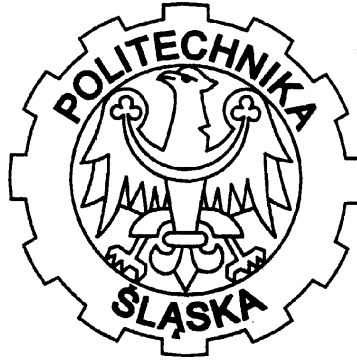


Politechnika Śląska w Gliwicach  
Wydział Automatyki, Elektroniki i Informatyki



# Projektowanie przemysłowych systemów komputerowych

*Projekt – Sprawozdanie*

*Gas Analyzer*

**Autorzy: Damian Karbowski, Grzegorz Powąła**  
**Informatyka, SSM3, grupa ISP1**  
**Prowadzący: dr inż. Jacek Stój**  
**Konsultant: mgr inż. Tomasz Kress**

11 lipca 2013

# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>2</b>
1.1	Geneza . . . . .	2
1.2	Temat . . . . .	2
1.3	Stanowisko . . . . .	2
1.3.1	Stanowisko prototypowe . . . . .	2
1.3.2	Stanowisko docelowe . . . . .	3
1.4	Analiza tematu . . . . .	3
1.5	Założenia . . . . .	4
1.6	Plan pracy . . . . .	5
<b>2</b>	<b>Specyfikacja wewnętrzna</b>	<b>6</b>
2.1	Baza danych . . . . .	7
<b>3</b>	<b>Instrukcja użytkownika</b>	<b>8</b>
3.0.1	Ekran powitalny . . . . .	8
<b>4</b>	<b>Podsumowanie</b>	<b>12</b>
4.1	Perspektywy rozwoju . . . . .	12
4.2	Wnioski . . . . .	12
<b>5</b>	<b>Bibliografia</b>	<b>13</b>
<b>6</b>	<b>Spis rysunków, tablic i kodów źródłowych</b>	<b>14</b>
6.1	Spis rysunków . . . . .	14
6.2	Spis tablic . . . . .	14
6.3	Spis kodów źródłowych . . . . .	14
<b>7</b>	<b>Załączniki</b>	<b>15</b>

# 1 Wstęp

## 1.1 Geneza

Tematem projektu, którego dotyczy to sprawozdanie jest: „Gas Analyzer”. Pomysł na projekt pojawił się w wyniku nawiązania przez nas współpracy z Zakładem Kotłów i Wytwornic Pary, a dokładnie Panem Tomaszem Kressem.

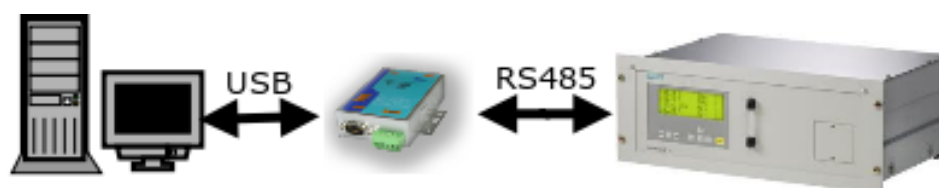
## 1.2 Temat

Głównymi celami pracy było napisanie oprogramowania umożliwiającego gromadzenie danych pomiarowych z kilku urządzeń firmy Siemens.

## 1.3 Stanowisko

W czasie realizacji projektu wykorzystywaliśmy 2 różne stanowiska. W pierwszej fazie projektu korzystaliśmy z uproszczonego stanowiska, które wyglądało jak na Rysunku 1. W dalszej fazie projektu, kiedy mieliśmy już przygotowaną i przetestowaną wersję podstawową współpracującą z jednym urządzeniem pomiarowym rozpoczęliśmy pracę na stanowisku docelowym składającym się z 4 urządzeń, które wyglądało jak na Rysunku 2.

### 1.3.1 Stanowisko prototypowe

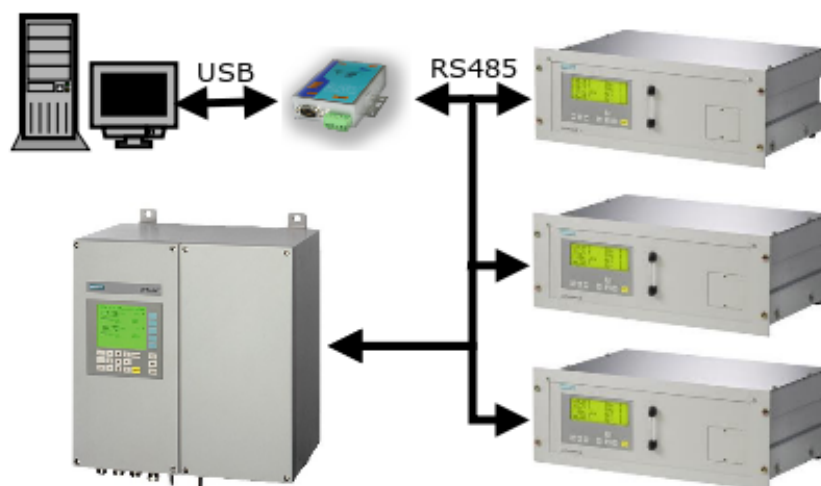


Rysunek 1: Schemat stanowiska prototypowego

Na potrzeby realizacji projektu stworzono stanowisko laboratoryjne, którego schemat przedstawia Rysunek 1. Składa się ono z:

- Komputera,
- Konwertera ATC-850,
- ULTRAMAT 23.

Komputery na, których powstała wersja rozwojowa projektu pracowały na systemach operacyjnych Linux Ubuntu w wersji 32 oraz 64 bitowej. Do połączenia komputera z urządzeniem ULTRAMAT 23 zastosowano izolowany konwerter USB do RS-232/422/485, moduł ATC-850 jest automatycznie wykrywany i instalowany jako standardowy port COM. Stosowane w tej fazie projektu urządzenie pomiarowe potrafiło mierzyć zawartość  $CO$ ,  $CO_2$ ,  $NO$  oraz  $O_2$ .



Rysunek 2: Schemat stanowiska docelowego

### 1.3.2 Stanowisko docelowe

Docelowo zrealizowany projekt ma być uruchamiany na stanowisku, którego schemat przedstawia Rysunek 2. Składa się ono z:

- Komputera,
- Konwertera ATC-850,
- 3x ULTRAMAT 23,
- ULTRAMAT 6.

Stanowisko docelowe różni się od stanowiska prototypowego po pierwsze systemem operacyjnym, który pracuje na komputerze i jest to Windows XP. Po drugie stanowisko docelowe posiada więcej urządzeń pomiarowych, a jest ich dokładnie cztery i mierzą wartości przedstawione w Tabeli 1.

Urządzenie	Wielkości mierzone
ULTRAMAT 6	$NH_3[vp\text{m}]$
ULTRAMAT 23	$CH_4[\%]$ , $CO[\%]$ , $CO_2[\%]$ , $O_2[\%]$
ULTRAMAT 23	$CO[ppm]$ , $CO_2[\%]$ , $NO[ppm]$ , $O_2[\%]$
ULTRAMAT 23	

Tablica 1: Urządzenia docelowe wraz z wartościami mierzonymi

## 1.4 Analiza tematu

Analiza tematu polegała przede wszystkim na zapoznaniu się z dokumentacjami urządzeń [1, 2]. Szczególnie istotnym, a w zasadzie najważniejszym punktem całej analizy były interfejsy i protokoły dostępne w obu typach urządzeń oraz w ewentualnych kolejnych urządzeniach tego producenta.

Analiza pozwoliła wytypować do dalszej analizy dwa protokoły:

1. PROFIBUS-DP/-PA
2. ELAN Network

Szczegółowa analiza rozwiązań opartych o oba protokoły komunikacyjne w dokumentacjach producenta [3, 4, 5] pozwoliła ustalić, że w przypadku PROFIBUSA można zastosować sterownik przemysłowy wyposażony w odpowiednie złącze komunikacyjne lub rozszerzony o odpowiedni moduł. Niestety jest to rozwiązanie drogie i mało elastyczne. Teoretycznie funkcjonują na rynku przejściówki PROFIBUS  $\Leftrightarrow$  USB, ale nie udało nam się znaleźć niczego aktualnego i godnego rozważań. Najistotniejszym punktem analizy był protokół ELAN Network, dostępna do niego dokumentacja [3] uświadomiła nam, że spełnia wszystkie nasze wymagania, a sprzęt potrzebny do uruchomienia jest tani i ogólnodostępny. Własna implementacja tego protokołu na podstawie dokumentacji i testów pozwala na pełną elastyczność i dopasowanie do naszych potrzeb. Poznanie tych podstaw pozwoliło dobrać technologię odpowiednią do realizacji projektu zgodnie z założeniami. Ostatecznie wybór padł na ELAN Network oraz stworzenie własnego oprogramowania w Javie.

## 1.5 Założenia

Oprogramowanie do zbierania danych pomiarowych powinno zostać stworzone przy użyciu technologii pozwalającej działać na różnych systemach operacyjnych bez skomplikowanych zabiegów. Funkcjonalności wchodzące w skład projektu, to:

- wykorzystanie jednego z dostępnych w urządzeniach protokołów,
- automatyczne wykrywanie podłączonych urządzeń,
- zarządzanie użytkownikami, tytułami naukowymi, miejscami, obiektami itd.
- wizualizacja bieżących pomiarów,
- wykrywanie i sygnalizacja problemów z urządzeniem,
- zapisywanie bieżących pomiarów ze wszystkich urządzeń jednocześnie,
- regulowany krok zapisu pomiarów do bazy,
- możliwość dodania komentarza do zapisywanego pomiaru,
- generowanie raportu z pomiaru jako plik arkusza kalkulacyjnego,
- generowanie raportu z pomiaru jako plik do wydruku z wynikami np. format PDF,
- konfiguracja nazwa urządzeń widocznych w aplikacji,
- ustawianie precyzji pomiarów dla danej wielkości mierzonej.

Powyżej zostały wymienione założenia podstawowe, jednak autorzy nie wykluczają zrealizowania dodatkowych zadań, które nie zostały zamieszczone w pierwotnej koncepcji realizacji projektu.

## 1.6 Plan pracy

Realizacja projektu została podzielona na następujące etapy:

- Przygotowanie stanowiska, zebranie odpowiednich materiałów i literatury,
- Analiza wymagań funkcjonalnych aplikacji,
- Projektowanie struktury oprogramowania i interfejsów wymiany danych,
- Implementacja,
- Testowanie i uruchamianie,
- Przedstawienie projektu i ewentualne korekty.

Powyższy plan pracy stanowił dla autorów wyznacznik kolejnych działań. Jednak powszechnie wiadomo, że w praktyce poszczególne punkty są wymienne i wpływają na siebie wzajemnie. Dodatkowo na potrzeby realizacji projektu powstał szczegółowy plan wraz z terminami oraz osobami odpowiedzialnymi za poszczególne zadania przedstawiony w Tabeli 2

Termin	Osoba	Zadanie
11.03 – 17.03	Wszyscy	Wybór tematu.
18.03 – 20.03	Wszyscy	Określenie celu i zakresu, przygotowanie harmonogramu, podział zadań.
21.03	Wszyscy	Analiza sprzętu oraz dokumentacji.
22.03 – 23.03	Wszyscy	Analiza oraz porównanie dopuszczalnych rozwiązań z wykorzystaniem protokołu ELAN lub Profibus.
24.03 – 25.03	Wszyscy	Analiza wybranego protokołu oraz potrzebnego sprzętu do połączenia z komputerem (np. konwerter RS-485 $\Leftrightarrow$ USB ).
25.03 – 02.04	Wszyscy	Implementacja wybranych fragmentów protokołu.
29.03 – 17.04	Damian	Przygotowanie podstawowej wersji interfejsu użytkownika, umożliwiającej przetestowanie implementacji protokołu.
03.04 – 18.04	Grzegorz	Rozwinięcie podstawowej wersji protokołu – interpretacja i przetwarzanie odbieranych danych.
20.04 – 01.05	Grzegorz	Stworzenie modelu bazy danych i połączenia ORM.
19.04 – 05.05	Damian	Wykrycie i wizualizacja struktury sieci oraz odbieranych danych.
03.05 – 06.05	Damian	Generowanie PDF.
04.05 – 10.05	Grzegorz	Generowanie XLS.
13.05 – 22.05	Grzegorz	Zarządzanie ustawieniami urządzeń.
27.05 – 05.06	Damian	Poprawki w GUI.
01.06 – 08.06	Wszyscy	Instrukcja użytkownika oraz dokumentacja.

Tablica 2: Szczegółowy plan pracy wraz z harmonogramem i osobami odpowiedzialnymi

## 2 Specyfikacja wewnętrzna

Oprogramowanie zostało stworzone w całości Javie. Dla ułatwienia kompilacji, zarządzanie zależnościami oraz wersjami zastosowano Apache Maven, które jest narzędziem automatyzującym budowę oprogramowania. Najważniejszymi bibliotekami wykorzystywanymi w projekcie są:

1. RXTX

W zasadzie najważniejsza biblioteka w całym projekcie wykorzystywana do komunikacji poprzez port szeregowy.

2. SWT: The Standard Widget Toolkit

Biblioteka wykorzystana do stworzenia GUI (graficzny interfejs użytkownika) aplikacji. Dostarcza sporą ilość gotowych komponentów, które trzeba odpowiednio oprogramować. Biblioteka jest zależna od architektury i systemu operacyjnego co zostało uwzględnione jako profile Mavena.

3. iText

Biblioteka iText służy głównie do tworzenia dokumentów PDF przez programy napisane w Javie. Jej dodatkowe możliwości to obsługa formatów RTF i HTML. Biblioteka została zastosowana do generowania raportu z pomiaru w formacie PDF.

4. Apache POI

Zbiór bibliotek do obsługi plików w formacie Microsoft OLE 2 z poziomu języka programowania Java. W naszym projekcie wykorzystujemy tylko HSSF, który umożliwia obsługę plików Microsoft Excel. Biblioteka została zastosowana do generowania raportu z pomiaru w formacie XLS.

5. Hibernate

Framework do realizacji warstwy dostępu do danych (ang. persistence layer). Zapewnia on przede wszystkim translację danych pomiędzy relacyjną bazą danych, a światem obiekowym (ang. O/R mapping). Opiera się na wykorzystaniu opisu struktury danych za pomocą języka XML, dzięki czemu można żłutować obiekty, stosowane w obiektowych językach programowania, takich jak Java bezpośrednio na istniejące tabele bazy danych.

6. dom4j

dom4j to kolejny projekt typu open-source. Jego API oparte jest na interfejsach. Korzysta z parsera SAX. Jego motywacja jest podobna jak JDOM: prostsze i lżejsze od DOM API, stworzone specjalnie dla języka Java. W projekcie wykorzystywany do odczytu oraz zapisu pliku zawierającego konfigurację urządzeń oraz precyzję pomiarów.

Maven umożliwia stworzenie profili, które wykonują różne zadania lub pozwalają rozróżnić odrębne niezależne przebiegi kompilacji. W naszym projekcie wykorzystaliśmy je do pobrania i dołączenia do pliku końcowego biblioteki SWT w wersji dla wybranego systemu operacyjnego i architektury. Dostępne profile Mavena:

1. win32 – Windows 32-bitowy
2. win64 – Windows 64-bitowy
3. lin32 – Linux 32-bitowy
4. lin64 – Linux 64-bitowy
5. mac32 – Mac OSX Cocoa 32-bitowy
6. mac64 – Mac OSX Cocoa 64-bitowy

Struktura projektu w formie diagramu:

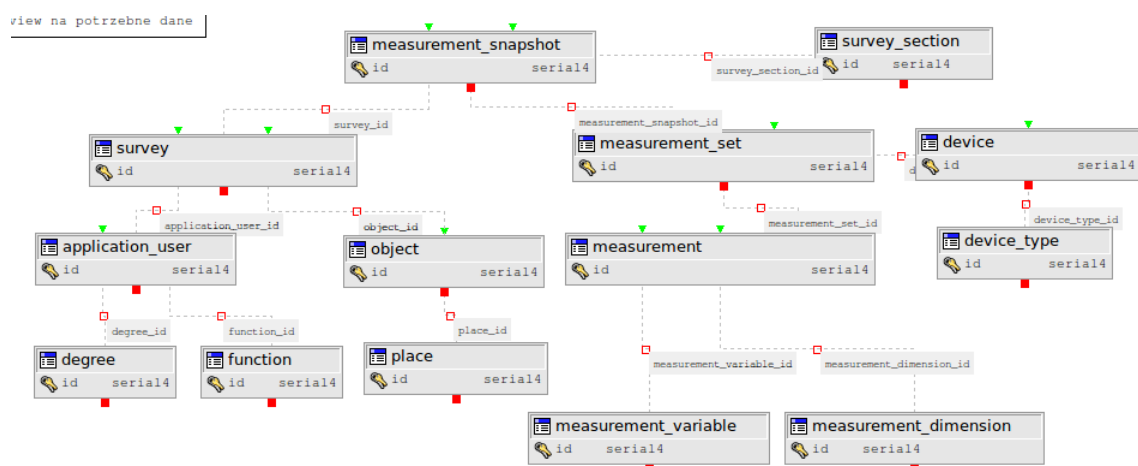
```
[node distance=2cm] (GasAnalyzer) [abstract, rectangle split, rectangle split parts=2]
    GasAnalyzer secondwersja 0.1.0 ; (AuxNode01) [text width=4cm, below=of
    GasAnalyzer] ; (ELANNetwork) [abstract, rectangle split, rectangle split parts=2,
    left=of AuxNode01] ELANNetwork secondwersja 0.1.0 ; (GasAnalyzerGUI)
    [abstract, rectangle split, rectangle split parts=2, right=of AuxNode01]
GasAnalyzerGUI secondwersja 0.1.0 ; [myarrow] (ELANNetwork.north) -- ++(0,0.8)
    -- (GasAnalyzer.south); [line] (ELANNetwork.north) -- ++(0,0.8) --
    (GasAnalyzerGUI.north);
```

Rysunek 3: Struktura projektu

## 2.1 Baza danych

W programie wykorzystujemy bazę PostgreSQL. Do obsługi w aplikacji wykorzystujemy omówioną już wcześniej bibliotekę Hibernate. Schemat bazy danych został stworzony w pgDesignerze i wygląda jak na Rysunku 4.





Rysunek 4: Schemat bazy danych

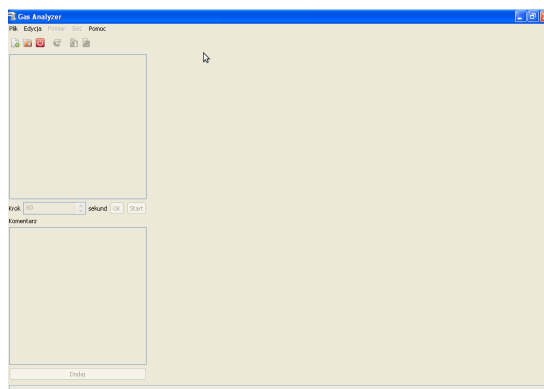
### 3 Instrukcja użytkownika



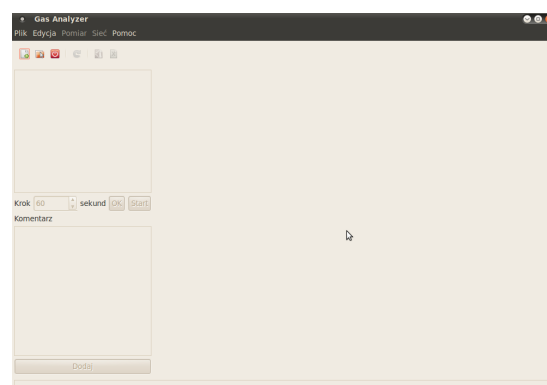
Rysunek 5: Okno ładowania

#### 3.0.1 Ekran powitalny

Bezpośrednio po uruchomieniu wizualizacji użytkownik zobaczy ekran powitalny taki jak na Rysunku 13 zawierający informacje o



(a) Windows



(b) Linux

Rysunek 6: Okno główne

(a) Windows

(b) Linux

Rysunek 7: Dodawanie nowego pomiaru

(a) Windows

(b) Linux

Rysunek 8: Okno wyboru daty

(a) Windows

(b) Linux

Rysunek 9: Dodawanie nowego miejsca

(a) Windows

(b) Linux

Rysunek 10: Błąd przy dodawaniu nowego miejsca

(a) Windows

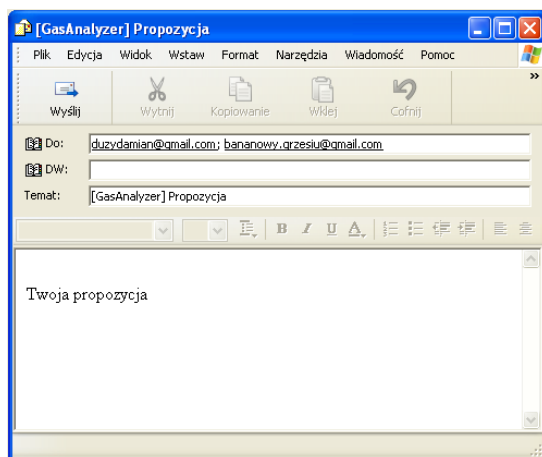
(b) Linux

Rysunek 11: Edytowanie istniejącego miejsca

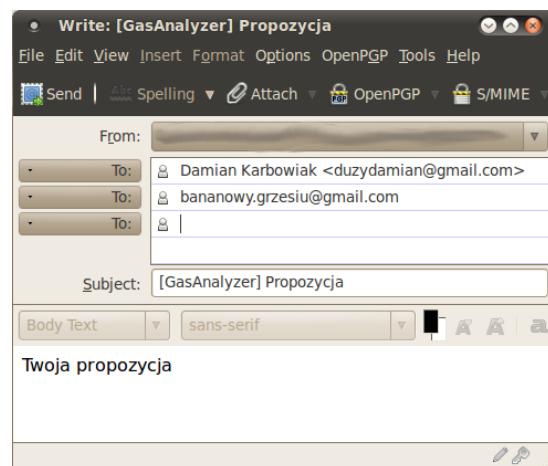
(a) Windows

(b) Linux

Rysunek 12: Otwieranie istniejącego pomiaru



(a) Windows



(b) Linux

Rysunek 13: Wysyłanie sugestii

## **4 Podsumowanie**

### **4.1 Perspektywy rozwoju**

Projekt jest bardzo perspektywiczny głównie dlatego, że w bieżącej części została zaimplementowana tylko znikoma część protokołu ELAN, a co za tym idzie można cały proces pomiarowy uskutecznić, uprościć oraz zautomatyzować w jeszcze większym stopniu.

### **4.2 Wnioski**

Głównymi celami pracy było napisanie oprogramowania gromadzącego dane z urządzeń pomiarowych.

## 5 Bibliografia

Literatura, która została wykorzystana przez autorów w czasie powstawania projektu, którą opisuje niniejsza dokumentacja.

- [1] Dokumentacja producenta: „*ULTRAMAT 23 Analizatory gazu dla tlenu i gazów pochłaniających podczerwień*”, luty 2001
- [2] Dokumentacja producenta: „*ULTRAMAT 6, OXYMAT6 Analizatory dla gazów absorbujących podczerwień i tlenu*”, styczeń 2001
- [3] Dokumentacja producenta: „*ELAN Interface Description*”, sierpień 2006
- [4] Dokumentacja producenta: „*STEP 7 AGA Gas Library - Applications Tools*”, listopad 2010
- [5] Dokumentacja producenta: „*GasAnalyzersCommunication*”, 2012

## 6 Spis rysunków, tablic i kodów źródłowych

### 6.1 Spis rysunków

Rysunek 1:	Schemat stanowiska prototypowego . . . . .	2
Rysunek 2:	Schemat stanowiska docelowego . . . . .	3
Rysunek 3:	Struktura projektu . . . . .	7
Rysunek 4:	Schemat bazy danych . . . . .	7
Rysunek 5:	Okno ładowania . . . . .	8
Rysunek 6:	Okno główne . . . . .	8
Rysunek 7:	Dodawanie nowego pomiaru . . . . .	9
Rysunek 8:	Okno wyboru daty . . . . .	9
Rysunek 9:	Dodawanie nowego miejsca . . . . .	9
Rysunek 10:	Błąd przy dodawaniu nowego miejsca . . . . .	10
Rysunek 11:	Edytowanie istniejącego miejsca . . . . .	10
Rysunek 12:	Otwieranie istniejącego pomiaru . . . . .	10
Rysunek 13:	Wysyłanie sugestii . . . . .	11

### 6.2 Spis tablic

Tablica 1:	Urządzenia docelowe wraz z wartościami mierzonymi . . . . .	3
Tablica 2:	Szczegółowy plan pracy wraz z harmonogramem i osobami odpowiedzialnymi . . . . .	5

### 6.3 Spis kodów źródłowych



## 7 Załączniki

- Oświadczenie o autorstwie,
- Płyta CD, na której znajdują się:
  - Pliki wykonywalne w wersji dla:
    - \* win32 – Windows 32-bitowy
    - \* win64 – Windows 64-bitowy
    - \* lin32 – Linux 32-bitowy
    - \* lin64 – Linux 64-bitowy
    - \* mac32 – Mac OSX Cocoa 32-bitowy
    - \* mac64 – Mac OSX Cocoa 64-bitowy,
  - Zdjęcia projektu,
  - Filmy prezentujące działanie projektu.