

Politechnika Śląska  
Wydział Automatyki, Elektroniki i Informatyki  
Informatyka



Mateusz Stolecki

# **Konwerter danych pochodzących terminali kas fiskalnych.**

projekt inżynierski

kierujący pracą: dr Karolina Nurzyńska

Gliwice, 30 listopada 2017



# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>1</b>
<b>2</b>	<b>Analiza tematu</b>	<b>3</b>
2.1	Wprowadzenie . . . . .	3
2.1.1	Procesor danych z terminali kas fiskalnych Aloha . . . . .	3
2.1.2	Umiejscowienie w systemie 360iQ . . . . .	3
2.1.3	Dane wejściowe . . . . .	5
2.1.4	Typy zdarzeń generowanych przez kasę Aloha . . . . .	6
2.1.5	Etap wstępnego przetwarzania . . . . .	9
2.1.6	Etap właściwego przetwarzania . . . . .	9
2.1.7	Przypadki niestandardowe . . . . .	10
2.2	Opis technologii i wykorzystanych narzędzi . . . . .	12
2.2.1	Środowisko pracy i język programowania . . . . .	12
2.2.2	Baza danych . . . . .	12
2.2.3	Moduły wspierające . . . . .	12
<b>3</b>	<b>Wymagania</b>	<b>15</b>
3.1	Wymagania funkcjonalne . . . . .	15
3.1.1	Moduł wstępnego przetwarzania danych - PreParser . . . . .	15
3.1.2	Moduł właściwego przetwarzania danych - Parser . . . . .	15
3.2	Wymagania нефункционалне . . . . .	16
3.2.1	Moduł wstępnego przetwarzania danych - PreParser . . . . .	16
3.2.2	Moduł właściwego przetwarzania danych - Parser . . . . .	16
3.3	Analiza przypadków użycia (diagramy UML) . . . . .	16
<b>4</b>	<b>Specyfikacja zewnętrzna</b>	<b>17</b>
4.1	Wymagania sprzętowe i programowe . . . . .	18
4.2	Instalacja . . . . .	18
4.2.1	Instalacja PreParser-a na kontrolerze u klienta . . . . .	18
4.2.2	Instalacja Parser-a po stronie administratora . . . . .	18
4.3	Obsługa systemu i administrowanie nim . . . . .	18
4.3.1	PreParser . . . . .	18
4.3.2	Parser . . . . .	18
4.4	Bezpieczeństwo . . . . .	18
4.5	Przykład działania . . . . .	18

<b>5</b>	<b>Specyfikacja wewnętrzna</b>	<b>19</b>
5.1	Architektura systemu . . . . .	19
5.2	Organizacja i struktura baz danych . . . . .	19
5.3	Wykorzystane biblioteki . . . . .	19
5.4	Model klas . . . . .	19
5.5	Szczegółowa analiza danych wejściowych . . . . .	19
5.6	Algorytmy . . . . .	19
5.7	Diagramy UML . . . . .	19
<b>6</b>	<b>Testowanie i uruchamianie</b>	<b>21</b>
6.1	PreParser . . . . .	21
6.2	Parser . . . . .	21
<b>7</b>	<b>Uwagi o przebiegu i wynikach prac</b>	<b>23</b>
7.1	Stopień realizacji zagadnienia . . . . .	23
7.2	Napotkane problemy . . . . .	23
7.3	Dalszy rozwój . . . . .	23
<b>8</b>	<b>Podsumowanie</b>	<b>25</b>

# Rozdział 1

## Wstęp

Celem pracy dyplomowej jest zaprojektowanie i implementacja systemu informatycznego, który umożliwiłby gromadzenie oraz przetwarzanie danych generowanych przez kasy sklepowe Aloha. System ten miałby na celu dostarczenie użytkownikowi informacji o funkcjonowaniu sklepu, w którym aplikacja została zainstalowana. Umożliwi to kompleksowy monitoring i analizę danych sprzedażowych. System będzie pobierał dane generowane przez kasę i poddawał je szczegółowej analizie, która pozwoli na tworzenie z nich struktur obrazujących rzeczywiste czynności wykonywane przy użyciu kas Aloha (szczególnie popularny model kasy w Stanach Zjednoczonych). Takie przetwarzanie przychodzących danych pozwoli użytkownikowi na wejście w dane szczegółowe konkretnej transakcji, zobaczenie materiału wideo z jej przebiegu oraz generowanie raportów i paragonów. Dzięki takim funkcjonalnościom system powinien umożliwić optymalizację sprzedaży oraz wprowadzanie oszczędności wynikających z usunięcia nieprawidłowości zaistniałych podczas pracy sklepu. Aplikacja będzie składała się z dwóch warstw:

- modułu wstępnego przetwarzania danych (PreParser-a)
- modułu właściwego przetwarzania danych (Parser-a)

PreParser będzie działał jako serwis po stronie użytkownika i wprowadzi konieczność jego instalacji na tzw. kontrolerze w sklepie użytkownika, czyli komputerze gromadzącym dane przychodzące z kas w danym sklepie i poddającym je etapowi wstępnej obróbki (preprasing) oraz wysyłającym dane wynikowe do bazy danych dostarczanej przez twórcę systemu, gdzie dalszym ich przetwarzaniem zajmie się Parser. Zadaniem modułu właściwego przetwarzania danych, będzie obróbka wstępnie przetworzonych informacji przez PreParser i zamiana ich na format mogący być analizowany przez system i zaprezentowany użytkownikowi. Pomysł na stworzenie takiego oprogramowania zrodził się z faktu zajmowania się tą tematyką zawodowo przez autora niniejszej pracy. Autor postanowił wykorzystać wiedzę i doświadczenie zdobyte podczas pracy zawodowej oraz studiów, by stworzyć system mający być częścią oprogramowania 360iQ dostarczanego przez firmę EZUniverse Inc. Dzięki regularnej pracy nad zagadnieniem podczas wykonywaniu obowiązków zawodowych, możliwe było dokładne dostosowanie systemu pod wymagania użytkownika oraz jego solidne przetestowanie przy działu z wykorzystaniem realnych danych i przypadków użycia. Kolejny rozdział ni-



Rysunek 1.1: Logo produktu 360iQ stworzonego przez firmę EZUniverse

niejszej pracy zawiera zarys problemu oraz informacje opisujące wykorzystane technologie. Rozdział trzeci wprowadza w temat wymagań funkcjonalnych i nie-funkcjonalnych oraz przypadków użycia. Następne rozdziały przybliżą tematykę specyfikacji zewnętrznej oraz wewnętrznej systemu. Poruszone zostaną zagadnienia związane z wykorzystanymi algorytmami oraz metodami radzenia sobie z problematycznymi danymi wejściowymi. Przedstawiony zostanie dodatkowo schemat bazy danych oraz nastąpi omówienie ważniejszych, ze względu na rolę i funkcjonalność klas. Dwa ostatnie rozdziały poruszą kwestię testowania aplikacji oraz omówiony zostanie przebieg prac i wyniki końcowe wraz z wizją dalszego rozwoju systemu.

## Rozdział 2

# Analiza tematu

Niniejszy rozdział omawia podstawowe zagadnienia związane z realizowanym projektem. Poruszona została kwestia umiejscowienia konwertera danych Aloha w całym systemie 360iQ oraz opisane zostały typy informacji przekazywane przez kasę.

### 2.1 Wprowadzenie

#### 2.1.1 Procesor danych z terminali kas fiskalnych Aloha

Terminale kasowe Aloha (rys. 2.1) są niezwykle popularne w Stanach Zjednoczonych. Umożliwiają one kompleksową obsługę transakcji w sklepie, wpłat oraz wypłat do kasy. Jedną z dodatkowych możliwości terminala jest również dostarczanie informacji o zalogowaniu się pracownika i jego wylogowaniu, co pozwala ustalić intensywność pracy danej osoby, bądź też faktyczne godziny w jakich pracuje. W systemie 360iQ kasy Aloha wykorzystywane są do sklepach sieci Burger King. Terminal kasy przy każdym tzw. zdarzeniu, wysyła informację do nasłuchującego go kontrolera. Zdarzenia te są prostymi akcjami wykonywanymi podczas użytkowania kasy np. dodanie nowego produktu, potwierdzenie przyjęcia płatności. Dzięki gromadzeniu tych akcji po stronie kontrolera istnieje możliwość połączenia ich w zbiory, które będą reprezentować transakcje, wpłaty, wypłaty oraz obecności pracowników. Łączeniem tych zdarzeń w całość zajmuje się moduł aplikacji zainstalowany na kontrolerze zwany PreParserem, który jako część systemu 360iQ funkcjonuje pod nazwą AlohaPreParser. Przetwarza on zdarzenia uwzględniając zawarte w nich informacje dotyczące numeru kasy, rachunku etc. Po wstępnym przetworzeniu danych zostają one wysłane do bazy ulokowanej po stronie administratora, gdzie przechwytywane są one przez Parser i dokonywana jest właściwa analiza danych i ich przetwarzanie w celu dostarczenia klientowi potrzebnych informacji.

#### 2.1.2 Umiejscowienie w systemie 360iQ

AlohaPreParser oraz AlohaParser są bezpośrednimi składowymi systemu systemu 360iQ dostarczanemu przez firmę EZUniverse (rys. 2.2). System ten składa się z całej infrastruktury nakierowanej na dostarczenie klientowi maksymalnej liczby informacji mogących wesprzeć funkcjonowanie biznesu klienta.



Rysunek 2.1: Przykładowy model kasy Aloha.



Rysunek 2.2: Logo firmy EZUniverse Inc.



1 

```
<SpyMessage TerminalID="3" EventTime="18:00:38"
EmployeeID="139" EmployeeName="JAMES BOND"
ManagerID="0" ManagerName="" TableID="3132860"
CheckID="3145860" TransactionTypeID="8"
TransactionType="ADD_ITEM" Description="SM FRY"
Amount="1.99" Quantity="1" Sender="192.168.0.101"
ReceivedOn="2017/03/05 00:00:52.042" />
```

Rysunek 2.3: Przykładowe zdarzenie - dane wejściowe Aloha.

Wspierane są duże ilości modeli kas, które mogą być zainstalowane w klepie klienta, między innymi:

- Aloha
- Micros
- Sicom
- Radiant
- Subshop2000
- ProfiTrack
- Panasonic7900
- Comtrex
- Gilbarco
- Par
- SubwayPOS

Dzięki wsparciu dla dużej ilości modeli kas oraz współpracy z największymi sieciami sklepów (np. Subway, Burger King) system 360iQ jest kompleksowa platformą do wspomagania biznesu klienta.

### 2.1.3 Dane wejściowe

Danymi wejściowymi dostarczonymi do kontrolera są pojedyncze zdarzenia zapisane w formacie XML. Format przychodzących danych omówiony zostanie na podstawie następującego zdarzenia (listing 2.3). Opisuje ono akcję dodania nowego przedmiotu do rachunku o numerze 3145860. Możemy na podstawie przedstawionego listingu ustalić również, że dane zdarzenie miało miejsce o godzinie 18:00:38 oraz przedmiot SM FRY został dodany do rachunku przez pracownika JAMES BOND. Na podstawie tak przedstawionych danych wejściowych moduł przetwarzania wstępnego jest w stanie budować całe zestawy transakcji, które zawierają szereg zdarzeń podobnych temu wyżej przedstawionemu. Dzięki tak pogrupowanym danym, zamiana zwykłych zdarzeń z terminala na struktury mogące być poddane analizie jest znacznie uproszczona. Terminal generuje również dodatkowe dane przechwytywane przez kontroler zwane słownikami.

Słowniki są zestawami danych zapisanych w formacie XML niosącymi informacje o specyficznych cechach danego sklepu, np. promocjach, identyfikatorach produktów, bądź podatkach. Przykładowe słowniki przedstawiono na listingach w skróconej formie, gdyż całe pliki potrafią zawierać znaczne ilości informacji (Kategorie produktów: 2.4, Lista promocji: 2.5)

i/Item<sub>i</sub>. Słowniki te gromadzone są przez moduł wstępnego przetwarzania danych i wykorzystywane w ich analizie i algorytmach mających na celu dokonanie odpowiednich poprawek w danych transakcyjnych (szczegółowo ten temat został poruszony w rozdziale traktującym o wykorzystanych algorytmach).

#### 2.1.4 Typy zdarzeń generowanych przez kasę Aloha

Kasa Aloha jest zdolna do wygenerowania szeregu typów zdarzeń. Są to między innymi:

- Zalogowanie pracownika w kasie
- Wylogowanie pracownika z kasy
- Przerwa w pracy
- Zakończenie przerwy w pracy
- Otwarcie kasy
- Dodanie nowego przedmiotu do transakcji
- Anulowanie dodanego przedmiotu
- Usunięcie przedmiotu z transakcji
- Otwarcie nowego zamówienia
- Drukowanie rachunku
- Zamknięcie zamówienia
- Ponowne otwarcie zamówienia
- Przyjęcie płatności
- Autoryzacja płatności
- Odrzucenie płatności
- Modyfikacja płatności
- Usunięcie płatności
- Dodaj promocje
- Usuń promocje
- Wartość podatku
- Wartość netto transakcji
- Wartość brutto transakcji

```

1  <ItemCategoryMessage>
2  <Item ItemId="112" ItemName="DELETE" Price="5.89">
3  <Category CategoryId="96" CategoryName="WHOPPER LINE"
   />
4  <Category CategoryId="68" CategoryName="SANDWICHES" />
5  <Category CategoryId="55" CategoryName="ALL ITEMS" />
6  <Category CategoryId="57" CategoryName="DISCOUNTABLE
   ITEMS" />
7  <Category CategoryId="56" CategoryName="ALL ITEMS
   EXCEPT CMB" />
8  <Category CategoryId="313" CategoryName="ANY SANDWICH"
   />
9  <Category CategoryId="1" CategoryName="FOOD" />
10 </Item>
11 <Item ItemId="2692" ItemName="LG FLOAT" Price="2.79">
12 <Category CategoryId="71" CategoryName="DRINKS" />
13 <Category CategoryId="55" CategoryName="ALL ITEMS" />
14 <Category CategoryId="57" CategoryName="DISCOUNTABLE
   ITEMS" />
15 <Category CategoryId="56" CategoryName="ALL ITEMS
   EXCEPT CMB" />
16 <Category CategoryId="1" CategoryName="FOOD" />
17 </Item>
18 <Item ItemId="392" ItemName="WRP HM CSP" Price="0">
19 <Category CategoryId="68" CategoryName="SANDWICHES" />
20 <Category CategoryId="310" CategoryName="Bearcat
   Sandwiches" />
21 <Category CategoryId="55" CategoryName="ALL ITEMS" />
22 <Category CategoryId="57" CategoryName="DISCOUNTABLE
   ITEMS" />
23 <Category CategoryId="56" CategoryName="ALL ITEMS
   EXCEPT CMB" />
24 <Category CategoryId="1" CategoryName="FOOD" />
25 </Item>
26 <Item ItemId="484" ItemName="WRP BLT CSP" Price="3.49"
   >
27 <Category CategoryId="55" CategoryName="ALL ITEMS" />
28 <Category CategoryId="57" CategoryName="DISCOUNTABLE
   ITEMS" />
29 <Category CategoryId="56" CategoryName="ALL ITEMS
   EXCEPT CMB" />
30 <Category CategoryId="25" CategoryName="ALL SALADS" />
31 <Category CategoryId="1" CategoryName="FOOD" />
32 </Item>
33 <Item ItemId="5748" ItemName="AVOCADO RAN" Price="0">
34 <Category CategoryId="55" CategoryName="ALL ITEMS" />
35 <Category CategoryId="57" CategoryName="DISCOUNTABLE
   ITEMS" />
36 <Category CategoryId="56" CategoryName="ALL ITEMS
   EXCEPT CMB" />
37 <Category CategoryId="1" CategoryName="FOOD" />
38 </Item>
39 </ItemCategoryMessage>

```

Rysunek 2.4: Przykładowy słownik produktów w formacie XML - wersja skrócona.

```

1 <PromotionMessage>
2 <Item ItemId="7592" ItemName="Items" Price="0">
3 <Promotion PromotionId="6485" PromotionVersion="1941"
   PromotionName="6485*$5C'wich ML 2" />
4 <Promotion PromotionId="118" PromotionVersion="999"
   PromotionName="Local Combos" />
5 <Promotion PromotionId="139" PromotionVersion="2655"
   PromotionName="Bacon King" />
6 <Promotion PromotionId="101" PromotionVersion="689"
   PromotionName="Whopper Combo" />
7 <Promotion PromotionId="101" PromotionVersion="744"
   PromotionName="Whopper Combo" />
8 <Promotion PromotionId="101" PromotionVersion="749"
   PromotionName="Whopper Combo" />
9 <Promotion PromotionId="110" PromotionVersion="2444"
   PromotionName="FISH COMBO" />
10 <Promotion PromotionId="282" PromotionVersion="169"
   PromotionName="Free Kids Combo" />
11 <Promotion PromotionId="104" PromotionVersion="694"
   PromotionName="Whopper Jr. Combo" />
12 <Promotion PromotionId="104" PromotionVersion="746"
   PromotionName="Whopper Jr. Combo" />
13 <Promotion PromotionId="108" PromotionVersion="693"
   PromotionName="Orig. Chicken Combo" />
14 <Promotion PromotionId="108" PromotionVersion="789"
   PromotionName="Orig. Chicken Combo" />
15 <Promotion PromotionId="160" PromotionVersion="900"
   PromotionName="Grilled Ckn Sw Combo" />
16 <Promotion PromotionId="160" PromotionVersion="2447"
   PromotionName="Grilled Ckn Sw Combo" />
17 <Promotion PromotionId="29142" PromotionVersion="160"
   PromotionName="$1.99 Kids Combo" />
18 <Promotion PromotionId="10012" PromotionVersion="947"
   PromotionName="5 FOR $4" />
19 <Promotion PromotionId="10012" PromotionVersion="1372"
   PromotionName="5 FOR $4" />
20 <Promotion PromotionId="10012" PromotionVersion="1866"
   PromotionName="5 FOR $4" />
21 <Promotion PromotionId="28802" PromotionVersion="1951"
   PromotionName="8802*$4 2Cwich ML" />
22 <Promotion PromotionId="147" PromotionVersion="747"
   PromotionName="Homestyle Cb" />
23 <Promotion PromotionId="9579" PromotionVersion="985"
   PromotionName="9579*2/$10 Wpr ML" />
24 <Promotion PromotionId="283" PromotionVersion="1351"
   PromotionName="Kids Breakfast Meal" />
25 <Promotion PromotionId="234" PromotionVersion="94"
   PromotionName="Dbl Biscuit Combo" />
26 <Promotion PromotionId="221" PromotionVersion="89"
   PromotionName="Crois 'wich Combo" />
27 <Promotion PromotionId="221" PromotionVersion="90"
   PromotionName="Crois 'wich Combo" />
28 <Promotion PromotionId="221" PromotionVersion="93"
   PromotionName="Crois 'wich Combo" />
29 </Item>
30 </PromotionMessage>

```

Rysunek 2.5: Przykładowy słownik listy promocji w formacie XML - wersja skrócona.

- Dodatek do potrawy

Jest to lista wszystkich obsługiwanych typów zdarzeń. Są one identyfikowane przez PreParser na podstawie przypisanych im unikalnych identyfikatorów. Istnieje jeden typ zdarzenia, który nie został uwzględniony w wyżej wymienionej liście, mianowicie zdarzenie nie będące elementem transakcji zakończonej sprzedażą. Takie przypadki to tzw. transakcje typu "NOSALE" i są one analizowane z wykorzystaniem szczególnego podejścia. Jest ono szerzej opisane w następnych rozdziałach.

### 2.1.5 Etap wstępnego przetwarzania

Każdy sklep korzystający z oprogramowania będącego przedmiotem niniejszej pracy wymaga instalacji specjalnego modułu działającego jako serwis na komputerze zwanym kontrolerem. Urządzenie to gromadzi zdarzenia generowane przez kasy działające w obrębie omawianego sklepu. Odebrane informacje są przechwytywane przez działający w systemie moduł wstępnego przetwarzania, który poddaje je wstępnej analizie. Możemy wyodrębnić kluczowe etapy składające się na przeprowadzenie wspomnianego procesu:

- Sprawdzanie danych pod kątem duplikatów ich i odpowiednie odfiltrowanie
- Analiza możliwości wystąpienia niezamkniętych lub błędnych transakcji ich odpowiednia obsługa
- Grupowanie zdarzeń przychodzących z kasy w kolekcje związane z jedną transakcją
- Poszukiwanie transakcji, które zostały ponownie i sprawdzanie czy wymagana jest korekcja cen produktów lub płatności
- Przetwarzanie całych grup transakcji, rozbudowa zdarzeń z wykorzystaniem algorytmów analizy promocji oraz kategorii produktów
- Konwersja zgrupowanych zdarzeń na obiekty XML (Transakcje, Wpłaty-/Wypłaty z kasy, Obecności pracowników)
- Wysyłanie XML-i do bazy danych w celu ich przetworzenia przez moduł właściwego przetwarzania

### 2.1.6 Etap właściwego przetwarzania

Po wstępnym przetworzeniu danych, gdy moduł wstępnego procesowania informacji przekaże je do bazy danych udostępnianej przez administratora systemu, są one przechwytywane przez działający w środowisku dostawcy usługi moduł właściwego przetwarzania danych. Pobiera on z tabel dane w formacie XML, które dzięki wstępnej pracy wykonanej przez PreParser są już odpowiednio pogrupowane i oznaczone. Umożliwia to Parserowi zejście poziom niżej w analizie danych, podczas gdy PreParser analizował zdarzenia pod kątem grupowania ich w struktury, Parser może analizować same zdarzenia i wyciągać z nich informacje, które są konieczne do zbudowania wyczerpującego opisu transakcji, bądź innego zdarzenia, które miało miejsce na kasie. Moduł ten analizuje

każde zdarzenie i krok po kroku tworzy obiekt, który na sam koniec procesowania danych zostanie umieszczony w wynikowej bazie. Udostępnia ona bezpośrednio dane mogące być analizowane przez klienta za pośrednictwem specjalnej aplikacji webowej będącej jest częścią systemu 360iQ. Schemat analizy danych przez moduł właściwego przetwarzania wygląda następująco:

- Sprawdzanie danych pod kątem duplikatów ich i odpowiednie odfiltrowanie
- Wykrywanie typu zdarzenia jakie zostało zgrupowane w wejściowy XML: (Transakcja, Wpłata/Wypłata, Obecność pracownika)
- Uruchamianie odpowiedniego schematu przetwarzania zależnie od wykrytego typu wiadomości
- Analiza danych w formacie XML krok po kroku i tworzenie cyfrowego modelu będącego odpowiednikiem realnej czynności wykonanej przez pracownika sklepu.
- Sprawdzanie poprawności stworzonych modeli i dokonywanie koniecznych poprawek, jak również oznaczenie tych, które zawierają wartości odbiegające od normy (ten fakt jest później odpowiednio prezentowany użytkownikowi w aplikacji webowej)
- Generowanie paragonów do wglądu przez klienta
- Wysyłanie przetworzony modeli do odpowiednich tabel wynikowej bazy danych w zależności od typu przetwarzanych wiadomości.

### 2.1.7 Przypadki niestandardowe

Terminale kasowe Aloha są dość bogate w przypadki, które można uznać za niestandardowe. Do najczęstszych takich sytuacji możemy zaliczyć:

- Wadliwe dane wejściowe
- Niezamknięte transakcje
- Dodawanie przedmiotów do rachunku z użyciem ich pełnej nazwy oraz późniejsze usuwanie wykorzystując ich skrócone nazwy
- Dodawanie promocji bez uwzględnienia odpowiednich przedmiotów, które dodana promocja powinna dodawać
- W regionach innych niż Stany Zjednoczone częstym problemem jest brak informacji o podatkach

#### Wadliwe dane wejściowe

Z racji faktu, że moduł wstępnego przetwarzania otrzymuje dane z terminali kasowych Aloha w formacie XML, często zdarza się sytuacja, że jest on wadliwy (np. ucięty), lub też występują jego duplikaty. W takich przypadkach PreParser ustawia odpowiednie statusy tych wiadomości (mogą być one potem zweryfikowane przez użytkownika) i pomija je w procedurze wstępnej analizy.

### Niezamknięte transakcje

Jednym z najrzadziej występujących przypadków niestandardowych jest sytuacja, gdy mamy do czynienia z niezamkniętą transakcją. Dzieje się to wtedy, gdy otwierana jest transakcja, lecz nie przychodzi zdarzenie oznaczające jej zamknięcie (w Aloha jest to: `ÓLOSECHECK`). W takiej sytuacji moduł wstępnego przetwarzania ustawiana taką transakcję w stan oczekiwania, który trwa 24 godziny. Jeśli do tego czasu przyjdzie zdarzenie zamykające daną transakcję, to zostanie ona zatwierdzona i przesłana do dalszej analizy, w przeciwnym razie nastąpi jej usunięcie z kolejki oczekujących i oznaczenie jako błędnej danej wejściowej, by użytkownik mógł samodzielnie się jej przyjrzyć.

### Dodawanie przedmiotów do rachunku z użyciem ich pełnej nazwy oraz późniejsze usuwanie wykorzystując ich skrócone nazwy

Przedmioty dodawane przez kasę Aloha do rachunków niestety często charakteryzują się niejednorodnym nazewnictwem. Ten sam przedmiot może być nazywany z wykorzystaniem:

- nazwy skróconej
- pełnej nazwy przedmiotu
- ID przedmiotu, bądź jego oznaczenia kodowego

Powoduje to problemy przy usuwaniu z rachunku już dodanych przedmiotów. Przedmiot dodany do transakcji pod nazwą X może być z niej usuwany pod nazwą Y. W celu uzyskania spójności i powiązania ze sobą przedmiotów wykorzystywany jest algorytm rozszerzania danych przekazywanych przez zdarzenie. Korzysta on ze słownika kategorii przedmiotów, który jest dostarczany raz dziennie przez właściciela sklepu jako specjalne zdarzenie terminala Aloha i przechowywany przez kontroler. Dzięki temu słownikowi możliwe jest powiązanie ze sobą tych samych przedmiotów występujących pod różną nazwą, wykorzystując fakt, że każdy przedmiot ma swoje unikalne ID, które jest udostępniane za pośrednictwem słownika kategorii produktów. Takie rozwiązanie gwarantuje, że w wynikowej transakcji przedmioty będą powiązane i usunięte w sposób prawidłowy niezależnie od zastosowania pełnej, bądź jego skróconej nazwy.

### Dodawanie promocji bez uwzględnienia odpowiednich przedmiotów, które dodana promocja powinna dodawać

Niezbyt przyjazną cechą terminala Aloha jest obsługa dodawania promocji bez uwzględnienia przedmiotów wchodzących w jej skład. W takim wypadku konieczne jest dodanie przedmiotów do transakcji, których dotyczy dana promocja, dzięki czemu zachowana zostanie zgodność danych z realnie sprzedanymi przedmiotami po stronie sklepu. Dodawanie przedmiotów przez moduł wstępnego przetwarzania realizowane jest przy użyciu słownika promocji dostarczanego przez klienta dla każdego sklepu. Pozwala on zidentyfikować nazwę promocji i rozszerzyć XML z transakcją o potrzebne przedmioty.

**W regionach innych niż Stany Zjednoczone częstym problemem jest brak informacji o podatkach**

Problem ten objawia się sytuacją, gdy w zdarzeniu wystawianym przez terminal mającym typ "TAX" pojawia się zerowa wartość podatku. W tej sytuacji, by uzyskać prawidłową cenę końcową z wliczoną już wartością podatku, konieczne jest wczytanie dodatkowego słownika przechowującego wartości podatków dla każdego z przedmiotów. Takie rozwiązanie pojawia się w sytuacji, gdy oprogramowanie działa w krajach gdzie system podatkowy znacząco różni się od tego, który obowiązuje w Stanach Zjednoczonych.

## **2.2 Opis technologii i wykorzystanych narzędzi**

### **2.2.1 Środowisko pracy i język programowania**

Projekt został zrealizowany przy użyciu technologii .NET oraz języka C#. Wielokrotnie podczas przeglądania kodu źródłowego projektu można również napotkać skrypty pisane w języku T-SQL, w celu komunikacji z bazą danych. Całość projektu została napisana w środowisku Visual Studio 2017, przy czym korzystano również z Visual Studio Code w celu analizy danych XML oraz szybkiego przeglądania plików tekstowych. Do administrowania i zarządzania bazą danych wykorzystano Microsoft SQL Server Management Studio 2016.

### **2.2.2 Baza danych**

Serwerem bazy danych był Microsoft SQL Server 2016 w wersji Express zainstalowany na lokalnym komputerze. Dokonano połączenia tabel koniecznych do pracy modułu wstępnego przetwarzania i tych niezbędnych do funkcjonowania jednostki realizującej właściwe procesowanie danych. Dzięki takiemu rozwiązaniu można było testować działanie całego systemu na lokalnym komputerze bez konieczności korzystania z osobnego komputera, który miałby pełnić funkcję kontrolera. Dane wykorzystywane w niniejszym projekcie są backupami realnych danych źródłowych zaczerpniętych z codziennej pracy systemu u klienta. Pozwala to testować i przedstawić pracę niniejszego projektu w jego naturalnych warunkach i w starciu z prawdziwymi danymi generowanymi przez sklepy w Stanach Zjednoczonych,

### **2.2.3 Moduły wspierające**

Opisywana w niniejszej pracy aplikacja jest wspierana przez następujące oprogramowanie, które jest częścią systemu 360iQ i zostało opracowane przez firmę EZUniverse:

- AlohaSpyRelay
- EZ360DataInterface

#### **AlohaSpyRelay**

AlohaSpyRelay jest aplikacją zapewniającą transmisję danych z kasy Aloha do kontrolera gdzie zainstalowany jest moduł wstępnego przetwarzania. Pobiera



ona dane bezpośrednio z terminala, zamienia ją na zdarzenia zapisane w formacie XML i przekazuje do kontrolera.

### **EZ360DataInterface**

EZ360DataInterface jest łącznikiem pomiędzy modulem wstępnego przetwarzania i częścią odpowiadającą za właściwe przetwarzanie. Po wstępnym przetworzeniu danych przez PreParser istnieje konieczność wysłania ich do bazy danych dostarczanej przez administratora systemu, tym właśnie zajmuje się opisywana aplikacja. Moduł wstępnego przetwarzania przygotowuje gotową paczkę danych do wysłania i powiadamia EZ360DataInterface, że są dane do wysyłki. Aplikacja przechwytuje dane od PreParsera i wysyła je zgodnie ze skonfigurowanymi w jej ustawieniach wytycznymi.



## Rozdział 3

# Wymagania

Rozdział ten opisuje wymagania konieczne do uruchomienia oprogramowania po stronie kontrolera jak i administratora aplikacji. Przedstawione zostały również przypadki użycia wraz ze stosownym komentarzem.

### 3.1 Wymagania funkcjonalne

#### 3.1.1 Moduł wstępnego przetwarzania danych - PreParser

- Serwis powinien być w stanie odczytywać pobrać dane z lokalnej bazy danych kontrolera gdzie przechowywane są informacje o zdarzeniach, które nadesłał terminal Aloha
- Powinna istnieć możliwość przetwarzania danych na etapie wstępnej analizy w celu eliminacji duplikatów i wadliwych wiadomości
- Kluczową kwestią jest zastosowanie algorytmów dodających potrzebne produkty z promocji oraz rozszerzające nazwy wspomnianych wcześniej produktów, by można było je łatwo ze sobą powiązać
- Serwis powinien informować użytkownika na bieżąco o stanie swojej aktywności za pośrednictwem komunikatów w pliku logów.
- Serwis powinien być zdolny grupować wiadomości w kolekcje i konwertować je na odpowiednio zdefiniowane XML-e, a następnie wysyłać je do bazy danych po stronie administratora, by mogły być one dalej analizowane

#### 3.1.2 Moduł właściwego przetwarzania danych - Parser

- Aplikacja powinna móc odczytywać dane z odpowiednich tabel w bazie danych, gdzie w sposób ciągły pojawiają się nowe dane przysyłane z bliżej nie określonej liczby kontrolerów, gdzie zostały one wstępnie obrobione przez moduł wstępnego przetwarzania
- Oprogramowanie musi wspierać różne typy przychodzących wiadomości (transakcje, wpłaty/wypłaty, obecności pracowników) i stosować dla nich właściwe formy analizy i przetwarzania

- Wynikowe dane powinny zostać zapisane w bazie danych, z której może korzystać aplikacja webowa dostarczana jako element systemu 360iQ i opracowana przez firmę EZUniverse i prezentująca wyniki użytkownikowi w przyjaznej dla niego formie. Rezultat przetwarzania powinien być możliwie jak najdokładniejszym odwzorowaniem zdarzenia jakie zaszło w danej chwili czasu na kasie w sklepie użytkownika. Ważne jest zachowanie odpowiednich stref czasowych i synchronizacja czasu wideo.
- Oprogramowanie powinno udostępniać prosty i intuicyjny interfejs użytkownika
- W przypadku wystąpienia błędu podczas analizy danych, aplikacja powinna zamieścić odpowiednią informację na ekranie interfejsu użytkownika

## 3.2 Wymagania niefunkcjonalne

### 3.2.1 Moduł wstępnego przetwarzania danych - PreParser

- Stabilnie działający serwis po stronie kontrolera
- Personalizacja za pomocą jednego pliku konfiguracyjnego
- Prosty sposób instalacji oraz włączania serwisu
- Szczegółowy system logów pomagający w diagnostyce usterek i monitoringu działania aplikacji
- Niskie wymagania sprzętowe, serwis powinien się uruchamiać i pracować bez widocznych strat na wydajności będąc zainstalowanym na stosunkowo taniej i mało wydajnej maszynie

### 3.2.2 Moduł właściwego przetwarzania danych - Parser

- Stabilnie działająca aplikacja desktopowa z intuicyjnym interfejsem użytkownika
- Pełne wsparcie dla przetwarzania danych z kas Aloha oraz integracja z pozostałymi parserami z innych kas działającymi w systemie
- Łatwość wystartowania i zastopowania aplikacji
- Dążenie do jak najniższego wykorzystania zasobów systemu, w którym aplikacja jest zainstalowana

## 3.3 Analiza przypadków użycia (diagramy UML)

## Rozdział 4

# Specyfikacja zewnętrzna

Rozdział ten opisuje wymagania sprzętowe i programowe, które należy spełnić, by zapewnić poprawne funkcjonowanie aplikacji. Poruszone zostały kwestie instalacji i administracji poszczególnymi modułami oprogramowania oraz temat zabezpieczeń.

## 4.1 Wymagania sprzętowe i programowe

## 4.2 Instalacja

### 4.2.1 Instalacja PreParser-a na kontrolerze u klienta

### 4.2.2 Instalacja Parser-a po stronie administratora

## 4.3 Obsługa systemu i administrowanie nim

### 4.3.1 PreParser

Startowanie serwisu

Zatrzymywanie serwisu

Deinstalacja serwisu

Diagnostyka i logowanie

### 4.3.2 Parser

Uruchamianie aplikacji

Zatrzymywanie aplikacji

Deinstalacja aplikacji

Diagnostyka i logowanie

## 4.4 Bezpieczeństwo

## 4.5 Przykład działania

## Rozdział 5

# Specyfikacja wewnętrzna

W niniejszym rozdziale podjęta została kwestia architektury oprogramowania. Przedstawione zostały schematy baz danych oraz zakres wykorzystywanych bibliotek. Rozpisano również model ważniejszych klas oraz szczegółowo omówione zostały dane wejściowe wraz algorytmami koniecznymi do ich odpowiedniej analizy.

### 5.1 Architektura systemu

### 5.2 Organizacja i struktura baz danych

### 5.3 Wykorzystane biblioteki

### 5.4 Model klas

### 5.5 Szczegółowa analiza danych wejściowych

### 5.6 Algorytmy

### 5.7 Diagramy UML

```
1 function map = CreateBinaryMap(image , P)
2     threshold = FindThreshold(image , P);
3     map = Binarize(image , threshold);
4     map = MorphologicalClosing(map);
5 end
```

Rysunek 5.1: Przykładowy listing.





## Rozdział 6

# Testowanie i uruchamianie

### 6.1 PreParser

### 6.2 Parser

Sposób testowania w ramach pracy (organizacja eksperymentów, przypadki testowe, wyniki, zakres testowania – pełny/niepełny)

Poniżej przykład tworzenia tabeli. Podobnie jak w przypadku rysunków, każdą tabelę należy opisać w treści pracy i odpowiednio się do niej odwołać.

Tablica 6.1: Nagłówek tabeli.

Pierwsza kolumna	Druga kolumna	Trzecia kolumna
3	4	5



## Rozdział 7

# Uwagi o przebiegu i wynikach prac

7.1 Stopień realizacji zagadnienia

7.2 Napotkane problemy

7.3 Dalszy rozwój



## Rozdział 8

# Podsumowanie

Przeczytaj poniższy tekst przed oddaniem pracy do sprawdzenia:

[W całej pracy należy zapewnić, że nie pojawiają się sieroty - pojedyncze litery na końcu linii. W tym celu należy korzystać z symbolu tyldy po każdej takiej literze.]

[Nie powinno być odstępów między akapitami. Jeżeli tak się dzieje, oznacza to złe ułożenie obrazków. Wtedy wystarczy określić, gdzie powinien taki obraz się znajdować (np. na górze/dole strony). Innym rozwiązaniem jest dodanie komendy 'newpage' na końcu strony.]

[Proszę nie używać słownictwa w języku obcym - większość słów ma swój polski odpowiednik.]

[Proszę nie używać apostrofów w roli cudzysłowia.]

[W przypadku stosowania różnych czcionek (kursywy czy pogrubień) proszę trzymać się jednej konwencji w całej pracy.]

[Wprowadzając pojęcie należy podać najpierw jego polską nazwę, następnie w nawiasie angielską i ewentualny skrót. Od takiej definicji skrótu można z niego korzystać w dalszej części pracy. Np. Jako algorytm uczący wybrałem metodę wektorów wspierających (ang. Support Vector Machine, SVM). SVM jest często stosowany do ...]





# Bibliografia