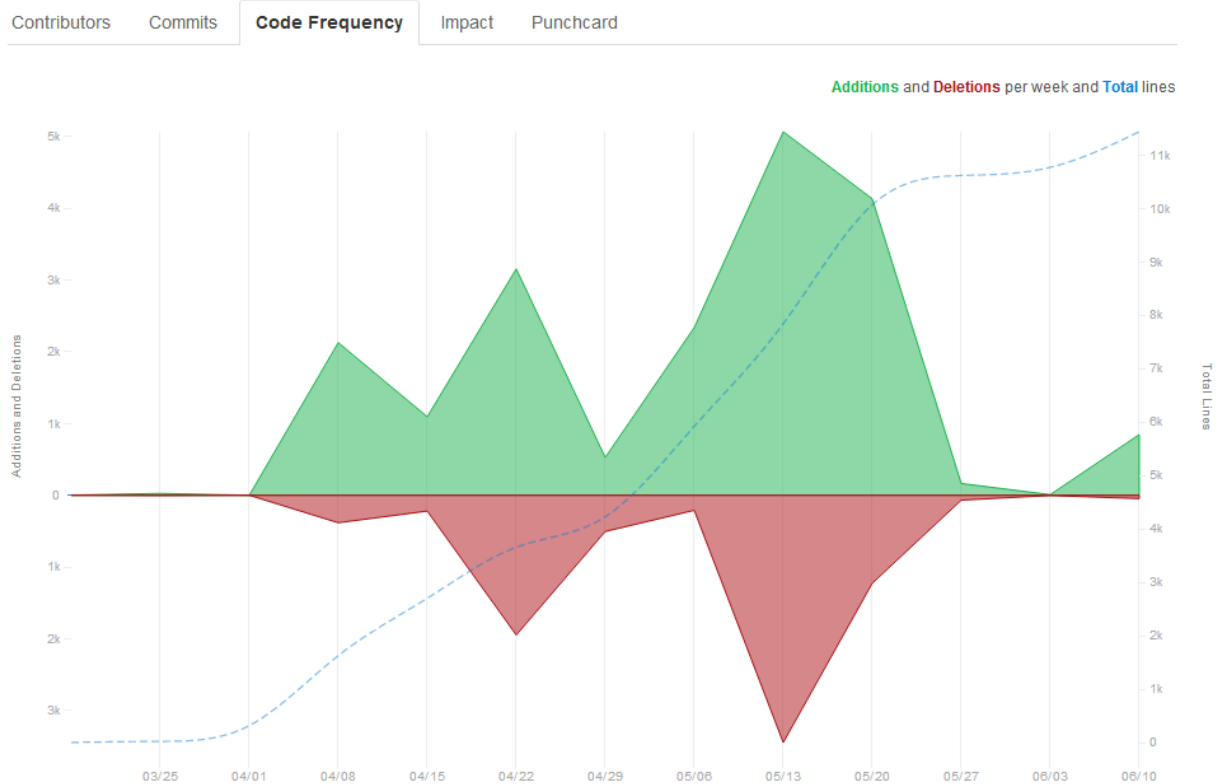
Test Results



Zaimplementowanych zostało 51 testów.

GITHUB

Wykres wszystkich, dodanych i usuniętych linii kodu.



W sumie ponad 11k linii.

Historia commitów



Zdecydowanie najwięcej przed spotkaniami z prowadzącym, ale trochę systematycznej pracy także się pojawiło.