

POLITECHNIKA KRAKOWSKA im. T. KOŚCIUSZKI

WYDZIAŁ INŻYNIERII ELEKTRYCZNEJ I KOMPUTEROWEJ

Projekt przedmiotu:

**Inżynieria Programowania**

Temat projektu:

**Wypożyczalnia sprzętu muzycznego**

**CIEPIELA ADRIAN**

**WĄTOR MICHAŁ**

Semestr 5

Rok akademicki 2016/2017

Spis treści

[1. Wstęp 4](#_Toc473139674)

[Cel i zakres projektu: 4](#_Toc473139675)

[2. Słownik danych 4](#_Toc473139676)

[3. Analiza biznesowa 6](#_Toc473139677)

[4. Analiza ryzyka oraz ograniczenia 7](#_Toc473139678)

[5. Ogólny opis 8](#_Toc473139679)

[6. Charakterystyka użytkowników 8](#_Toc473139680)

[Administracja sklepu 8](#_Toc473139681)

[Klient 9](#_Toc473139682)

[7. Główne funkcje produktu: 9](#_Toc473139683)

[Funkcje obsługiwane przez administrację: 9](#_Toc473139684)

[Funkcje obsługiwane przez klienta: 9](#_Toc473139685)

[8. Wymagania funkcjonalne i niefunkcjonalne 10](#_Toc473139686)

[Wymagania funkcjonalne: 10](#_Toc473139687)

[Wymagania niefunkcjonalne: 10](#_Toc473139688)

[9. Diagramy operacyjne 11](#_Toc473139689)

[Diagram ERD 11](#_Toc473139690)

[Diagram DFD 12](#_Toc473139691)

[Diagram ELH 14](#_Toc473139692)

[Diagram STD 15](#_Toc473139693)

[10. Proces projektowania aplikacji 16](#_Toc473139694)

[Analiza wymagań projektu / aplikacji 16](#_Toc473139695)

[Specyfikacja wymagań 16](#_Toc473139696)

[Tworzenie aplikacji 16](#_Toc473139697)

[Testowanie 17](#_Toc473139698)

[Wdrożenie aplikacji 17](#_Toc473139699)

[Konserwacja i opieka nad aplikacją 17](#_Toc473139700)

[Rozbudowa aplikacji 17](#_Toc473139701)

[11. Scenariusze testowe 18](#_Toc473139702)

# Wstęp

## Cel i zakres projektu:

Celem projektu jest zaprojektowanie i zaimplementowanie aplikacji desktopowej, która będzie umożliwiała obsługę i zarządzanie wypożyczalnią sprzętu muzycznego. Aplikacja będzie posiadała funkcjonalności odpowiadające za pracę zarówno administracji sklepu jak i klienta. Funkcjonalność oferowana przez aplikację będzie wystarczająca, aby klient był w stanie w prosty i intuicyjny sposób wypożyczyć sprzęt muzyczny z poziomu aplikacji, bez konieczności dodatkowego kontaktowania się z pracownikami wypożyczalni. Wraz z aplikacją powstanie również baza danych niezbędna do przechowywania danych używanych przez aplikację.

Technologie jakie zostały użyte do zrealizowania tego projektu:

* aplikacja desktopowa – język obiektowy Java (javaFX – graficzny interfejs użytkownika),
* baza danych – język bazodanowy Oracle SQL .

# Słownik danych

**Account** – konto w aplikacji przypisane do konkretnego uzytkownika, zapewniające funkcjonalność zależną od rodzaju konta (konto użytkownika, konto administratora)

**Admin** – osoba posiadająca uprawnienia administratora w aplikacji

**API** - Interfejs programistyczny danej funkcjonalności. Udostępnia zestaw metod które są wystarczające do kompleksowej obsługi danej części aplikacji, np. DatabaseAPI – klasa oferująca metody do obsługi bazy danych.

**Client** – osoba korzystająca z aplikacji w celu wypożyczenia sprzętu muzycznego

* id – klucz główny, unikatowy numer klienta, składający się z maksymalnie 10 cyfr
* firstName – imię podane przez klienta, maksymalnie 20 liter
* lastName – nazwisko podane przez klienta, maksymalnie 30 liter
* address – adres klienta, maksymalnie 50 znaków

**Database** –baza danych aplikacji. Przechowuje informacje o klientach, sprzęcie oraz rezerwacjach. Pozwala na odczyt i zapis danych z poziomu aplikacji.

**Equipment** – sprzęt muzyczny przeznaczony do wypożyczenia

* id – klucz główny, unikatowy numer instrumentu, składający się z maksymalnie 10 cyfr
* type – typ instrumentu
* brand – marka instrumentu
* model – model instrumentu
* yearOfProduction – rok produkcji instrumentu
* pricePerDay – opłata dzienna za wypożyczenie instrumentu
* description – opis instrumentu

**Order** – zamówienie, realizacja konkretnej rezerwacji

**Registration** – proces tworzenia konta w aplikacji

**Reservation** – rezerwacja sprzętu dokonywana przez klienta

* id –klucz główny, unikatowy numer rezerwacji, składający się z maksymalnie 10 cyfr
* startDate – data rozpoczęcia wypożyczenia
* durationDays – długość wypożyczenia, składająca się z maksymalnie 3 cyfr
* typeOfReception – typ utworzonego rachunku, składający się z maksymalnie 20 znaków
* typeOfReturn – sposób zwrotu wypożyczonych instrumentów
* paymentMethod – sposób płatności
* clientId – klucz obcy

# Analiza biznesowa

Przegląd stron internetowych krakowskich wypożyczalni sprzętu muzycznego pokazuje brak aplikacji, która umożliwiałaby dostęp do różnorodnego sprzętu w jednym miejscu. Lokalne wypożyczalnie dają możliwość dostępu jedynie do pewnego zestawu sprzętu. Na przykład:

* wypożyczalnia umożliwia dostęp tylko do sprzętu typowo estradowego/nagłośnieniowego. Brakuje możliwości wypożyczania samych instrumentów,
* wypożyczalnia umożliwia dostęp do mocno ograniczonego zestawu poszczególnych rodzajów sprzętu, np. tylko jedna perkusja, tylko jeden zestaw kolumn estradowych. Brakuje możliwości wybrania sprzętu dostosowanego ściśle do naszych potrzeb.

Drugim istotnym brakiem w aplikacjach krakowskich wypozyczalni jest brak interfejsu umożliwiającego dokonywanie zamówień. Strony te umożliwiają jedynie obejrzenie dostępnego asertymentu oraz zapoznanie się z jego cennikiem. Nie pozwalają jednak na dokonie samej akcji wypożyczenia sprzętu, ani dalszych związanych z tym rzeczy:

* wybór okresu wypożyczenia,
* ilości instrumentów,
* sposobu odbioru / zwrotu sprzętu.

Wszystkie te czynności zmuszeni jesteśmy ustalać poprzez kontakt telefoniczny lub mailowy. W dzisiejszych czasach, gdzie przyzwyczajeni jesteśmy, że zakupy możemy zrobić za pomocą kilku kliknięć myszką, takie braki w funkcjonalnościach są istotnymi niedogodnościami, które mogą mieć duży wpływ na prosperowanie tych usług na rynku.

# Analiza ryzyka oraz ograniczenia

Aplikacja do poprawnej pracy wymaga stałego połączenia z internetem co definiuje pierwsze i główne ograniczenie. Bez stałego dostępu do połączenia sieciowego, aplikacja nie będzie w stanie zapewnić swoich podstawowych funkcjonalności.

Istotną rzeczą jest też skuteczne rozgraniczenie funkcji administracji sklepu od funkcji z których korzysta klient. Dostęp do funkcji administracyjnych możliwy bedzie tylko po zalogowaniu się na odpowiednie, stworzone do tych celów konto. Osoba nie posiadająca dostepu do tego konta nie będzie mogła korzystać z żadnych dostarczanych przez nie funkcjonalności. Dzieki temu podejściu otrzymamy zabezpieczenie kluczowych danych (np. danych osobowych klientów) przed niepożądanym ujawnieniem oraz zablokujemy możliwość ich modyfikacji przez nieuprawnione do tego osoby.

Ponadto należy zabezpieczyć sie przed zmianami które są uprawnione ale niepoprawne. Nie jest możliwa całkowita kontrola nad tego typu błedami, natomiast można zapewnić, że dane zawsze są przetwarzana zgodnie z założeniami. Trzeba zadbać, aby wprowadzenie błędnych danych nigdy nie mogło spowodować zatrzymania działania aplikacji lub niestabilną jej pracę. Aby to osiągnąć należy uwzględnić w programie sprawdzanie zgodności typów oraz rozmiaru wszystkich wprowadzanych danych.

Aplikacja musi się dobrze skalować. Powinna zawsze, niezależnie od ilości jednocześnie używających jej osób działać płynnie. Aby to zapewnić serwer, którego używa aplikacja musi działać na sprzęcie, który umożliwi płynne działanie aplikacji przy różnych poziomach obciążenia serwera.

Należy dodatkowo zabezpieczać dane aplikacji wykonując ich regularną archiwizację. Działanie to powinno zapobiec możliwości całkowitej utraty swoich danych, lub czasowemu braku dostępu do nich w przypadku awarii serwera.

# Ogólny opis

Użytkownik rejestruje się do aplikacji tworząc swoje własne konto (w przypadku posiadania już konta nie jest to potrzebne). Następnie po zalogowaniu się na swoje konto uzyskuje dostęp do aplikacji. Użytkownik w tym momencie może albo przejrzeć sprzęt oferowany przez wypożyczalnię, albo przejść do zamawiania konkretnego sprzętu. W pierwszym przypadku klient wybierając odpowiednie opcje może przejrzeć interesujące do pozycje. W drugim wybiera rodzaj sprzętu oraz czas na jaki chce dokonać wypożyczenia. Po wybraniu wymaganych pól aplikacja zwraca informację o cenie oraz pozwala potwierdzić wypożyczenie.

Administracja sklepu, po zalogowaniu się na konto posiadające uprawnienia administracyjne ma możliwość dodawania nowego sprzętu, aktualizacji informacji lub usunięcia konkretnego sprzętu.

# Charakterystyka użytkowników

Z aplikacji będzie mogła korzystać każda osoba która posiada komputer z dostępem do Internetu. Administracja loguje się na swoje konto z uprawnieniami administracyjnymi. Każdy użytkownik musi natomiast utworzyć swoje własne konto, przez które dostaje dostęp do aplikacji.

## Administracja sklepu

* zajmuje się realizacją zamówień:
* przygotowanie sprzętu do wydania,
* kontrola dokonania opłaty przez klienta,
* pracuje w magazynie,
* program obsługuje myszką i klawiaturą,
* posiada wiedzę na temat obsługi aplikacji,
* posiada wiedzę na temat posiadanego sprzętu muzycznego,
* udziela informacji na temat dostępnego sprzetu klientom,
* dba o stan instrumentów.

## Klient

* wybiera interesujący go sprzęt muzyczny,
* określa okres wypożyczenia,
* dokonuje opłaty,
* odbiera sprzęt,
* po upływie czasu wypożyczenia, zwraca sprzęt.

# Główne funkcje produktu:

## Funkcje obsługiwane przez administrację:

* dodawanie sprzętu,
* ustalanie dostępności sprzętu,
* ustalanie cen.

## Funkcje obsługiwane przez klienta:

* przeglądanie dostępnego sprzętu,
* zamówienie konkretnych pozycji,
* określenie ilości wypożyczanego sprzętu,
* wybór okresu na jaki wypożyczony będzie sprzęt,
* wybór sposobu odbioru sprzętu,
* wybór sposobu oddania sprzętu,
* określenie sposobu płatności,
* dokonywanie rezerwacji sprzętu.

# Wymagania funkcjonalne i niefunkcjonalne

## Wymagania funkcjonalne:

* wybór rodzaju sprzętu,
* wybór ilości sprzętu
* określanie długości czasu wypożycznia,
* określenie terminu rozpoczęcia wypożyczenia
* wyświetlanie informacji o sprzęcie,
* wyświetlanie podsuwowania zamówienia,
* możliwość uzyskania dodatkowych informacji (kontakt klienta z administracją wypożyczalni),
* określenie sposobu płatności,
* wprowadzanie nowych pozycji,
* aktualizacja informacji o sprzęcie,
* aktualizacja informacji o dostęponości sprzętu,
* wprowadzanie i aktualizacja cen.

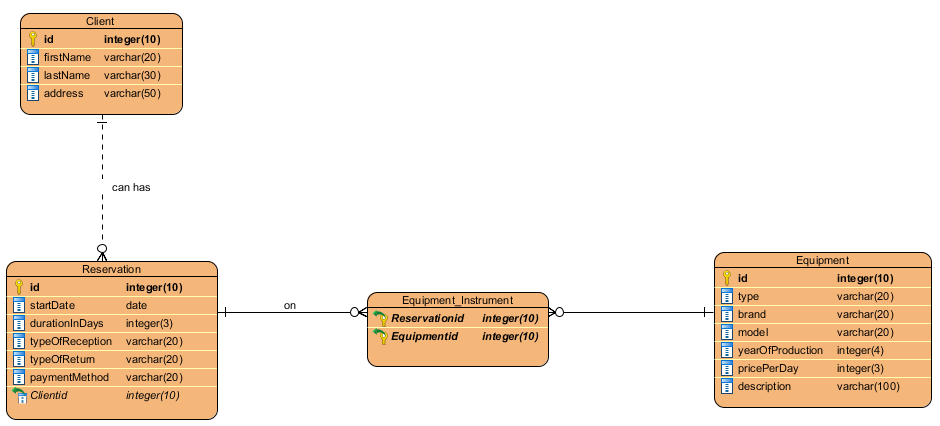
## Wymagania niefunkcjonalne:

* przydatność: wzrost efektywności i szybkości pracy wypożyczalni,
* bezpieczęństwo: oddzielny panel aplikacji dla admina – zapezpieczenie konta przez hasło,
* dostępność: system musi być dostępny bez przerwy   
  w godzinach pracy wypożyczalni, prace administracyjne będą odbywać się po godzinach funkcjonowania wypożyczalni,
* wydajność: aplikacja działa sprawnie i bez długiego oczekiwania na reakcje nawet przy dużej ilości użytkowników,
* skalowalność: system powinien łatwo dać się rozszerzać   
  i mieć możliwość zmiany oraz wprowadzanie nowych funkcji,
* platforma: oprogramowanie - system Windows, Linux,
* integracja z innymi systemami: aplikacja współpracuje   
  z systemem bazodanowym.

# Diagramy operacyjne

## Diagram ERD

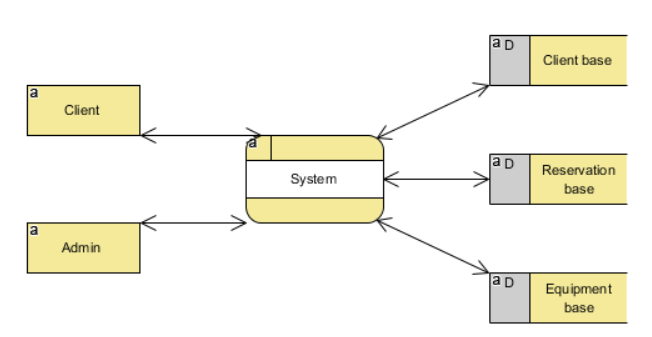
ERD (Entity-Relationship Diagram, diagram związków encji) przedstawia modelowanie danych w systemie.



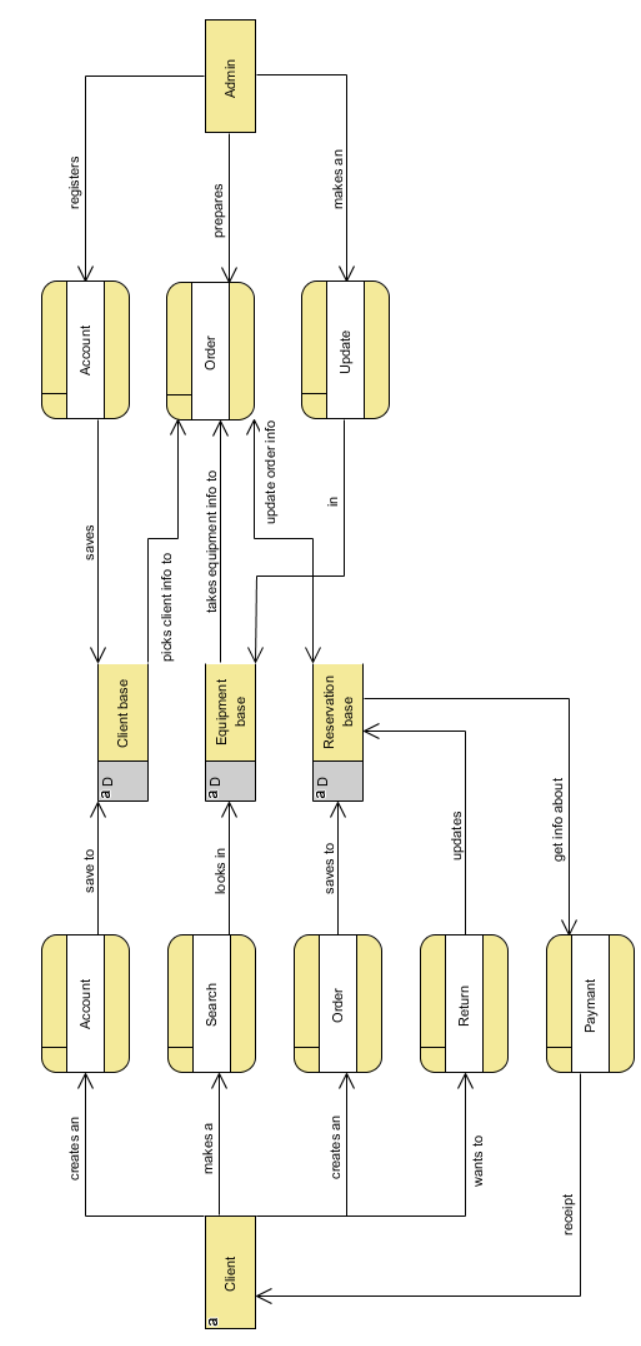
## Diagram DFD

DFD (Data Flow Diagram, diagram przepływu danych) - graf, którego węzły są zwykle procesami, a łuki przepływami danych. Opisuje transformacje danych wewnątrz systemu.

**Poziom 0 – diagram kontekstowy -** określa granicę systemu, czyli system widziany jako czarna skrzynka z zaznaczonymi obiektami zewnętrznymi:



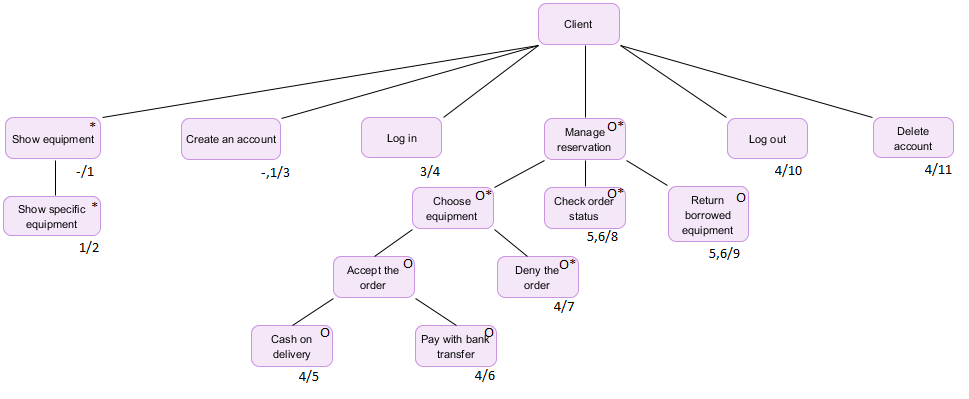
**Poziom 1 – diagram systemowy –** wyróżnia podstawowe procesy systemu oraz magazyny i przepływy danych:



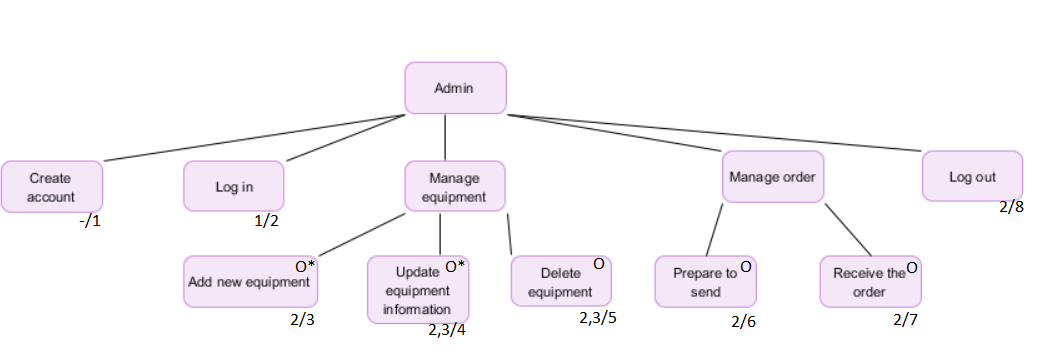
## Diagram ELH

ELH (Entity Life History, diagram historii życia obiektu) – ukazuje zmiany stanu danych w czasie, inaczej mówiąc jest to diagram życia encji.

**klient**

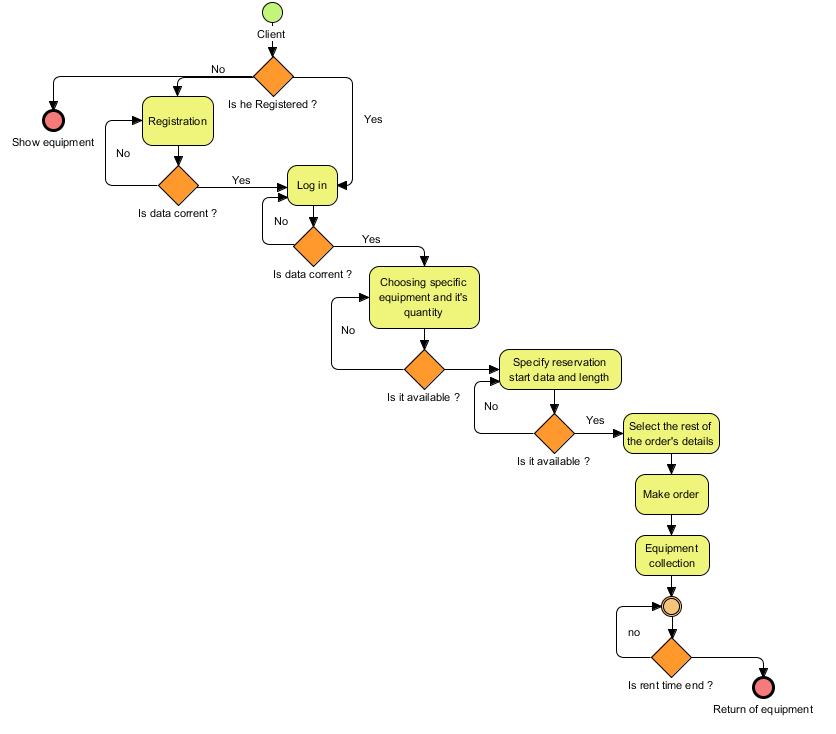


**admin**



## Diagram STD

STD (state diagram, diagram stanów) – pokazuje możliwe stany obiektów i systemu.



# Proces projektowania aplikacji

## Analiza wymagań projektu / aplikacji

Przed przystąpieniem do realizacji projektu wykonaliśmy tzw. „obeznanie rynku”, które pomogłoby nam zorientować się w potrzebach potencjalnych użytkowników aplikacji. Na rynku dostrzegliśmy małą ilość wypożyczalni sprzętu muzycznego, przez co zwróciliśmy też uwagę na wypożyczalnie innego typu. Przeglądając dostępne na rynku wypożyczalni decydowaliśmy o przydatności i łatwości obsługi poszczególnych funkcji.

Realizację projektu rozpoczęliśmy od opisania celów projektu i wymagań. Na tym etapie powstał zarys funkcjonalności aplikacji.

## Specyfikacja wymagań

Kolejnym etapem było rozpisanie szczegółowych wymagań, w których zapisaliśmy istotne dla naszej aplikacji funkcje. Pozwoliło nam to na łatwe zarządzenie projektem w kolejnych etapach. Naszym celem było stworzenie aplikacji która będzie intuicyjna do użytku dla każdego bez względu na wiek i umiejętność korzystania z podobnych aplikacji. Aplikacja wspomaga pracę od strony użytkownika, poprzez możliwość wypożyczenia sprzętu muzycznego bez wychodzenia z domu, jak i korzystna jest dla właściciela powodując przyciągnięcie potencjalnych klientów.

## Tworzenie aplikacji

Prace nad aplikacją toczyły się na podstawie wcześniej utworzonej specyfikacji. Tworzenie aplikacji rozpoczęliśmy od stworzenia GUI (Graphical User Interface), które następnie połączyliśmy z bazą danych za pomocą kontrolerów.

## Testowanie

Testy odbywały się na każdym etapie, między innymi testowaliśmy poszczególne funkcje poza główną aplikacją, sprawdzając czy są one odpowiedni utworzone. Kolejnym etapem testów było sprawdzenie czy zaimplementowane funkcje odpowiednio działają w aplikacji. Ostatecznym etapem testów było przebycie drogi od uruchomienia aplikacji do złożenia zamówienia.

## Wdrożenie aplikacji

Etap ten powinien odbyć się przy udziale głównego administratora wydelegowanego przez właściciela wypożyczalni, który w przyszłości będzie dbał nad zarządzaniem zasobami sklepu. Pokazanie każdej funkcji umożliwia „wyłapanie” pozostałych błędów i naprawienie ich przed ostatecznym wdrożeniem aplikacji. Pozwoli to na odebranie od zamawiającego aplikację opinii dotyczących ostatecznego wyglądu i funkcjonalności aplikacji.

## Konserwacja i opieka nad aplikacją

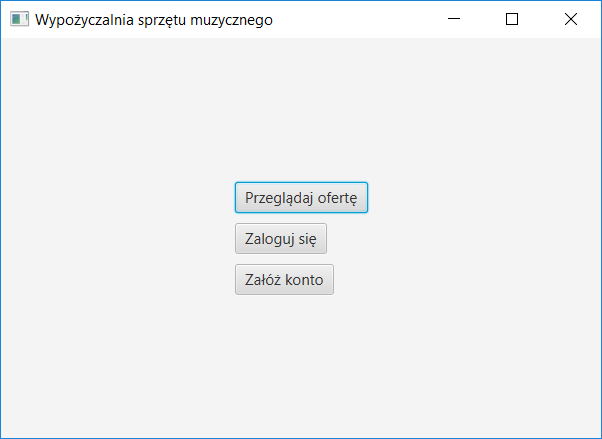
Opiekę nad sprawnym działaniem i ewentualną likwidacją błędów aplikacji przez obowiązujący czas umowy sprawuje dostawca. Natomiast za raportowanie błędów odpowiedzialny jest administratora po stronie wypożyczalni.

## Rozbudowa aplikacji

Aplikacja jest w łatwy sposób rozszerzalna przez co możliwym jest dodanie funkcji w przyszłości nie ingerując w całą strukturę aplikacji. Umożliwia to zmianę team’u deweloperskiego jak i ogranicza koszty, które mogłyby powstać w przypadku tworzenia nowej aplikacji.

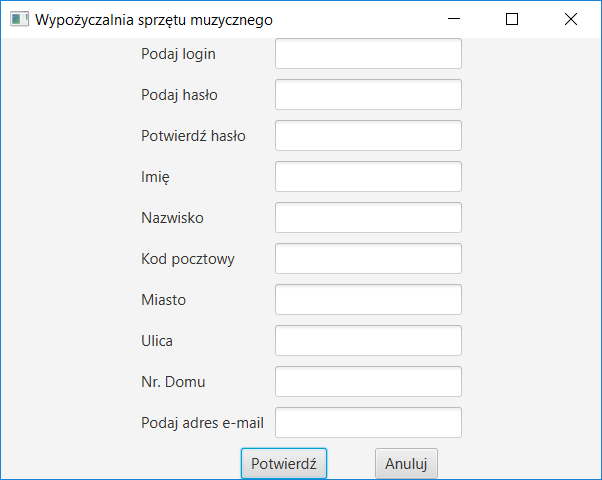
# Scenariusze testowe

* 1. **Ekran powitalny**

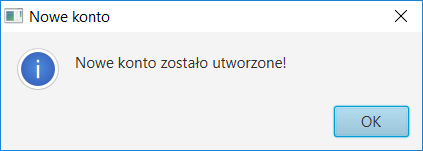
****

* + 1. **Rejestracja użytkownika**

1. Użytkownik uruchamia aplikację.
2. Pojawia się „ekran powitalny”.
3. Przechodzi do zakładki „Załóż konto”.
4. Otwiera się okno z danymi do formularza:



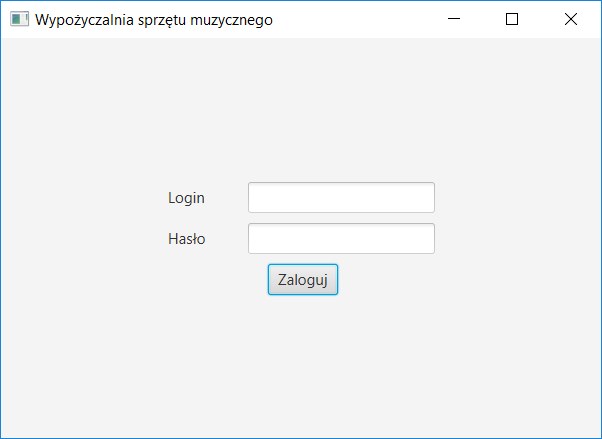
1. Wypełnia wszystkie wymagane dane.
2. Naciska przycisk „Potwierdź”.
3. Wyświetla się komunikat:



1. Po zamknięciu okna poprzez kliknięcie przycisku „OK”, użytkownik może się zalogować do konta.

**Sytuacje wyjątkowe:**

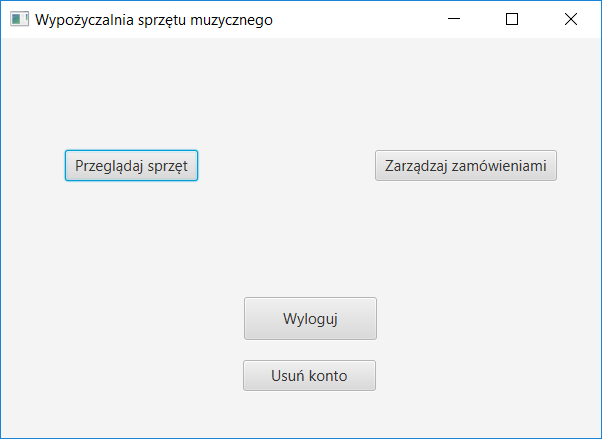
1. Użytkownik nie podał wszystkich wymaganych danych.
2. Istnieje już użytkownik posiadający taki login.
3. Dane są niepoprawne (np. wartości z pola „Potwierdź hasło” nie zgadzają się z wartościami z pola „Podaj hasło”).
   * 1. **Logowanie użytkownika**

****

1. Użytkownik uruchamia aplikację.
2. Wybiera zakładkę „Zaloguj się” z Ekranu powitalnego.
3. Podaje login oraz hasło w odpowiednich polach (wybrane podczas rejestracji).
4. Naciska przycisk „Zaloguj”.
5. Przechodzi do menu użytkownika.

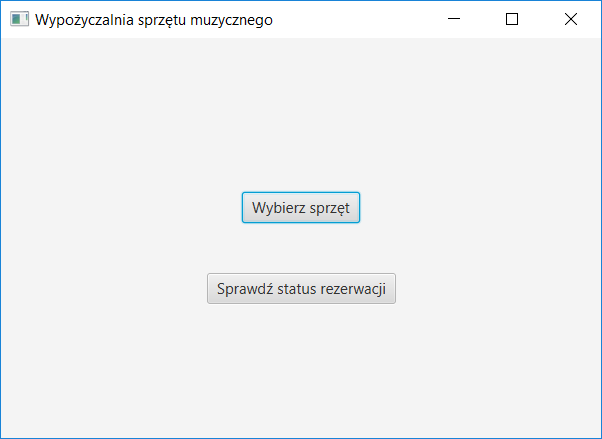
**Sytuacje wyjątkowe:**

1. Użytkownik nie podał wszystkich wymaganych danych.
2. Niepoprawne dane: nie istniej konto z takimi loginem lub złe hasło.
   1. **Panel użytkownika**

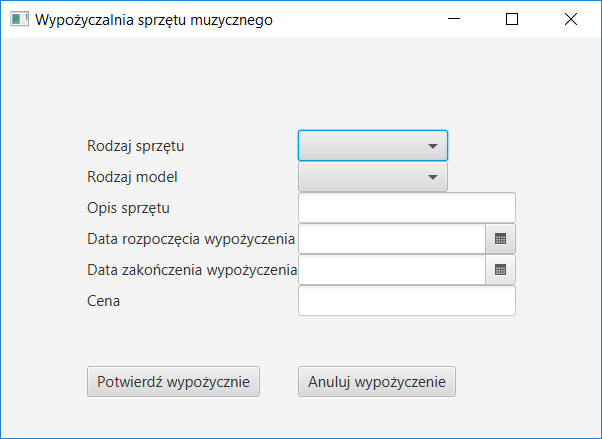


* + 1. **Wypożyczenie sprzętu**

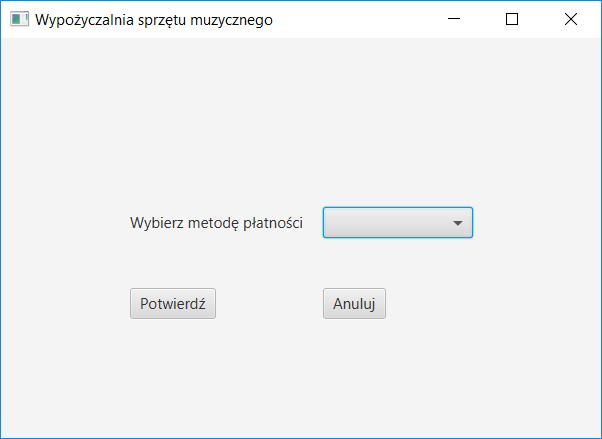
1. Użytkownik klika przycisk „Zarządzaj zamówieniami”.
2. Pojawia się nowe okno:



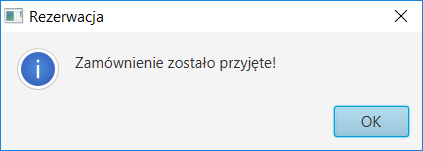
1. Użytkownik klika przycisk „Wybierz sprzęt”
2. Pojawia się nowe okno:



1. Wypełnianie są wszystkie wymagane dane.
2. Użytkownik naciska przycisk „Potwierdź wypożyczenie”.
3. Pojawia się nowe okno:

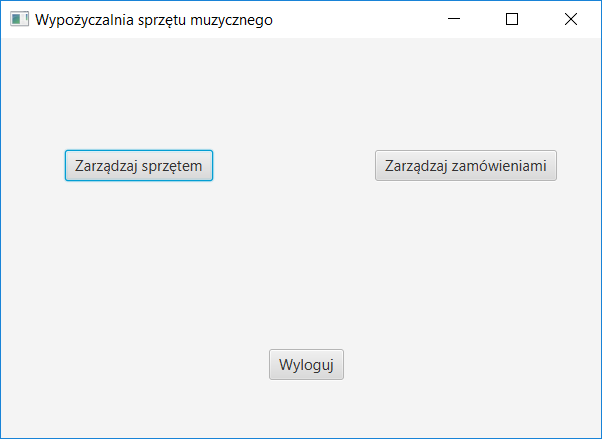


1. Wybiera się metodę płatności z rozwijanego menu.
2. Użytkownik klika przycisk „Potwierdź”.
3. Pojawia się nowe okno:



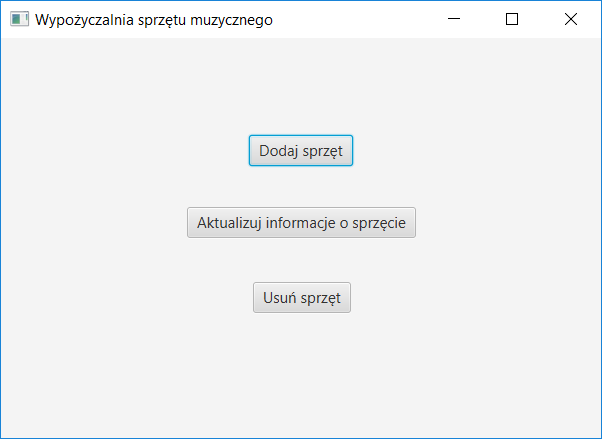
**Sytuacje wyjątkowe:**

1. Użytkownik nie podał wszystkich wymaganych danych w którymś z okien.
   1. **Panel administratora**

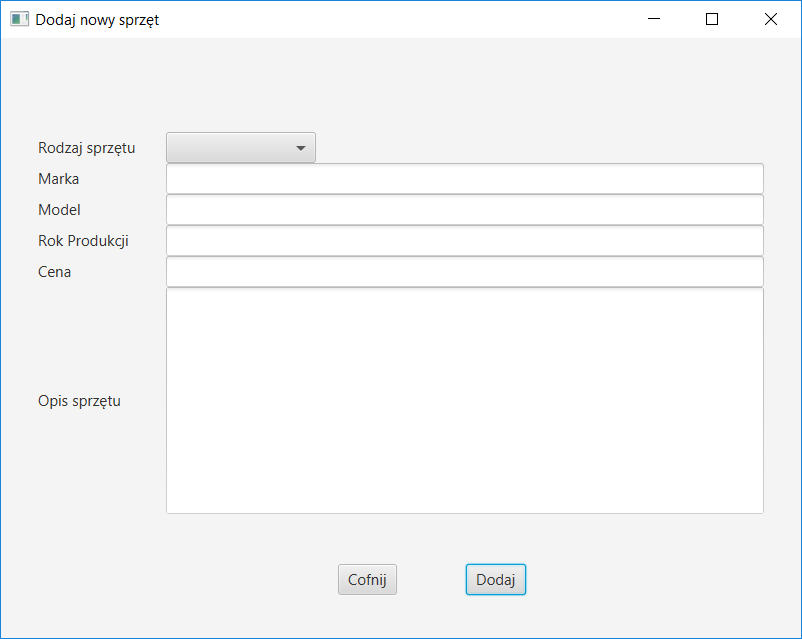
****

* + 1. **Dodanie nowego sprzętu**

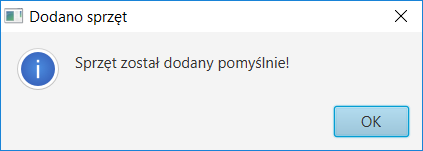
1. Administrator klika przycisk „Zarządzaj sprzętem”.
2. Otwiera się nowe okno:



1. Administrator klika przycisk „Dodaj sprzęt”.
2. Otwiera się nowe okno:



1. Wypełniane są wszystkie pola
2. Administrator klika przycisk „Dodaj”
3. Pojawia się nowe okno:



**Sytuacje wyjątkowe:**

1. Administrator nie podał wszystkich wymaganych danych.