

# ŚLEDZENIE STANU I RUCHU JEDNOSTEK LOGISTYCZNYCH W MAGAZYNIE ZA POMOCĄ WIRTUALNEGO MODELU 3D

*W artykule omówiona została implementacja unikalnego oprogramowania wspomagającego zarządzanie magazynem za pomocą interaktywnego, trójwymiarowego modelu magazynu. Program łączy systemy RTLS i WMS, wykorzystując dane pochodzące z obu źródeł do reprezentacji stanu magazynu w przejrzystym i intuicyjnym interfejsie.*

## WSTĘP

Rosnąca przepustowość łączy internetowych oraz rozwój standardów i możliwości przeglądarek WWW dają nam dzisiaj możliwość budowy zupełnie nowej klasy rozwiązań informatycznych, dostępnych on-line i wygodnych dla użytkownika. Internet przestał już być zbiorem statycznych stron zawierających teksty, zdjęcia i media do pobrania lub obejrzenia i wysłuchania w streamingu. Rewolucja pod hasłem „Web 2.0” dała użytkownikom możliwość o wiele bardziej rozbudowanej interakcji z serwisami WWW. Przede wszystkim powszechnie zaczęto wykorzystywać dwukierunkową komunikację asynchroniczną i dynamiczną manipulację strukturą dokumentu HTML. Dzięki temu wiele stron internetowych nabrało cech zarezerwowanych dotąd dla aplikacji stacjonarnych, jak np.: ładowanie danych z serwera i wysyłanie danych na serwer bez konieczności przeladowywania całej strony, uruchamianie akcji inicjowanych przez serwer, użycie modalnych okienek do edycji danych i dynamiczna reakcja na akcje użytkownika.

Rozpoczynając prace nad projektem integrującym systemy RTLS (ang. Real Time Location System) i WMS (ang. Warehouse Management System) w nowym magazynie podjęliśmy decyzję o wykorzystaniu do tego celu aplikacji webowej, która w założeniu miała mieć kilka istotnych przewag nad aplikacją desktopową:

- lokalizacja aplikacji na serwerze zamiast na wielu komputerach firmy;
- uniknięcie kłopotliwego i kosztownego serwisowania aplikacji w wielu różnych środowiskach użytkowników;
- łatwa aktualizacja, jednoczesna dla wszystkich użytkowników;
- pełna kontrola nad środowiskiem działania aplikacji;

Zbudowany w ramach tego projektu system Warehouse3D służy jako łącznik pomiędzy systemem Mojix Star (RTLS), a systemem ExpertWMS, prezentując jednocześnie informacje o bieżącej lokalizacji jednostek logistycznych i położeniu wózków na trójwymiarowej wizualizacji magazynu.

## 1. PRZYGOTOWANIE ŚRODOWISKA I NARZĘDZI

### 1.1. System lokalizacyjny

Niezbędnym elementem całego projektu jest infrastruktura dostarczająca informację o bieżącej i rzeczywistej fizycznej lokalizacji wybranych obiektów w magazynie. Taką rolę spełniają systemy RTLS. Spośród dostępnych na rynku systemów RTLS wybrany został Star firmy Mojix, który zapewnia optymalny w tym zastosowaniu stosunek kosztów do możliwości. Szczególnie istotna jest tu

możliwość wykorzystania do lokalizacji pasywnych znaczników RFID, co w kontekście konieczności znakowania jednostek logistycznych pozwala na znaczne ograniczenie kosztów zakupu znaczników. Dostajemy przy tym precyzję pozycjonowania na poziomie do 1 metra, co dla projektowanego zastosowania jest całkowicie wystarczające.

Dla pokrycia całego obszaru magazynu wymagane było zainstalowanie ponad 500 anten zasilających i jednej anteny odbiorczej.

Wynikiem działania Stara są raporty odczytanych lokalizacji, dostarczane przez serwer Star w formie dokumentów XML za pośrednictwem web service typu REST [4, 7].

### 1.2. Aplikacja WMS

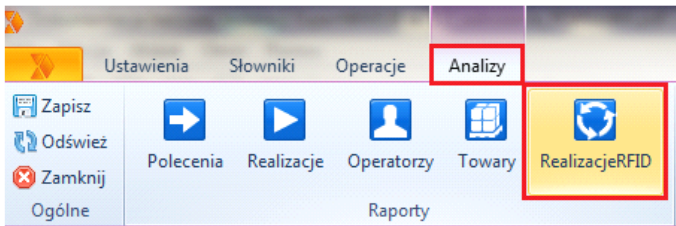
WMS pełni rolę nadrzędnego źródła informacji o jednostkach logistycznych. Pozostaje on też odpowiedzialny za ewidencję przyjęć, wydań, przesunięć i stanów magazynowych. WMS będzie też otrzymywał potwierdzenia realizacji zleceń, pochodzące z odczytu przez RTLS rzeczywistej pozycji znacznika, znajdującego się na jednostce logistycznej.

Dla tego zastosowania został wybrany ExpertWMS firmy Dataconsult, który zapewnia możliwość budowy interfejsu wymiany danych na poziomie bazy danych oraz umożliwia obsługę użytkowników przez interfejsy WWW i mobilny.

Z funkcjonalnego punktu widzenia system ExpertWMS wspiera zarządzanie magazynem od momentu przyjęcia towarów przez zatowarowanie stref kompletacji, aż po procesy wydań.

Użytkownicy w systemie ExpertWMS realizują operacje takie jak: przyjęcie towaru, przesunięcie jednostki logistycznej, zebranie jednostki logistycznej do wydania, załadunek jednostek logistycznych oraz inne. Każda operacja podczas której, oklejona odpowiednią etykietą ze znacznikiem RFID, jednostka logistyczna przesuwana jest w magazynie z jednej lokalizacji na drugą rejestrowana jest w systemie jako realizacja. Każda realizacja/ruch w magazynie eksportowany będzie z systemu WMS do tabel przejściowych zgodnie z ustalonymi regułami podczas uszczegółowienia danych dotyczących integracji ExpertWMS i STAR RFID.

W systemie WMS utworzona zostanie kompozycja, w której z poziomu aplikacji ExpertWMS Panel upoważniony do tego użytkownik będzie miał możliwość podglądu potwierdzonych przesunięć jednostek w magazynie. Kompozycja „Realizacje RFID” (Rys. 1.) w systemie WMS umieszczona została w menu Operacje Analizy, przycisk [Realizacje RFID].



Rys. 1. Interfejs użytkownika ExpertWMS

Każda realizacja, to znaczy każdy transport jednostki logistycznej z punktu A do punktu B będzie potwierdzany przez system STAR. Zatwierdzona realizacja wysyłana będzie do tabel przejściowych, a następnie system Warehouse3D będzie na podstawie danych pochodzących z systemu STAR weryfikował czy dane przesunięcie jest zgodne z zarejestrowanym ruchem. Wynik weryfikacji będzie odsyłany jako informacja zwrotna do tabel przejściowych.

### 1.3. Wybór środowiska i zestawu oprogramowania narzędziowego 3D

W wyniku przeprowadzonej analizy wymagań dla powstającego systemu określona została technologia wykonania aplikacji Warehouse3D:

- aplikacja Web, oparta o serwer MS IIS
- .Net Framework 4.5
- Baza danych SQL Server
- Entity Framework
- architektura MVC
- Web Services REST
- HTML 5
- jQuery
- Bootstrap
- wizualizacja magazynu dostępna przez przeglądarkę WWW

Wykonane zostały analizy i testy następujących dostępnych silników grafiki 3D, działających w przeglądarkach WWW: Unity 5, Unreal 4, Three.js, Babylon.js, Voxel.js, KickJS, Turbulenz, PlayCanvas, Spark Engine, minko, Delight, SpiderGL, OSG JS, JS3D, Pre3D, X3DOM, C3DL, SceneJS, PhiloJS, Ambiera CopperLicht, GLGE, CubicVRjs, Gladius, Goo Create, Phoria.js, Cube Engine, Famo.us, AwayJS, Blend4web, Godot Engine, ShiVa3D, HaxorEngine.

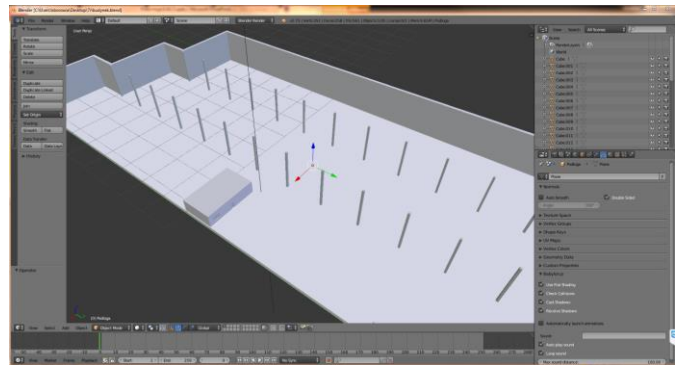
Spośród wymienionych silników został wybrany Babylon.js [5], który zapewnia odpowiednią dostępność, wydajność, możliwości dalszego rozwoju oprogramowania oraz bogatą dokumentację i narzędzia pomocnicze ułatwiające proces tworzenia oprogramowania.

Ponadto do tworzenia brył 3D został wybrany program Blender na licencji GPL, oferujący duże możliwości importu i eksportu danych oraz dobrej jakości dokumentację.

## 2. BUDOWA SYSTEMU

### 2.1. Modelowanie przestrzeni magazynu.

Na podstawie projektu CAD magazynu, dostarczonego przez 7R Logistic, utworzono w programie Blender (Rys. 2.) model przestrzeni magazynowej. Model odwzorowuje dokładnie układ magazynu, rozmieszczenie filarów i budynków biurowych wewnątrz. Dodatkowo kolor i faktura ścian oraz podłogi zostały dobrane tak, żeby zapewnić jak najlepszy kontrast i czytelność w przeglądarce WWW.

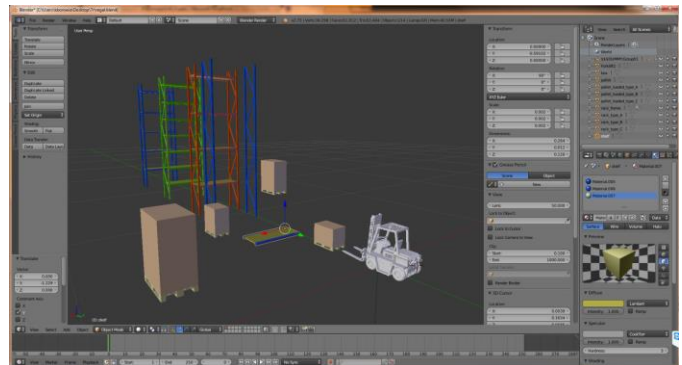


Rys. 2. Budowa przestrzeni magazynu w programie Blender

### 2.2. Warehouse3D – wizualizacja magazynu

Wizualizacja magazynu generowana jest na stronie internetowej serwisu Warehouse3D i jest dostępna za pomocą dowolnej przeglądarki internetowej obsługującej bibliotekę WebGL. Wykorzystano w tym celu silnik 3D Babylon.js oraz język skryptowy javascript.

Zgodnie z dostarczonym przez 7R Logistic projektem wykonano modele 3D jednostek logistycznych oraz trzech typów regałów magazynowych używanych w magazynie (Rys. 3.). Modele zostały wykonane w sposób umożliwiający reprezentowanie różnych typów regałów.



Rys. 3. Budowa elementów wyposażenia magazynu w programie Blender

Następnie zaimplementowano tabele bazy danych przechowujące informację o dokładnej lokalizacji każdego miejsca magazynowego w przestrzeni 3D wraz z jego typem, nazwą i adresem zawierającym nr rzędu, kolumny i poziomu. Na podstawie dostarczonego planu zaregalowania zostały utworzone i umieszczone w tabeli definicje 17 281 miejsc magazynowych (Rys. 4.).

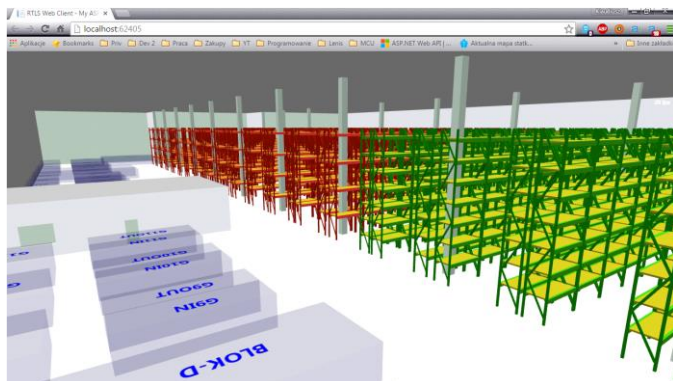
iloc_id	iloc_name	iloc_row	iloc_column	iloc_level	iloc_x1	iloc_y1	iloc_z1	iloc_x2	iloc_y2	iloc_z2	iloc_type	
1	17280	R57K25P00	57	25	0	2.46500000	1.28400000	0.00000000	2.55600000	1.40400000	0.17000000	1
2	17281	R57K26P00	57	26	0	2.55600000	1.28400000	0.00000000	2.64600000	1.40400000	0.17000000	1
3	17282	R57K27P00	57	27	0	2.64600000	1.28400000	0.00000000	2.73600000	1.40400000	0.17000000	1
4	17283	R57K25P01	57	25	1	2.46500000	1.28400000	0.19000000	2.55600000	1.40400000	0.36000000	1
5	17284	R57K26P01	57	26	1	2.55600000	1.28400000	0.19000000	2.64600000	1.40400000	0.36000000	1
6	17285	R57K27P01	57	27	1	2.64600000	1.28400000	0.19000000	2.73600000	1.40400000	0.36000000	1
7	17286	R57K25P02	57	25	2	2.46500000	1.28400000	0.41000000	2.55600000	1.40400000	0.58000000	1
8	17287	R57K26P02	57	26	2	2.55600000	1.28400000	0.41000000	2.64600000	1.40400000	0.58000000	1
9	17288	R57K27P02	57	27	2	2.64600000	1.28400000	0.41000000	2.73600000	1.40400000	0.58000000	1
10	17289	R57K25P03	57	25	3	2.46500000	1.28400000	0.63000000	2.55600000	1.40400000	0.80000000	1
11	17290	R57K26P03	57	26	3	2.55600000	1.28400000	0.63000000	2.64600000	1.40400000	0.80000000	1
12	17291	R57K27P03	57	27	3	2.64600000	1.28400000	0.63000000	2.73600000	1.40400000	0.80000000	1
13	17292	R57K25P04	57	25	4	2.46500000	1.28400000	0.85000000	2.55600000	1.40400000	1.02000000	1
14	17293	R57K26P04	57	26	4	2.55600000	1.28400000	0.85000000	2.64600000	1.40400000	1.02000000	1
15	17294	R57K27P04	57	27	4	2.64600000	1.28400000	0.85000000	2.73600000	1.40400000	1.02000000	1
16	17275	R57K22P00	57	22	0	2.74800000	1.28400000	0.00000000	2.83800000	1.40400000	0.17000000	1
17	17276	R57K23P00	57	23	0	2.83800000	1.28400000	0.00000000	2.92800000	1.40400000	0.17000000	1
18	17277	R57K24P00	57	24	0	2.92800000	1.28400000	0.00000000	3.01800000	1.40400000	0.17000000	1

Rys. 4. Fragment tabeli przechowującej lokalizacje miejsc magazynowych w rzeczywistej przestrzeni magazynu 7R

Do budowy strony WWW użyty został standard HTML5, dzięki czemu możliwe było pokazanie sceny trójwymiarowej za pomocą elementu canvas języka HTML5 [1]. Strona budowana jest przez

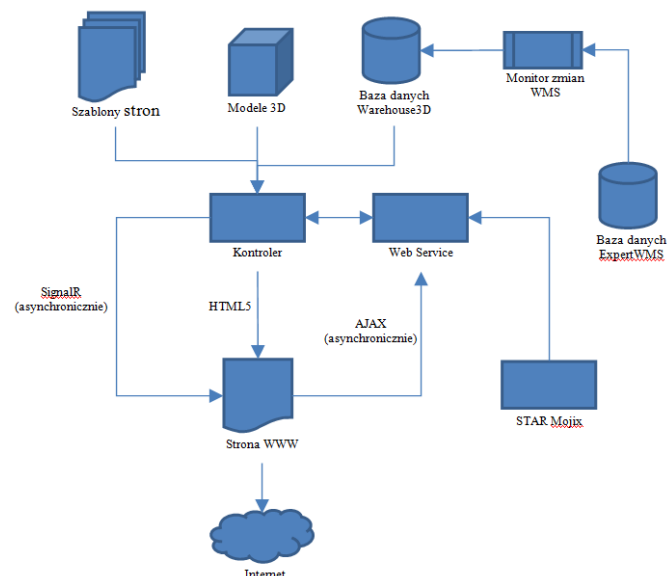
serwer IIS dynamicznie, na żądanie pochodzące z przeglądarki. Oprogramowanie serwera zostało wykonane w architekturze MVC (Model-View-Controller). W celu wygenerowania strony serwer łączy ze sobą informacje pochodzące z bazy danych MS SQL Server, predefiniowane modele 3D oraz szablony stron zapisane z użyciem technologii Razor w języku C# [8]. Pozwala to na automatyczne wygenerowanie skryptów javascript [2], definiujących obiekty 3D, ich zachowanie na stronie WWW oraz obsługujących interakcję z użytkownikiem. Modele 3D użyte w wizualizacji zostały zbudowane w programie Blender [3], a ich rozmieszczenie i zachowanie jest determinowane przez własności zapisane w tabelach bazy danych (Rys. 5).

Elementy wizualizacji można podzielić na dwie grupy: elementy statyczne i dynamiczne. Elementami statycznymi są: konstrukcja magazynu, rozmieszczenie stref, regały magazynowe i oświetlenie. Elementy dynamiczne to jednostki logistyczne oraz wózki widłowe.



**Rys.5.** Wygląd wizualizacji wykorzystującej zamodelowaną przestrzeń magazynu oraz elementy wyposażenia rozmieszczone zgodnie z danymi pobranymi z bazy danych.

Elementy statyczne są generowane i rysowane na wizualizacji tylko raz w czasie otwierania strony WWW w przeglądarce, ponieważ ich układ i własności nie zmieniają się w czasie pracy z programem.



**Rys.6.** Elementy systemu

W przypadku elementów dynamicznych zachodzi konieczność zmiany ich lokalizacji na wizualizacji w celu odzwierciedlenia bieżą-

cego stanu magazynowego oraz informacji pochodzących z systemu STAR Mojix (Rys. 6.).

Dwukierunkowa i asynchroniczna komunikacja strony WWW z serwerem bez konieczności przeładowywania całej strony jest realizowana za pomocą bibliotek: AJAX (dla komunikacji strona WWW – serwer) oraz SignalR (dla komunikacji serwer – strona WWW).

### 2.3. Warehouse3D – serwer

Częścią oprogramowania serwera Warehouse3D jest moduł monitorujący zmiany w tabelach przejściowych systemu ExpertWMS. Każda zmiana lokalizacji jednostki logistycznej w WMS jest wykrywana przez ten moduł i informacja o nowej lokalizacji jest przesyłana za pomocą biblioteki SignalR do wszystkich klientów (przeglądarek) podłączonych aktualnie do serwera. W przeglądarce, za pomocą skryptów javascript, zmieniana jest lokalizacja jednostki logistycznej na nowe miejsce magazynowe.

Warehouse3D posiada również moduł web service, który służy do pobierania danych z systemu STAR Mojix. Dane w postaci raportów w formacie XML [6] są przetwarzane asynchronicznie i odczytywana jest z nich informacja o bieżącym, rzeczywistym położeniu jednostek logistycznych oznakowanych etykietami RFID. Odczytane dane służą do potwierdzenia wykonania zlecenia pochodzącego z WMS.

ExpertWMS umieszcza w tabelach przejściowych informację o rozpoczętej realizacji zlecenia przesunięcia jednostki logistycznej, a Warehouse3D po odczytaniu tej informacji rozpoczyna proces śledzenia wskazanej jednostki logistycznej. Przez określony w konfiguracji czas, serwer monitoruje raporty STAR pod kątem śledzonego numeru SSCC i zapisuje jego pozycję w bazie danych. Jeśli przed upływemadanego czasu lokalizacja znacznika pokryje się z lokalizacją w skazaną w zleceniu ExpertWMS, do tabeli przejściowej zostanie zwrócona informacja o udanej realizacji zlecenia. W przeciwnym wypadku, po upływie czasu, Warehouse3D zwróci informację o braku realizacji zlecenia.

### 2.4. Implementacja klienta WWW i komunikacji asynchronicznej z serwerem.

Wykonana została aplikacja WWW w architekturze MVC. W widokach zaimplementowano skrypty Javascript, które realizują funkcje komunikacji z serwerem:

- ładowanie modeli 3D z serwera
- odbiór danych o lokalizacjach miejsc magazynowych i ich konwersję do układu współrzędnych modelu 3D
- asynchroniczne, z wykorzystaniem biblioteki Ajax, odbieranie danych o zmianie lokalizacji wózków widłowych i jednostek logistycznych pochodzących z systemu Mojix
- asynchroniczne, z wykorzystaniem biblioteki Ajax, pobieranie danych o zmianie lokalizacji jednostek logistycznych pochodzących z systemu ExpertWMS
- asynchroniczne wysyłanie do serwera zapytań o cechy wskazanego elementu i odbieranie tych informacji
- asynchroniczne uruchamianie inwentaryzacji jednostek logistycznych

Po stronie serwera został zaimplementowany moduł obserwujący zmiany w tabelach przejściowych ExpertWMS oraz zmiany lokalizacji wynikające z raportów systemu Mojix. Moduł ten komunikuje się asynchronicznie ze skryptami klienckimi działającymi w przeglądarce w celu aktualizacji widoku i przekazywania bieżących informacji użytkownikowi. Dzięki temu użytkownik ma do dyspozycji w swojej przeglądarce WWW zawsze aktualny stan magazynu. Pozwala to też na wizualizację ruchu wózków widłowych zgodnie z

bieżącymi informacjami przekazywanymi przez system lokalizacyjny Mojix.

## 2.5. Inwentaryzacja

Do realizacji funkcji inwentaryzacji został wykorzystany system Mojix. Uruchomienie inwentaryzacji następuje asynchronicznie przez skrypt klienta. Inwentaryzacja została zaimplementowana jako kontroler działający asynchronicznie na serwerze. W celu odczytania wszystkich jednostek logistycznych w magazynie przez system Mojix, w pierwszej kolejności moduł inwentaryzacji uruchamia za pomocą Web Service konfigurację Mojix włączającą jednocześnie wszystkie anteny nadawcze. Po włączeniu anten program gromadzi informacje o wszystkich odczytanych jednostkach logistycznych. Zakończenie inwentaryzacji jest inicjowane przez użytkownika. Dane o spisanych jednostkach logistycznych są przekazywane asynchronicznie do klienta i wyświetlane przez skrypt na stronie internetowej, skąd można je zapisać w formie listy.

## PODSUMOWANIE

Zadanie integracji systemów WMS oraz RTLS w nowoczesnym magazynie operatora logistycznego zrealizowane zostało w oparciu o autorskie rozwiązanie, którego podstawą było wykorzystanie bibliotek WebGL oraz narzędzi stosowanych do projektowania grafiki trójwymiarowej w grach komputerowych. Dzięki temu, zdefiniowane na wstępie wymagania dotyczące dostępności systemu dla użytkowników końcowych, łatwości utrzymania, aktualizacji oraz serwisu zostały zrealizowane. Trójwymiarowa wizualizacja, dostępna za pośrednictwem praktycznie dowolnej przeglądarki internetowej, pozwala na obserwację w czasie rzeczywistym aktualnej sytuacji w magazynie, ale pozwala również na sprawdzenie informacji logistycznych, wyszukanie dowolnej jednostki logistycznej a nawet inwentaryzację. Przygotowana aplikacja, poza wizualizacją, umożliwia wymianę danych pomiędzy systemami WMS i RTLS, pozwalając na wykorzystanie informacji o lokalizacji obiektów w systemie WMS oraz dostarczając danych źródłowych do systemu RTLS.

## BIBLIOGRAFIA

1. World Wide Web Consortium (W3C): HTML5 W3C Recommendation [online], wyd. 28 października 2014 r., dostępny w Internecie: <https://www.w3.org/TR/html5>
2. Ecma International: Standard ECMA-262 ECMAScript 2015 Language Specification [online], wyd. czerwiec 2015 r., dostępny w Internecie: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>
3. Blender Foundation: Blender Reference Manual [online], [dostęp: 22 września 2015], dostępny w Internecie: <https://www.blender.org/manual>
4. World Wide Web Consortium (W3C): Web Services Architecture [online], wyd. 11 stycznia 2004 r., dostępny w Internecie: <https://www.w3.org/TR/ws-arch>
5. The jQuery Foundation: jQuery API Documentation [online], [dostęp: 29 września 2015], dostępny w Internecie: <https://api.jquery.com>
6. World Wide Web Consortium (W3C): Extensible Markup Language (XML) 1.0 (Fifth Edition) [online], wyd. 26 listopada 2008 r., dostępny w Internecie: <https://www.w3.org/TR/xml>
7. Ecma International: Standard ECMA-404 The JSON Data Interchange Format [online], wyd. październik 2013, dostępny w Internecie: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>
8. Microsoft Corporation: C# Reference [online], [dostęp: sierpień 2015], dostępny w Internecie: <https://msdn.microsoft.com/en-us/library/618ayhy6.aspx>

### Tracking the state and movement of logistics units in stock with virtual 3d model

*The task of WMS and RTLS systems integration in a modern warehouse of logistics operator was conducted on a proprietary solution, which is based on the use of a WebGL library and the tools usually used to design three-dimensional graphics in computer games. As a result, the outset requirements for system defined as good availability for end users, ease of maintenance, updates and maintenance were successfully realized. Three-dimensional visualization, available virtually through any web browser allows you to see in the real time the current situation inside warehouse, but also allows you to check the logistics information, search for any logistics unit and even perform inventory process. Application, in addition to visualization, enables the exchange of data between the WMS and RTLS, allowing for the use of information about the location of objects in WMS and providing source data to the RTLS system.*

Autorzy:

dr inż. **Michał Grabia** - Instytut Logistyki i Magazynowania, Laboratorium Technologii Identyfikacyjnych  
mgr inż. **Tomasz Markowski** – Instytut Logistyki i Magazynowania, Laboratorium Technologii Identyfikacyjnych  
mgr **Klaudiusz Borowiak** - Instytut Logistyki i Magazynowania, Centrum Informatyki