15.12.2021r.

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

Sprawozdanie z testów

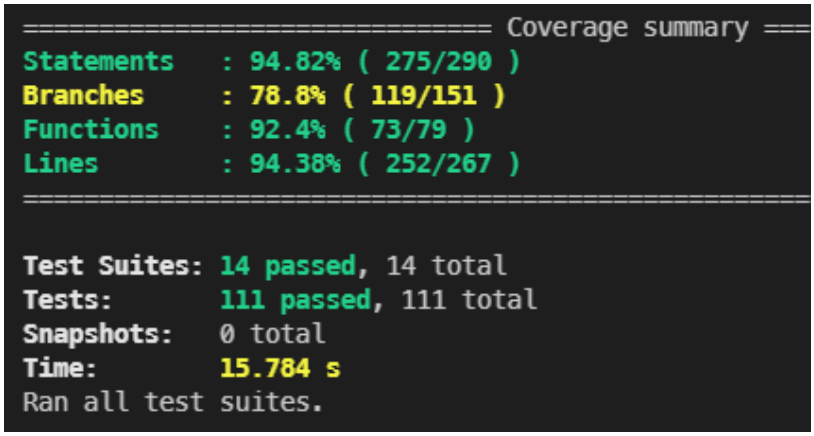
Autorzy: Krzysztof Smogór, Piotr Widomski
Promotor: Dr inż. Marek Kozłowski

Spis treści

1 Testy jednostkowe

1.1 Testy jednostkowe frontendu

Do testowania aplikacji klienckiej oraz panelu administratora wykorzystaliśmy testy jednostkowe. Testowane były zarówno serwisy, jak i komponenty odpowiadające za widoki. Dzięki testowaniu komponentów w zakresie DOMa uzyskaliśmy testowanie UI, które dopełnione zostało testami E2E.



Rysunek 1: Podsumowanie testów jednostkowych aplikacji klienckiej

1.2 Biblioteka modelu systemu

Klasy implementujące model systemu wydzielone zostały do osobnej biblioteki, która importowana jest do modułów jako pakiet nuget. Biblioteka ta zawiera proste klasy, służące do przechowywania informacji o stanie systemu. Do przetestowania jej użyliśmy testów jednostkowych, które badają poprawność działania funkcji udostępnianych przez klasy.

1.3 Moduł nadzorcy oraz serwera wirtualizacji

Moduły te są głównymi elementami systemu. Testowanie ich zostanie przeprowadzone za pomocą testów jednostkowych. Testowane będą wszystkie serwisy, z zastosowaniem mocków.

2 Testy integracyjne

2.1 Testy integracji z libvirtem oraz vagrantem

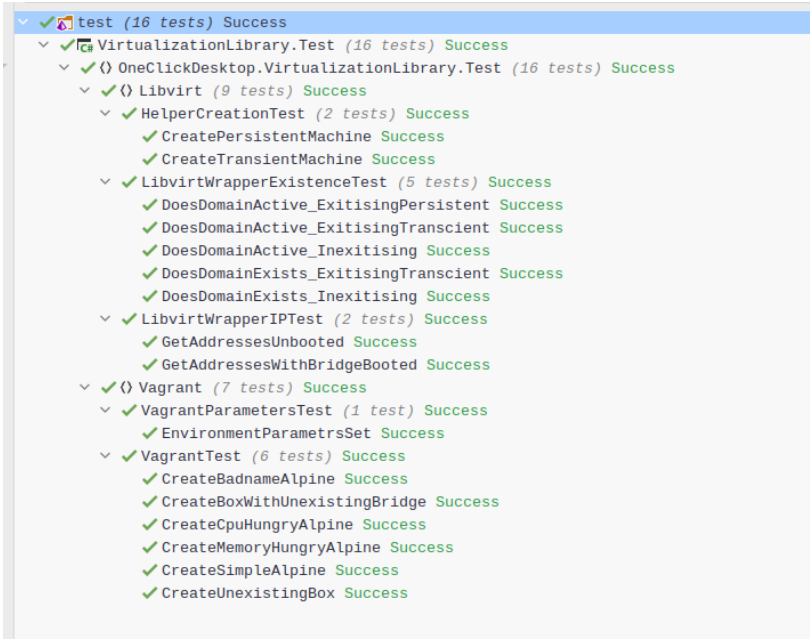
Przy integracji systemu z libvirtem oraz vagrantem musielismy sprawdzic następujące funkcjonalności:

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 2: 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

Planujemy wykonać następujące testy akceptacyjne:

4.1 Serwer wirtualizacji wyłącza się przy braku komunikacji z jakimkolwiek nadzorcą

Przy starcie samodzielnego serwera wirtualizacji i wykryciu braku komunikacji z jakimkolwiek nadzorcą (poprzez brokera wiadomości) serwer powinien się wyłączyć z błędem.

Kroki:

1. Włącz serwer wirtualizacji.
2. Wyświetl błąd i zakończ działanie.

4.2 Serwer wirtualizacji wyłącza się przy utracie komunikacji z nadzorcami

Po poprawnym starcie systemu z 1 nadzorca oraz 1 serwerem wirtualizacji serwer powinien wyłączyć się po wyłączeniu się ostatniego nadzorcy.

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.
3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na prawidłowy start systemu.
5. Wyłącz nadzorcę lub brokery wiadomości.
6. Poczekaj na wykrycie braku nadzorców (brak otrzymanych wiadomości)
7. Wyświetl błąd i zakończ działanie.

4.3 Typowe użycie systemu przez użytkownika

Użytkownik podłącza się do systemu składającego się z 1 nadzorcy oraz 1 serwera wirtualizacji, gdzie działa przynajmniej jedna wolna maszyna. Powinien prawidłowo otrzymać sesje oraz po odłączeniu się maszyna powinna zostać wyłączona po 15 minutach(czas konfigurowalny)

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.
3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na uruchomienie przynajmniej jednej maszyny wirtualnej.
5. Poproś o sesję poprzez aplikację kliencką.
6. Podłącz się poprzez RDP do uzyskanej maszyny.
7. Zakończ sesję z poziomu aplikacji.
8. Po określonym czasie maszyna powinna się wyłączyć.

4.4 Typowe użycie systemu przez użytkownika przy awarii nadzorcy

Użytkownik podłącza się do systemu składającego się z 2 nadzorców oraz 1 serwera wirtualizacji, gdzie działa przynajmniej jedna wolna maszyna. Powinien prawidłowo otrzymać sesje oraz po odłączeniu się i awarii nadzorcy, z którym rozmawiał, przy szybkim powrocie powinien uzyskać swoja poprzednia maszynę.

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz dwie aplikacje nadzorców.
3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na uruchomienie przynajmniej jednej maszyny wirtualnej.
5. Poproś o sesję poprzez aplikację kliencką.
6. Podłącz się poprzez RDP do uzyskanej maszyny.
7. Zakończ sesję z poziomu aplikacji.
8. Wyłącz tego nadzorcę, z którym klient się komunikował.
9. Poproś ponownie o sesję poprzez aplikację kliencką(powinien uzyskać tą sama maszynę)
10. Podłącz się poprzez RDP do uzyskanej maszyny.

4.5 Podłączenie nowego serwera wirtualizacji w czasie działania systemu

W trakcie działania systemu nowy serwer wirtualizacji powinien zostać włączony do modelu nadzorców.

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.

3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na prawidłowy start systemu.
5. Włącz kolejna instancje serwera wirtualizacji.
6. Poczekaj na aktualizację modelu.
7. Sprawdź w panelu administratora, że dwa serwery sa w modelu.

4.6 Podłączenie nowego nadzorcy w czasie działania systemu

W trakcie działania systemu nowy nadzorca powinien posiadać taki sam model jak aktualnie działający

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.
3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na prawidłowy start systemu.
5. Włącz kolejna instancje nadzorcy.
6. Poczekaj na aktualizację modelu.
7. Sprawdź model na pierwszym nadzorcy poprzez panel administratora.
8. Sprawdź model na drugim nadzorcy poprzez panel administratora.

4.7 Odnotowanie utraty serwera wirtualizacji

W trakcie działania systemu, przy utracie serwera wirtualizacji, nadzorcy powinni usunąć go z modelu.

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.
3. Włącz dwie aplikacje serwera wirtualizacji.
4. Poczekaj na prawidłowy start systemu.
5. Wyłącz jeden z serwerów wirtualizacji.
6. Poczekaj na odnotowanie straty.
7. Sprawdź model poprzez panel administratora.

4.8 Odnotowanie utraty serwera wirtualizacji

W trakcie działania systemu nadzorca powinien odnotować brak komunikacji z serwerami wirtualizacji poprzez wspólna kolejkę. W efekcie powinien wyczyścić model.

Kroki:

1. Włącz brokera wiadomości wewnętrznego oraz zewnętrznego.
2. Włącz aplikację nadzorcy.
3. Włącz aplikację serwera wirtualizacji.
4. Poczekaj na prawidłowy start systemu.
5. Wyłącz serwer wirtualizacji.
6. Poczekaj na odnotowanie straty.
7. Sprawdź model poprzez panel administratora.

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.