

# MATERIAL ORDER

Dokumentacja Techniczna

Microsoft Power Platform

<b>Wersja</b>	1.0.0.9
<b>Platforma</b>	Power Platform
<b>Komponenty</b>	Canvas App, Dataverse, Power Automate
<b>Autor</b>	Patryk Lyszkowski

## Spis Treści

Spis Treści.....	2
1. Przegląd Aplikacji .....	4
1.1. Główna Funkcjonalność .....	4
1.2. Architektura .....	4
1.3. Stack Techniczny .....	4
2.1. Tabela: CompanyUser .....	5
2.2. Tabela: Project .....	5
2.3. Tabela: Order .....	6
2.4. Tabela: OrderItem.....	6
2.5. Relacje Między Tabelami .....	7
2.6. Cykl Życia Statusu Zamówienia .....	7
3. Przepływy Power Automate.....	8
3.1. Flow: NewOrderAdded .....	8
3.1.1. Konfiguracja Triggera .....	8
3.1.2. Logika Workflow .....	8
3.1.3. Struktura Adaptive Card.....	8
3.1.4. Obsługa Akcji.....	9
3.2. Flow: OrderResubmitted.....	10
3.2.1. Konfiguracja Triggera .....	10
3.2.2. Logika Workflow .....	10
4. Aplikacja Material Order (Canvas App).....	11
4.1. Architektura Aplikacji.....	11
4.1.1. Ekrany.....	11
4.1.2. Zmienne Globalne .....	12
4.2. Zaawansowane Wzorce PowerFX .....	13
4.2.1. Statystyki w zależności od roli użytkownika .....	13
4.2.2. Tworzenie zamówienia z listą materiałów .....	15
4.2.3. Filtrowanie zamówień z optymalizacją wydajności .....	17
4.2.4. Dodawanie i usuwanie pozycji zamówienia.....	18
Jak działa lista materiałów: .....	18
5. Wymagania Instalacji .....	19
5.1. Licencje.....	19
5.2. Konfiguracja Środowiska.....	19

5.3. Konfiguracja Bezpieczeństwa.....	19
6. Przykład Użycia.....	20
6.1. Scenariusz: Zamówienie Materiałów Budowlanych .....	20
6.2. Przebieg Krok Po Kroku.....	20
Krok 1: Tworzenie Zamówienia.....	20
Krok 2: Notyfikacja o Zatwierdzeniu .....	20
Krok 3: Przegląd Przez Kierownika .....	20
Krok 4: Realizacja Przez Dostawcę .....	20
6.3. Ścieżka Alternatywna: Zażąданie Zmiany .....	21
7. Wsparcie Techniczne .....	21

## 1. Przegląd Aplikacji

Material Order to system zarządzania zamówieniami materiałów w Microsoft Power Platform. Aplikacja automatyzuje proces od momentu zgłoszenia zapotrzebowania, przez zatwierdzenie przez kierownika projektu, aż po realizację przez dostawcę.

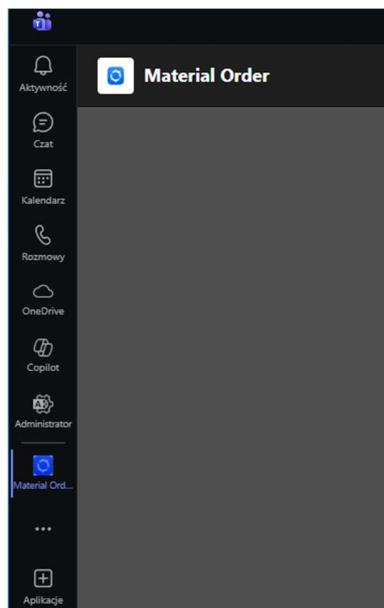
### 1.1. Główna Funkcjonalność

System obsługuje:

- Wielopozytyczne zamówienia materiałów z trackingiem ilości
- Warunkowe procesy zatwierdzeń bazujące na konfiguracji projektu
- Powiadomienia w czasie rzeczywistym w Teams przez Adaptive Cards
- Dostęp do aplikacji w zależności od roli (Supervisor/Manager/Supplier)
- Zarządzanie cyklem życia zamówienia

### 1.2. Architektura

Główny punkt dostępu: aplikacja osadzona w Microsoft Teams. Użytkownicy pracują z Canvas App bezpośrednio w Teams bez konieczności instalacji Power Apps. Aplikacja wykorzystuje Microsoft Teams do uwierzytelniania i dostarczania powiadomień.



Screenshot 1: Material Order w lewym panelu Teams - aplikacja dostępna bezpośrednio w interfejsie

### 1.3. Stack Techniczny

Warstwa	Technologia
Prezentacji	Canvas App w Teams

Logiki Biznesowej	Power Automate cloud flows (automated triggers)
Danych	Dataverse - relacyjna baza danych
Integracji	Teams Connector (Adaptive Cards), Dataverse Connector

## 2. Model Danych

System wykorzystuje cztery niestandardowe tabele w Dataverse. Wszystkie tabele automatycznie śledzą historię zmian: daty utworzenia i modyfikacji (CreatedOn, ModifiedOn), autorów (CreatedBy, ModifiedBy) oraz właściciela (Owner).

### 2.1. Tabela: CompanyUser

Przechowuje użytkowników systemu z kontrolą dostępu opartą na rolach.

Kolumna	Typ	Wymagane	Uwagi
name	Text (100)	Tak	Pole Primary Name
fullname	Text (200)	Tak	Nazwa wyświetlna
role	Choice	Tak	Employee/Manager/Supplier
isactive	Choice	Nie	Active/Inactive

### 2.2. Tabela: Project

Projekty budowlane. Ustawienie w projekcie decyduje, czy zamówienie wymaga akceptacji kierownika.

Kolumna	Typ	Wymagane	Uwagi
name	Text (100)	Tak	Identyfikator projektu
address	Text (500)	Tak	Adres dostawy materiałów
insurancecompany	Text (200)	Nie	Nazwa projektu
requiresapproval	Two Options	Tak	Czy zamówienia wymagają zatwierdzenia

### 2.3. Tabela: Order

Główna tabela z zamówieniami materiałów. Rejestruje wszystkie etapy realizacji zamówienia.

Kolumna	Typ	Wymagane	Uwagi
name	Autonumber	Auto	Numer zamówienia (ORD-{SEQNUM})
project	Lookup	Tak	→ project
status	Choice	Tak	7 stanów (patrz §2.6)
priority	Choice	Tak	Low/Medium/High/Urgent
requestedby	Lookup	Tak	→ companyuser (Employee)
assignedto	Lookup	Tak	→ companyuser (Supplier)
approvedby	Lookup	Nie	→ companyuser (Manager)
deliveryaddress	Text (500)	Nie	Dziedziczone z projektu

### 2.4. Tabela: OrderItem

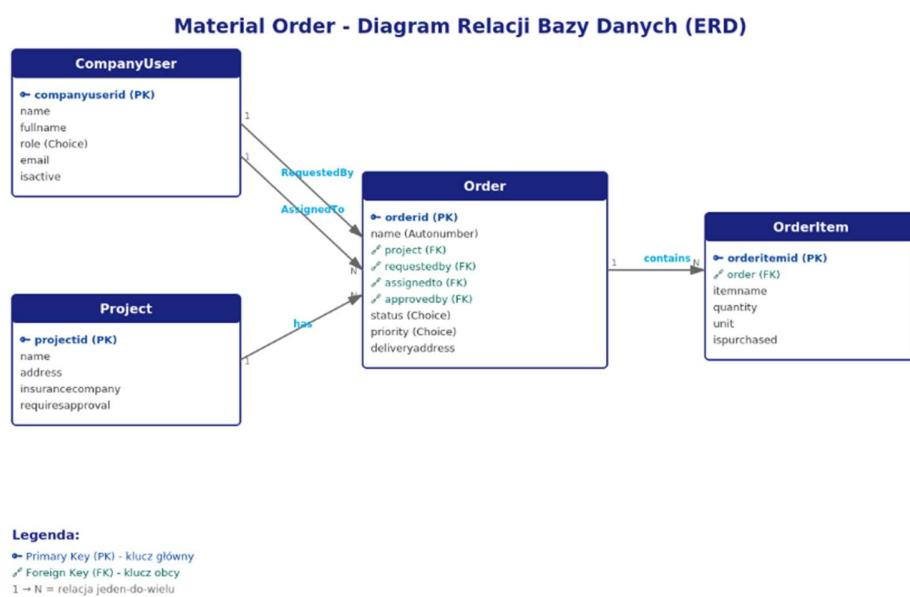
Pozycje zamówienia. Relacja: 1:N z Order.

Kolumna	Typ	Wymagane	Uwagi
order	Lookup	Tak	→ order
itemname	Text (200)	Tak	Opis materiału
quantity	Whole Number	Tak	Ilość pozycji
unit	Text (50)	Tak	kg, m, szt., itd.
ispurchased	Two Options	Nie	Czy dostawca zakupił materiał

## 2.5. Relacje Między Tabelami

Kluczowe relacje między tabelami:

Z	Do	Typ	Przy usuwaniu
Project	Order	1:N	Blokada
Order	OrderItem	1:N	Usuwa wszystko
CompanyUser	Order (RequestedBy)	1:N	Blokada
CompanyUser	Order (AssignedTo)	1:N	Blokada



Screenshot 2: Diagram relacji między tabelami w bazie danych

## 2.6. Cykl Życia Statusu Zamówienia

Zamówienia przechodzą przez poniższe etapy:

Status	Opis
Draft	Stan początkowy. Pracownik może edytować.
Submitted	Złożone do zatwierdzenia. Uruchamia workflow jeśli projekt wymaga akceptacji.
Approved	Kierownik zatwierdził. Gotowe do przypisania dostawcy.
Rejected	Kierownik odrzucił.
Changes Requested	Kierownik zażądał modyfikacji. Wraca do Draft.
In Progress	Dostawca aktywnie realizuje zamówienie.
Completed	Wszystkie pozycje zakupione i dostarczone.

### 3. Przepływy Power Automate

System wykorzystuje dwa cloud flows z triggerami Dataverse i akcjami Teams.

#### 3.1. Flow: NewOrderAdded

Automatyczny flow triggerowany przy tworzeniu Order.

##### 3.1.1. Konfiguracja Triggera

Trigger: Utworzenie nowego rekordu w tabeli Orders Flow odpala się natychmiast, gdy pracownik wyśle zamówienie.

##### 3.1.2. Logika Workflow

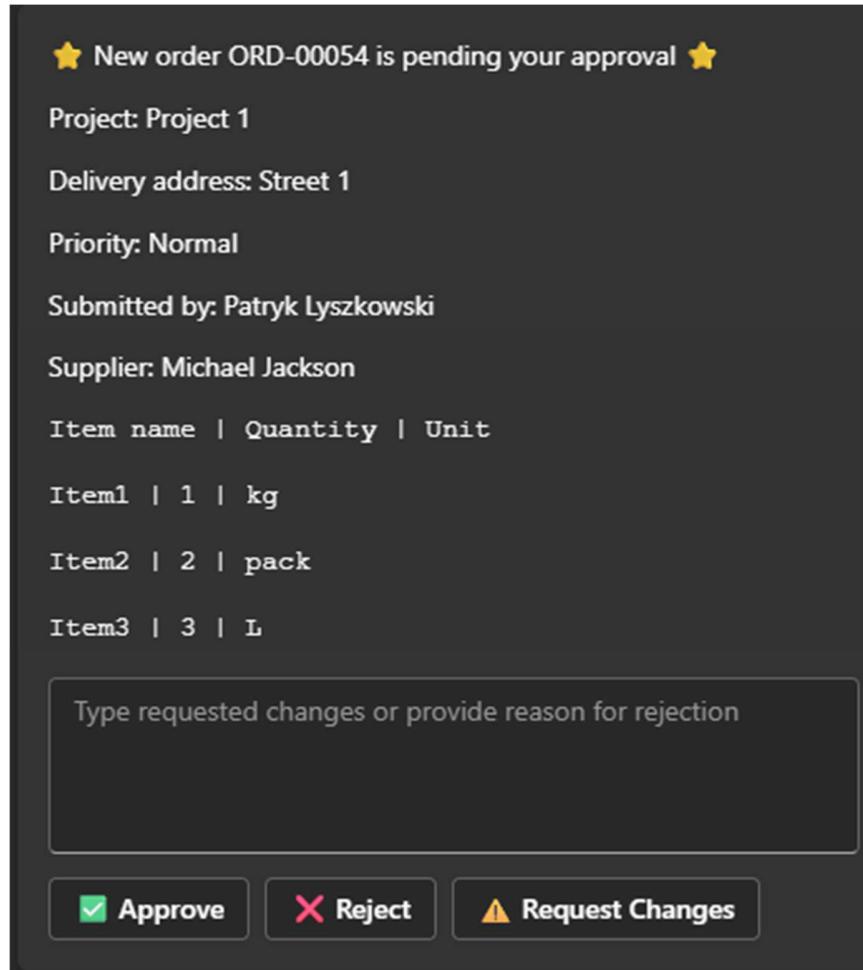
1. Pobiera dane projektu i sprawdza czy wymaga zatwierdzenia (pole requiresapproval)
2. Jeśli tak - zbiera informacje o pracowniku, dostawcy i kierowniku
3. Wczytuje wszystkie pozycje zamówienia (OrderItems)
4. Formatuje listę materiałów w czytelną tabelę
5. Wysyła Adaptive Card do Teams z 3 akcjami: Zatwierdź / Odrzuć / Wymagaj Zmiana

##### 3.1.3. Struktura Adaptive Card

Karta zawiera dynamiczną treść i trzy akcje do wyboru:

```
{
  "type": "AdaptiveCard",
  "body": [
    {
      "type": "TextBlock",
      "text": " OrderNumber ${OrderNumber}"
    },
    {
      "type": "TextBlock",
      "text": "Project: ${ProjectName}"
    },
    {
      "type": "TextBlock",
      "text": " Priority: ${Priority}"
    },
    {
      "type": "TextBlock",
      "text": "${ProductTable}"
    },
    {
      "type": "Input.Text",
      "id": "notes",
      "isMultiline": true
    }
  ],
  "actions": [
    {
      "type": "Action.Submit",
      "title": "Approve",
      "data": {"action": "approve"}
    }
  ]
}
```

```
{
  "type": "Action.Submit",
  "title": "Reject",
  "data": {"action": "reject"}
},
{
  "type": "Action.Submit",
  "title": "Request Changes",
  "data": {"action": "change"}
}
]
```



Screenshot 3: Adaptive Card w Teams - kierownik może zatwierdzić, odrzucić lub zażądać zmian

### 3.1.4. Obsługa Akcji

Co się dzieje po kliknięciu przycisku:

- Zatwierdź → Zamówienie otrzymuje status "Approved", zapisuje się kto i kiedy zatwierdził
- Odrzuć → Status zmienia się na "Rejected", zapisuje się powód odrzucenia
- Wymagaj Zmian → Zamówienie wraca do statusu "Draft" z notatką co poprawić

### **3.2. Flow: OrderResubmitted**

Obsługuje ponowne wysłanie po zażądanych zmianach.

#### **3.2.1. Konfiguracja Triggera**

Trigger: Zmiana statusu zamówienia na "Submitted"

Flow uruchamia się gdy pracownik poprawił zamówienie i ponownie kliknął "Wyślij do zatwierdzenia".

Kierownik otrzymuje nowe powiadomienie w Teams z zaktualizowanymi danymi. Logika identyczna jak NewOrderAdded.

#### **3.2.2. Logika Workflow**

Identyczna jak NewOrderAdded, ale uruchamia się tylko przy zmianie statusu na Submitted. Zapobiega duplikatom powiadomień przy tworzeniu.

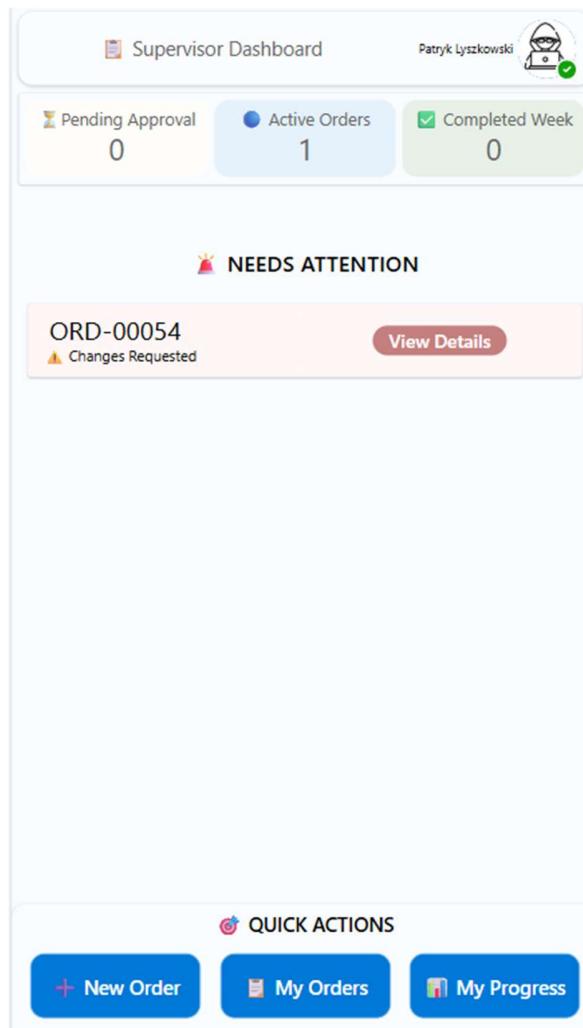
## 4. Aplikacja Material Order (Canvas App)

Canvas App osadzona w Microsoft Teams. Interfejs dostosowuje się do roli użytkownika (Supervisor/Manager/Supplier).

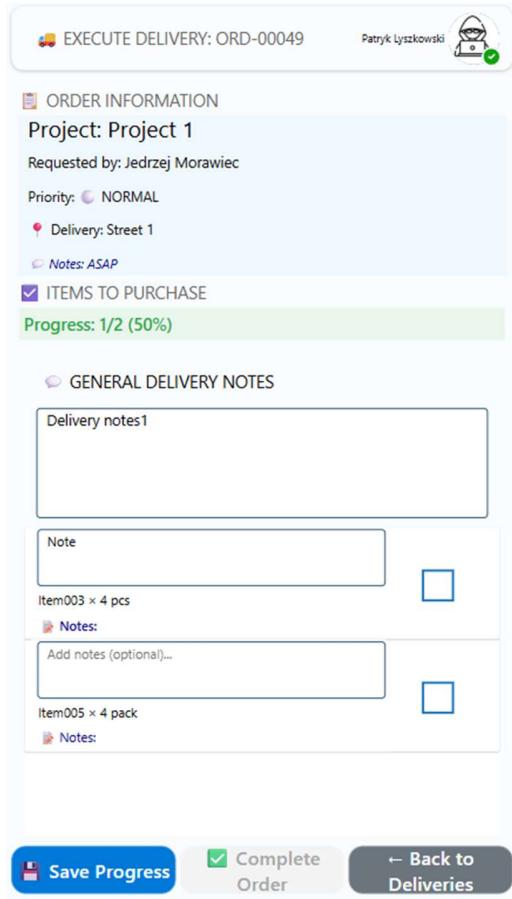
### 4.1. Architektura Aplikacji

#### 4.1.1. Ekrany

- **HomeScreen:** Statystyki zamówień i nawigacja do wszystkich funkcji aplikacji
- **NewOrderScreen:** Formularz z wyborem projektu, dostawcy i pozycjami materiałów
- **OrderDetailScreen:** Podgląd zamówienia z historią zmian statusu
- **SupplierExecuteScreen:** Realizacja zamówienia z oznaczaniem zakupionych materiałów (tylko Supplier)
- **ProjectManagementScreen:** Tworzenie i edycja projektów (tylko Manager)



Screenshot 4: Ekran Główny (Dashboard) - widok dla Supervisor ze statystykami zamówień



Screenshot 7: Ekran Dostawcy - realizacja zamówienia z oznaczaniem zakupionych materiałów

Aplikacja zawiera również ekran pomocnicze np. do zarządzania użytkownikami i ustawieniami.

#### 4.1.2. Zmienne Globalne

Przykładowe zmienne ustawiane przy starcie aplikacji:

```
Set(varCurrentUserEmail, User().Email);
Set(varCurrentUser,
    LookUp(
        CompanyUsers,
        Email = varCurrentUserEmail && IsActive = true
    )
);
// Sprawdzenie uprawnień
If(
    IsBlank(varCurrentUser),
    // Brak dostępu
    Set(varAccessDenied, true);
    Notify(
        "Brak dostępu - Skontaktuj się z Administratorem",
        NotificationType.Error
    ),
)
```

```
// Sprawdzenie roli użytkownika
Switch(
    varCurrentUser.Role,
    'Role (CompanyUsers)'.Supervisor,
        Set(varUserRole, "Supervisor"),
    'Role (CompanyUsers)'.Manager,
        Set(varUserRole, "Manager"),
    'Role (CompanyUsers)'.Supplier,
        Set(varUserRole, "Supplier"),
    Set(varUserRole, "Unknown")
)
);
```

Aplikacja ustawia ponad 30 zmiennych globalnych obejmujących:

- Sprawdzanie czy użytkownik jest aktywny i ma dostęp do systemu
- Blokowanie dostępu z komunikatami błędów dla nieuprawnionych użytkowników
- Zamianę ról systemowych na nazwy tekstowe (Supervisor, Manager, Supplier)
- Pełny system jasnego i ciemnego motywów z paletami kolorów
- Zmienne dla wyglądu interfejsu (kolory ramek, cieni, statusów)

## 4.2. Zaawansowane PowerFX

Dla developerów - przykłady zaawansowanych technik użytych w aplikacji.

### 4.2.1. Statystyki w zależności od roli użytkownika

Home Screen OnVisible:

```
// Pobieranie użytkownika z walidacją
Set(varCurrentUser,
    LookUp(
        CompanyUsers,
        Email = varCurrentUserEmail && IsActive = true
    )
);

// Konwersja enum Role na tekst
Switch(
    varCurrentUser.Role,
    'Role (CompanyUsers)'.Supervisor,
        Set(varUserRole, "Supervisor"),
    'Role (CompanyUsers)'.Manager,
        Set(varUserRole, "Manager"),
    'Role (CompanyUsers)'.Admin,
        Set(varUserRole, "Admin"),
    'Role (CompanyUsers)'.Supplier,
        Set(varUserRole, "Supplier"),
    Set(varUserRole, "Unknown")
);

// Statystyki dla Supervisor
If(
    varUserRole = "Supervisor",
        Set(varMyPendingApproval,
```

```
CountRows(
    Filter(
        Orders,
        RequestedBy.Email = varCurrentUser.Email &&
        'Status (status)' = 'Status (Orders)'.Pending Approval'
    )
);
);

Set(varNeedsAttention,
    Filter(
        Orders,
        RequestedBy.Email = varCurrentUser.Email &&
        (
            'Status (status)' = 'Status (Orders)'.Changes Requested' ||
            'Status (status)' = 'Status (Orders)'.Rejected ||
            (
                'Status (status)' = 'Status (Orders)'.Partially Completed' &&
                StartedOn < varTwoDaysAgo
            )
        )
    )
);
);
```

Użyte rozwiązania:

- Zamiana ról z formatu systemowego na czytelny tekst za pomocą Switch
- Automatyczne liczenie zamówień z określonym statusem (CountRows + Filter)
- Złożone warunki filtrowania - można sprawdzać wiele rzeczy naraz
- Sprawdzanie dat - system wykrywa zamówienia, które się opóźniają

#### 4.2.2. Tworzenie zamówienia z listą materiałów

Screenshot 6: Formularz Nowego Zamówienia - dodawanie materiałów

SupervisorNewOrder - przycisk Submit:

```
// KROK 1: Zapisanie głównego zamówienia
Set(varNewOrder,
Patch(
    Orders,
    Defaults(Orders),
{
    Project: LookUp(Projects, Name = varSelectedProject.Name),
    RequestedBy: LookUp(CompanyUsers, Email = varCurrentUser.Email),
    AssignedTo: dd_NewOrderSupplier.Selected,
    Priority:
        Switch(
            varSelectedPriority,
            "Urgent", 'Priority (Orders)').Urgent,
            "Normal", 'Priority (Orders)').Normal,
            "Low", 'Priority (Orders)').Low
        ),
    'Status (status)':
        If(
            varSelectedProject.RequiresApproval,
            'Status (Orders)'.Pending Approval',
            'Status (Orders)'.Approved
        ),
    
```

```

        SubmittedOn: Now()
    }
);

// KROK 2: Zapisanie listy materiałów
ForAll(
    colOrderItems,
    Patch(
        OrderItems,
        Defaults(OrderItems),
        {
            Order: varNewOrder,
            ItemName: ItemName,
            Quantity: Quantity,
            Unit:
                Switch(
                    Unit,
                    "pcs", 'Unit (OrderItems)'.pcs,
                    "kg", 'Unit (OrderItems)'.kg,
                    "m", 'Unit (OrderItems)'.m,
                    "L", 'Unit (OrderItems)'.L
                ),
            IsPurchased: false
        }
    )
);

// KROK 3: Powiadomienie
Notify(
    "✓ Order " & varNewOrder.OrderNumber & " created! " &
    If(
        varSelectedProject.RequiresApproval,
        "Waiting for manager approval.",
        "Sent directly to supplier."
    ),
    NotificationType.Success,
    4000
);

```

#### Kluczowe elementy:

- Zmienna varNewOrder przechowuje utworzone zamówienie
- Dane pobierane z bazy przez LookUp (bezpieczniejsze)
- Automatyczna konwersja wartości tekstowych na systemowe
- Status zależny od konfiguracji projektu (wymaga zatwierdzenia?)
- ForAll tworzy wszystkie pozycje naraz
- Treść powiadomienia zmienia się dynamicznie

#### 4.2.3. Filtrowanie zamówień z optymalizacją wydajności

The screenshot shows a user interface for managing orders. At the top, there's a header with 'ALL ORDERS' and a user profile for 'Patryk Lyszkowski'. Below the header, a summary box displays the total number of orders (9), pending orders (2), active orders (5), and completed orders (1). A 'FILTERS' section allows filtering by Status (All), Supervisor (empty dropdown), Project (Project 1), and Search (Search by order number, supervisor, project...). The main area lists several orders with their details:

- ORD-00054 • Project 1  
Supervisor: Patryk Lyszkowski  
Status: Unknown • Priority: NORMAL  
Supplier: Michael Jackson • Items: 3  
Created: 04 Feb 2026, 19:53 (14h ago)
- ORD-00053 • Project 1  
Supervisor: Jedrzej  
Status: Approved • Priority: NORMAL  
Supplier: Michael Jackson • Items: 2  
Created: 27 Jan 2026, 20:56 (9 days ago)
- ORD-00052 • Project 1  
Supervisor: Jedrzej  
Status: Approved • Priority: NORMAL  
Supplier: Michael Jackson • Items: 2  
Created: 27 Jan 2026, 20:55 (9 days ago)
- ORD-00051 • Project 1

A 'Home Screen' button is located at the bottom left.

Screenshot 7: Lista zamówień z filtrowaniem

ManagerAllOrdersScreen - Lista zamówień z inteligentnym filtrowaniem:

```
SortByColumns(
    Filter(
        Filter(
            Orders,
            // Pierwszy Filtr
            Switch(
                varFilterStatus,
                "All", true,
                "Pending Approval",
                false
            )
        )
    )
)
```

```

        'Status (status)' = 'Status (Orders)'.Pending Approval',
        "Approved",
        'Status (status)' = 'Status (Orders)'.Approved,
        "In Progress",
        'Status (status)' = 'Status (Orders)'.In Progress',
        "Completed",
        'Status (status)' = 'Status (Orders)'.Completed,
        true
    )
),
// Drugi Filtр
(varFilterSupervisor = "All" || RequestedBy.FullName = varFilterSupervisor) &&
(varFilterProject = "All" || Project.Name = varFilterProject) &&
(IsBlank(varOrderSearch) ||
    OrderNumber in varOrderSearch ||
    RequestedBy.FullName in varOrderSearch)
),
"CreatedOn",
Descending
)

```

Jak to działa:

- Dwa filtry po kolej
- pierwszy w bazie danych, drugi w aplikacji
- Pierwszy zmniejsza ilość danych pobranych z serwera (szybsze działanie)
- Switch dla statusów zamiast If - lepsza wydajność
- Wyszukiwanie i filtrowanie po użytkowniku/projektie działa lokalnie
- Zabezpieczenie przed błędami przy pustym polu wyszukiwania
- Sortowanie po dacie utworzenia (najnowsze na górze)

#### 4.2.4. Dodawanie i usuwanie pozycji zamówienia

SupervisorNewOrder - Zarządzanie pozycjami zamówienia:

```

// Inicjalizacja pustej kolekcji
Clear(colOrderItems);

// Dodawanie pozycji przez użytkownika (przycisk Add Item)
Collect(
    colOrderItems,
    {
        ItemName: txt_SingleItemNameNewOrder.Text,
        Quantity: Value(txtQuantityNewOrder.Text),
        Unit: ddUnitNewOrder.Selected.Value
    }
);

// Usuwanie pozycji (przycisk Delete w gallery)
Remove(colOrderItems, ThisItem);

```

Jak działa lista materiałów:

- Clear() czyści listę materiałów przed rozpoczęciem nowego zamówienia
- Collect dodaje nową pozycję z danymi wpisanymi przez użytkownika
- Value() zamienia tekst z pola ilości na liczbę

- Remove usuwa wybraną pozycję z listy (przycisk w wierszu)

## 5. Wymagania Instalacji

Wymagania wstępne dla wdrożenia:

### 5.1. Licencje

- Microsoft 365 E3/E5 lub Business Premium
- Power Apps licencja per-user lub per-app
- Power Automate premium (dla connectora Dataverse)
- Pojemność bazy Dataverse

### 5.2. Konfiguracja Środowiska

6. Utworzenie środowiska Production z Dataverse
7. Import solution (MaterialOrder\_1\_0\_0\_9.zip)
8. Konfiguracja connection references:
  - Dataverse connector
  - Teams connector
9. Update Teams channel IDs w flows (patrz §3.1.1)
10. Aktywacja flows: NewOrderAdded, OrderResubmitted
11. Dodanie app do Teams: App Studio → Upload custom app
12. Wypełnienie rekordów CompanyUser z emailami i rolami użytkowników

### 5.3. Konfiguracja Bezpieczeństwa

System kontroluje dostęp do danych w zależności od roli użytkownika:

- **Rola Employee:** Widzi tylko swoje zamówienia
- **Rola Supplier:** Widzi i może edytować zamówienia, które są do niego przypisane
- **Rola Manager:** Ma pełny dostęp do wszystkich zamówień i projektów

Bezpieczeństwo zaimplementowane przez:

- Canvas App filtry dane według roli (patrz §4.2.1)
- Dataverse kontroluje dostęp na poziomie bazy danych
- Właściciel rekordu może udostępnić go innym użytkownikom

## 6. Przykład Użycia

Pełny proces zamówienia - od złożenia do realizacji:

### 6.1. Scenariusz: Zamówienie Materiałów Budowlanych

Kontekst:

- **Projekt:** Remont Magazynu (Legnica)
- **Pracownik:** Jan Kowalski (Supervisor)
- **Kierownik:** Anna Nowak (Manager)
- **Dostawca:** BuildMat Sp. z o.o. (Supplier)

### 6.2. Przebieg Krok Po Kroku

#### Krok 1: Tworzenie Zamówienia

Jan otwiera Material Order w Teams → Nowe Zamówienie → Wybiera:

- Projekt: Remont Magazynu
- Supplier: BuildMat Sp. z o.o.
- Priorytet: Wysoki
- Pozycje: 50kg Cement portlandzki, 200szt Cegła czerwona

Kliką 'Wyślij do zatwierdzenia' → Status zmienia się na Submitted.

#### Krok 2: Notyfikacja o Zatwierdzeniu

Flow NewOrderAdded triggeruje się → Sprawdza Project.RequiresApproval = true → Wysyła Adaptive Card do kanału Teams.

Anna otrzymuje notyfikację w Teams: 'Nowe zamówienie ORD-00042 czeka na Twoje zatwierdzenie'

#### Krok 3: Przegląd Przez Kierownika

Anna kliką Adaptive Card → Przegląda szczegóły zamówienia → Kliką 'Zatwierdź' → Flow aktualizuje:

Status: Approved  
ApprovedBy: Anna Nowak  
ApprovedOn: 2025-02-01T14:23:00Z

#### Krok 4: Realizacja Przez Dostawcę

Użytkownik BuildMat otwiera Material Order w Teams → Widzi ORD-00042 w przypisanych zamówieniach → Zmienia status na 'In Progress' → Oznacza pozycje jako zakupione → Zmienia status na 'Completed'.

### 6.3. Ścieżka Alternatywna: Zażądane Zmiany

Jeśli Anna kliknie 'Request Changes' i wprowadzi 'Wymagane potwierdzenie adresu dostawy':

Status: Draft

RequestChanges: Wymagane potwierdzenie adresu dostawy

Jan widzi zamówienie wróciło do Drafts z feedbackiem → Aktualizuje adres dostawy → Wysyła ponownie → Flow OrderResubmitted triggeruje się → Anna dostaje nową kartę zatwierdzenia.

## 7. Wsparcie Techniczne

W sprawach implementacji lub customization:

<b>Developer</b>	Patryk Lyszkowski
<b>GitHub</b>	<a href="https://github.com/patryk-lyszkowski">github.com/patryk-lyszkowski</a>
<b>LinkedIn</b>	<a href="https://linkedin.com/in/patryk-lyszkowski">linkedin.com/in/patryk-lyszkowski</a>

© 2025 Patryk Lyszkowski. Wszelkie prawa zastrzeżone.