14.12.2021r.

System do zdalnej pracy w środowisku graficznym wykorzystujący maszyny wirtualne QEMU z akceleracja sprzętową

Sprawozdanie z testów

Autorzy: Krzysztof Smogór, Piotr Widomski
Promotor: Dr inż. Marek Kozłowski

Spis treści

1 Testy jednostkowe

2 Testy integracyjne

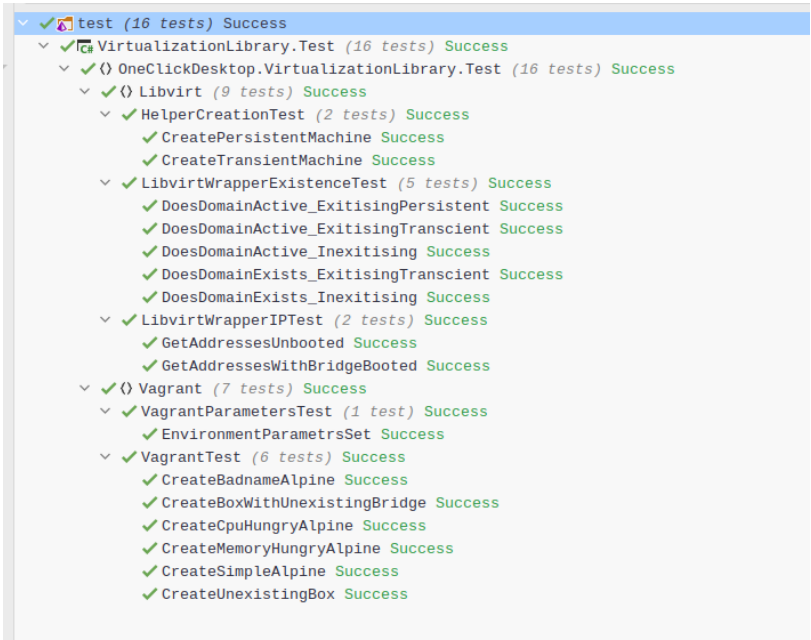
2.1 Testy integracji z libvirtem oraz vagrantem

Przy integracji systemu z libvirtem oraz vagrantem musielismy sprawdzic następujące funkcjonalności:

1. Włączanie maszyn poprzez vagranta
2. Wyłączanie maszyn poprzez vagranta
3. Sprawdzanie czy maszyna jest uruchomiona poprzez libvirta
4. Pobranie adresu IP uruchomionej maszyny wirtualnej

Aby testy miały sens potrzebowaliśmy konfiguracji XML do tworzenia maszyn *transient* oraz gotowego lekkiego vagrant-boxa przy tworzeniu maszyn *persistence*. Przy testach libvirta wykorzystaliśmy obraz live ArchLinuxa, którego uruchamiamy w minimalnie przygotowanej konfiguracji. Do testów vagranta skorzystaliśmy z lekkiego obrazu "generic/alpine38". Przy testach konfiguracji sieciowej utworzyliśmy bridge sieciowy, który jest uwzględniony w konfiguracji uruchamianych maszyn wirtualnych.

Aby upewnić się, że wszystko działa prawidłowo skorzystaliśmy z mechaniki testów jednostkowych NUnit przy jednocześnie uruchomionym daemonie libvirta.



Rysunek 1: Lista testów sprawdzających integrację z libvirtem i vagrantem

2.2 Testy integracyjne z RabbitMQ

Podczas tworzenia projektu wydzieliliśmy osobny moduł, którego celem jest obłożenie biblioteki API RabbitMQ w interfejs, który umożliwi wygodne użytkowanie brokera

z poziomu głównych modułów systemu. Jako iż biblioteka ta nie posiada wewnętrznej logiki, a jedynie wywołuje odpowiednie funkcje brokera wiadomości, przetestowana została z użyciem testów integracyjnych. Testowana funkcjonalność obejmowała:

1. Tworzenie punktów wymiany (exchange)
2. Tworzenie kolejek i podłączanie ich do punktów wymiany
3. Wysyłanie i odbieranie wiadomości (wraz z zaimplementowanym mechanizmem deserializacji)
4. Wysyłanie wiadomości do konkretnych odbiorców
5. Wykrywanie braku odbiorców

W tym celu wykorzystaliśmy metodę analogiczną do testów z libvirtem, czyli mechaniki testów jednostkowych NUnit przy jednocześnie uruchomionym brokerze RabbitMQ.

3 Testy E2E

Aplikacja kliencka oraz panel administratora posiadają proste testy E2E, z wykorzystaniem platformy Cypress, spełniające równocześnie po części rolę testów UI. Testy te skupiają się na pojedynczych ekranach aplikacji oraz jej działaniu z perspektywy użytkownika.

Planujemy dodanie testów E2E realizujących wielokrokowe scenariusze testowe, zaczynające się od zalogowania do aplikacji.

4 Scenariusze akceptacyjne

5 Aktualny stan testów

Na ten moment następujące elementy systemu posiadają testy automatyczne (wraz z typem testów):

- Aplikacja kliencka (testy jednostkowe, UI, proste E2E)
- Panel administratora (testy jednostkowe, UI, proste E2E)
- Moduł serwera wirtualizacji (testy integracyjne libvirt)
- Biblioteka modelu systemu (testy jednostkowe)
- Biblioteka brokera wiadomości (testy integracyjne)

Testy jednostkowe do modułów:

- Nadzorca
- Serwer wirtualizacji

są w trakcie powstawania. Powstaną również testy E2E, które również realizować część scenariuszy akceptacyjnych.