

# Analiza rozwiązania OpenStack<sup>1</sup>/Murano<sup>2</sup>

---

<sup>1</sup> <https://www.openstack.org/>

<sup>2</sup> <https://wiki.openstack.org/wiki/Murano>

<b>Streszczenie</b>	<b>3</b>
Cel	3
Sposób realizacji	3
Rezultat	3
<b>Analiza funkcji</b>	<b>3</b>
Opis OpenStack/Murano	3
Lista wymaganych funkcji	4
Przebieg eksperymentu sprawdzającego funkcje	4
Rezultat eksperymentu	5
<b>Integracja z innymi pomysłami</b>	<b>5</b>
Uruchamianie w innych środowiskach	5
Wirtualizacja za pomocą Docker'a	5
<b>Analiza jakości</b>	<b>6</b>
Zakres analizy	6
Pobieżna analiza jakości	7

# Streszczenie

## Cel

Weryfikacja, czy system OpenStack/Murano można wykorzystać jako marketplace aplikacji.gov.pl biorąc pod uwagę:

- jego funkcje;
- spójność z innymi rozwiązaniami zaplanowanymi w aplikacji.gov.pl;
- jego jakość.

## Sposób realizacji

Analiza funkcji polegała na:

- opracowaniu listy funkcji, które musi spełniać docelowy marketplace;
- ocenie systemu OpenStack/Murano zgodnie z wybranymi i opisanymi na liście kryteriami.

Analiza jakości polegała na:

- określeniu, które części systemu OpenStack'a zostaną przeanalizowane;
- zgromadzeniu listy kryteriów, które musi spełniać system, aby być dobrej jakości;
- ocenie systemu;
- podsumowaniu oceny poszczególnych części i wystawieniu zbiorczej oceny.

## Rezultat

Mimo, że system OS/Murano ma wymagane od marketplace funkcje, nie rekomendujemy wykorzystania systemu OpenStack/Murano jako marketplace aplikacji.gov.pl.

Analizowany system jest rozwiązaniem kompleksowym - świadczy to na jego niekorzyść w kontekście wykorzystania w projekcie aplikacji.gov.pl. Ta jego cecha w praktyce oznacza, że trudno będzie użyć jego poszczególnych części osobno. Utrudni to m.in. uruchomienie platformy w niestandardowych konfiguracjach (np. podczas testów lub w przypadku, gdy instytucja będzie miała niestandardowe potrzeby).

Szczegółowa analiza jakości systemu OS/ Murano nie została przeprowadzona, ponieważ poprzedzające ją kroki pokazały, że analizowany system nie nadaje się do wykorzystania. Podprojekty projektu OpenStack wyglądają na dobrze napisane.

## Analiza funkcji

### Opis OpenStack/Murano

OpenStack to system zarządzający zasobami w systemie komputerowym (np. mocą obliczeniową, pamięcią, dyskami, sieciami). Na wejściu daje mu się „pod opiekę” działające maszyny (*compute nodes*). Następnie zleca mu się uruchomienie określonych instancji

serwerów. Wtedy uruchamiane są na *compute nodes* z użyciem technik wirtualizacji odpowiednie instancje. OpenStack współpracuje z wieloma mechanizmami dzielenia zasobów pomiędzy instancje oraz izolacji pomiędzy instancjami. Np. może uruchamiać maszyny wirtualne za pomocą różnych narzędzi (qemu, VMware, ...). Za pomocą komponentu nova-docker może także uruchamiać instancje używając systemu kontenerów docker, jednak rozwój tego komponentu został wstrzymany. W rezultacie komponent ten nie powinien być używany w nowych systemach. Istnieją też inne możliwości, niektóre wymienione w README repozytorium <https://github.com/openstack/nova>.

OpenStack posiada funkcję tworzenia *projektów* - części odizolowanych pomiędzy sobą, wirtualnych serwerowni. System uprawnień umożliwia stworzenie użytkownika i nadanie mu uprawnień do zarządzania określonymi *projektami*. Istnieje możliwość nałożenia limitów na poszczególne zasoby używane konkretnym *projekcie* (np „co najwyżej 14 rdzeni, 123GB RAMu, 3 sieci lokalne”).

Komponent Murano dobudowuje nad tym interfejs użytkownika, dzięki któremu wewnątrz *projektu* można stworzyć *środowisko* składające się z aplikacji uruchomionych na maszynach wirtualnych. Komponowanie środowiska polega na „wyklikaniu” - użytkownik wybiera aplikacje z listy gotowych wzorców aplikacji, wybiera parametry maszyny z listy dostępnych konfiguracji, wybiera obraz systemu operacyjnego także z listy. Do skonfigurowania *środowiska* potrzeba znajomości zależności w systemach informatycznych - trzeba np wiedzieć, ile RAMu i rdzeni potrzeba dla instancji, na której będzie uruchomiona aplikacja X, trzeba wiedzieć czy przydzielić tej aplikacji publiczne IP.

## Lista wymaganych funkcji

Na wcześniejszym etapie prac badawczo-rozwojowych opracowano listę funkcji, które musi spełniać marketplace aplikacji gov.pl:

[https://docs.google.com/document/d/1wS\\_gOiy7rVnC0oPm-ojxKFz37bA92\\_Gyo9oe-0QS64k](https://docs.google.com/document/d/1wS_gOiy7rVnC0oPm-ojxKFz37bA92_Gyo9oe-0QS64k)

## Przebieg eksperymentu sprawdzającego funkcje

Instancję OpenStacka uruchomiono z użyciem devstack<sup>3</sup>. Nie jest to instalacja produkcyjna. Na instancję składają się standardowo instalowane komponenty :

- keystone
- glance
- nova
- cinder
- neutron
- horizon

oraz komponenty zainstalowane dodatkowo:

- murano
- heat

Po zainstalowaniu została przeprowadzona próba polegająca na:

---

<sup>3</sup> <https://docs.openstack.org/developer/devstack/>

Z konta administratora systemu

- stworzeniu nowego *projektu* z określonymi limitami zasobów
- stworzeniu użytkownika, który ma prawa zarządzać tylko ww projektem

Z konta nowego użytkownika

- z wykorzystaniem interfejsu graficznego Murano stworzeniu i skonfigurowaniu środowiska składającego się z jednego serwera HTTP
- uruchomieniu środowiska

## Rezultat eksperymentu

Testowa instancja miała wszystkie wymagane funkcje.

## Integracja z innymi pomysłami

### Uruchamianie w innych środowiskach

Biorąc pod uwagę założenia platformy aplikacje.gov.pl można przyjąć z dużym prawdopodobieństwem, że docelowo w systemie będzie istniała potrzeba uruchomienia platformy, lub jej części poza chmurą. Na przykład deweloper aplikacji umieszczanej na platformie lub deweloper samej platformy powinien mieć możliwość testowego uruchomienia napisanego przez siebie kodu. Aby mógł to zrobić musi uruchomić platformę lokalnie, na komputerze, nad którym ma pełną kontrolę. Taka potrzeba może zaistnieć również w sytuacji, w której firma prywatna X chciałaby korzystać z systemu EZD/EOD dostępnego w ramach aplikacje.gov.pl, ale nie ma ona możliwości kupienia/otrzymania dostępu do chmury państwowej. Dobrze by było gdyby firma X mogła uruchomić produkcyjną instancję platformy wraz z EZD/EOD na swoich serwerach. W miejsce przykładowej firmy prywatnej X można podstawić również urząd, który zajmuje się bardzo wrażliwymi danymi. Taki urząd powinien mieć możliwość uruchomienia instancji platformy na serwerach odizolowanych fizycznie od internetu.

OpenStack/Murano nie zapewnia łatwego uruchamiania w różnych środowiskach. Nie pozwala też na uruchomienie części projektu - np jednej instancji, bez interfejsu do dodawania instancji. Jest przeznaczony do użycia jako całość.

Problemem przy małych instalacjach OpenStack/Murano (np. instalacja deweloperska) jest duży narzut na zużycie pamięci RAM. Podczas testów uruchomiony OpenStack bez żadnych aplikacji działających w jego ramach zużywał około 5GB pamięci RAM.

### Wirtualizacja za pomocą Docker'a

Z powodu dużego narzutu na użycie RAMu, w przypadku gdy dużo małych serwisów byłoby uruchamiane każdy w swojej maszynie wirtualnej zdecydowano, że należy zweryfikować, czy w chmurze koordynowanej przez OpenStack jest możliwość uruchomienia instancji kontenerów dockera.

Komponent nova-docker pozwala na uruchamianie instancji kontenerów dockera jako maszyn wirtualnych. To rozwiązanie wygląda obiecująco, jednak rozwój projektu został wstrzymany, więc nie powinien być używany.

Inną możliwością jest komponent Magnum<sup>4</sup>. Na żądanie administratora zleca on uruchomienie maszyn wirtualnych, które razem tworzą klastr uruchamiający kontenery. Klastr może być kontrolowany przez jedną z trzech technologii: Docker Swarm, Kubernetes, Apache Mesos. Wszystkie trzy mogą używać docker-engine do uruchamiania kontenerów. Administrator podczas tworzenia klastra wybiera liczbę maszyn i ich parametry.

Uruchamianie OpenStacka na maszynie wirtualnej z komponentem Magnum, za pomocą devstacka jest niewspierane. Testowa instancja OpenStacka została uruchomiona na maszynie wirtualnej. Z tego powodu testy Magnum nie zostały przeprowadzone (nie był to też cel opisywanej części badań).

Klastry stworzone przez Magnum nie podlegają zarządzaniu przez Murano. Murano uruchamia „wyklikane” komponenty jako instancje maszyn wirtualnych. Murano nie jest przeznaczone do zarządzania kontenerami działającymi wewnątrz klastra np Docker Swarm.

Wirtualizacja z użyciem docker-engine wygląda obiecująco, wyklucza jednak wykorzystanie Murano. Wykorzystanie OpenStacka wymagane było tylko po to, aby można było użyć komponentu Murano. Jeśli zostanie podjęta decyzja o użyciu docker-engine argumenty za użyciem OpenStack/Murano przedstawione w tym dokumencie nie będą aktualne.

## Analiza jakości

Wnikliwa analiza jakości systemu nie została wykonana ponieważ:

- OpenStack ma cechy, które uniemożliwiają jego wyboru niezależnie od wyniku analizy jakości;
- wnikliwa analiza jest pracochłonna.

## Zakres analizy

Części OpenStack’a, które są niezbędne, aby uruchomić projekt Murano muszą zostać poddane analizie. Analizie zostaną poddane następujące części:

- keystone (<https://github.com/openstack/keystone>)
- glance (<https://github.com/openstack/glance>)
- nova (<https://github.com/openstack/nova>)
- cinder (<https://github.com/openstack/cinder>)
- neutron (<https://github.com/openstack/neutron>)
- horizon (<https://github.com/openstack/horizon>)
- heat (<https://github.com/openstack/heat>)
- murano
  - murano (<https://github.com/openstack/murano>)
  - murano-agent (<https://github.com/openstack/murano-agent>)

---

<sup>4</sup> <https://wiki.openstack.org/wiki/Magnum>

## Pobieżna analiza jakości

Komponenty OpenStacka są tworzone zgodnie z wykorzystaniem dobrych praktyk, takich jak: stosowanie polityki zależności programistycznych, używanie oprogramowania Gerrit (systemu do review kodu), stosowanie polityki wycofywania funkcji, wydzielanie wspólnego kodu do bibliotek (m.in. oslo), dzielenie projektu na wiele podprojektów, stosowanie określonego codestyle. Są to przesłanki świadczące o dobrej jakości produktu.

Projekt nie jest porzucony - tzn. system jest rozwijany i poprawiany. Repozytoria podprojektów OpenStack cechują się dużą aktywnością.

OpenStack ma instrukcję instalacji. Istnieje skrypt do automatycznego instalowania deweloperskiej wersji systemu (devstack) oraz instrukcja jego użycia. Oprócz tego istnieje obszerna instrukcja instalacji produkcyjnej.

Komponent Murano posiada instrukcję tworzenia aplikacji zdolnych do uruchomienia za pomocą Murano.