|  |  |
| --- | --- |
| POLITECHNIKA WARSZAWSKA  Wydział Elektroniki i Technik Informacyjnych | Rok akademicki  2013/2014 |

**PRACA DYPLOMOWA MAGISTERSKA**

Adam Turski

**Architektura i implementacja skalowalnej aplikacji Java EE z użyciem technologii RFID**

Praca wykonana pod kierunkiem

dr inż. Piotra Witońskiego

........................................

ocena pracy

........................................

podpis Przewodniczącego Komisji

Warszawa 2013

# Życiorys

**Adam Turski**

Urodziłem się 24 marca 1986 roku w Warszawie. W latach 2001-2006 uczęszczałem do Technikum Mechatronicznego im. Stanisława Staszica w Siedlcach. Po ukończeniu technikum rozpocząłem studia 1-go stopnia w Instytucie Informatyki Akademii Podlaskiej w Siedlcach. W roku 2009 obroniłem swoją pracę licencjacką oraz rozpocząłem uzupełniające studia magisterskie na Wydziale Elektroniki i Technik Informacyjnych. Z początkiem roku 2013 zatrudniłem się w firmie Enigma SOI jako programista Javy. Zdobytą na studiach wiedzę wykorzystuje na co dzień w swojej pracy zawodowej.

Prywatnie interesuje się kinematografią.

........................................

podpis studenta

**EGZAMIN DYPLOMOWY**

Złożył egzamin dyplomowy w dniu ............................................................................... 200.....r.

z wynikiem ..................................................................................................................................

Ogólny wynik studiów .................................................................................................................

Dodatkowe wnioski i uwagi Komisji ............................................................................................

.....................................................................................................................................................

.....................................................................................................................................................

# Streszczenie

Java Enterprise Edition jest serwerową platformą do tworzenia aplikacji w języku Java opartych o wielowarstwową architekturę komponentową. Dzięki niej proces wytwarzania aplikacji webowych sprowadza się do wykorzystania gotowych, sprawdzonych komponentów. Nasze siły wówczas możemy skupić na realizacji założonego celu biznesowego. Takie podejście poprawia jakość oprogramowania oraz skraca czas jego realizacji. Otwarta architektura pozwala obniżyć całkowity koszt projektu jednocześnie ogranicza ryzyko uzależnienia od jednego dostawcy.

RFID jest technologią identyfikacji obiektów z wykorzystaniem fal radiowych. Dane identyfikacyjne są zapisane na etykiecie (transponder), która posiada wbudowaną antenę. Czytnik przy pomocy anteny nadajnika odczytuje dane z transpondera na odległość. Technika ta, w odróżnieniu od innych metod identyfikacyjnych np. kodów kreskowych, pozwala zautomatyzować proces identyfikacji oraz ograniczyć prawdopodobieństwo błędu ludzkiego

Celem mojej pracy magisterskiej było stworzenie skalowalnego systemu JEE z wykorzystaniem techniki RFID. Założenie to zrealizowałem tworząc system śledzenia przesyłek opatrzonych transponderami. System ten z powodzeniem może zostać wykorzystany w firmach z branż pocztowych oraz kurierskich.

**Abstract**

Java Enterprise Edition is a server platform for building Java applications based on the multi-tier component's architecture . With this platform, process of producing a web application comes down to use , proven components . Then we can focus our forces on achieving the planned businnes purpose. This approach improves software quality and reduces the time of its implementation. The open architecture allows you to reduce the overall cost of the project at the same time reduces the risk of dependence on a single supplier .

RFID is a technology for identifying objects using radio waves. Identification data are stored in the tag ( transponder), which has a built-in antenna . Reader with a transmitter antenna reads the transponder at a distance. This technique , in contrast to other methods of identification such as bar codes, allows to automate the identification process and reduce the likelihood of human error.

The purpose of my thesis was to create a scalable JEE system with the use of RFID technology. This assumption I realized by creating a system tracking shipments marked with RFID tags. This system can be successfully used in companies with postal and courier industries .

# Spis treści

[Życiorys 2](#_Toc372923070)

[Streszczenie 3](#_Toc372923071)

[Spis treści 5](#_Toc372923072)

[1 Wstęp 6](#_Toc372923073)

[Cel i zakres pracy 7](#_Toc372923074)

[2 Wprowadzenie do systemów JEE oraz technologii RFID 8](#_Toc372923075)

[2.1 8](#_Toc372923076)

[3 Opis systemu 9](#_Toc372923077)

[3.1 Wymagania funkcjonalne 9](#_Toc372923078)

[3.1.1 Wymagania funkcjonalne dla użytkownika 11](#_Toc372923079)

[3.1.2 Wymagania funkcjonalne serwisów 11](#_Toc372923080)

[4 Architektura 13](#_Toc372923081)

[4.1 Warstwa bazodanowa 13](#_Toc372923082)

[4.1.1 Projekt bazy danych 13](#_Toc372923083)

# Wstęp

W dzisiejszych czasach bardzo popularne jest powiedzenie "*Kto stoi w miejscu ten się cofa*". Chociaż prawda ta była, jest i będzie aktualna to jednak obecne czasy najlepiej to odzwierciedlają. Dużo firm traci swoją pozycję na rynku z powodu braku zmian w procesach firmy, począwszy od procesu technologicznego po wizerunek firmy. Techniki, które wczoraj były wystarczające, dziś wymagają zmiany.

Drugim popularnym powiedzeniem

Niniejsza praca magisterska opisuje architekturę oraz implementację skalowalnego systemu zrealizowanego w standardzie JAVA EE[[1]](#footnote-1) z użyciem technologii RFID[[2]](#footnote-2). Aplikacje wykonane w standardzie JEE są opartego na wielowarstwowej architekturze komponentowej.//TODO(więcej o technologii JEE)

Technologia RFID, czyli identyfikacja za pomocą "tagów" //TODO(co to jest). Pozwala ona na zastąpienie innych typów identyfikacji np. kodów kreskowych, co przekłada się //TODO(na co).

## Cel i zakres pracy

Głównym celem niniejszej pracy było zaprojektowanie oraz implementacja skalowalnego systemu z wykorzystaniem technologii RFID w oparciu o technologię Java Enterprise Edition. Aby zrealizować to założenie został opracowany system wspomagający śledzenie przesyłek HERMES wykorzystujący transpondery jako znaczniki przesyłek.

W pierwszych rozdziałach zostaną omówione technologie JEE oraz RFID, a także proces śledzenia przesyłek. Kolejne przedstawiają architekturę oraz implementację systemu.

Rozdział 1. opisuje serwerową platformę programistyczną JEE jako standard tworzenia aplikacji opartych o wielowarstwową architekturę komponentową. Przedstawia technologię RFID w świetle obecnych zastosowań.

Rozdział 2. analizuje proces śledzenia przesyłek, a także dostępnych na rynku podobnych rozwiązań.

Rozdział 3. opisuje koncepcję oraz projekt platformy HERMES. Zostaną omówione wymagania stawiane systemowi, przedstawiony zostanie model konceptualny.

Rozdział 4. przedstawia narzędzia oraz biblioteki za pomocą których wynikowy system został zrealizowany.

Rozdział 5.

# Wprowadzenie do systemów JEE oraz technologii RFID

## 

# Projekt systemu

Systemem opisywanym w tej pracy magisterskiej będzie system śledzenia przesyłek HERMES. System ten pozwoli Aplikacja ta służy do rejestracji oraz śledzenia przesyłek.

TODO: Więcej co to bedzie robilo

System jest dedykowany dla firm z branż pocztowych oraz kurierskich. Pomysł na stworzenie takiego systemu przyszedł autorowi na myśl podczas pobytu w punkcie pocztowym i obserwacji nadawaniu paczek wykorzystując technologię kodów kreskowych. Technologia kodów kreskowych w dobie dzisiejszych możliwości ustępuje technologii RFID, jednocześnie wymaga większego zaangażowania zasobów ludzkich przy obsłudze takiego rodzaju identyfikacji.

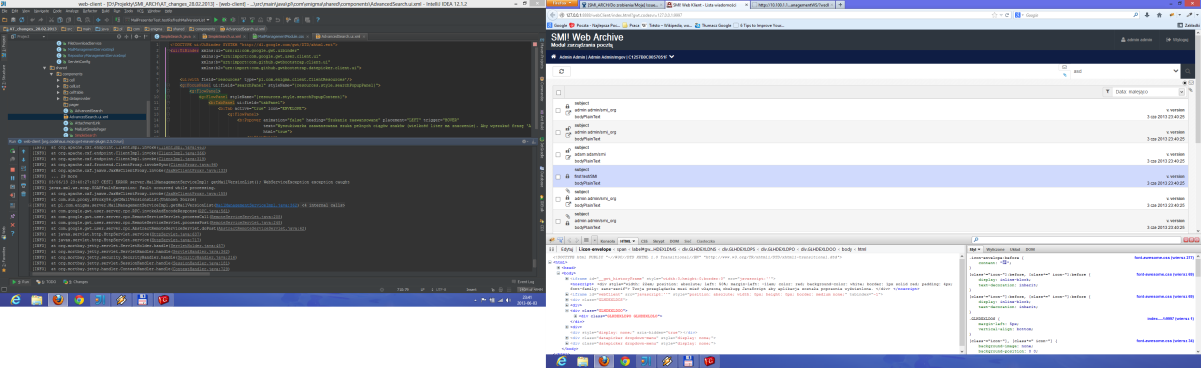
Aby lepiej zrozumieć

TODO: WIecej dla kogo to ejst

## Wymagania funkcjonalne

Analiza wymagań funkcjonalnych pozwala na zidentyfikowanie i opisanie oczekiwanego zachowania systemu. W myśl jednej z definicji, wymagania funkcjonalne to stwierdzenie, jakie usługi ma oferować system, jak ma reagować na określone dane wejściowe oraz jak ma się zachowywać w określonych sytuacjach.

Zebrane wymagania funkcjonalne zostaną przedstawione poniżej na diagramie przypadków użycia (Diagram przypadków użycia).



Rysunek Diagram przypadków użycia

Najważniejsze przypadki użycia systemu przedstawione zostały poniżej w formie formularzy wymagań funkcjonalnych.

Tabela Opis wymagań funkcji Rejestruj przesyłkę

|  |  |
| --- | --- |
| **Nazwa funkcji** | **Rejestruj przesyłkę** |
| Opis | Funkcja pozwalająca na rejestrację przesyłki w systemie |
| Dane wejściowe | Numer identyfikacyjny przesyłki |
| Źródło danych wejściowych | Pracownik poczty |
| Wynik | Rejestracja przesyłki w systemie |
| Warunek wstępny | Przesyłka została dostarczona do punktu pocztowego w celu nadania |
| Warunek końcowy | Przesyłka została otagowana znacznikiem RFID |
| Powód | Śledzenie przesyłki |

Tabela Opis wymagań funkcji Zmień status przesyłki

|  |  |
| --- | --- |
| **Nazwa funkcji** | **Zmień status przesyłki** |
| Opis | Funkcja pozwalająca na zmianę statusu przesyłki |
| Dane wejściowe | Numer identyfikacyjny przesyłki, nowy status |
| Źródło danych wejściowych | System HERMES, pracownik poczty, |
| Wynik | Zaktualizowanie statusu przesyłki w systemie |
| Warunek wstępny | Wprowadzony został numer identyfikacyjny przesyłki oraz nowy status |
| Warunek końcowy | Przesyłka jest aktywna |
| Powód | Zmiana statusu przesyłki |

Tabela Opis wymagań funkcji Sprawdź status przesyłki

|  |  |
| --- | --- |
| **Nazwa funkcji** | **Sprawdź status przesyłki** |
| Opis | Funkcja pozwalająca na sprawdzenie statusu przesyłki |
| Dane wejściowe | Numer identyfikacyjny przesyłki |
| Źródło danych wejściowych | Nadawca przesyłki |
| Wynik | Informacja o statusie przesyłki |
| Warunek wstępny | Wprowadzony został numer identyfikacyjny przesyłki |
| Warunek końcowy | Wprowadzono dane weryfikacyjne |
| Powód | Sprawdzenie statusu przesyłki |

### Wymagania funkcjonalne dla użytkownika

Wymagania funkcjonalne dla użytkownika to:

* Sprawdzenie statusu przesyłki

### Wymagania funkcjonalne serwisów

Wymagania funkcjonalne dla serwisów to:

* zmiana statusu przesyłki

# Architektura

System HERMES posiada warstwową architekturę typową dla aplikacji Java EE.

Tutaj powinien pojawić się obrazek struktury systemu!!

## Warstwa bazodanowa

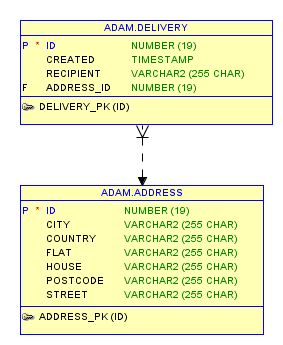
Najniższą warstwą systemu HERMES jest warstwa bazodanowa. Zastosowany został relacyjny silnik bazy danych ORACLE 11g Express Edition. Rozwiązanie to, będące jednocześnie najbardziej popularne i wiodące na rynku, pozwala na darmowe budowanie jak i użytkowanie aplikacji z pewnymi ograniczeniami:

* możliwość zainstalowania jednej bazy danych na pojedynczej maszynie,
* maksymalny rozmiar danych użytkownika do 11GB,
* maksymalne zużycie pamięci do 1GB,
* użycie tylko jednego procesora maszyny.

Na potrzeby tworzonego systemu wymagania te nie uniemożliwiają pracy systemu. Jednak w środowisku produkcyjnym takie ograniczenia są nie do przyjęcia, przez co potrzebne jest wykupienie licencji oraz migracja do wersji płatnej. Plusem tego rozwiązania jest to, że producent gwarantuje nam bezproblemową migrację do produktu płatnego.

### Projekt bazy danych

Baza danych została zaprojektowana z myślą przechowywania danych dotyczących systemu (konfiguracja, połączenia,) jak również danych logicznych (przesyłki, statusy). Struktura bazy zostanie przedstawiona w formie dwóch diagramów.



# Implementacja

1. Java EE - Java Enterprise Edition [↑](#footnote-ref-1)
2. RFID - Radio Frequency ID [↑](#footnote-ref-2)