

ID	Oceniany element	4 osoby (nie zmieniać)
1	Kod jest pod kontrolą systemu zarządzania wersjami (Github) i prowadzący ma do niego dostęp	10
2	Kod określający inkrement produktu jest oznaczony w repozytorium (tag = numer wersji) oraz odpowiednia wersja w pom.xml	10
3	Aplikacja buduje się w sposób automatyczny (kompilacja, stworzenie paczki do dystrybucji)	5
4,1	Skonfigurowano środowisko ciągłej integracji i prowadzący ma do niego dostęp (serwer CI reaguje na zmiany w repozytorium kodu i uruchamia proces budowania)	5
5,1	Dokumentacja generuje się w sposób automatyczny podczas budowy oprogramowania	5
5,2	Przynajmniej 1 klasa / interfejs posiada pełną dokumentację techniczną* (javadoc, itp.)	10
5,3	Przynajmniej 5 klas / interfejsów posiada pełną dokumentację techniczną* (javadoc, itp.)	10
5,4	Przynajmniej 7 klas / interfejsów posiada pełną dokumentację techniczną* (javadoc, itp.)	5
6	Skonfigurowano środowisko do zarządzania rejestrem produktu i rejestrem sprintu, prowadzący ma do niego dostęp (np. Github Issues, Github Projects, Trello)	10
7,1	W programie użyto biblioteki SL4J do logowania wykonania na poziomie DEBUG i INFO	0
8,1	Zaprojektowano model klas (dla własnych klas, jeśli więcej niż 10 to ograniczyć do najważniejszych klas) z wykorzystaniem diagramu klas UML i umieszczono go w repozytorium lub na wiki projektu	10
8,2	Opracowano diagram sekwencji obrazujący sekwencje przetwarzania wybranej operacji począwszy od otrzymania żądania REST do zwrócenia wartości w kontrolerze REST	0

9,1	Podczas implementacji wykorzystano sugerowany wzorzec projektowy w proponowanym zakresie: TextTransformer - Dekorator wykorzystany do dynamicznego tworzenia obiektu transformującego; Scenario Quality Checker - Wizytator będzie użyty do analizy scenariuszy i pod-scenariuszy; Building Info - Kompozyt do reprezentacji struktury pomieszczeń oraz wywoływania operacji mających bezpośredni związek ze strukturą danych (obliczenia powierzchni, kubatury, zużycia energii, mocy oświetlenia). Wizytator do funkcji raportujących, Bootstrap Builder - budowniczy do podzielenia budowy strony na fragmenty (np. nagłówke, treść, stopka), JSON tools - wzorzec dekorator do dynamicznego tworzenia wyjścia na bazie poszczególnych opcji (np. filtrowanie, minifikacja)	10
10,1	<b>Stworzono min. 1 klasę testującą i 5 metod testujących**</b>	40
10,2	Stworzono min. 2 klasy testujące i 10 metod testujących**	30
10,3	Stworzono min. 2 klasy testujące i 20 metod testujących**	20
10,4	Stworzono min. 3 klasy testujące i 25 metod testujących**	
10,5	<b>Przetestowano przynajmniej 1 wywołań metod z zewnętrznych klas za pomocą mock obiektów</b>	40
10,6	Przetestowano przynajmniej 5 wywołań metod z zewnętrznych klas za pomocą mock obiektów	30
10,7	Przetestowano przynajmniej 10 wywołań metod z zewnętrznych klas za pomocą mock obiektów	20
10,8	Przetestowano przynajmniej 15 wywołań metod z zewnętrznych klas za pomocą mock obiektów	
1,1	Wprowadzono rejestr produktu do narzędzia zarządzania (np. Trello)	10
1,2	Uszeregowano rejestr produktu według ważności	0
2,1	Przeniesiono do rejestru sprintu elementy rej. produktu, które mają być zrealizowane w sprincie	10
2,2	Dodano zadania do poszczególnych elementów rej. produktu realizowanych w sprincie	10
3,1	<b>Akcje naprawcze ze retrospektywy poprzedniego sprintu zostały spisane</b>	20
3,2	Wdrożenie akcji naprawczych było monitorowane (np. pojawiły się zadania w sprint backlogu)	10
1 Inkrement zawiera implementację elementów rejestru produktu o wartości 4 BV		40

2	Inkrement zawiera implementację elementów rejestru produktu o wartości 7 BV	30
3	Inkrement zawiera implementację elementów rejestru produktu o wartości 9 BV	

5 osoby (nie zmieniać)	Punkty zdobyte przez zespół (wpisać tutaj)	Informacje dodatkowe - należy zamienić tekst objaśnienia na wymagane informacje zgodnie z opisem
10	10	<a href="https://github.com/Danieluss/IOIO">https://github.com/Danieluss/IOIO</a>
10	10	Done. Patrz: <a href="https://github.com/Danieluss/IOIO/blob/master/pom.xml">https://github.com/Danieluss/IOIO/blob/master/pom.xml</a>
5	5	W tym kontekście korzystając z buildów travisa zapewniamy, że: - nasze gałęzi w repozytorium githuba kompilują się automatycznie w celu: - stworzenia paczki roboczej (niewidoczne dla niedeweloperów) - odpalenia testów (pull requestów nie można zmergować do mastera, jeżeli nie są przejrzane i zaakceptowane przez kogoś innego lub nie przechodzą wyżej wspomniane testy) - paczka dystrybucyjna tworzy się tylko dla tagów (github release <a href="https://github.com/Danieluss/IOIO/releases">https://github.com/Danieluss/IOIO/releases</a> )
5	5	<a href="https://travis-ci.org/Danieluss/IOIO?utm_medium=notification&amp;utm_source=">https://travis-ci.org/Danieluss/IOIO?utm_medium=notification&amp;utm_source=</a> środowisko na wszystko pięknie reaguje, jak @wyżej
5	5	poza tym jest dodatkowo osadzana na github.pages strony naszego projektu, o tu <a href="https://danieluss.github.io/IOIO/docs/apidocs/">https://danieluss.github.io/IOIO/docs/apidocs/</a>
10	10	com.ioio.jsontools.core.service.whitespace.JsonMinifier
10	10	com.ioio.jsontools.core.service.whitespace.JsonMaxifier com.ioio.jsontools.core.service.filter.JsonFilterModifier com.ioio.jsontools.core.service.filter.JsonFilter com.ioio.jsontools.core.service.JsonModifierDecorator
5	5	com.ioio.jsontools.core.service.JsonModifierImpl com.ioio.jsontools.core.service.JsonModifier
10	10	Rejestr produktu znajduje się tutaj: <a href="https://github.com/Danieluss/IOIO/issues">https://github.com/Danieluss/IOIO/issues</a> Rejestr sprintu i aktualny stan zadań znajduje się tutaj: <a href="https://github.com/Danieluss/IOIO/projects/3">https://github.com/Danieluss/IOIO/projects/3</a>
0	0	Wraz z specjalnymi aspektami do logowania wywołań metod na różnych poziomach za pomocą samych anotacji.
10	10	Nasz wspaniały diagram znajduje się tutaj: <a href="https://github.com/Danieluss/IOIO/blob/master/core-uml.png">https://github.com/Danieluss/IOIO/blob/master/core-uml.png</a>
5	0	-- 4 osobowy zespół

10	10	Nasz dekorator najlepiej widać na diagramie: <a href="https://github.com/Danieluss/IOIO/blob/master/core-uml.png">https://github.com/Danieluss/IOIO/blob/master/core-uml.png</a> Opis endpointa, który obsługuje zapytania tworzące dynamicznie łańcuch dekoratorów znajduje się w readme wraz z przykładem użycia: <a href="https://github.com/Danieluss/IOIO/blob/master/readme.md">https://github.com/Danieluss/IOIO/blob/master/readme.md</a>
40	40	com.ioio.jsontools.core.service.CoreControllerTest com.ioio.jsontools.core.service.JsonFilterDataConfigStrategyServiceTest i jeszcze 3 inne klasy, łącznie 40 metod testujących
30	30	
20	20	
30	0	-- 4 osobowy zespół
40	40	com.ioio.jsontools.core.service.CoreControllerTest com.ioio.jsontools.core.service.JsonMinifierTest, 10 wywołań metod z zewnętrznych klas
30	30	
20	20	
30	0	-- 4 osobowy zespół
10	10	Rejestr produktu znajduje się tutaj: <a href="https://github.com/Danieluss/IOIO/issues">https://github.com/Danieluss/IOIO/issues</a>
0	0	
10	10	widoczne w tym miejscu: <a href="https://github.com/Danieluss/IOIO/projects/3">https://github.com/Danieluss/IOIO/projects/3</a>
10	10	<a href="https://github.com/Danieluss/IOIO/projects/3">https://github.com/Danieluss/IOIO/projects/3</a>
20	20	<a href="https://github.com/Danieluss/IOIO/blob/master/trudne_sprawy.md">https://github.com/Danieluss/IOIO/blob/master/trudne_sprawy.md</a>
10	10	@up oraz tablica projektu.
40	40	<ul style="list-style-type: none"> <li>- Jako twórca zewnętrznego oprogramowania, mogę korzystać z dostępnych funkcji zdalnie poprzez REST, aby móc zintegrować narzędzie z innymi aplikacjami.</li> <li>- Jako programista mogę uzyskać zminifikowaną strukturę w formacie JSON na podstawie pełnego zapisu w formacie JSON, aby zmniejszyć rozmiar danych.</li> <li>- Jako programista mogę uzyskać pełną strukturę w formacie JSON ze zminifikowanego zapisu w formacie JSON, aby polepszyć czytelność danych.</li> <li>- Jako programista mogę uzyskać strukturę w formacie JSON zawierającą tylko określone własności, aby uprościć strukturę</li> <li>- Jako programista mogę uzyskać strukturę w formacie JSON zawierającą tylko określone własności, aby uprościć strukturę</li> </ul>

30	30	<ul style="list-style-type: none"> <li>- Jako programista mogę uzyskać strukturę w formacie JSON nie zawierającą określonych własności, aby uprościć strukturę</li> <li>- Jako programista mogę porównać dwa teksty (np. reprezentujące strukturę danych w formatach JSON) oczekując, że aplikacja pokaże linijki, w których występuje różnica.</li> <li>- Jako użytkownik mogę skorzystać z aplikacji za pomocą interfejsu użytkownika, niestety co do tego zadania nie zapytaliśmy o kryteria akceptacji</li> <li>- Moduł interfejsu użytkownika</li> </ul>
35	0	<Lista ukończonych elementów Rejestru Produktu>