

Kurier

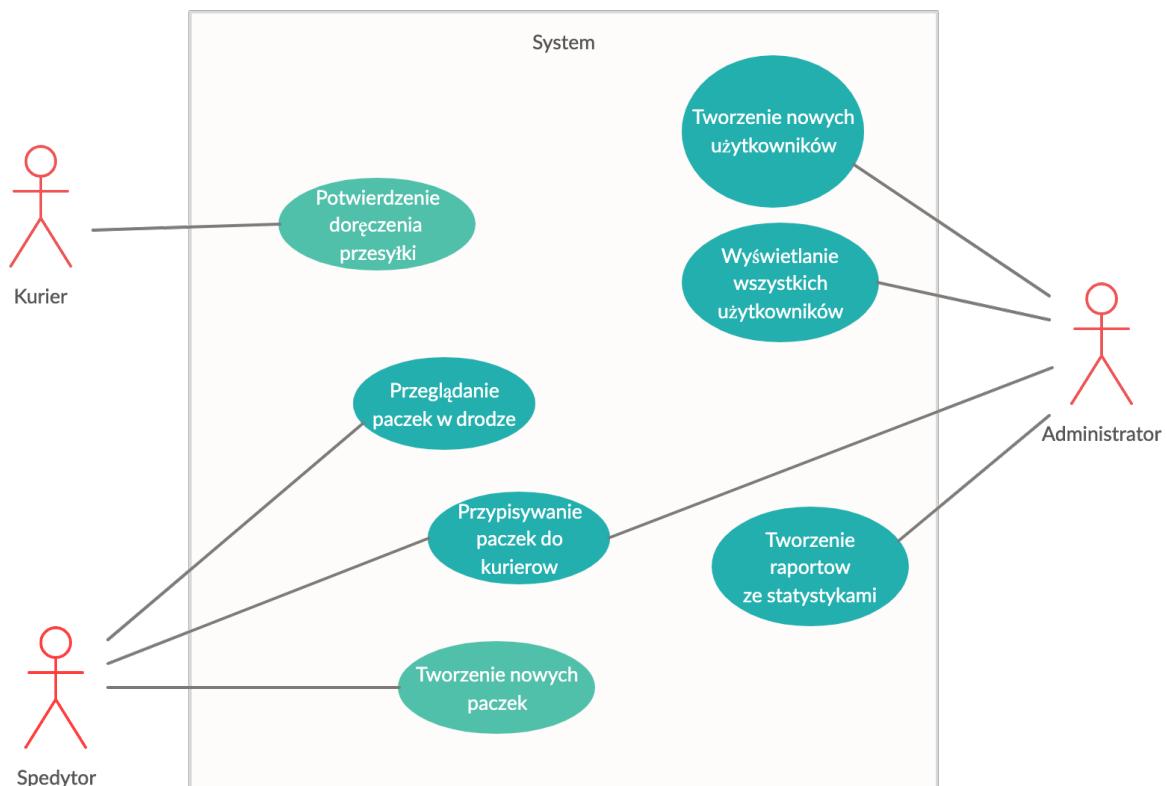
1 SPIS TREŚCI

2	Wprowadzenie	2
3	Wymagania funkcjonalne	2
4	Opis wymagań	4
5	Architektura	6
6	Diagram bazy danych.....	7
7	Zawartość bazy danych.....	8
8	Diagram klas	9
9	Autoryzacja i autentykacja.....	0
10	Przykładowe widoki aplikacji	1
11	Podsumowanie	4

2 WPROWADZENIE

Aplikacja kurier stworzona została na potrzeby wspomagania pracy kurierskiej. Program przeznaczony jest dla małych, średnich jak i dużych instytucji. Głównym założeniem programu jest maksymalne przyśpieszenie procesów wymiany informacji pomiędzy pracownikami firmy. Aplikacja pozwala w szybki i łatwy sposób prowadzić ewidencję przesyłek, śledzić ich status oraz zarządzać nimi.

3 WYMAGANIA FUNKCJONALNE



1) Tworzenie nowych użytkowników

Administrator ma możliwość dodawania nowych użytkowników o typie kurier oraz spedytor

2) Wyświetlanie wszystkich użytkowników

Administrator ma możliwość wyświetlenia wszystkich użytkowników w systemie

3) Przypisywanie paczek do kurierów

Administrator i spedytor może zarządzać pracą kurierów i przypisywać im paczki które należy obsłużyć

4) Tworzenie raportów

Administrator ma możliwość generowania raportów pdf ze statystykami dostarczonych paczek

5) Przeglądanie paczek

Spedytor ma możliwość przeglądania paczek które zostały przypisane do kurierów

6) Tworzenie paczek

Spedytor ma możliwość tworzenia nowych paczek specyfikując ich nazwę i rozmiar

7) Potwierdzenie doręczenia

Kurier po dostarczeniu przesyłki potwierdza w systemie wykonanie zadania

4 OPIS WYMAGAŃ

ID	1
NAZWA	System musi umożliwiać dodawanie nowych użytkowników
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	2
NAZWA	System musi umożliwiać usuwanie użytkowników
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	3
NAZWA	System musi umożliwiać edytowanie użytkowników
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	4
NAZWA	System musi umożliwiać nadawanie uprawnień użytkownikom
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	5
NAZWA	Spedytorzy mogą tworzyć nowe zadania
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	6
NAZWA	Spedytorzy mogą przypisywać zadania do kurierow
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	7
NAZWA	Administrator może tworzyć podsumowanie pracy wszystkich kurierów
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

ID	8
NAZWA	Kurier musi mieć możliwość potwierdzić że paczka została doręczona
PRIORYTET	WYSOKI
TYP	FUNKCJONALNE

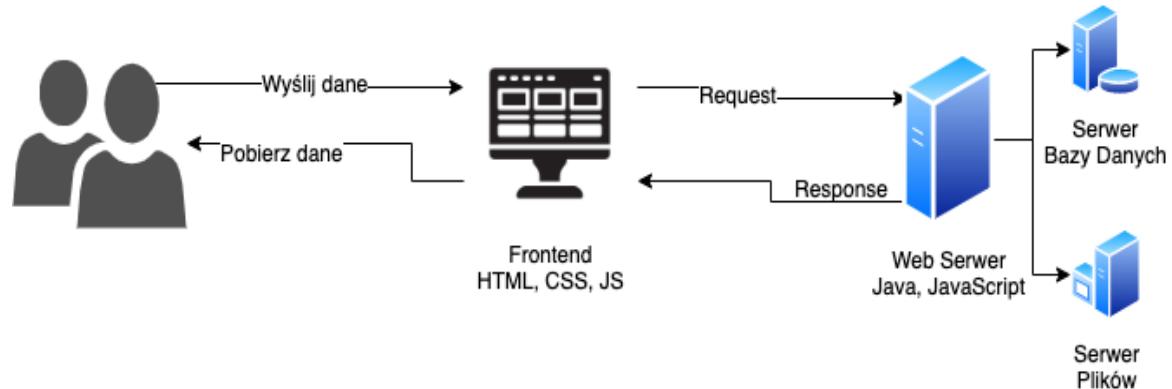
ID	9
NAZWA	Aplikacja skaluje się dla małych i dużych firm
PRIORYTET	WYSOKI
TYP	NIEFUNKCJONALNE

ID	10
NAZWA	Aplikacja zabezpieczona przed nieautoryzowanym dostępem
PRIORYTET	WYSOKI
TYP	NIEFUNKCJONALNE

5 ARCHITEKTURA

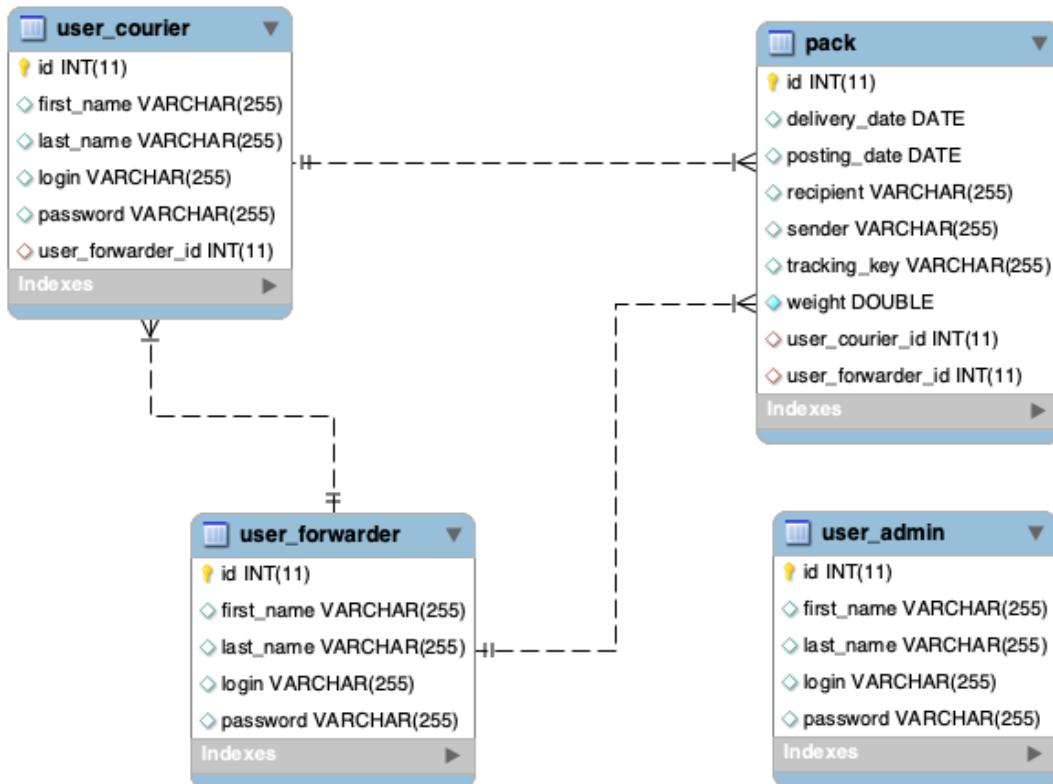
Wybrana architektura przedstawia aplikacje internetową. Jest to program, który pracuje na serwerze i komunikuje się z użytkownikiem poprzez sieć komputerową z wykorzystaniem przeglądarki internetowej. Przeglądarka internetowa w takim przypadku pełni funkcję interfejsu użytkownika. W aplikacji internetowej zakłada się interakcję z użytkownikiem, korzystanie z baz danych i innych usług, często rozproszonych, umieszczonych na wielu różnych serwerach. Istotnymi cechami aplikacji internetowej są łatwość i szybkość dotarcia do informacji (dowolny komputer podłączony do Internetu lub sieci LAN i przeglądarka stron internetowych) oraz bezpieczeństwo danych (login i hasło, szyfrowanie połączenia oraz system uprawnień). Zalety wybranego rozwiązania:

- **dostępność dla wszystkich zainteresowanych bez ograniczeń związanych z czasem i miejscem,**
- **praktycznie brak konieczności instalowania dodatkowego oprogramowania - aby korzystać z aplikacji internetowej wystarczy przeglądarka internetowa (czasem tylko z dodatkowymi pluginami),**
- **brak konieczności zakupu dodatkowego sprzętu komputerowego - aplikacja internetowa może zostać zainstalowana na serwerze dostępnym w Internecie,**
- **łatwość utrzymania i modernizacji, ponieważ zmiany i aktualizacje w aplikacji są wykonywane na serwerze bez udziału użytkowników aplikacji,**
- **łatwość integracji aplikacji internetowej z innymi usługami, np. stroną internetową,**
- **niższe koszty wytworzenia, uruchomienia i utrzymania aplikacji internetowej w porównaniu z innymi rozwiązaniami - większość technologii stosowanych do wytworzenia aplikacji internetowych jest bezpłatna.**



6 DIAGRAM BAZY DANYCH

Przy modelowaniu baz danych możemy posłużyć się notacją graficzną modelowania danych – diagramem związków encji (ERD, ang. Entity-Relationship Diagram). Jest to model sieciowy opisujący na wysokim poziomie abstrakcji dane, które są przechowywane w systemie. Model ERD budowany jest przez analityka. Służy on do zobrazowania w sposób zrozumiały zarówno dla projektanta, jak i osoby niemającej wykształcenia informatycznego (np. klienta) obiektów i związków zachodzących w projektowanej dziedzinie problemowej. Model ERD nie jest związany z konkretną implementacją systemu (np. na serwerze MS SQL czy Oracle), choć jego odmiany mogą zawierać informacje specyficzne dla danego języka lub środowiska implementacyjnego. Staje się on wówczas modelem projektowym



7 ZAWARTOŚĆ BAZY DANYCH

Aplikacja na starcie generuje przykładowe dane wejściowe, z poziomu bazy danych można to zrobić używając następujących komend.

```
SELECT * FROM courierService.user_admin;
```

```
'1','admin','admin','6000','6000'
```

```
SELECT * FROM courierService.user_courier;
```

```
'1','userCourier- 0','userCourier- 0','4000','4000','1'
```

```
'2','userCourier- 1','userCourier- 1','4001','4001','1'
```

```
'3','userCourier- 2','userCourier- 2','4002','4002','1'
```

```
'4','userCourier- 3','userCourier- 3','4003','4003','1'
```

```
'5','userCourier- 4','userCourier- 4','4004','4004','1'
```

```
SELECT * FROM courierService.user_forwarder;
```

```
'1','userForwarder','userForwarder','5000','5000'
```

```
SELECT * FROM courierService.pack;
```

```
'1',NULL,'2020-01-13','pack- 0','pack- 0',NULL,'0','3','1'
```

```
'2',NULL,'2020-01-13','pack- 1','pack- 1',NULL,'0','1','1'
```

```
'3','2020-01-13','2020-01-12','pack- 2','pack- 2',NULL,'0','2','1'
```

```
'4',NULL,'2020-01-13','pack- 3','pack- 3',NULL,'0','1','1'
```

```
'5','2020-01-13','2020-01-12','pack- 4','pack- 4',NULL,'0','2','1'
```

```
'6',NULL,'2020-01-13','pack- 5','pack- 5',NULL,'0',NULL,'1'
```

```
'7',NULL,'2020-01-13','pack- 6','pack- 6',NULL,'0','2','1'
```

```
'8',NULL,'2020-01-13','pack- 7','pack- 7',NULL,'0',NULL,'1'
```

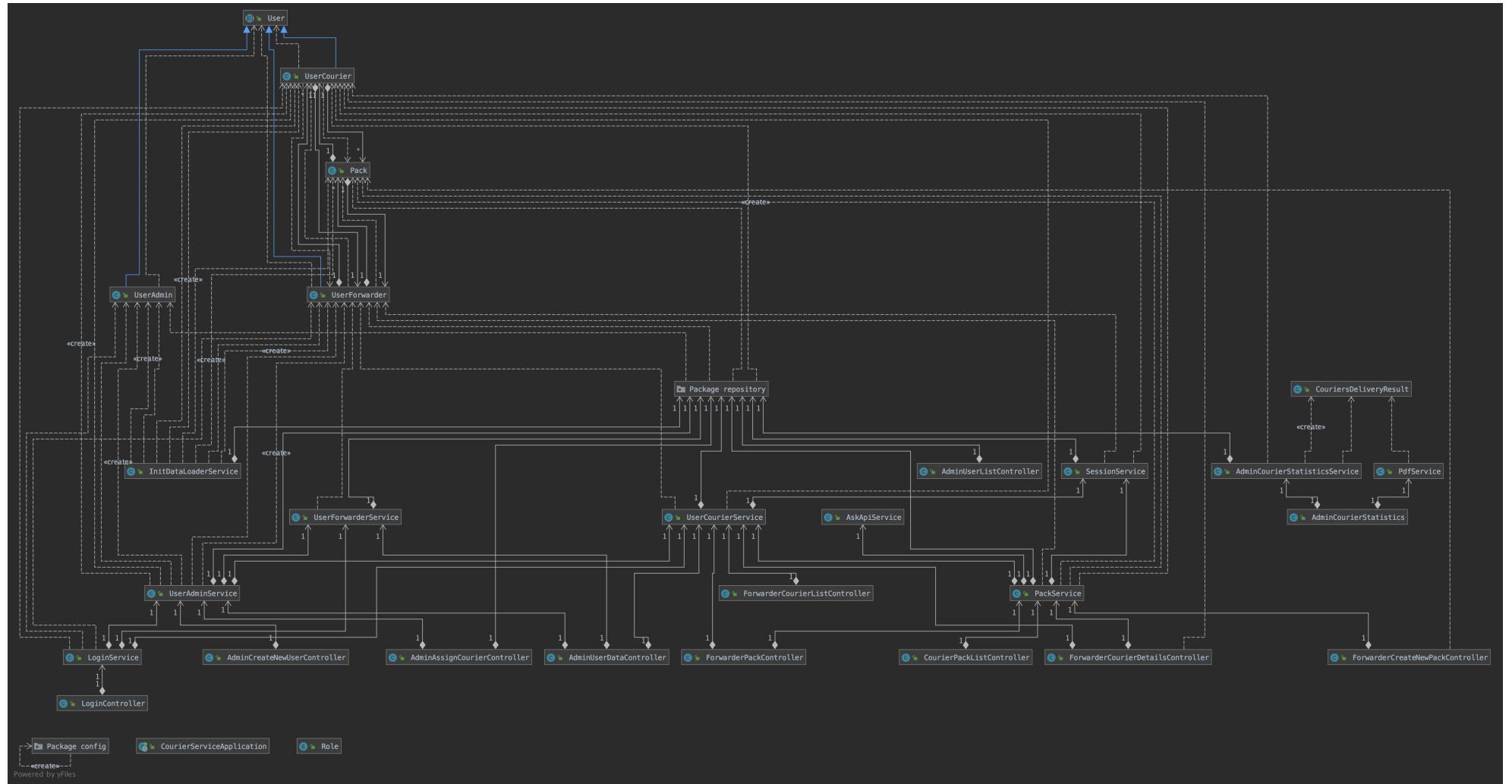
```
'9',NULL,'2020-01-13','pack- 8','pack- 8',NULL,'0',NULL,'1'
```

```
'10',NULL,'2020-01-13','pack- 9','pack- 9',NULL,'0','1','1'
```

Klasa odpowiedzialna za inicjalizację danych przy starcie aplikacji to: `InitDataService.class`

8 DIAGRAM KLAS

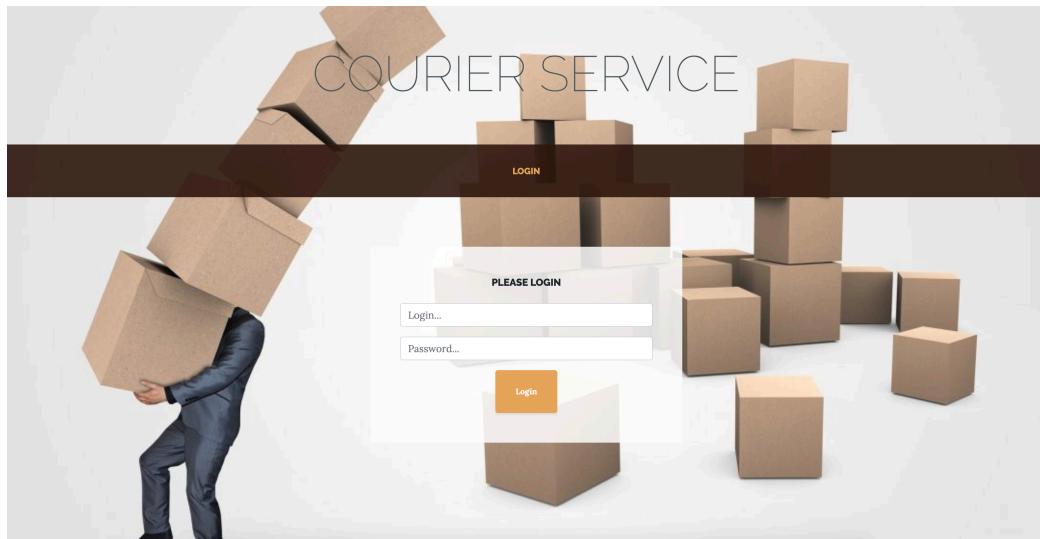
Diagram klas obrazuje pewien zbiór klas, interfejsów i kooperacji oraz związki między nimi. Jest on grafem złożonym z wierzchołków (klas, interfejsów, kooperacji) i łącz (reprezentowanych przez relacje). Diagram klas stanowi opis statyki systemu, który uwypukla związki między klasami, pomijając pozostałe charakterystyki. Najsilniej prezentuje on więc strukturę systemu, stanowiąc podstawę dla jego konstrukcji. W modelowaniu złożonych systemów nie mamy obowiązku przedstawiania ich struktury na jednym diagramie. Powinniśmy pamiętać o tym, że złożenie wszystkich diagramów, a właściwie ich elementów i relacji, stanowi kompletny model. Możemy zatem przyjąć, że podzbiory zbioru klas użyte na diagramach klas są wybierane celowo i stanowią wynik decyzji zarówno analitycznych, jak i projektowych. Tak na przykład, diagramy klas stanowiące wynik decyzji analitycznych tworzą tzw. widoki klas biorących udział w realizacji danego przypadku użycia (ang. view of participating class – VOPC) i powiązania między nimi. Zasadniczo jednak diagramy klas służą do zobrazowania statycznych aspektów perspektywy projektowej, w której bierze się pod uwagę wymagania funkcjonalne systemu – usługi, jakie system powinien udostępniać swoim użytkownikom.



9 AUTORYZACJA I AUTENTYKACJA

Do autoryzacji i autentykacji aplikacja wykorzystuje Spring Security. Baza danych przechowuje dane użytkowników wraz z przypisanymi do nich rolami. W aplikacji wyróżnionych jest kilka ról. Dostęp do poszczególnych widoków posiadają użytkownicy z określonymi rolami.

10 PRZYKŁADOWE WIDOKI APLIKACJI



Rysunek 1 Moduł logowania

A screenshot of the same Courier Service application showing the "USER LIST" tab selected in the navigation bar. The navigation bar also includes "CREATE NEW USER", "ASSIGN COURIER", "COURIER STATISTICS", and "LOGOUT". The main content area displays a table with columns "First name", "Last name", and "Packages delivered". The data is as follows:

First name	Last name	Packages delivered
userCourier- 0	userCourier- 0	0
userCourier- 1	userCourier- 1	2
userCourier- 2	userCourier- 2	0
userCourier- 3	userCourier- 3	0
userCourier- 4	userCourier- 4	0
aaa	aaa	0

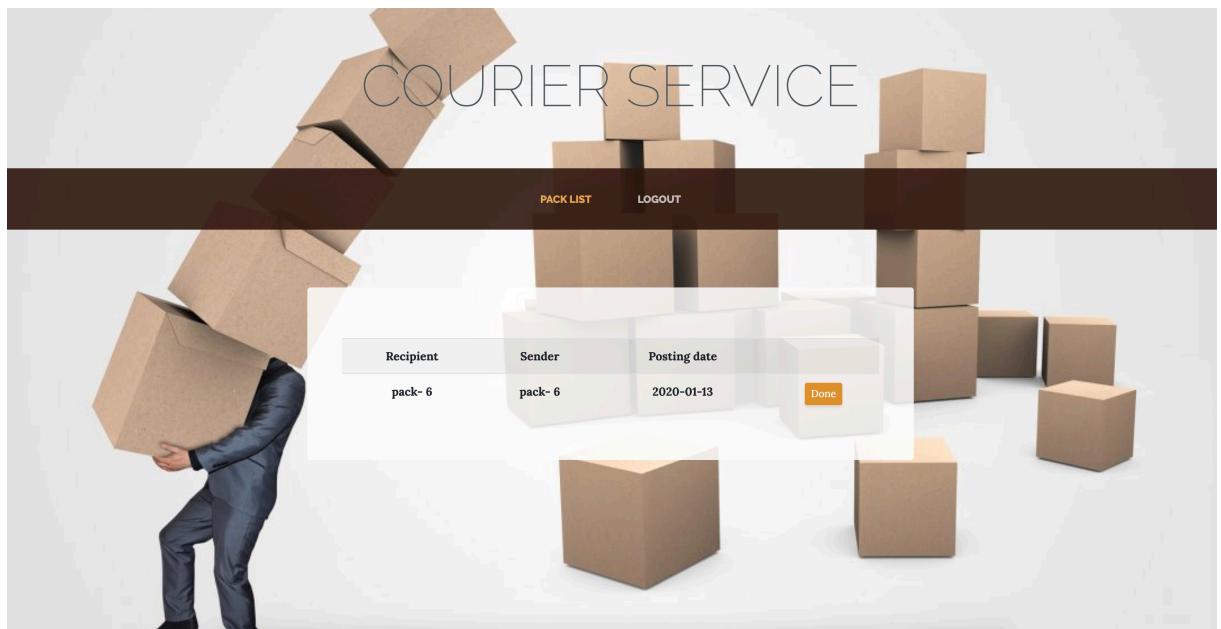
[Generate report](#)

Rysunek 2 Lista kurierów

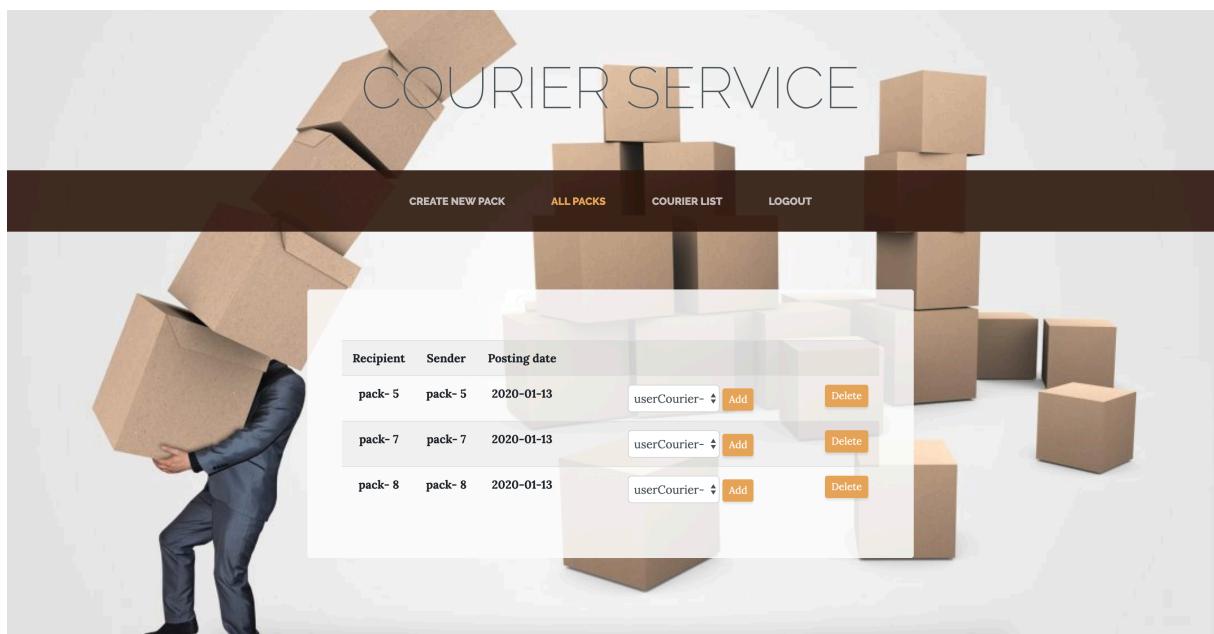
Courier delivery result report

Number	First name	Last name	Packages delivered
1	userCourier- 0	userCourier- 0	0
2	userCourier- 1	userCourier- 1	0
3	userCourier- 2	userCourier- 2	0
4	userCourier- 3	userCourier- 3	0
5	userCourier- 4	userCourier- 4	0
			0

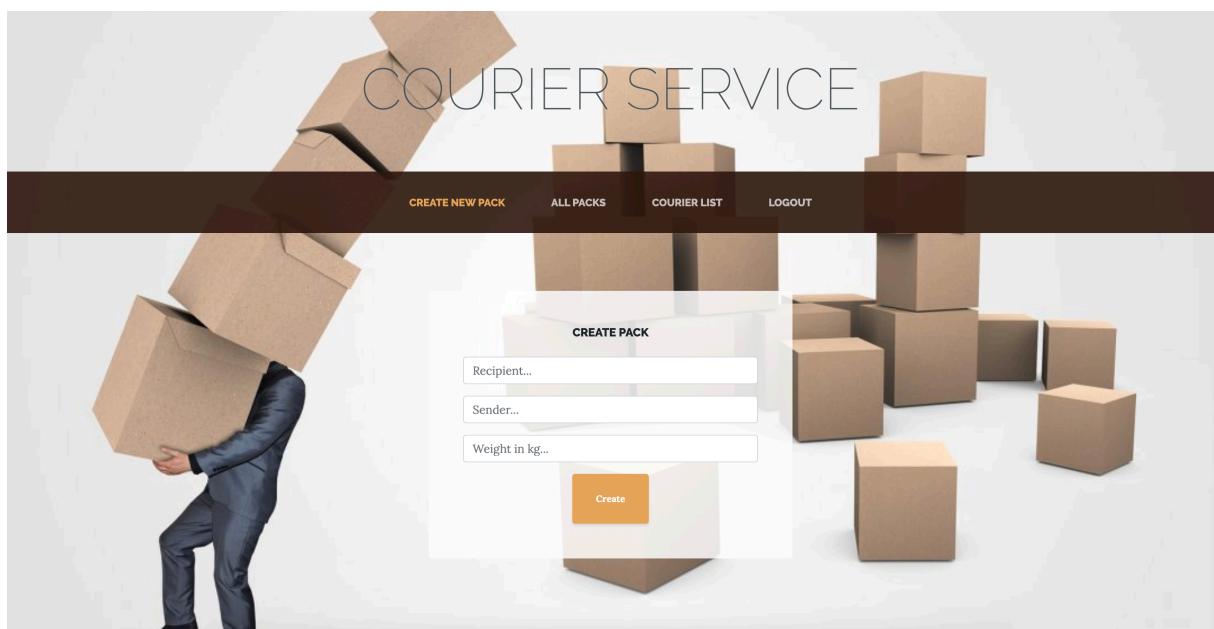
Rysunek 3 Wygenerowany raport



Rysunek 4 Potwierdzenie dostarczenia paczki



Rysunek 5 Przypisanie paczki do kuriera



Rysunek 6 Dodanie nowej paczki

11 PODSUMOWANIE

Prowadzenie firmy to nie tylko realizacja usług, ale również rozliczenia z klientami, odbiorcami oraz pracownikami. Nasz system ułatwia i automatyzuje prace w tym zakresie. Nad usystematyzowanymi procesami można szybko zapanować i łatwiej je skontrolować. Rosnąca liczba zamówień to dodatkowe zadania przy ich obsłudze: ewidencja zleceń, przekazywanie informacji, koordynacja realizacji usługi. Ktoś musi tę pracę wykonać, ale nie musi ona być czasochłonna. Ze względu na wybraną architekturę aplikacja jest dostępna na wszystkie platformy co czyni to narzędzie jeszcze bardziej uniwersalne. Do prawidłowego działania potrzebujemy jedynie sprawnej przeglądarki internetowej. Przedstawiona aplikacja ukazuje podstawową funkcjonalność, którą napisać może nawet małe biuro programistyczne.