

# Projekt bazy danych dla aplikacji internetowej

Zarządzanie firmą świadczącą usługi foto-video

Łukasz Setlak  
2024-12-13

## Spis treści

1.	OPIS PROJEKTU .....	2
2.	GŁÓWNE FUNKCJE SYSTEMU .....	2
3.	DIAGRAM .....	3
4.	STRUKTURA BAZY DANYCH.....	4
4.1.	Tabele użytkownika .....	4
4.1.1.	users.....	4
4.1.2.	roles .....	4
4.1.3.	user_role .....	5
4.1.4.	messages.....	5
4.1.5.	reservation_employee .....	6
4.2.	Tabele rezerwacji .....	7
4.2.1.	resrvation .....	7
4.2.2.	status.....	8
4.2.3.	service_type .....	8
4.2.4.	reservation_type.....	9
4.2.5.	reservation_details .....	9
4.2.6.	details_wedding.....	10
4.2.7.	details_photoshoot.....	11
4.2.8.	details_baptism.....	11
4.2.9.	details_other.....	12
4.2.10.	reservation_cancellation .....	12
4.3.	Tabele wyposażenia .....	13
4.3.1.	equipment.....	13
4.3.2.	equipment_type .....	13
4.3.3.	equipment_manufacturer.....	14
4.3.4.	equipment_compability .....	14
4.3.5.	equipment_condition_log.....	15
4.3.6.	reservation_equipment .....	15
5.	MODUŁY .....	16
5.1.	Moduł użytkownika .....	16
5.2.	Moduł wyposażenia .....	17
5.3.	Moduł rezerwacji.....	18
6.	PROCEDURY .....	20
6.1.	AddReservation .....	20
6.2.	AddDetailsWedding.....	21
6.3.	AddDetailsPhotoshoot .....	21
6.4.	AddDetailsBaptism .....	21
6.5.	AddDetailsOther.....	22
6.6.	AssignEmployeeToReservation .....	22
6.7.	CancelReservation.....	23
7.	WIDOKI.....	24
7.1.	ActiveUsersWithRoles .....	24
7.2.	ReservationView .....	24
7.3.	EmployeeReservationsView .....	25
8.	TRIGGERY .....	26
8.1.	PreventUserDeletion .....	26
8.2.	LogEquipmentConditionChange.....	26

## **1. Opis projektu**

System umożliwia zarządzanie rezerwacjami usług fotograficznych i filmowych, wspiera organizację pracowników oraz komunikację między klientami a firmą. Umożliwia także śledzenie dostępnego w firmie sprzętu i jego stanu. Został stworzony z myślą o firmie skupiającej się na realizowaniu zleceń ślubnych, ale dzięki modułowej strukturze nie ogranicza się wyłącznie do nich i pozwala na łatwą rozbudowę.

## **2. Główne funkcje systemu**

### **1. Zarządzanie użytkownikami i rolami**

- Rejestrowanie klientów oraz pracowników
- Nadawanie ról i uprawnień
- Dezaktywacja konta

### **2. Rezerwacja usług**

- Dodawanie nowych rezerwacji z uwzględnieniem szczegółowych informacji specyficznych dla wybranego typu zlecenia
- Przypisywanie statusów rezerwacji oraz możliwość jej anulowania

### **3. Komunikacja**

- Wysyłanie i odbieranie wiadomości między klientami a pracownikami
- Historia korespondencji

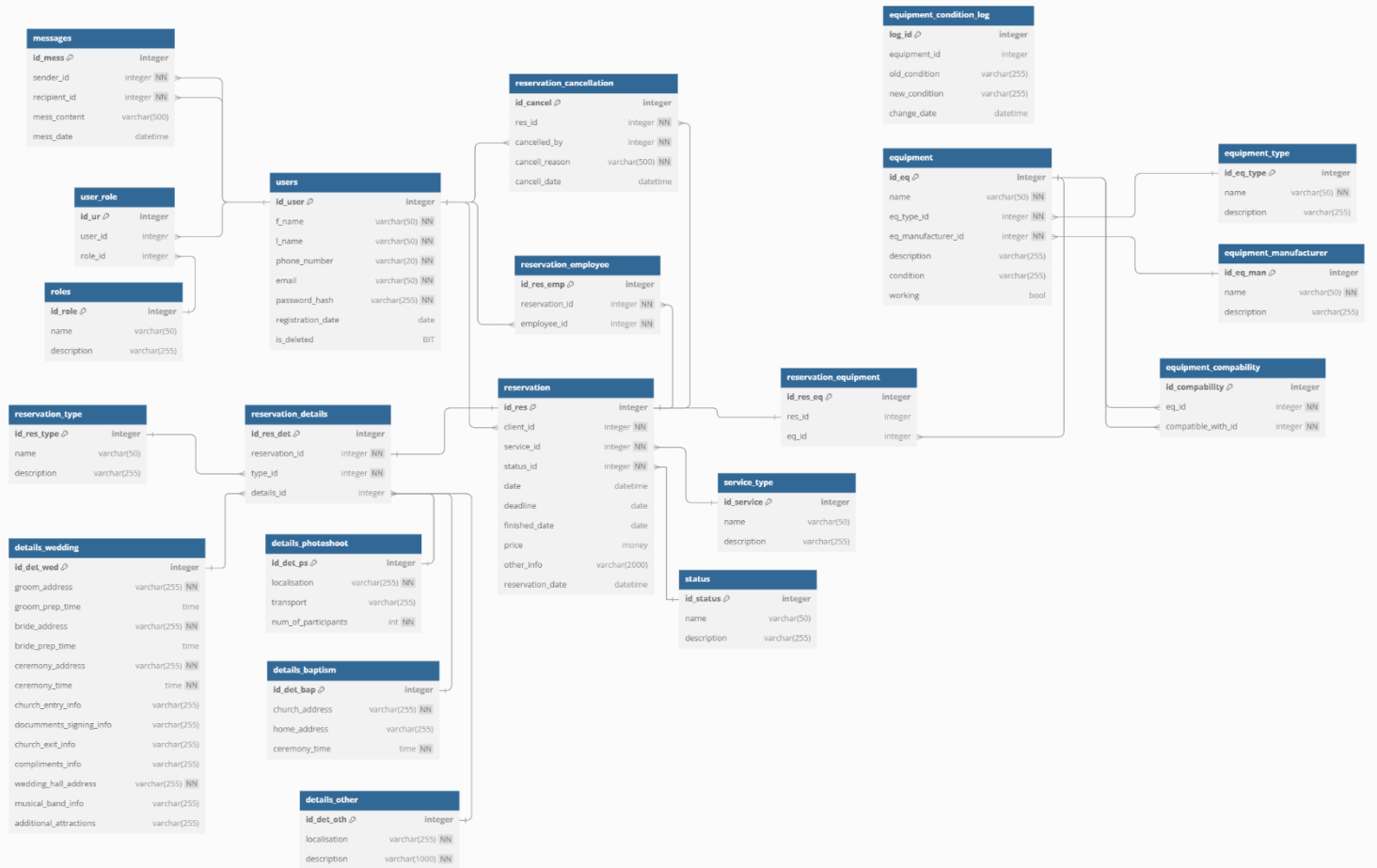
### **4. Zarządzanie wyposażeniem**

- Dodawanie i edycja wyposażenia firmy
- Monitorowanie stanu technicznego sprzętu oraz historia jego użytkowania
- Możliwość określenia i sprawdzania kompatybilności poszczególnych części wyposażenia
- Przypisywanie wybranego wyposażenia do konkretnego zlecenia

### **5. Zarządzanie pracownikami**

- Przypisywanie pracowników do konkretnych rezerwacji.
- Możliwość wyboru przez pracownika sprzętu potrzebnego do realizacji zlecenia.

### 3. Diagram



## 4. Struktura bazy danych

### 4.1. Tabele użytkownika

#### 4.1.1. users

Tabela przechowuje dane użytkowników systemu.

- id\_user – unikalny identyfikator użytkownika (klucz główny).
- f\_name, l\_name – imię i nazwisko użytkownika.
- phone\_number – numer telefonu (unikalny).
- email – adres email (unikalny).
- password\_hash – zaszyfrowane hasło.
- registration\_date – data rejestracji.
- is\_deleted – status użytkownika (0: aktywny/ 1: usunięty).

```
CREATE TABLE users (  
    id_user INT PRIMARY KEY IDENTITY,  
    f_name VARCHAR(50) NOT NULL,  
    l_name VARCHAR(50) NOT NULL,  
    phone_number VARCHAR(20) NOT NULL UNIQUE,  
    email VARCHAR(50) NOT NULL UNIQUE,  
    password_hash VARCHAR(255) NOT NULL,  
    registration_date DATETIME DEFAULT CURRENT_TIMESTAMP,  
    is_deleted BIT DEFAULT 0  
)
```

	id_user	f_name	l_name	phone_number	email	password_hash	registration_date	is_deleted
1	1	Jan	Kowalski	123456789	jan.kowalski@example.com	hashed_password_1	2024-12-05 15:36:19.547	0
2	2	Anna	Nowak	987654321	anna.nowak@example.com	hashed_password_2	2024-12-05 15:36:19.547	0
3	3	Piotr	Zielinski	555666777	piotr.zielinski@example.com	hashed_password_3	2024-12-05 15:36:19.547	0

#### 4.1.2. roles

Lista ról przypisywanych użytkownikom

- id\_role – identyfikator roli (klucz główny).
- name – nazwa roli (np. "fotograf").
- description – opis roli.

```
CREATE TABLE roles (  
    id_role INT PRIMARY KEY IDENTITY,  
    name VARCHAR(50) NOT NULL,  
    description VARCHAR(255)  
)
```

	id_role	name	description
1	1	administrator	użytkownik z uprawnieniami do zarządzania użytkownik...
2	2	fotograf	użytkownik będący pracownikiem, mogący wykonyw...
3	3	kamerzysta	użytkownik będący pracownikiem, mogący wykonyw...
4	4	klient	użytkownik mający możliwość korzystania z czatu i dok...

#### 4.1.3. user\_role

Tabela wiążąca użytkowników z ich rolami. Jeden użytkownik może mieć więcej niż jedną rolę.

- id\_ur - identyfikator (klucz główny).
- user\_id - identyfikator (klucz główny).
- role\_id - identyfikator roli (klucz obcy do roles).

```
CREATE TABLE user_role (  
    id_ur INT PRIMARY KEY IDENTITY,  
    user_id INT,  
    role_id INT,  
  
    FOREIGN KEY (user_id) REFERENCES users(id_user),  
    FOREIGN KEY (role_id) REFERENCES roles(id_role)  
)
```

	id_ur	user_id	role_id
1	1	1	1
2	2	2	3
3	3	2	2
4	4	3	4

#### 4.1.4. messages

Przechowuje wiadomości przesyłane między użytkownikami.

- id\_mess - identyfikator wiadomości (klucz główny).
- sender\_id, recipient\_id - identyfikatory nadawcy i odbiorcy (klucze obce do users).
- mess\_content - treść wiadomości.
- mess\_date - data wysłania wiadomości.

```
CREATE TABLE messages (  
    id_mess INT PRIMARY KEY IDENTITY,  
    sender_id INT NOT NULL,  
    recipient_id INT NOT NULL,  
    mess_content VARCHAR(500),  
    mess_date DATETIME DEFAULT CURRENT_TIMESTAMP,  
  
    FOREIGN KEY (sender_id) REFERENCES users(id_user),  
    FOREIGN KEY (recipient_id) REFERENCES users(id_user)  
)
```

	id_mess	sender_id	recipient_id	mess_content	mess_date
1	1	3	1	Witam, chciałbym zarezerwować termin na sesję fo...	2024-12-03 12:46:20.000
2	2	1	3	Oczywiście, proszę utworzyć nową rezerwację, a n...	2024-12-03 12:49:57.000
3	3	3	4	Cześć Łukasz, mam pytanie odnośnie rezerwacji se...	2024-12-04 09:15:00.000

#### 4.1.5. *reservation\_employee*

Lista pracowników (użytkownicy z odpowiednią rolą, np. fotograf lub kamerzysta) przypisanych do danej rezerwacji zlecenia.

- `id_res_emp` – identyfikator (klucz główny)
- `reservation_id` – identyfikator rezerwacji (klucz obcy do `reservation`).
- `Employee_id` – identyfikator użytkownika (klucz obcy do `users`).

```
CREATE TABLE reservation_employee (  
  id_res_emp INT PRIMARY KEY IDENTITY,  
  reservation_id INT NOT NULL,  
  employee_id INT NOT NULL,  
  
  FOREIGN KEY (reservation_id) REFERENCES reservation(id_res),  
  FOREIGN KEY (employee_id) REFERENCES users(id_user)  
)
```

	id_res_emp	reservation_id	employee_id
1	1	1	4

## 4.2. Tabele rezerwacji

### 4.2.1. reservation

Tabela przechowuje wszystkie rezerwacje zleceń.

- id\_res – identyfikator rezerwacji (klucz główny).
- client\_id – identyfikator klienta (klucz obcy do users).
- service\_id – typ usługi (klucz obcy do service\_type).
- status\_id – status rezerwacji (klucz obcy do status).
- date – data realizacji usługi.
- deadline – termin realizacji zlecenia.
- finished\_date – data zakończenia zlecenia.
- price – koszt usługi.
- other\_info – dodatkowe informacje.
- reservation\_date – data utworzenia rezerwacji.

```
CREATE TABLE reservation (  
  id_res INT PRIMARY KEY IDENTITY,  
  client_id INT NOT NULL,  
  service_id INT NOT NULL,  
  status_id INT NOT NULL DEFAULT 1,  
  date DATETIME,  
  deadline DATE,  
  finished_date DATE,  
  price MONEY,  
  other_info VARCHAR(2000),  
  reservation_date DATETIME DEFAULT CURRENT_TIMESTAMP,  
  
  FOREIGN KEY (client_id) REFERENCES users(id_user),  
  FOREIGN KEY (service_id) REFERENCES  
service_type(id_service),  
  FOREIGN KEY (status_id) REFERENCES status(id_status)  
)
```

	id_res	client_id	service_id	status_id	date	deadline	finished_date	price	other_info	reservation_date
1	1	3	1	1	NULL	NULL	NULL	500.00	Sesja zdjęciowa narzeczeńska w plenerze, zależy n...	2024-12-05 15:36:19.563
2	2	3	3	1	2025-06-15 00:00:00.000	NULL	NULL	8000.00	Reportaż FOTO-VIDEO na weselu	2024-12-05 15:36:19.570
3	3	5	1	2	2024-09-20 00:00:00.000	NULL	NULL	1000.00	Ceremonia chrztu, fotografia w kościele i w domu.	2024-12-05 15:36:19.570



#### 4.2.2. status

Lista statusów przypisywanych rezerwacji.

- id\_status – identyfikator statusu (klucz główny).
- name – nazwa statusu.
- description – opis statusu.

```
CREATE TABLE status (  
  id_status INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  description VARCHAR(255)  
)
```

	id_status	name	description
1	1	Oczekujące	Zlecenie oczekuje na przypisanie odpowiednich pra...
2	2	W trakcie ustaleń	Przypisano odpowiednich pracowników, trwa ustala...
3	3	Anulowano	Zlecenie zostało anulowane przed rozpoczęciem je...
4	4	Potwierdzone	Data i lokalizacja zlecenia zostały ustalone z kliente...
5	5	W trakcie postprodukcji	materiał zdjęciowy/filmowy został wykonany i jest w...
6	6	Gotowy	Gotowy materiał jest gotowy do odbioru

#### 4.2.3. service\_type

Lista typów zlecenia przypisanych do rezerwacji.

- id\_service – identyfikator typu zlecenia (klucz główny).
- name – nazwa typu zlecenia.
- description – opis typu zlecenia.

```
CREATE TABLE service_type (  
  id_service INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  description VARCHAR(255)  
)
```

	id_service	name	description
1	1	FOTO	Zlecenie do którego wykonania potrzebny jest foto...
2	2	VIDEO	Zlecenie do którego wykonania potrzebny jest kam...
3	3	FOTO-VIDEO	Zlecenie do którego wykonania potrzebny jest zaró...

#### 4.2.4. reservation\_type

Lista typów wydarzenia wybranego przez klienta przy dokonaniu rezerwacji. (Np. ślub, sesja zdjęciowa, chrzest, inne).

- id\_res\_type – identyfikator typu rezerwacji (klucz główny).
- name – nazwa typu rezerwacji.
- description – opis typu rezerwacji.

```
CREATE TABLE reservation_type (  
  id_res_type INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  description VARCHAR(255)  
)
```

	id_res_type	name	description
1	1	wedding	Rezerwacja zlecenia ślubnego, przygotowania, cere...
2	2	photoshoot	Rezerwacja sesji plenerowej, obejmująca różnorodn...
3	3	baptism	Rezerwacja zlecenia z ceremonii chrztu dziecka.
4	4	other	Inny typ rezerwacji, który nie pasuje do wcześniej w...

#### 4.2.5. reservation\_details

Tabela zapisująca wybór typu zlecenia z tabeli reservation\_type do konkretnej rezerwacji. (Np. typ rezerwacji: ślub, id rezerwacji:9)

- id\_res\_det – identyfikator (klucz główny).
- reservation\_id – identyfikator rezerwacji (klucz obcy do reservation).
- type\_id – identyfikator typu zlecenia (klucz obcy do reservation\_type).
- details\_id – identyfikator tabeli rozszerzającej detale dotyczące rezerwacji. Wybierany na podstawie type\_id. Nie jest kluczem obcym, ponieważ rekord odpowiedniej tabeli \*\_details zostanie utworzony dopiero po wyborze typu. ID zostanie wpisane do pola za pomocą SCOPE\_IDENTITY(), a kolizji między użytkownikami możemy uniknąć stosując transakcję.

```
CREATE TABLE reservation_details (  
  id_res_det INT PRIMARY KEY IDENTITY,  
  reservation_id INT NOT NULL,  
  type_id INT NOT NULL,  
  details_id INT,  
  
  FOREIGN KEY (reservation_id) REFERENCES reservation(id_res),  
  FOREIGN KEY (type_id) REFERENCES reservation_type(id_res_type)  
)
```

	id_res_det	reservation_id	type_id	details_id
1	1	1	2	1
2	2	2	1	1

#### 4.2.6. details\_wedding

Tabela rozszerzająca dane dotyczące rezerwacji zlecenia dostępna po wybraniu określonego typu rezerwacji (Dla wedding: id\_res\_type=1).

- id\_det\_wed – identyfikator (klucz główny).
- groom\_address – adres Pana Młodego.
- groom\_prep\_time – czas rozpoczęcia przygotowań u Pana Młodego
- bride\_address – adres Pani Młodej
- bride\_prep\_time – czas rozpoczęcia przygotowań u Pani Młodej
- ceremony\_address – adres miejsca w którym odbędzie się ceremonia.
- ceremony\_time – godzina ceremonii.
- church\_entry\_info – dodatkowe informacje dotyczące wejścia do kościoła/miejsca ceremonii (Np. czy wchodzi razem, czy też Pan Młody czeka przy ołtarzu, a Pani Młoda wchodzi z tatą).
- documents\_signing\_info – informacja czy dokumenty będą podpisywane przed, w trakcie, czy po ceremonii.
- church\_exit\_info – dodatkowe informacje dotyczące wyjścia z kościoła/miejsca ceremonii (Np. czy będzie confetti).
- compliments\_info – informacje dotyczące składania życzeń (Np. po ceremonii / na sali / po obiedzie.
- wedding\_hall\_address – adres sali weselnej.
- musical\_band\_info – nazwa zespołu/ oprawy muzycznej.
- additional\_attractions – informacje dotyczące innych przewidzianych tego dnia atrakcji dodatkowych (Np. fotobudka, zimne ognie).

```
CREATE TABLE details_wedding (  
  id_det_wed INT PRIMARY KEY IDENTITY,  
  groom_address VARCHAR(255) NOT NULL,  
  groom_prep_time TIME,  
  bride_address VARCHAR(255) NOT NULL,  
  bride_prep_time TIME,  
  ceremony_address VARCHAR(255) NOT NULL,  
  ceremony_time TIME NOT NULL,  
  church_entry_info VARCHAR(255),  
  documents_signing_info VARCHAR(255),  
  church_exit_info VARCHAR(255),  
  compliments_info VARCHAR(255),  
  wedding_hall_address VARCHAR(255) NOT NULL,  
  musical_band_info VARCHAR(255),  
  additional_attractions VARCHAR(500)  
);
```

	id_det_wed	groom_address	groom_prep_time	bride_address	bride_prep_time	ceremony_address	ceremony_time	church_entry_info
1	1	ul. Kwiatowa 10, Warszawa	09:00:00.0000000	ul. Wesola 5, Warszawa	10:30:00.0000000	Kościół św. Jana, Warszawa	14:00:00.0000000	Wejście osobno. Pan młody czeka przy ołtarzu, a ...

#### 4.2.7. details\_photoshoot

Tabela rozszerzająca dane dotyczące rezerwacji zlecenia dostępna po wybraniu określonego typu rezerwacji (Dla photoshoot: id\_res\_type=2).

- id\_det\_ps – identyfikator (klucz główny).
- localisation – adres lub nazwa miejsca w którym odbędzie się sesja zdjęciowa/ filmowa.
- transport – informacja dotycząca sposobu transportu (Np. czy klient odbiera ekipę, czy też pracownicy powinni dojechać we własnym zakresie za dodatkową opłatą).
- num\_of\_participants – liczba osób, które będą fotografowane lub nagrywane.

```
CREATE TABLE details_photoshoot (  
    id_det_ps INT PRIMARY KEY IDENTITY,  
    localisation VARCHAR(255) NOT NULL,  
    transport VARCHAR(255),  
    num_of_participants INT NOT NULL  
)
```

	id_det_ps	localisation	transport	num_of_participants
1	1	Góra Zborów	Dojazd osobno, samochodem prywatnym	2

#### 4.2.8. details\_baptism

Tabela rozszerzająca dane dotyczące rezerwacji zlecenia dostępna po wybraniu określonego typu rezerwacji (Dla baptism: id\_res\_type=3).

- id\_det\_bap – identyfikator (klucz główny).
- church\_address – adres kościoła.
- home\_address – adres domu w którym odbędą się przygotowania
- ceremony\_time – godzina rozpoczęcia mszy.

```
CREATE TABLE details_baptism (  
    id_det_bap INT PRIMARY KEY IDENTITY,  
    church_address VARCHAR(255) NOT NULL,  
    home_address VARCHAR(255),  
    ceremony_time TIME NOT NULL  
)
```

	id_det_bap	church_address	home_address	ceremony_time
1	1	Kościół św. Anny, Kraków	ul. Lipowa 15, Kraków	12:00:00.0000000

#### 4.2.9. details\_other

Tabela rozszerzająca dane dotyczące rezerwacji zlecenia dostępna po wybraniu określonego typu rezerwacji (Dla other: id\_res\_type=4). Jeśli żaden z przewidzianych typów rezerwacji nie spełnia oczekiwań użytkownika, to może on wybrać opcję „inne” i opisać specyfikę zlecenia.

- id\_det\_oth – identyfikator (klucz główny).
- localisation – adres lub nazwa miejsca gdzie ma zostać wykonane zlecenia.
- description – opis zlecenia.

```
CREATE TABLE details_other (  
  id_det_oth INT PRIMARY KEY IDENTITY,  
  localisation VARCHAR(255) NOT NULL,  
  description VARCHAR(1000) NOT NULL  
)
```

	id_det_oth	localisation	description
1	1	Restauracja "Złoty Smak", Poznań	Rodzinne przyjęcie urodzinowe, zdjęcia przy sto...

#### 4.2.10.reservation\_cancellation

Tabela zapisująca anulowanie rezerwacji. Zapisany zostaje powód anulowania, data oraz kto je wykonał. Status rezerwacji w tabeli reservation zostaje zmieniony na anulowany (status\_id=3).

- id\_cancell – identyfikator (klucz główny)
- res\_id – identyfikator rezerwacji (klucz obcy do tabeli reservation).
- cancelled\_by – identyfikator użytkownika (klucz obcy do tabeli users)
- cancell\_reason – powód anulowania rezerwacji
- cancell\_date – data anulowania, pobierania automatycznie w momencie anulowania rezerwacji.

```
CREATE TABLE reservation_cancellation (  
  id_cancell INT PRIMARY KEY IDENTITY,  
  res_id INT NOT NULL,  
  cancelled_by INT NOT NULL,  
  cancell_reason VARCHAR(500) NOT NULL,  
  cancell_date DATETIME DEFAULT CURRENT_TIMESTAMP,  
  
  FOREIGN KEY (res_id) REFERENCES reservation(id_res),  
  FOREIGN KEY (cancelled_by) REFERENCES users(id_user)  
)
```

	id_cancell	res_id	cancelled_by	cancell_reason	cancell_date
1	1	5	7	Odwalamy sesję ze względu na zmianę planów.	2024-12-05 15:36:19.573

### 4.3. Tabele wyposażenia

#### 4.3.1. equipment

Tabela z listą wyposażenia firmy.

- id\_eq – identyfikator sprzętu (klucz główny).
- name – nazwa sprzętu.
- eq\_type\_id – typ sprzętu (klucz obcy do tabeli equipment\_type).
- eq\_manufacturer\_id – producent sprzętu (klucz obcy do tabeli equipment\_manufacturer).
- description – opis sprzętu.
- condition – stan techniczny sprzętu.
- working – oznaczenie bitowe wskazujące, czy dane urządzenie jest działające (1 - działa, 0 – nie działa)

```
CREATE TABLE equipment (  
  id_eq INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  eq_type_id INT NOT NULL,  
  eq_manufacturer_id INT NOT NULL,  
  description VARCHAR(255),  
  condition VARCHAR(255),  
  working BIT DEFAULT 1, --1: working, 0: not working  
  
  FOREIGN KEY (eq_type_id) REFERENCES equipment_type(id_eq_type),  
  FOREIGN KEY (eq_manufacturer_id) REFERENCES  
equipment_manufacturer(id_eq_man)  
)
```

	id_eq	name	eq_type_id	eq_manufacturer_id	description	condition	working
1	1	Canon EOS R5	1	1	Bezusterkowy aparat fotograficzny o wysokiej rozdzielczości	Nowy	1
2	2	Sony Alpha 7 III	1	2	Profesjonalny bezusterkowiec z matrycą pełnoklatkową	Używany	1
3	3	Sigma 35mm f/1.4 DG HSM	2	4	Obiektyw stałogniskowy o dużej jasności, idealny do...	Nowy	1
4	4	Tamron 28-75mm f/2.8 Di III RXD	2	5	Zoom o stałej jasności, idealny do fotografii krajobraz...	Używany	1
5	5	YANGNUO YN560 IV Flash	4	3	Lampa błyskowa do aparatów, z możliwością bezprzewodowego...	Nowy	1

#### 4.3.2. equipment\_type

Tabela z listą kategorii sprzętu (Np. obiektyw, aparat, statyw).

- id\_eq\_type – identyfikator kategorii sprzętu (klucz główny).
- name – nazwa kategorii.
- description – opis kategorii.

```
CREATE TABLE equipment_type (  
  id_eq_type INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  description VARCHAR(255)  
)
```

	id_eq_type	name	description
1	1	Body	Podstawowy element aparatu, zawierająca matrycę i...
2	2	Lens	Obiektyw, element aparatu odpowiedzialny za skupie...
3	3	Tripod	Statyw fotograficzny, używany do stabilizacji aparatu...
4	4	Flash	Lampa błyskowa, służąca do doświetlania sceny w ...

#### 4.3.3. *equipment\_manufacturer*

Tabela z listą producentów sprzętu (Np. Canon, Sony)

- id\_eq\_type – identyfikator producenta sprzętu (klucz główny).
- name – nazwa producenta.
- description – opis producenta

```
CREATE TABLE equipment_manufacturer (  
  id_eq_man INT PRIMARY KEY IDENTITY,  
  name VARCHAR(50) NOT NULL,  
  description VARCHAR(255)  
)
```

	id_eq_man	name	description
1	1	Canon	Lustrzanki, bezlusterkowce i obiektywy używane prz...
2	2	Sony	Bezlusterkowce i obiektywy używane przez nas do ...
3	3	YANGNUO	Lampy błyskowe i obiektywy
4	4	Sigma	Obiektywy
5	5	Tamron	Obiektywy

#### 4.3.4. *equipment\_compability*

Tabela przechowująca informację, jakie sprzęty są ze sobą kompatybilne (Np. lampa błyskowa firmy YONGNUO może być kompatybilna z aparatami Canon i Sony)

- id\_compability – identyfikator kompatybilności (klucz główny).
- eq\_id – identyfikator sprzętu (klucz obcy do tabeli equipment).
- compatible\_with\_id – identyfikator sprzętu z którym jest kompatybilny (klucz obcy do tabeli equipment).

```
CREATE TABLE equipment_compability (  
  id_compability INT PRIMARY KEY IDENTITY,  
  eq_id INT NOT NULL,  
  compatible_with_id INT NOT NULL,  
  
  FOREIGN KEY (eq_id) REFERENCES equipment(id_eq),  
  FOREIGN KEY (compatible_with_id) REFERENCES equipment(id_eq)  
)
```

	id_compability	eq_id	compatible_with_id
1	1	3	1
2	2	4	2
3	3	5	1
4	4	5	2

#### 4.3.5. *equipment\_condition\_log*

Tabela przechowująca zmiany stanu sprzętu. W momencie gdy stan sprzętu zostanie zmieniony, to poprzedni stan zostanie zapisany w tabeli, a nowy stan zostanie zaktualizowany w tabeli equipment i zapisany również do equipment\_condition\_log.

- log\_id – identyfikator (klucz główny).
- equipment\_id – identyfikator sprzętu (klucz obcy do tabeli equipment).
- old\_condition – opis poprzedniego stanu sprzętu.
- new\_condition – opis nowego stanu sprzętu.

```
CREATE TABLE equipment_condition_log (  
  log_id INT IDENTITY PRIMARY KEY,  
  equipment_id INT,  
  old_condition VARCHAR(255),  
  new_condition VARCHAR(255),  
  change_date DATETIME  
);
```

#### 4.3.6. *reservation\_equipment*

Tabela obsługująca listę sprzętu przypisanego do danej rezerwacji.

- id\_res\_eq – identyfikator (klucz główny).
- res\_id – identyfikator rezerwacji (klucz obcy do tabeli reservation).
- eq\_id – identyfikator sprzętu (klucz obcy do tabeli equipment).

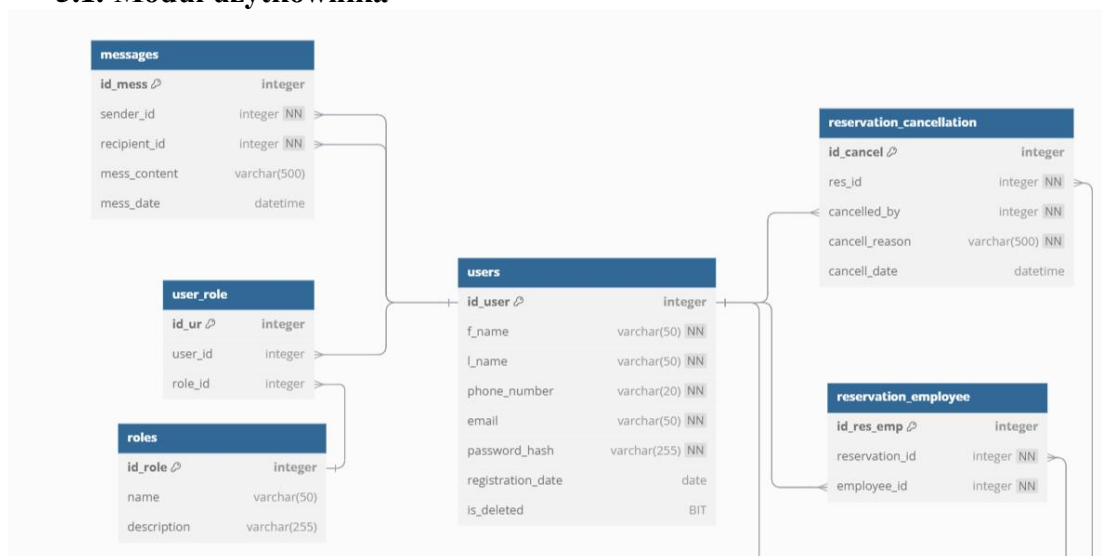
```
CREATE TABLE reservation_equipment (  
  id_res_eq INT PRIMARY KEY IDENTITY,  
  res_id INT NOT NULL,  
  eq_id INT NOT NULL,  
  
  FOREIGN KEY (res_id) REFERENCES reservation(id_res),  
  FOREIGN KEY (eq_id) REFERENCES equipment(id_eq)  
)
```

	id_res_eq	res_id	eq_id
1	1	1	1
2	2	1	3



## 5. Moduły

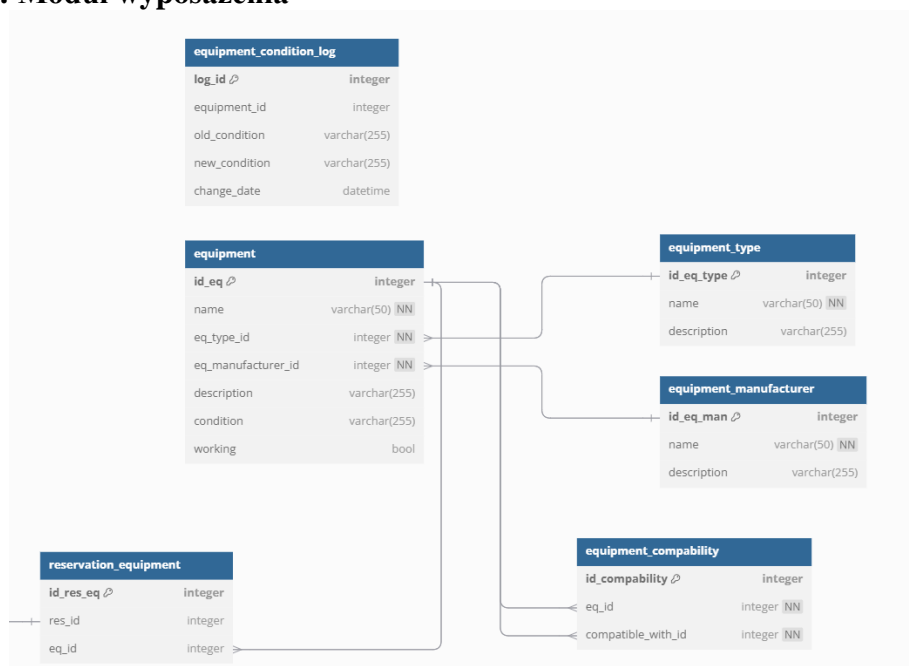
### 5.1. Moduł użytkownika



Moduł użytkownika jest odpowiedzialny za zarządzanie użytkownikami oraz ich interakcjami w systemie. Zawiera funkcjonalności umożliwiające przechowywanie danych użytkowników, przypisywanie ról, komunikację oraz organizację pracy personelu.

- **Zarządzanie użytkownikami**  
Moduł umożliwia rejestrację i przechowywanie podstawowych informacji o użytkownikach, takich jak dane kontaktowe (e-mail, numer telefonu), hasło oraz status konta. Dzięki temu system może obsługiwać zarówno klientów, jak i pracowników.
- **Przypisywanie ról i uprawnień**  
Dzięki mechanizmowi ról, system pozwala na przypisywanie określonych uprawnień użytkownikom. Przykładowo, klient ma dostęp do tworzenia rezerwacji, natomiast pracownik może je obsługiwać. Administratorzy mogą zarządzać całym systemem. Ponieważ jeden użytkownik może mieć kilka ról, moduł pozwala na elastyczne zarządzanie dostępem.
- **Komunikacja między użytkownikami**  
Moduł umożliwia wysyłanie wiadomości między użytkownikami systemu, np. klientami i pracownikami. Klienci mogą kontaktować się z pracownikami w sprawie szczegółów zlecenia, a pracownicy mogą przekazywać ważne informacje dotyczące realizacji usług.
- **Organizacja pracy personelu**  
System pozwala na przypisywanie pracowników do konkretnych rezerwacji. Dzięki temu można śledzić, którzy pracownicy obsługują dane zlecenie. System pozwala na zaangażowania kilku osób, np. fotografów, kamerzystów czy asystentów w obrębie jednego zlecenia.

## 5.2. Moduł wyposażenia



Moduł wyposażenia w systemie jest odpowiedzialny za zarządzanie sprzętem, jego kategoryzację, monitorowanie stanu oraz przypisywanie go do rezerwacji. Moduł ten umożliwia efektywne zarządzanie zasobami technicznymi firmy oraz kontrolę ich użycia w realizacji usług. Ponadto system pozwala na kontrolę kompatybilności pomiędzy poszczególnymi sprzętami.

- **Katalogowanie sprzętu**

Moduł umożliwia tworzenie katalogu wyposażenia, który zawiera informacje o sprzęcie dostępnym w firmie, takim jak aparaty, obiektywy, lampy błyskowe czy statywy, ale może być to również dowolny inny typ wyposażenia, taki jak samochody. Sprzęt jest klasyfikowany według różnych kategorii oraz producentów, co pozwala na przejrzyste i logiczne uporządkowanie zasobów.

- **Monitorowanie kompatybilności sprzętu**

System umożliwia określenie, z jakimi innymi urządzeniami dany sprzęt jest kompatybilny. Przykładowo, statyw konkretnej firmy może być kompatybilny z aparatami różnych producentów. Dzięki temu pracownicy mogą łatwo sprawdzić, który sprzęt można ze sobą łączyć, co jest istotne podczas planowania realizacji zlecenia.

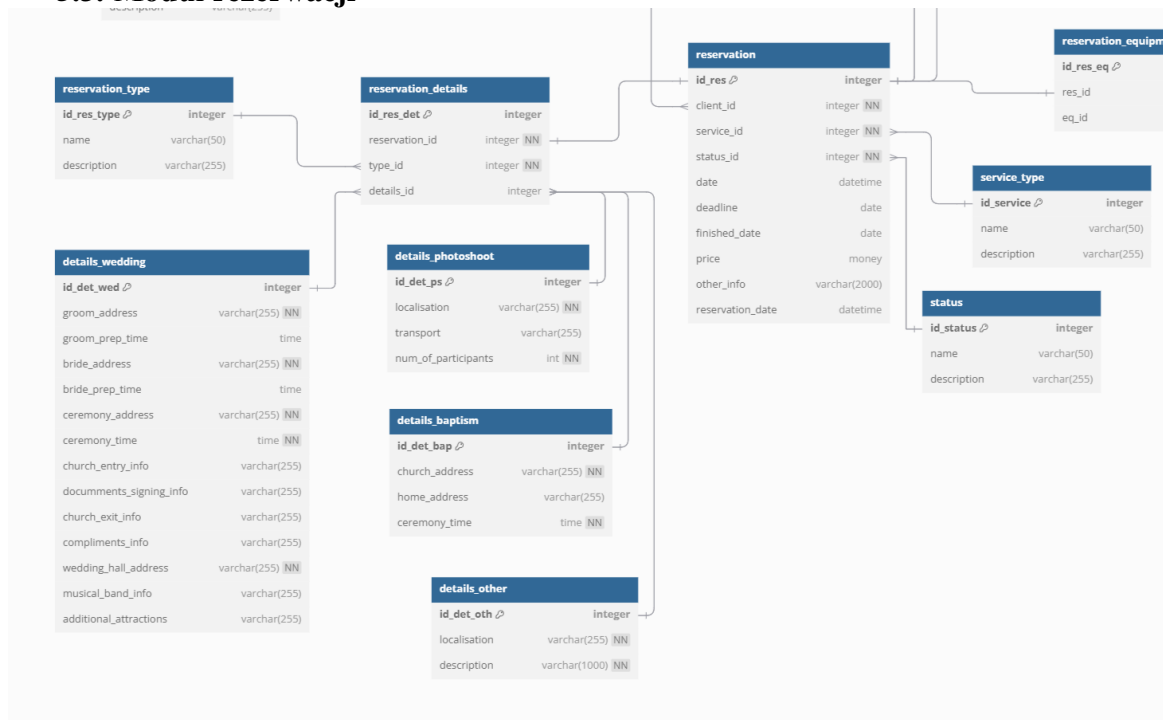
- **Zarządzanie stanem technicznym sprzętu**

Moduł pozwala na monitorowanie stanu technicznego wyposażenia. Przechowuje pole tekstowe zawierające opis stanu urządzenia. System umożliwia logowanie zmian stanu sprzętu, takich jak awarie, naprawy czy konserwacja. Dzięki temu można śledzić historię stanu technicznego każdego urządzenia.

- **Przypisywanie sprzętu do rezerwacji**

System umożliwia przypisanie konkretnego sprzętu do zleceń rezerwacji. Dzięki temu można zapewnić dostępność niezbędnego wyposażenia podczas realizacji zleceń.

### 5.3. Moduł rezerwacji



Moduł rezerwacji umożliwia klientom i pracownikom zarządzanie zleceniami oraz śledzenie ich statusu. Pozwala na efektywne planowanie, organizację i realizację usług oferowanych przez firmę, takich jak sesje zdjęciowe, śluby, czy inne wydarzenia. Po uzupełnieniu podstawowych danych rezerwacji, wspólnych dla każdego typu zlecenia, system prosi użytkownika o sprecyzowanie jakiego typu wydarzenia będzie dotyczyć rezerwacja. Na podstawie tej informacji zostanie wyświetlony odpowiedni formularz z danymi charakterystycznymi dla danego typu zlecenia.

- **Tworzenie i zarządzanie rezerwacjami**

Moduł pozwala na tworzenie nowych rezerwacji, gdzie klient może zdefiniować szczegóły zlecenia, takie jak typ usługi (np. sesja zdjęciowa, reportaż ślubny) oraz preferowaną datę i godzinę. Dodatkowo rezerwacja zawiera informacje o dodatkowych wymaganiach klienta, takich jak lokalizacja, cena, czy szczegółowy opis oczekiwań.

- **Śledzenie statusu rezerwacji**

Każda rezerwacja posiada przypisany status, który pozwala na monitorowanie jej postępu. Statusy mogą obejmować różne etapy, takie jak: oczekujące (domyślny status po utworzeniu rezerwacji, przed przypisaniem pracowników do jej wykonania), w trakcie realizacji, postprodukcja, zakończone i anulowane. Ułatwia to organizację w realizacji zleceń i informuje klienta o postępach w pracy.

- **Przypisanie typu rezerwacji i szczegółowych danych**  
System pozwala określenie typu wydarzenia (np.: ślub, sesja zdjęciowa) oraz przypisanie na tej podstawie tabeli ze szczegółowymi danymi charakterystycznymi dla danego typu.
- **Anulowanie rezerwacji**  
System pozwala na anulowanie rezerwacji przez klienta lub pracownika. Podczas anulowania zapisywane jest kto dokonał anulowania, przyczyna anulowania oraz jej data.

## 6. Procedury

### 6.1. AddReservation

#### Opis:

Dodaje nową rezerwację do systemu wraz z podstawowymi informacjami, przypisując typ zlecenia do tabeli reservation\_details i zwracając ID tego wpisu jako parametr wyjściowy.

#### Parametry wejściowe:

- @client\_id INT – ID klienta, który tworzy rezerwację.
- @service\_id INT – ID usługi powiązanej z rezerwacją.
- @date DATETIME – Data realizacji rezerwacji.
- @other\_info VARCHAR(2000) – Dodatkowe informacje o rezerwacji.
- @type\_id INT – Typ zlecenia, przypisany do tabeli reservation\_details.

#### Parametry wyjściowe:

- @reservation\_details\_id INT OUTPUT – ID wpisu w tabeli reservation\_details.

#### Opis działania:

1. Sprawdza, czy użytkownik ma rolę klienta (ID: 4) lub administratora (ID: 1).
2. Dodaje nową rezerwację do tabeli reservation.
3. Dodaje rekord do tabeli reservation\_details powiązany z utworzoną rezerwacją.
4. Zwraca ID wpisu z tabeli reservation\_details jako parametr wyjściowy.

#### Komunikaty:

- Sukces: "Rezerwacja zlecenia została dodana."
- Błąd: "Brak uprawnień do dodania rezerwacji."

#### Przykład wywołania:

```
DECLARE @reservation_details_id INT;

-- Wywołanie procedury AddReservation z przekazaniem parametrów i odbiorem
@reservation_id
EXEC AddReservation
    @client_id = 6,
    @service_id = 1,
    @date = '2024-12-05',
    @other_info = 'Informacje dodatkowe...',
    @type_id = 4,
    @reservation_details_id = @reservation_details_id OUTPUT; --zapisanie id
do @reservation_id na wyjściu

-- Wywołanie procedury AddDetailsOther z przekazaniem @reservation_id
EXEC AddDetailsOther
    @id_reservation_details = @reservation_details_id,
    @localisation = 'ul. Nowa 5, Wrocław', -- Lokalizacja
    @description = 'Event w plenerze z widokiem na jezioro'; -- Opis
```

## 6.2. AddDetailsWedding

### Opis:

Dodaje szczegóły rezerwacji typu "wedding" do tabeli details\_wedding oraz aktualizuje ID w tabeli reservation\_details.

### Parametry wejściowe:

- @id\_reservation\_details INT – ID z tabeli reservation\_details.
- Szczegółowe dane (adresy, godziny, informacje o ceremonii, atrakcje, itp.).

### Opis działania:

1. Dodaje szczegóły do tabeli details\_wedding.
2. Pobiera ID dodanego rekordu.
3. Aktualizuje kolumnę details\_id w tabeli reservation\_details.

## 6.3. AddDetailsPhotoshoot

### Opis:

Dodaje szczegóły rezerwacji typu "photoshoot" do tabeli details\_photoshoot oraz aktualizuje ID w tabeli reservation\_details.

### Parametry wejściowe:

- @id\_reservation\_details INT – ID z tabeli reservation\_details.
- @localisation VARCHAR(255) – Lokalizacja sesji zdjęciowej.
- @transport VARCHAR(255) – Szczegóły transportu.
- @num\_of\_participants INT – Liczba uczestników.

### Opis działania:

1. Dodaje szczegóły do tabeli details\_photoshoot.
2. Pobiera ID dodanego rekordu.
3. Aktualizuje kolumnę details\_id w tabeli reservation\_details.

## 6.4. AddDetailsBaptism

### Opis:

Dodaje szczegóły rezerwacji typu "baptism" do tabeli details\_baptism oraz aktualizuje ID w tabeli reservation\_details.

### Parametry wejściowe:

- @id\_reservation\_details INT – ID z tabeli reservation\_details.
- @home\_address VARCHAR(255) – Adres domu.
- @church\_address VARCHAR(255) – Adres kościoła.
- @ceremony\_time TIME – Godzina ceremonii.

### Opis działania:

1. Dodaje szczegóły do tabeli details\_baptism.
2. Pobiera ID dodanego rekordu.
3. Aktualizuje kolumnę details\_id w tabeli reservation\_details.

## 6.5. AddDetailsOther

### Opis:

Dodaje szczegóły rezerwacji typu "other" do tabeli details\_other oraz aktualizuje ID w tabeli reservation\_details.

### Parametry wejściowe:

- @id\_reservation\_details INT – ID z tabeli reservation\_details.
- @localisation VARCHAR(255) – Lokalizacja.
- @description VARCHAR(500) – Opis szczegółów.

### Opis działania:

1. Dodaje szczegóły do tabeli details\_other.
2. Pobiera ID dodanego rekordu.
3. Aktualizuje kolumnę details\_id w tabeli reservation\_details.

## 6.6. AssignEmployeeToReservation

### Opis:

Przypisuje pracownika do konkretnej rezerwacji, sprawdzając jego rolę i zgodność z typem usługi.

### Parametry wejściowe:

- @reservation\_id INT – ID rezerwacji.
- @employee\_id INT – ID pracownika.

### Opis działania:

1. Sprawdza, czy pracownik istnieje w systemie.
2. Pobiera typ usługi przypisanej do rezerwacji.
3. Weryfikuje, czy pracownik ma odpowiednią rolę do przypisanego typu usługi.
4. Jeśli weryfikacja jest poprawna, dodaje wpis do tabeli reservation\_employee.

### Komunikaty:

- Sukces: "Pracownik został przypisany do zlecenia."
- Błąd: "Wybrany użytkownik nie ma wymaganych uprawnień."

### Przykład wywołania:

```
EXEC AssignEmployeeToReservation
    @reservation_id = 1,
    @employee_id = 4;
```

## 6.7. CancelReservation

### Opis:

Anuluje rezerwację, sprawdzając uprawnienia osoby anulującej.

### Parametry wejściowe:

- @res\_id INT – ID rezerwacji.
- @cancelled\_by INT – ID użytkownika anulującego.
- @cancell\_reason VARCHAR(500) – Powód anulowania.

### Opis działania:

1. Pobiera ID użytkownika, który stworzył rezerwację.
2. Sprawdza, czy użytkownik anulujący jest:
  - Twórcą rezerwacji,
  - Przypisanym pracownikiem,
  - Administratorem.
3. Jeśli warunki są spełnione, aktualizuje status rezerwacji na anulowany i zapisuje dane w tabeli reservation\_cancellation.

### Komunikaty:

- Sukces: "Rezerwacja anulowana pomyślnie."
- Błąd: "Nie masz uprawnień do anulowania tej rezerwacji."

### Przykład wywołania:

```
EXEC CancelReservation
    @res_id = 1,
    @cancelled_by = 1,
    @cancell_reason = 'Admin action';
```



## 7. Widoki

### 7.1. ActiveUsersWithRoles

#### Opis:

Widok ActiveUsersWithRoles łączy informacje o użytkownikach z przypisanymi im rolami. Wyświetla wyłącznie aktywne konta (czyli takie, które nie zostały oznaczone jako usunięte).

#### Zastosowanie:

- Służy do przeglądania listy aktywnych użytkowników wraz z ich rolami w systemie.
- Może być wykorzystany do raportowania lub zarządzania uprawnieniami użytkowników.

#### Struktura:

- id\_user - ID użytkownika.
- f\_name - Imię użytkownika.
- l\_name - Nazwisko użytkownika.
- Email - Adres e-mail użytkownika.
- Role\_name - Nazwa roli przypisanej użytkownikowi.

#### Przykład wywołania:

	id_user	f_name	l_name	email	role_name
1	1	Jan	Kowalski	jan.kowalski@example.com	administrator
2	2	Anna	Nowak	anna.nowak@example.com	kamerzysta
3	2	Anna	Nowak	anna.nowak@example.com	fotograf
4	3	Piotr	Zielinski	piotr.zielinski@example.com	klient
5	4	Łukasz	Setlak	lukaset@example.com	fotograf
6	5	Marta	Wiśniewska	marta.wisniewska@example.com	klient
7	6	Krzysztof	Nowicki	krzysztof.nowicki@example.com	klient
8	7	Ewa	Kaczmarek	ewa.kaczmarek@example.com	klient

### 7.2. ReservationView

#### Opis:

Widok ReservationView łączy informacje o rezerwacjach z danymi klientów, typami zleceń, statusami oraz dodatkowymi informacjami o rezerwacji. Jest używany do przeglądania szczegółów wszystkich rezerwacji w systemie.

#### Zastosowanie:

- Umożliwia przeglądanie szczegółowych informacji o rezerwacjach.
- Przydatny do raportowania oraz analizy danych dotyczących zleceń.

#### Struktura:

- id\_res – ID rezerwacji.
- client\_name – Imię i nazwisko klienta.
- service\_type – Typ usługi.
- status – Aktualny status rezerwacji.
- price – Cena wykonania zlecenia.
- other\_info – Dodatkowe informacje dotyczące zlecenia.
- reservation\_date – Data utworzenia rezerwacji.

#### Przykład wywołania:

	id_res	client_name	service_type	status	price	other_info	reservation_date
1	1	Piotr Zielinski	FOTO	Oczekujące	500,00	Sesja zdjęciowa narzeczeńska w plenerze, zależy n...	2024-12-05 15:36:19.563
2	2	Piotr Zielinski	FOTO-VIDEO	Oczekujące	8000,00	Reportaż FOTO-VIDEO na weselu	2024-12-05 15:36:19.570
3	3	Marta Wiśniewska	FOTO	W trakcie ustaleń	1000,00	Ceremonia chrztu, fotografia w kościele i w domu.	2024-12-05 15:36:19.570
4	4	Krzysztof Nowicki	FOTO	W trakcie ustaleń	1500,00	Uroczystość rodzinna, wymagane zdjęcia grupowe i...	2024-12-05 15:36:19.570
5	5	Ewa Kaczmarek	VIDEO	Anulowano	1200,00	Sesja fotograficzna w parku miejskim, odwołana prz...	2024-12-05 15:36:19.570

### 7.3. EmployeeReservationsView

#### Opis:

Widok EmployeeReservationsView zawiera informacje o wszystkich rezerwacjach, do których zostali przypisani pracownicy (np. fotografowie lub kamerzyści). Wyświetla dane rezerwacji, klienta, statusu oraz przypisanego pracownika.

#### Zastosowanie:

- Umożliwia przeglądanie rezerwacji w kontekście przypisanych do nich pracowników.
- Przydatny do zarządzania harmonogramami pracy pracowników.
- Pomocny w raportowaniu zleceń realizowanych przez poszczególnych pracowników.

#### Struktura:

- reservation\_id – ID rezerwacji.
- client\_name – Imię i nazwisko klienta.
- service\_type – Typ usługi.
- status – Aktualny status rezerwacji.
- price – Cena wykonania zlecenia.
- other\_info – Dodatkowe informacje dotyczące zlecenia.
- reservation\_date – Data utworzenia rezerwacji.
- Employee\_name – Imię i nazwisko przypisanego pracownika.

	reservation_id	client_name	service_type	status	price	other_info	reservation_date	employee_name
1	1	Piotr Zielinski	FOTO	Oczekujące	500,00	Sesja zdjęciowa narzeczeńska w plenerze, zależy ...	2024-12-05 15:36:19.563	Łukasz Setlak

## 8. Triggery

### 8.1. PreventUserDeletion

#### Opis:

Trigger PreventUserDeletion zapobiega usunięciu konta użytkownika, jeśli posiada on aktywne rezerwacje. Aktywne rezerwacje to takie, których status nie jest ustawiony na "Anulowano" (status 3) ani "Gotowy" (status 6). Zamiast fizycznego usunięcia, użytkownik zostaje oznaczony jako usunięty (is\_deleted = 1).

#### Zastosowanie:

- Chroni przed usunięciem konta klienta z otwartymi lub w trakcie realizacji rezerwacjami.
- Utrzymuje integralność danych rezerwacji.

#### Działanie:

1. Sprawdza, czy użytkownik usuwany (id\_user z tabeli deleted) ma przypisane rezerwacje o statusie innym niż 3 lub 6.
2. Jeśli tak – blokuje operację usunięcia i zwraca komunikat: *"Nie można usunąć konta z aktywną rezerwacją."*
3. Jeśli użytkownik nie ma aktywnych rezerwacji – oznacza go jako usuniętego poprzez ustawienie is\_deleted = 1

#### Przykład wywołania:

```
DELETE FROM users WHERE id_user = 2;
```

### 8.2. LogEquipmentConditionChange

#### Opis:

Trigger LogEquipmentConditionChange zapisuje zmiany stanu sprzętu (pole condition w tabeli equipment) do tabeli logów equipment\_condition\_log.

#### Zastosowanie:

- Służy do monitorowania historii zmian stanu sprzętu.
- Pomaga w śledzeniu awarii, napraw lub innych operacji wpływających na stan sprzętu.

#### Działanie:

1. Wyzwala się po aktualizacji tabeli equipment.
2. Sprawdza, czy pole condition zostało zaktualizowane.
3. Jeśli tak – wstawia do tabeli equipment\_condition\_log:
  - ID sprzętu (equipment\_id),
  - poprzedni stan (old\_condition),
  - nowy stan (new\_condition),
  - datę zmiany (change\_date).

#### Przykład wywołania:

```
UPDATE equipment  
SET condition = 'Damaged'  
WHERE id_eq = 5;
```