

# Sauron

System Alarmujący o Uczniach Robiących Oszustwa Naukowe

Patryk Mroczyński  
Jakub Wiśniewski  
Daniel Stańczak  
Oskar Rutkowski

# Spis zagadnień

- Wymagania aplikacji
- Wybrane technologie
- Architektura systemu



# Wymagania aplikacji

## Aplikacja kliencka

- uruchamia się przy starcie systemu
- działa jako proces w tle
- wysyła informacje o aktualnych procesach i przeglądanych stronach na serwer
- pobiera informacje o whitelistach w momencie, gdy włączy się aplikacja prowadzącego
- filtruje aktualne procesy i strony według whitelisty ustalonej przez prowadzącego
- w razie naruszenia, blokuje komputer i wysyła informację do aplikacji prowadzącego
- w przypadku próby wyłączenia aplikacji monitorującej, wysyła informację do aplikacji prowadzącego oraz wyłącza komputer




# Wymagania aplikacji

## Aplikacja zarządcza

- możliwość wyboru klienta, którego procesy chce zobaczyć
- możliwość utworzenia whitelisty
- możliwość nadania interdyktu
- możliwość zapisu aktualnego klienta, który popełnił wykroczenie (screenshot oraz aktualne procesy)
- pierwszy prowadzący udostępnia swoje whitelisty, a następni podłączeni, są jedynie w trybie obserwatorów

## Aplikacja serwerowa

- udostępnienie RESTowego API
  - obsługa noSQLowej bazy danych
- 

# Wybrane technologie

- Vue.js
- Python  $\geq 3.5$  (psutil, PyMongo, CherryPy)
- MongoDB



# Architektura systemu

Architektura klient-serwer: architektura systemu komputerowego, umożliwiająca podział zadań. Serwer zapewnia usługi dla *klientów*, zgłaszających do serwera żądania obsługi. Po stronie serwera będą przetrzymywane informacje o procesach oraz ewentualnych ścieżkach do zrzutów ekranów podejrzanego klienta. Aplikacja “**Sauron**” wysyła żądania do serwera o listy procesów oraz zakładki w przeglądarkach internetowych klientów. Po stronie podłączonych klientów, działają aplikacje klienckie “**Nazgûl**”, które działają w tle oraz zbierają informacje o aktualnie działających procesach oraz włączonych zakładkach internetowych. Następnie wysyłają zebrane informacje na serwer. Na serwerze będzie działać noSQL’owa baza danych, przechowująca logi o procesach użytkowników w danej sesji.





Dziękujemy za uwagę!