



# PLATFORMA **HUMAN BASED COMPUTATION**

Tomasz Rogozik (tr249051)

# HUMAN-BASED COMPUTATION – PRZYPOMNIENIE

- ⊙ **Human-based computation** – delegowanie zadań „obliczeniowych” do wykonania przez grupę ludzi
- ⊙ **Przykłady zadań:**
  - ⊙ etykietowanie zdjęć
  - ⊙ znajdowanie różnic w obrazkach, tekstach
  - ⊙ zbieranie informacji marketingowych
  - ⊙ kategoryzowanie zawartości
  - ⊙ rozpoznawanie pisma

# CEL DO OSIĄGNIĘCIA

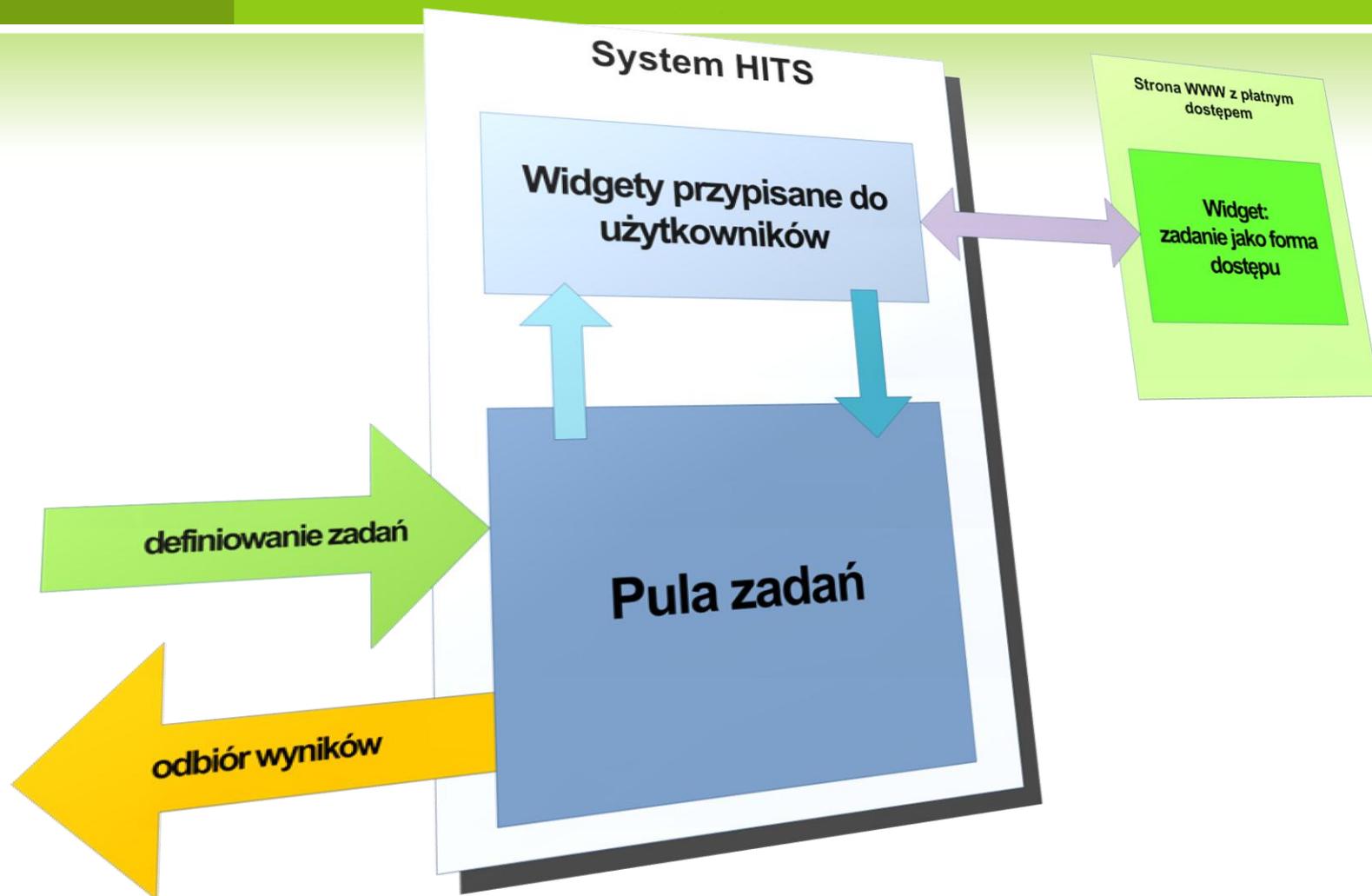
- 🎯 Budowanie unikalnych baz wiedzy przy **niskim** nakładzie finansowym



# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## JAK? - PLATFORMA OBLICZEŃ



- ⊙ Jak zapewnić wyświetlanie zadań i ich obsługę na stronie użytkownika w prosty sposób?

## ROZWIĄZANIE:

zbudować dynamiczną bibliotekę Javascript dołączaną w prosty sposób

```
<form method="post" id="hitsForm">  
  <script type="text/javascript"  
    src="http://hits.org.pl/Widget/challenge.aspx?key=klucz_publiczny">  
  </script>  
</form>
```

- ⊙ **Jak zapewnić możliwość przesyłania wyniku rozwiązanego zadania nie korzystając z bezpośredniego połączenia przeglądarki z serwerem HITS?**

### **ROZWIĄZANIE:**

- zbudować otwartą bibliotekę dołączaną przez użytkowników do ich dynamicznych aplikacji webowych
- zapewnić łatwość przeniesienia implementacji biblioteki do innych języków i technologii
- wykorzystać webservices jako metodę komunikacji niezależną od wybranej technologii

### ◎ Jak zapewnić odporność na podszywanie się pod użytkowników?

#### ROZWIĄZANIE:

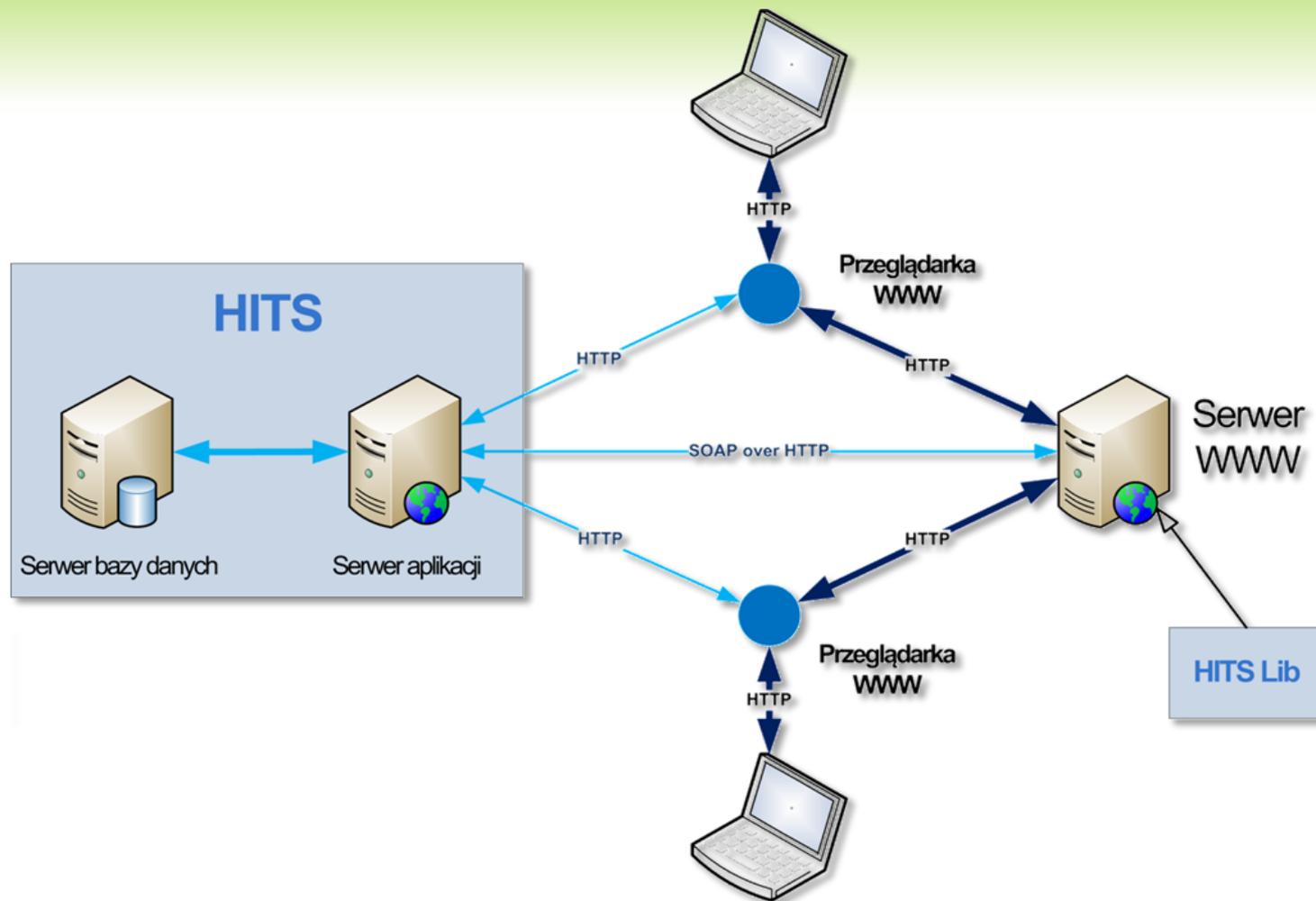
- autoryzacja asymetryczna (klucz publiczny i klucz prywatny)
- widget posiada zapisany klucz publiczny
- biblioteka server-side posiada skonfigurowany klucz prywatny
- serwer HITS autoryzuje użytkownika na podstawie pary kluczy



# HITS

HUMAN INTELLIGENCE TASK SYSTEM

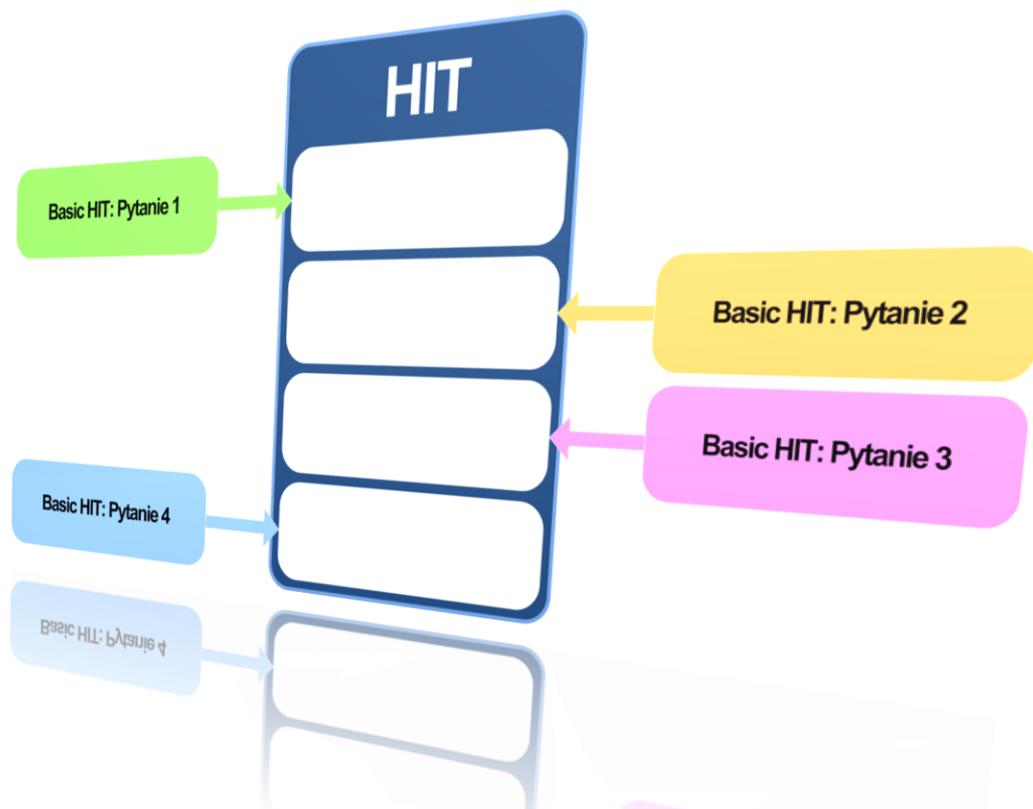
## ARCHITEKTURA ROZWIĄZANIA



- ⊙ **Jakie dane są niezbędne do przechowania? Jak przechowywać zadania pozostawiając dużo swobody użytkownikom je definiującym?**

## ROZWIĄZANIE:

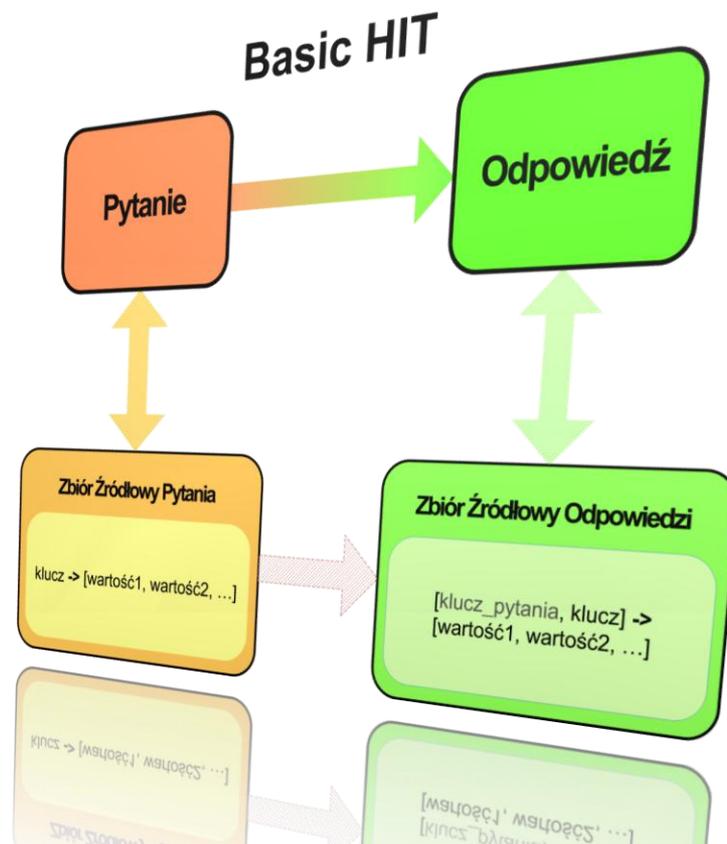
- zadanie (*HIT*) składa się z zadań prostych (*BasicHIT*)



- ⊙ **Jakie dane są niezbędne do przechowania? Jak przechowywać zadania pozostawiając dużo swobody użytkownikom je definiującym?**

## ROZWIĄZANIE:

- zadania proste wykorzystują zbiory danych jako źródła treści pytań i listy dostępnych odpowiedzi oraz docelowe bazy wiedzy
- zadania proste, poprzez rozwiązania, mogą generować nowe elementy zbioru źródłowego pytania



### ◎ Jak zatem przechowywać zbiory danych?

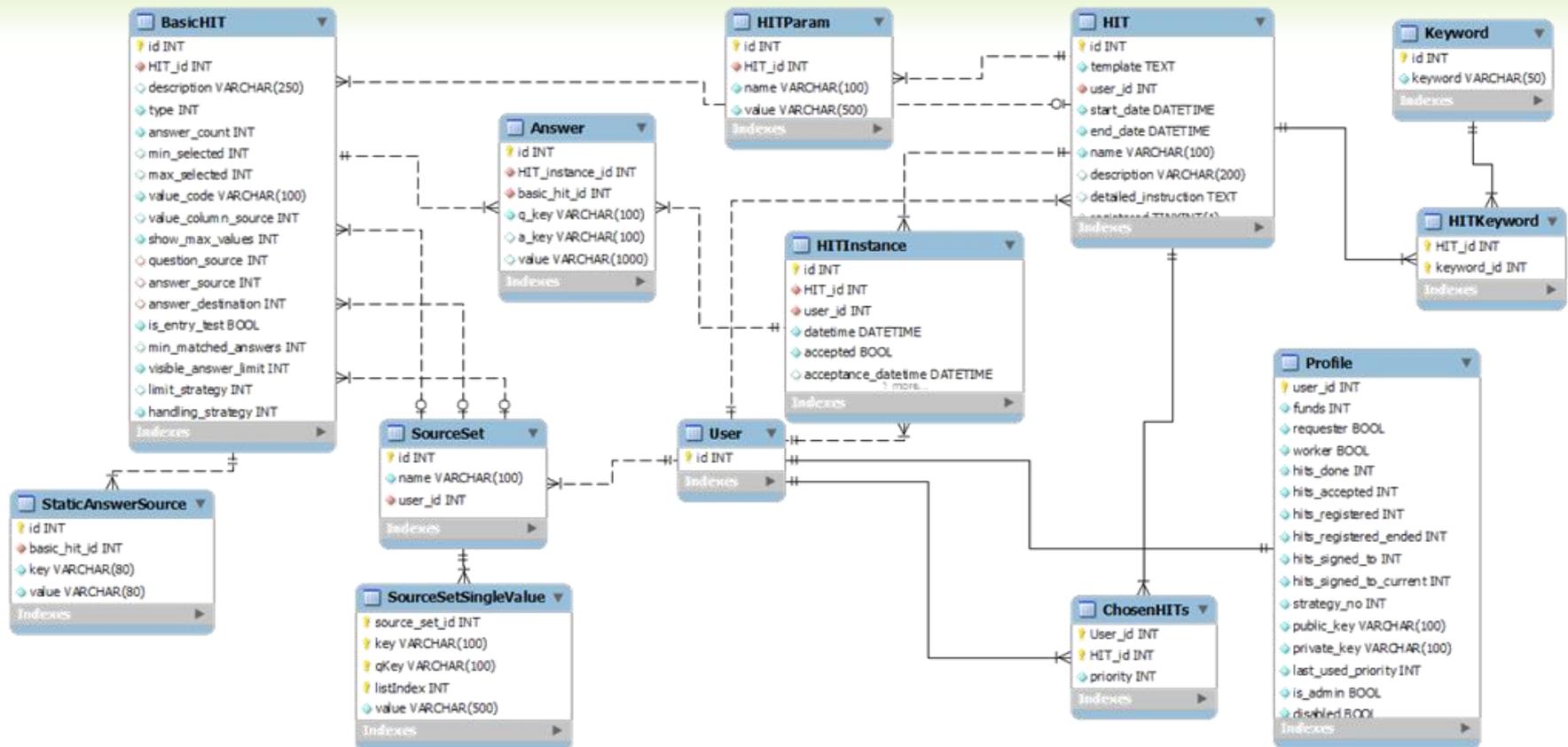
#### ROZWIĄZANIE:

- elementom nadajemy unikalne klucze, umożliwiające ich identyfikację
- elementy zbioru są tak naprawdę strukturami zawierającymi napisy
- modelujemy wykorzystywane zbiory jako zbiory list danych
- listy mogą mieć dowolną długość
- znaczenie kolejnych elementów listy definiujemy w zadaniu

# HITS

HUMAN INTELLIGENCE TASK SYSTEM

# MODEL DANYCH



- ③ **Do widgetu może być przypisanych wiele zadań. Skąd wiadomo, którego użyć?**
- ③ Wprowadzono strategie doboru zadań (interfejs strategii). Interfejs implementują strategie losowa (***Random***) i zgodna z listą priorytetów strategia kolejkowa (***PriorityRoundRobin***)
- ③ Użytkownik decyduje poprzez GUI, której strategii chce użyć dla swojego widgetu

- Podczas ładowania widgetu tworzony jest egzemplarz zadania. Gdy użytkownik nie wypełni zadania, w bazie danych pozostaje pusty egzemplarz. Jak rozwiązać ten problem?



egzemplarz odpowiedzi



tabela tymczasowa



docelowa baza



kosz

# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## TECHNOLOGIA

- ⊙ Serwer aplikacji – **MS IIS 7.0**
- ⊙ Baza danych **MySQL 5.1** z **InnoDB**
- ⊙ **MS.NET 3.5** Framework
  - ⊙ **ASP.NET, ASP** Web Services
  - ⊙ **C#**
  - ⊙ **LINQ-to-SQL** na dbLINQ
- ⊙ **jQuery**



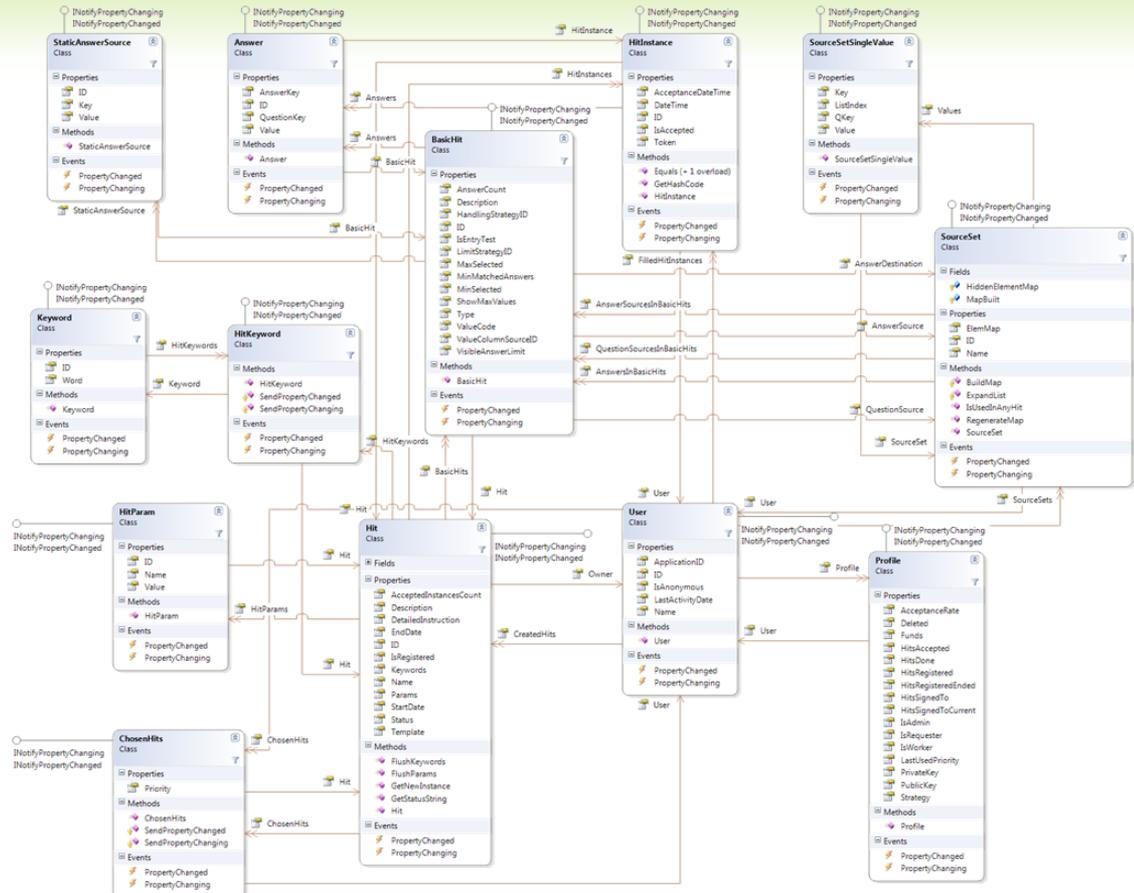
# HITS

HUMAN INTELLIGENCE TASK SYSTEM

# WYKONANE DO TEJ PORY



## ORM, schemat bazy

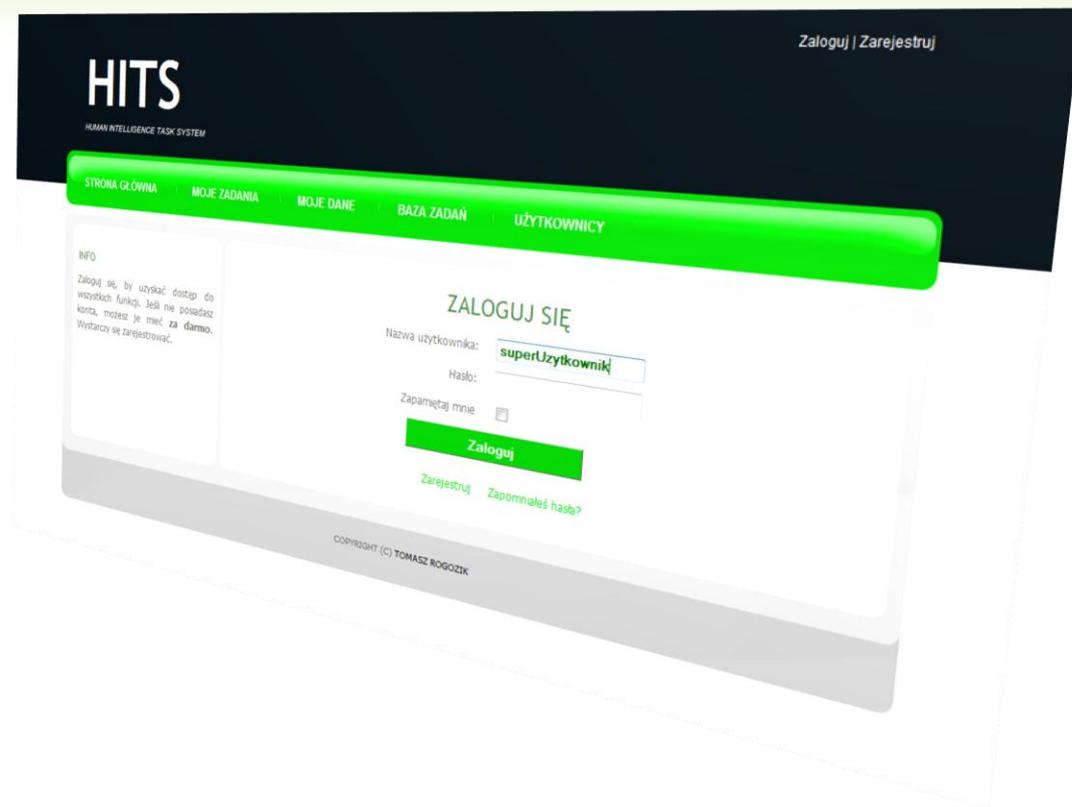


# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## WYKONANE DO TEJ PORY

- ① ORM, schemat bazy
- ① Administracja (rejestracja, logowanie, przywracanie hasła, itd.)



# HITS

HUMAN INTELLIGENCE TASK SYSTEM

# WYKONANE DO TEJ PORY

- 🎯 ORM, schemat bazy
- 🎯 Administracja (rejestracja, logowanie, przywracanie hasła, itd.)
- 🎯 Przeglądanie użytkowników

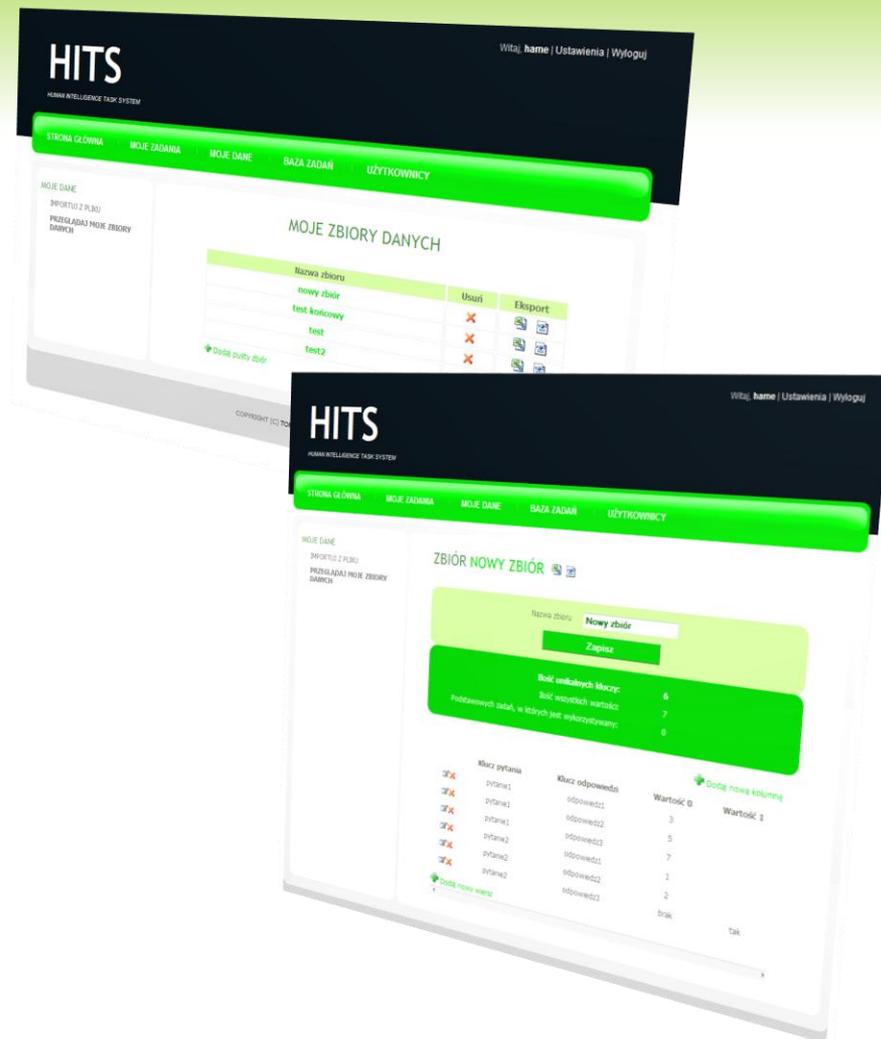


# HITS

HUMAN INTELLIGENCE TASK SYSTEM

# WYKONANE DO TEJ PORY

- ① ORM, schemat bazy
- ① Administracja (rejestracja, logowanie, przywracanie hasła, itd.)
- ① Przeglądanie użytkowników
- ① Dodawanie/modyfikacja/usunięcie zbiorów danych



# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## WYKONANE DO TEJ PORY

- ⦿ **Import zbiorów z CSV, eksport zbiorów do CSV i XML**



# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## WYKONANE DO TEJ PORY

- ⦿ Import zbiorów z CSV, eksport zbiorów do CSV i XML
- ⦿ Ładujący się widget wraz z połączeniem z biblioteką i serwerem HITS (po webservices)

Dostęp chroniony. Rozwiąż zadanie lub wyślij SMS o treści DOSTEP na numer 7999.

OK

powered by HITS

# HITS

HUMAN INTELLIGENCE TASK SYSTEM

## WYKONANE DO TEJ PORY

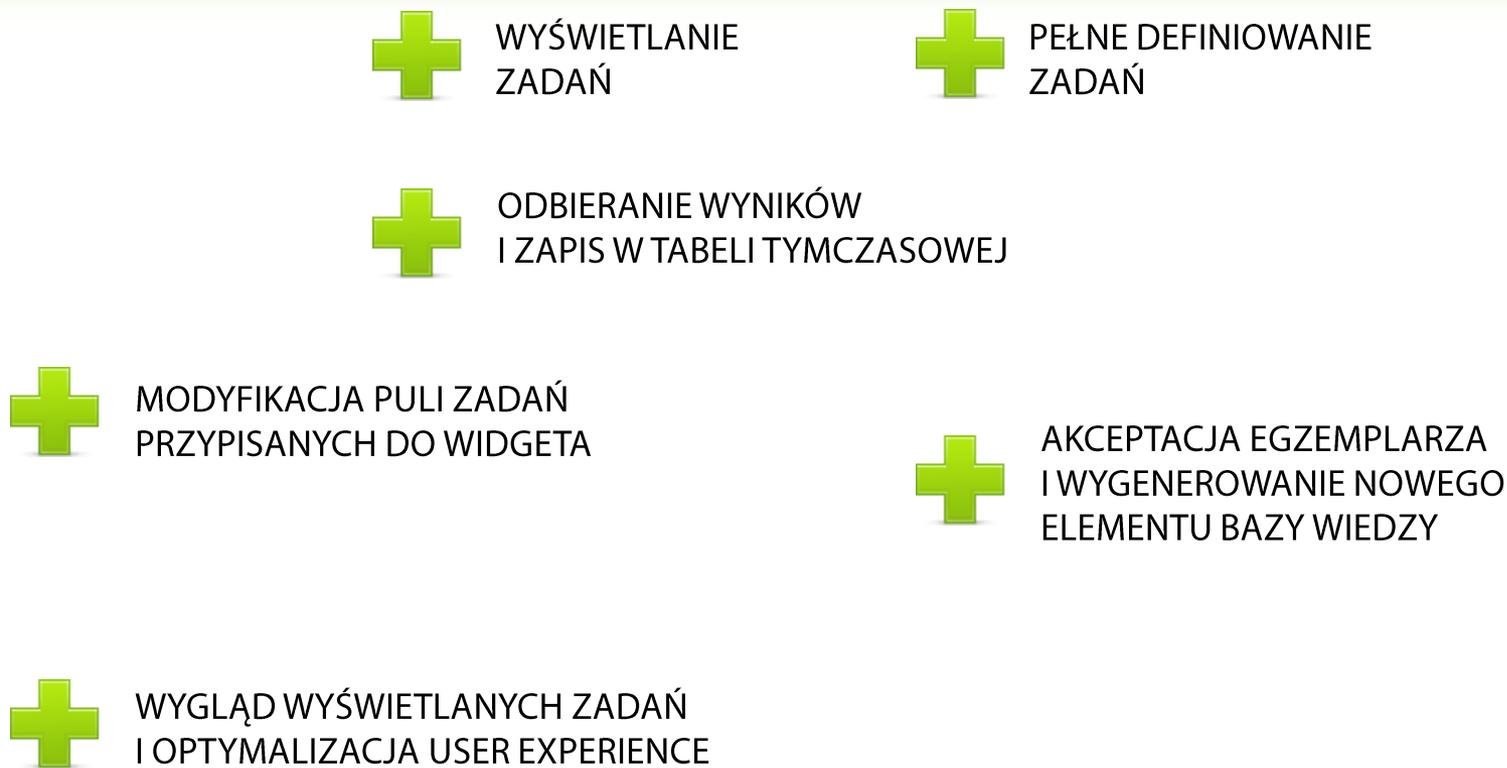
- ⊙ Import zbiorów z CSV, eksport zbiorów do CSV i XML
- ⊙ Ładujący się widget wraz z połączeniem z biblioteką i serwerem HITS (po webservices)
- ⊙ + dużo innych małych, ale niemniej przydatnych rzeczy  
w tej chwili 10875 LOC

# HITS

HUMAN INTELLIGENCE TASK SYSTEM

HUMAN INTELLIGENCE TASK SYSTEM

PRIORYTET



TRUDNOŚĆ IMPLEMENTACJI (czasochłonność)

**DZIĘKUJĘ**

 [Tomasz@Rogozik.pl](mailto:Tomasz@Rogozik.pl)