QT PLANNER

*Planner akademicki obsługujący format iCal*

***Autorzy:***

***Anna Krzak***

***Łukasz Rylik***

***Piotr Suckert***

Spis treści

[1. Planowanie 3](#_Toc361258474)

[1.1. Cel i zakres działania programu 3](#_Toc361258475)

[1.2. Opis systemu biznesowego 3](#_Toc361258476)

[1.2.1. Aktorzy systemu biznesowego 3](#_Toc361258477)

[1.2.2. Lista funkcji realizowanych przez projektowany system informatyczny 3](#_Toc361258478)

[1.3. Format iCalendar 3](#_Toc361258479)

[1.3.1. Opis standardu 3](#_Toc361258480)

[1.3.2. Modyfikacja standardu 3](#_Toc361258481)

[1.3.3. Przykład użycia standardu 4](#_Toc361258482)

[1.4. Diagram przypadków użycia 4](#_Toc361258483)

[1.4.1. Opis zdarzeń aplikacji z punktu widzenia aktorów 5](#_Toc361258484)

[1.5. Perspektywa pojęciowa 5](#_Toc361258485)

[2. Analiza 6](#_Toc361258486)

[2.1. Model przypadków użycia 6](#_Toc361258487)

[2.1.1. Przypadek użycia Wyświetl wydarzenia 6](#_Toc361258488)

[2.1.2. Przypadek użycia Ustawienia 6](#_Toc361258489)

[2.1.3. Przypadek użycia Aktualizuj 6](#_Toc361258490)

[2.1.4. Przypadek użycia Sprawdź aktualizacje 6](#_Toc361258491)

[2.2. Perspektywa specyfikacyjna 7](#_Toc361258492)

[2.3. Minimalne wymagania środowiskowe: 7](#_Toc361258493)

[2.3.1. Wersja desktop 7](#_Toc361258494)

[2.3.2. Docelowa wersja mobilna 7](#_Toc361258495)

[3. Implementacja 8](#_Toc361258496)

[3.1. Diagram stanów 8](#_Toc361258497)

[3.2. Perspektywa implementacyjna – diagram klas 9](#_Toc361258498)

[4. Testowanie 10](#_Toc361258499)

[5. Umowa licencyjna: 10](#_Toc361258500)

# Planowanie

## Cel i zakres działania programu

Celem naszego programu jest proste i przejrzyste przedstawienie aktualnego planu zajęć. Aplikacja, na życzenie, automatycznie powiadamia użytkownika o zmianach przez porównanie planu z poprzednią wersją (wiek pliku danych), użytkownik aktualizuje kalendarz ręcznie lub automatycznie. Dodatkowo Użytkownik ma możliwość filtrowania danych - np. pomijanie danego przedmiotu.

## Opis systemu biznesowego

Aplikacja skierowana do studentów, uczniów i wykładowców, których uczelnia/szkoła udostępnia plan zajęć w postaci elektronicznej (wymagany format iCal). Oprogramowanie jest dystrybuowane w postaci zamkniętej aplikacji na licencji Freeware.

### Aktorzy systemu biznesowego

Za aktora uważamy (a)user’a, który za pomocą aplikacji komunikuję się z (b)silnikiem przetwarzającym dane kalendarzowe w aplikacji. Trzecim aktorem systemu jest (c)serwer publikujący dane. W szczególnym przypadku aplikacja może działać w oparciu o lokalny plik danych, z pominięciem komunikacji internetowej.

### Lista funkcji realizowanych przez projektowany system informatyczny

* wyświetlanie planu dla użytkownika
* aktualizacja danych
* ustawianie filtra

## Format iCalendar

### Opis standardu

iCalendar to standard dla wymiany informacji kalendarzowych opracowanym przez IETF. Jest on czasami nazywany iCal, od nazwy aplikacji firmy Apple, która jest jedną z implementacji tego standardu.

iCalendar pozwala użytkownikom wymieniać między sobą dane kalendarzowe, czyli informacje o terminach (np. datach spotkań – w celu ustalenia pasującego czasu i miejsca) i zadaniach do wykonania – np. przesyłając je za pośrednictwem poczty elektronicznej. Odbiorca wiadomości z danymi iCalendar (za pomocą odpowiedniego programu) może natychmiast łatwo odpowiedzieć – zaakceptować termin lub zaproponować inny.

### Modyfikacja standardu

Założenia niezbędne do poprawnego odczytu przez aplikację:

* W polu tytułowym wydarzenia używamy zapisu rodzaj=przedmiot, np. W=Fizyka co oznacza odpowiednio wykład z Fizyki
* W polu miejsce wydarzenia używamy zapisu miejsce=sala, np. WSTI=zach5, co oznacza odpowiednio miejsce i sale
* W polu opisu wydarzenia wstawiamy prowadzącego

### Przykład użycia standardu

Przykładowy zapis w formacie iCal:

BEGIN:VEVENT

DTSTART;TZID=Europe/Warsaw:20130512T140000

DTEND;TZID=Europe/Warsaw:20130512T153000

RRULE:FREQ=MONTHLY;UNTIL=20130711T215959Z;BYMONTHDAY=12

EXDATE;TZID=Europe/Warsaw:20130612T140000

DTSTAMP:20130629T213812Z

UID:0h4rgfe8dt1d029dv1q8uh8dro@google.com

CREATED:20130629T195733Z

DESCRIPTION:Jarosław Karcewicz

LAST-MODIFIED:20130629T202019Z

LOCATION:ŚTZN=Aula

SEQUENCE:1

STATUS:CONFIRMED

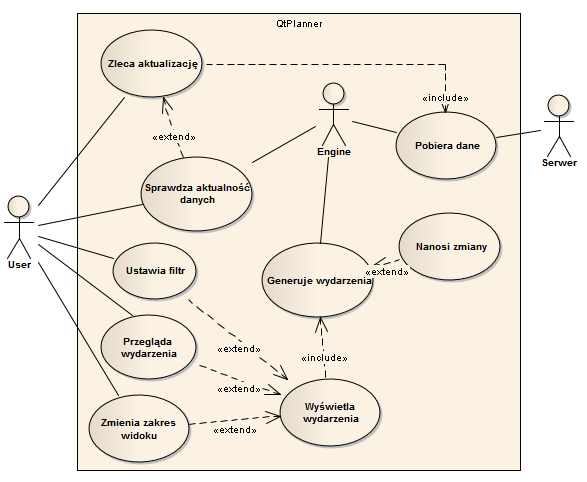
SUMMARY:W=Sieciowe systemy operacyjne

TRANSP:OPAQUE

END:VEVENT

Powyższy kod zostanie przetworzony na wydarzenie powtarzające się 12 dnia każdego miesiąca o godzinie 14-tej, od 2013-05-12 do 2013-07-11. W projektowanym planerze wyświetli się jako wykład z przedmiotu „Sieciowe systemy operacyjne”, prowadzony przez dra Jarosława Karcewicza i oznakowane będzie data ostatniej aktualizacji 2013-06-29, godzina 20:20. Pozostałe dane nie będą przetwarzane.

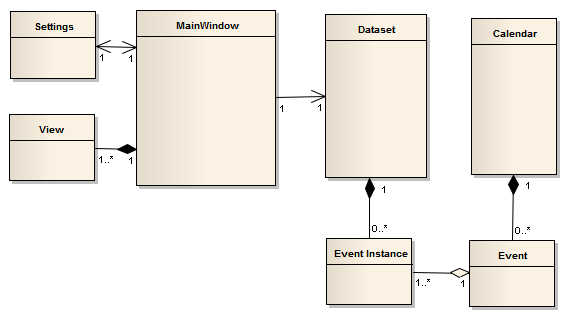
## Diagram przypadków użycia



### Opis zdarzeń aplikacji z punktu widzenia aktorów

* User
  + Wyświetlenie widoku wydarzeń
  + Zmiana zakresu widoku
  + Pytanie o aktualizację
  + Wymuszenie aktualizacji
  + Zmiana ustawień
* Serwer
  + Obsługa zapytania o plik danych
  + Obsługa zapytania o datę modyfikacji pliku
* Engine
  + Przetworzenie pliku danych na zestaw wydarzeń
  + Uwidocznienie zmian w przypadku aktualizacji
  + Wysłanie zapytania o plik danych

## Perspektywa pojęciowa



Trzonem części wizualnej aplikacji jest główne okno, reprezentowane przez klasę **MainWindow**, która przekazuje dane i agreguje jeden lub więcej widoków o różnych ramach czasowych. Klasa widoku (**View**), ma świadomość istnienia klasy rodzica i automatycznie informuje go o konieczności odświeżenia okna.

Okno główne pozwala również na zmianę ustawień aplikacji (klasa **Settings**), tu również komunikacja odbywa się dwustronnie, ponieważ zmiana ustawień może być powodem konieczności wygenerowania nowych danych.

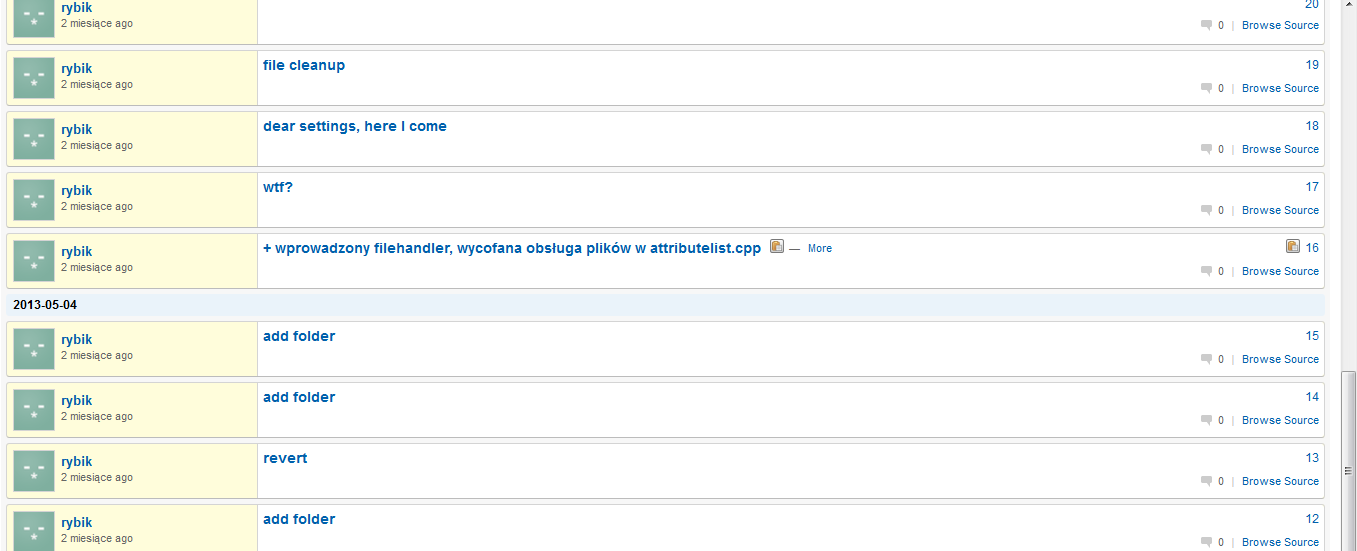
Danych do widoków dostarcza klasa **Dataset**, agregująca wszystkie wystąpienia (**EventInstance**) wszystkich wydarzeń w kalendarzu. Wystąpienia wydarzeń są generowane z obiektów wydarzeń (**Event**), przechowywanych w klasie **Calendar**.

Klasa **Calendar** pozyskuje dane przez sparsowanie pliku .ics wykorzystując klasę narzędziową.

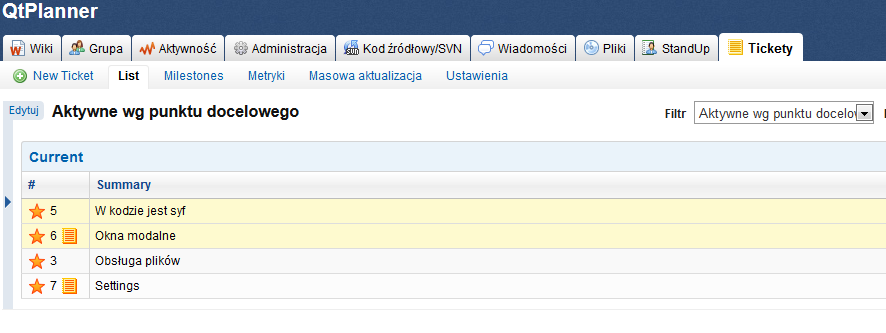
## Konserwacja kodu

W celu umożliwienia pracy zespołowi, pliki przechowywane są w zdalnym repozytorium na serwerze assembla.com ze wsparciem systemu kontroli wersji na silniku SVN

### System kontroli wersji oraz repozytorium – SVN



### System ticketów i zleceń – Assembla



# Analiza

## Model przypadków użycia

### Przypadek użycia Wyświetl wydarzenia

Na ekranie wyświetla się okno aplikacji zawierającej 3 zakładki (Dzień, Tydzień, Miesiąc) przedstawiające listę wydarzeń w odpowiednich ramach czasowych, licząc od bieżącej daty wprzód.

#### Po uruchomieniu programu

Po uruchomieniu, aplikacja wyświetla ostatnio wybrana zakładkę z przypadku ogólnego 2.1.1

#### Po zmianie zakładki

Po zmianie zakładki, aplikacja wyświetla nowo wybraną zakładkę i wybór ten będzie pamiętany do kolejnej zmiany, niezależnie ew. ponownego uruchomienia aplikacji.

#### Po powrocie z okna ustawień filtra

Po ustawieniu i zatwierdzeniu opcji filtrowania, aplikacja ponownie wygeneruje zestaw wydarzeń z uwzględnieniem filtra oraz odświeży listy wydarzeń na zakładkach, wracając do widoku 2.1.1

### Przypadek użycia Ustawienia

Po kliknięciu „Opcje” i kolejno „Ustawienia” użytkownikowi ukazuję okienko z możliwymi do wyboru ustawieniami:

* Ustawienia filtra wydarzeń
* Zezwolenie na cykliczne sprawdzanie aktualności danych (komunikacja z serwerem)

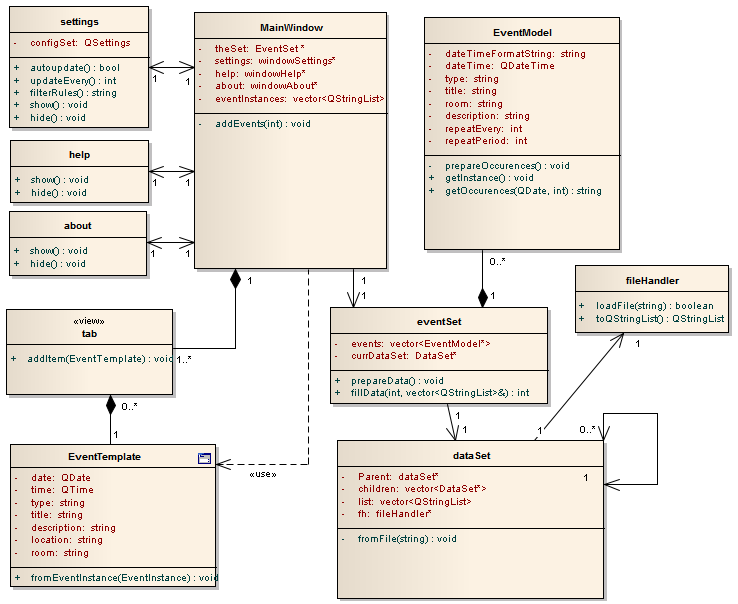
### Przypadek użycia Aktualizuj

Aplikacja łączy się z serwerem i pobiera plik z danymi kalendarza. Nanosi informacje o zmianach i generuje dane do widoków.

### Przypadek użycia Sprawdź aktualizacje

Aplikacja łączy się z serwerem, sprawdza datę pliku na serwerze. W razie potrzeby pobiera plik i sprawdza zmiany w poszczególnych wydarzeniach. Pyta o nadpisanie nowymi danymi.

## Perspektywa specyfikacyjna



## Minimalne wymagania środowiskowe:

### Wersja desktop

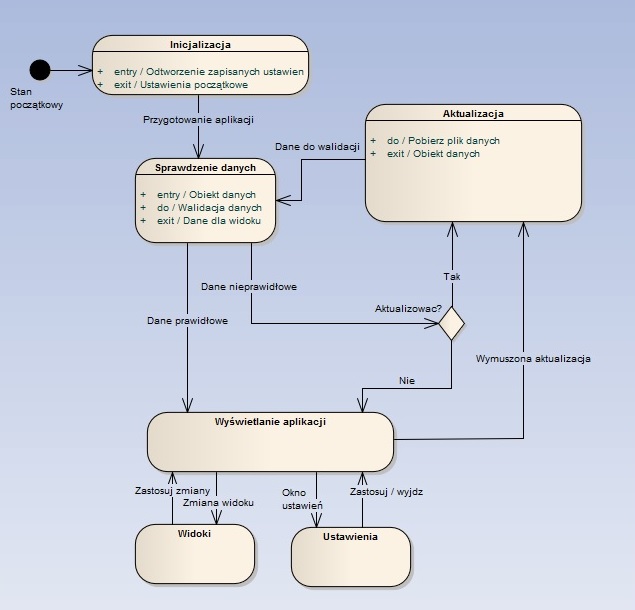
* System Windows 7
* Minimum 50MB miejsca na dysku
* Minimum 256MB pamięci RAM
* Biblioteka QT 5.x
* Pakiet C++ redistributable
* Ekran min 640x480 – 256 kolorów

### Docelowa wersja mobilna

* System iOS lub Android 4.x
* Minimum 15MB miejsca na dysku
* Ekran o rozdzielczości min. 320 x 480

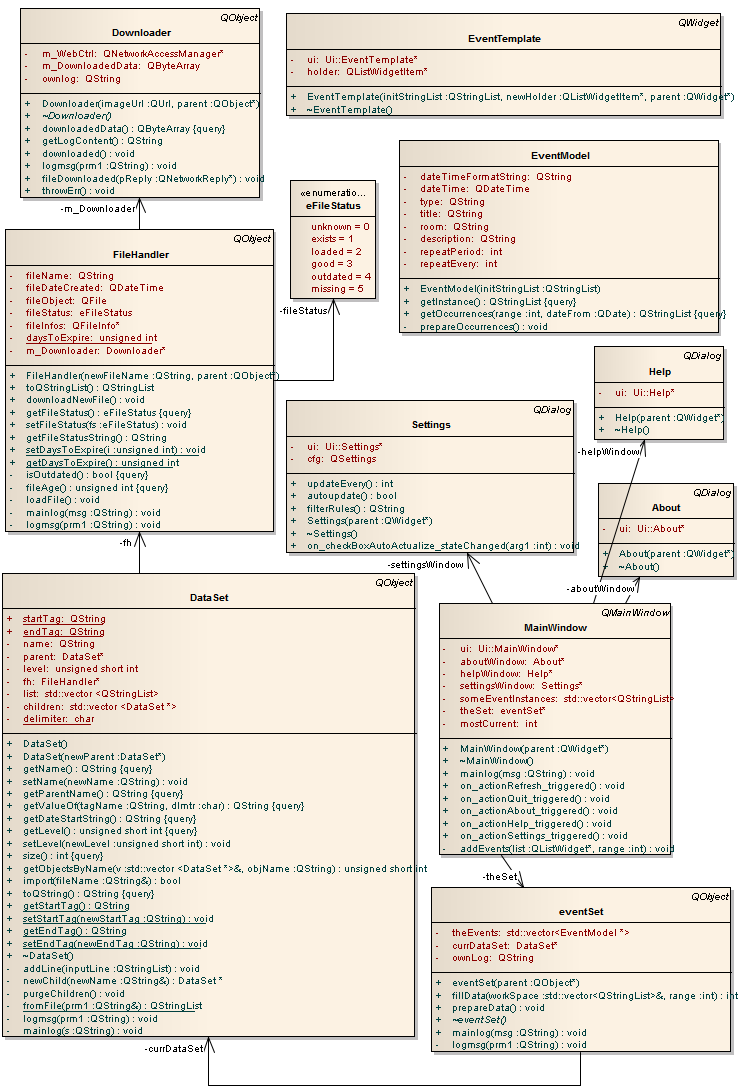
# Implementacja

## Diagram stanów



Rysunek przedstawia diagram stanów dla prezentowanego systemu.

## Perspektywa implementacyjna – diagram klas



# Testowanie

|  |  |  |  |
| --- | --- | --- | --- |
| C:\Users\Piotrek\Downloads\działa1.png | C:\Users\Piotrek\Downloads\działa2.png |  |  |
| Widok dnia | Widok tygodnia | Okno ustawień | Okno „about” |

# Umowa licencyjna:

1. **POSTANOWIENIA UMOWY**  
   Twórca oświadcza, iż przysługuje mu całość majątkowych praw autorskich do oprogramowania QT Planner (zwanego dalej Oprogramowaniem), jak również iż majątkowe prawa autorskie do Oprogramowania nie zostały obciążone na rzecz jakichkolwiek osób trzecich w sposób uniemożliwiający zawarcie niniejszej Umowy.
2. Twórca oświadcza, że korzystanie przez Nabywcę z Oprogramowania nie będzie naruszało jakichkolwiek praw własności intelektualnej osób trzecich. W przeciwnym wypadku Twórca zobowiązuje się do pokrycia wszelkich odszkodowań i kosztów w związku z dochodzeniem przez osoby trzecie takich roszczeń od Nabywcy.
3. Twórca udziela Nabywcy niewyłącznej licencji na korzystanie z Oprogramowania (Licencja) na polach eksploatacji obejmujących:
   1. w zakresie utworów będących programami komputerowymi (w tym w szczególności pliki wykonywalne, interfejsy, skrypty itp.):
      1. trwałe lub czasowe zwielokrotnienie programu komputerowego w całości lub w części jakimikolwiek środkami i w jakiejkolwiek formie;
      2. tłumaczenia, przystosowywania, zmiany układu lub jakichkolwiek innych zmian w programie komputerowym, z zachowaniem praw osoby, która tych zmian dokonała,
      3. rozpowszechnianie, w tym użyczenie lub najem, programu komputerowego lub jego kopii,
   2. w zakresie utworów nie będących programami komputerowymi, a w szczególności w zakresie dokumentacji Oprogramowania:
      1. w zakresie utrwalania i zwielokrotniania utworu - wytwarzanie dowolną techniką egzemplarzy utworu, w tym techniką drukarską, reprograficzną, zapisu magnetycznego oraz techniką cyfrową;
      2. w zakresie obrotu oryginałem albo egzemplarzami, na których utwór utrwalono - wprowadzanie do obrotu, użyczenie lub najem oryginału albo egzemplarzy;
      3. w zakresie rozpowszechniania utworu w sposób inny niż określony w pkt poprzedzającym - publiczne wykonanie, wystawienie, wyświetlenie, odtworzenie oraz nadawanie i reemitowanie, a także publiczne udostępnianie utworu w taki sposób, aby każdy mógł mieć do niego dostęp w miejscu i w czasie przez siebie wybranym.
4. Licencja obejmuje prawo Nabywcy do korzystania z Oprogramowania na terytorium Polski oraz poza jej granicami – wedle wyboru Nabywcy.
5. Nabywca będzie uprawniony do przeniesienia Licencji na rzecz osób trzecich oraz udzielania sublicencji w odniesieniu do Oprogramowania oraz Licencji, pod warunkiem iż taka osoba trzecia zaakceptuje warunki niniejszej Umowy.
6. Wraz z udzieleniem Licencji, w ramach wynagrodzenia wskazanego w punkcie 9, Twórca udziela Nabywcy prawa do zezwalania na wykonywanie zależnych praw autorskich.
7. Twórca gwarantuje, że osoby uprawnione z tytułu osobistych praw autorskich do Oprogramowania nie będą wykonywać takich praw w stosunku do Nabywcy, jego następców prawnych i ich licencjobiorców.
8. Wraz z Oprogramowaniem Twórca przekaże Nabywcy kompletne kody źródłowe Oprogramowania wraz z ich opisem oraz pełną dokumentację Oprogramowania. Dokumentacja zawierać będzie w szczególności wszelkie informacje pozwalające na samodzielne korzystanie i dokonywanie dalszych zmian programów komputerowych przez Nabywcę.
9. Z tytułu udzielenia Licencji Nabywca zapłaci na rzecz Twórcy jednorazowe wynagrodzenie w wysokości [wysokość wynagrodzenia] (słownie – [kwota słownie]).
10. Wynagrodzenie wskazane w punkcie 9 jest wynagrodzeniem netto, do którego Twórca doliczy podatek VAT w obowiązującej wysokości. Wynagrodzenie będzie płatne w terminie [termin zapłaty wynagrodzenia] dni od daty podpisania niniejszej umowy przelewem na rachunek bankowy Twórcy o numerze [numer rachunku bankowego].
11. Wraz z udzieleniem Licencji, w ramach wynagrodzenia wskazanego w punkcie 9, Twórca przenosi własność (sprzedaż) nośników, na których Oprogramowanie zostało utrwalone.
12. Udzielenie Licencji oraz przejście na Nabywcę własności nośników wskazanych w punkcie 11 nastąpi z chwilą zawarcia niniejszej Umowy.
13. Niniejsza Umowa zostaje zawarta na czas nieokreślony lat i może zostać wypowiedziana przez każdą ze Stron nie wcześniej, niż po upływie okresu 5 lat od jej zawarcia. Okres wypowiedzenia niniejszej Umowy wynosić będzie 5 lat ze skutkiem na koniec roku kalendarzowego.
14. W sprawach nieuregulowanych niniejszą Umową stosuje się przepisy prawa polskiego. Wszelkie spory pomiędzy Stronami rozstrzygane przez sądy właściwe miejscowo dla siedziby Nabywcy.
15. Niniejsza Umowa została sporządzona w dwóch jednobrzmiących egzemplarzach, po jednym dla każdej ze Stron.