

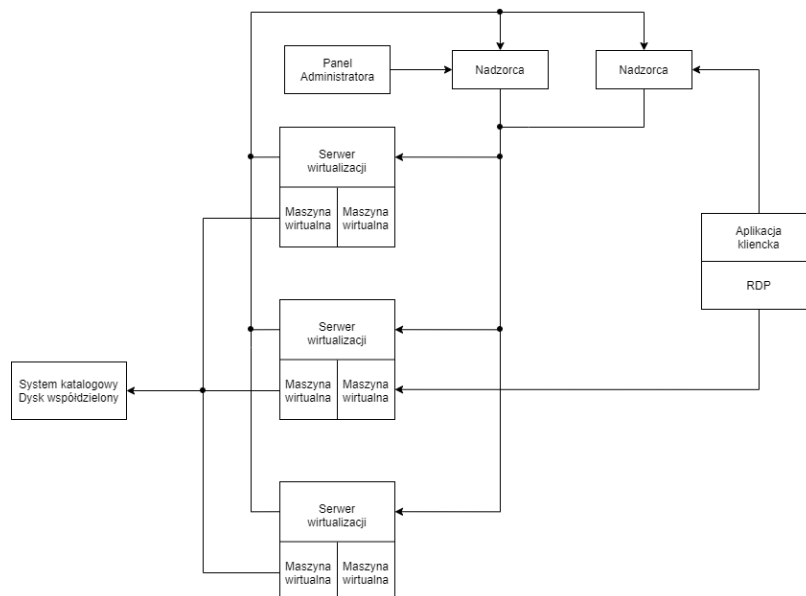
# 1 BLa bla bla - wymagane punkty jak z deliverable-one (MODULARNOSC!!!)

## 2 Architektura systemu

Przedstawiony system składa się z następujących modułów:

- nadzorcy,
- serwer wirtualizacji,
- brokera wiadomości,
- aplikacji klienckiej,
- panelu administratora,
- systemu katalogowego,
- dysku współdzielonego.

Schematyczny obraz systemu przedstawia poniższy rysunek.



Rysunek 1: Schematyczna architektura systemu

Z założenia system powinien móc skalować się w dwóch wymiarach, to znaczy:

1. Zwiększanie liczby serwerów wirtualnych - zwiększenie liczby sesji dla użytkowników.
2. Zwiększenie liczby nadzorców - zwiększenie liczby obsługiwanych klientów jednocześnie.

Szczegółowe opisy poszczególnych modułów będą omówione w następnych rozdziałach: Opis tworzonych modułów oraz Opis zewnętrznie dostarczonych modułów.

## 3 Opis tworzonych modułów

### 3.1 Nadzorca

### 3.2 Serwer wirtualizacji

### 3.3 Aplikacja kliencka

### 3.4 Panel administratora

## 4 Opis zewnętrznie dostarczonych modułów

### 4.1 Broker wiadomości

Komunikacje wewnątrz systemu, czyli pomiędzy serwerami wirtualizacji oraz nadzorcami, będzie realizowali poprzez kolejki wiadomości. Wiadomości będą przekazywane pomiędzy modułami przez brokera, którego nie będziemy implementować. Zdecydowaliśmy się na skorzystanie z systemu RabbitMQ, który zapewni niezawodność komunikacji pomiędzy modułami. Pozostaje jednak problem hazardu oraz wyścigów, które zostaną wyeliminowane w logice komunikacji nadzorcy oraz serwera wirtualizacji.

Aby system mógł funkcjonować zdefiniowane zostaną kolejki:

- (I) Kolejka kończąca się na każdym z serwerów wirtualizacji i dla każdego z nich wiadomości są powielane. Służy ona do wysyłania próśb od nadzorców do serwerów wirtualizacji.
- (II) Kolejka kończąca się na każdym z nadzorców i dla każdego z nich wiadomości są powielane. Służy ona do wysyłania informacji do nadzorców o zmianie stanu w serwerze wirtualizacji. Dodatkowo może służyć do wymiany danych pomiędzy nadzorcami.
- (III) Kolejka kończąca się wyłącznie na pojedynczym serwerze wirtualizacji. Liczba kolejek zgadza się z liczbą serwerów wirtualizacji aktywnych w systemie. Służą one do sprawdzania, czy serwer wirtualizacji nadal pracuje po drugiej stronie. Skorzystamy z funkcjonalności kolejek na wyłączność (Exclusive Queue<sup>1</sup>). Każda z nich powinna mieć nazwę jaka przedstawi się serwer wirtualizacji.

Powyższe 3 grupy kolejek umożliwią prawidłowe działanie systemu. Każdy z modułów utworzy odpowiednie kolejki w trakcie uruchamiania. Jedynym wymogiem prawidłowego uruchomienia komunikacji jest dostępny przez wszystkie serwery wirtualizacji oraz nadzorców proces brokera.

### 4.2 Dysk sieciowy

Usługa wspólnej przestrzeni dyskowej jest potrzebna do uzyskania niezależności wyboru maszyny wirtualnej od folderu użytkownika. Każda maszyna wirtualna powinna przy starcie otrzymać adres oraz dane dostępowe do takiego dysku sieciowego. Dane zostaną dostarczone poprzez wykonanie Ansible playbooka po uruchomieniu maszyny.

W przypadku maszyn wirtualnych uruchamiających system Linux potrzebne są dane do połączenia przez protokół NFS, a przy systemie Windows dane do protokołu SAMBA. Niezależnie od protokołu dane nie mogą się różnić (struktura katalogów oraz zawartość plików).

---

<sup>1</sup>Opis zachowania kolejek na wyłączność

### 4.3 System katalogowy

Usługa systemu katalogowego jest potrzebna do uzyskania niezależności wyboru maszyny wirtualnej od danych logowania do systemu uruchomionego na maszynie wirtualnej. Każda maszyna wirtualna powinna przy starcie otrzymać adres oraz dane dostępne do takiego systemu katalogowego. Dane zostaną dostarczone poprzez wykonanie Ansible playbooka po uruchomieniu maszyny.

Aby system katalogowy był kompatybilny z systemami Windows oraz Linux uruchamianymi na maszynie wirtualnej skorzystamy z protokołu OpenLDAP do uzyskiwania danych o użytkownikach.

## 5 Komunikacja

5.1 Komunikacja użytkownika z systemem - REST API

5.2 Komunikacja wewnątrz systemu - broker wiadomości

## 6 Diagramy

6.1 Diagramy stanów

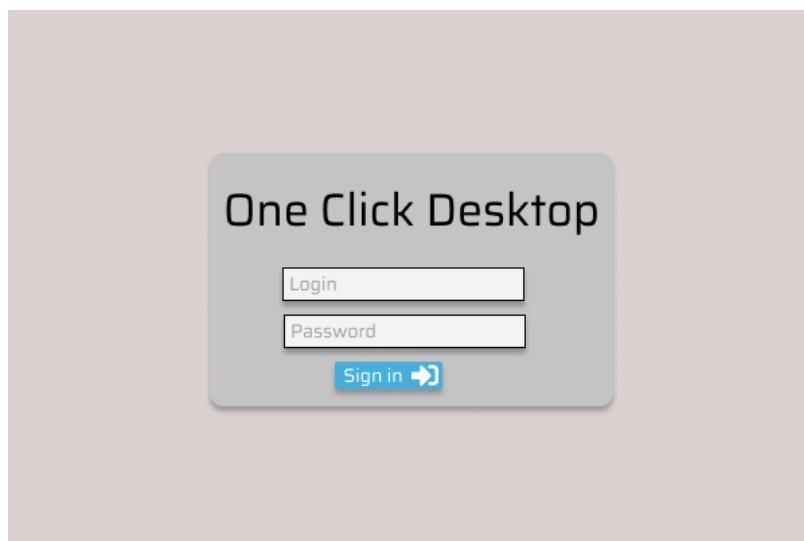
6.2 Diagramy aktywności

6.3 Diagramy klas

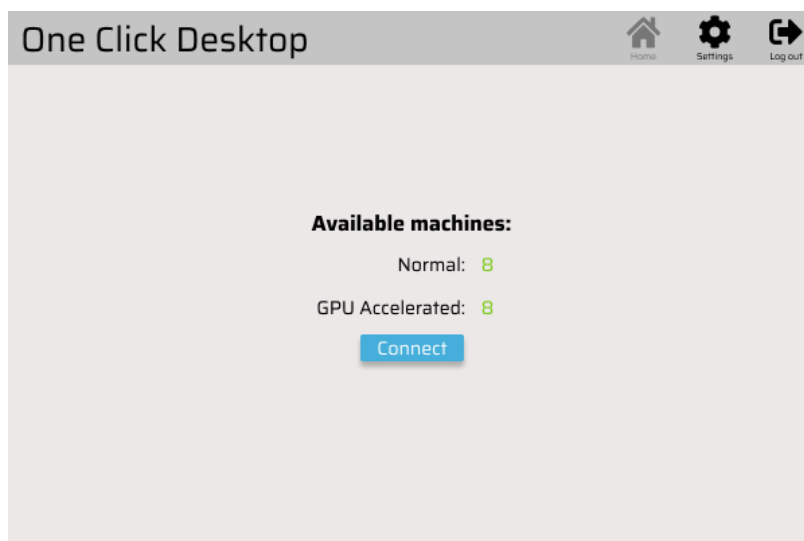
6.4 Diagramy sekwencji

## 7 Interfejs użytkownika

### 7.1 Aplikacja kliencka



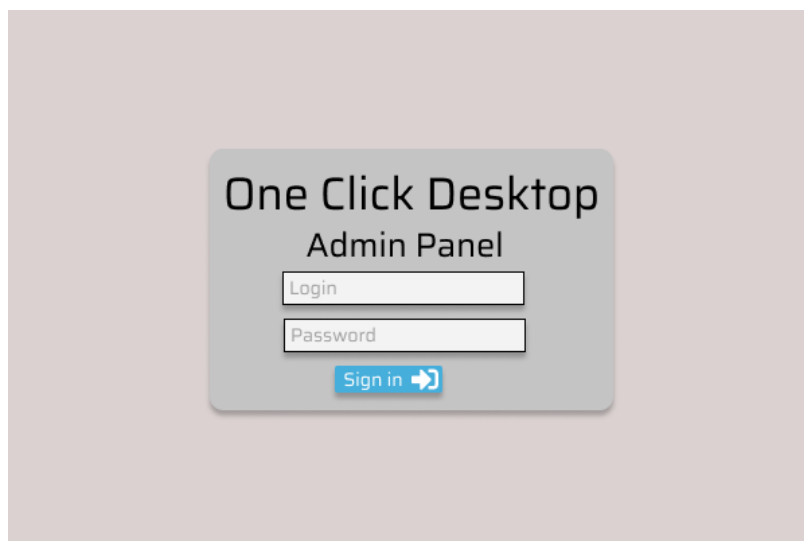
Rysunek 2: Ekran logowania



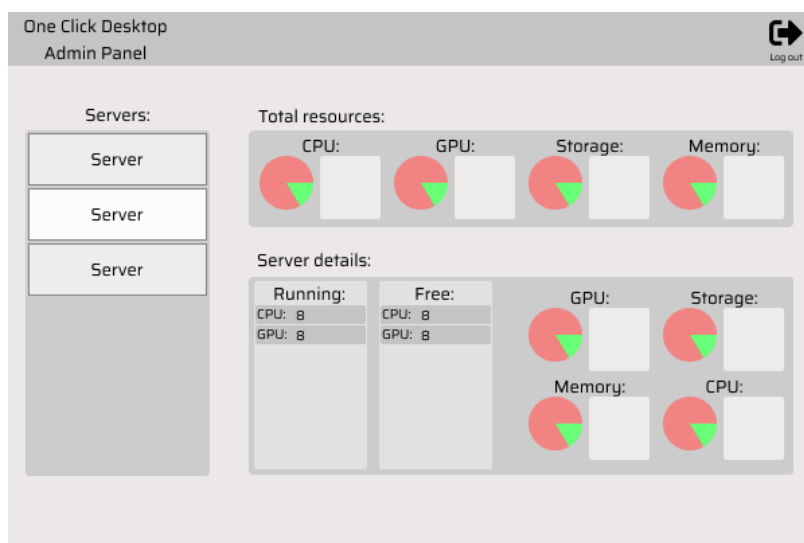
Rysunek 3: Główny widok zawierający dostępność maszyn

## 7.2 Panel administratora

Panel administratora posiada skromny interfejs umożliwiający zalogowanie się oraz podgląd zużycia zasobów.



Rysunek 4: Ekran logowania



Rysunek 5: Widok zużycia zasobów

Na tym widoku możemy zobaczyć zużycie zasobów globalne oraz dla każdego z serwerów wirtualizacji. Dodatkowo dla serwera wyświetlona jest również ilość działających i możliwych do uruchomienia maszyn każdego z typów.



## 8 Zewnętrzne narzędzia

### 8.1 Ansible

Ansible zostanie wykorzystany w systemie do zaaplikowania zmiennej konfiguracji do każdej uruchamianej wirtualnej maszyny. Podstawowo playbook będzie zawierać informacje o:

1. Dane dostępowe do dysku sieciowego oraz wykorzystany protokół
2. Dane dostępowe do usługi katalogowej
3. TODO: dopisać wszystkie potrzebne konfiguracje

Playbook można rozszerzać o potrzebne dane zależne od użycia.

### 8.2 Vagrant

Vagrant zostanie wykorzystany w celu łatwej parametryzacji oraz powtarzalnego tworzenia maszyn wirtualnych z przygotowanego wcześniej obrazu systemu. Głównie wykorzystany będzie mechanizm Vagrantboxów, które są obrazami wcześniej przygotowanego systemu operacyjnego. Aby system działał prawidłowo obraz systemu zamknięty w Vagrantboxie musi spełniać następujące warunki:

1. Użytkownicy muszą być pobierani z usługi katalogowej.
2. Katalogi domowe użytkowników muszą być na dysku sieciowym.
3. TODO: dopisać wszystkie potrzebne wymagania

### 8.3 Libvirt z QEMU

Libvirt połączony z QEMU będzie wykorzystany do zarządzania maszynami wirtualnymi uruchamianymi na serwerze wirtualizacji. Umożliwi on:

1. Tworzenie maszyn wirtualnych.
2. Uruchamianie maszyn wirtualnych.
3. Przyporządkowanie zasobów maszynom wirtualnym (w tym kraty graficzne).
4. Wyłączanie maszyn wirtualnych.
5. Sprawdzanie, czy maszyna działa na serwerze wirtualizacji.

## 9 Wybrana technologia