

# SD SYSTEM

Autorzy: Szymon Jańczak, Przemysław Jancewicz

## 1. Opis projektu

SD System służy do zarządzania zgłoszeniami pracowników oraz świadczenia usług wsparcia technicznego. Aplikacja umożliwia użytkownikom wewnętrznym danej firmy dostęp do odpowiednich funkcji zgodnie z ich uprawnieniami. Rolą administratora będzie zarządzanie systemem zgłoszeń pod kątem edycji statusu zgłoszeń lub priorytetu, przeglądania zamkniętych zgłoszeń w archiwum, wyświetlenia szczegółów lub usunięcie zgłoszenia w panelu koordynatora. Klienci końcowi mogą utworzyć zlecenie, śledzić ich status. Pracownik może sprawdzić zgłoszenie w panelu użytkownika w zakładce „Moje zgłoszenia”.

## 2. Funkcje

- Tworzenie zgłoszeń,
- Śledzenie zgłoszeń, sprawdzanie szczegółów,
- Zarządzanie zgłoszeniami (odczyt, edycja, usuwanie, filtrowanie, sortowanie), □  
Logowanie do panelu administratora, bądź użytkownika.
- Przeglądanie zamkniętych zgłoszeń w archiwum

## 3. Użyte technologie

- .NET Core 8
- Entity Framework Core
- MS SQL Server
- Microsoft Identity

## 4. Instalacja

- a) Sklonuj/pobierz repozytorium <https://github.com/SzymonJ1/Sd-System>
- b) Po uruchomieniu projektu w Visual Studio 2022 wejdź w zakładkę **Tools -> NuGet Package Manager -> Package Manager Console**
- c) W konsoli wpisz **Update-Database**
- d) Uruchom aplikację
- e) Dane logowania do testowego konta administratora: email – admin@example.com, hasło Admin123!

Administrator tylko zarządza zgłoszeniami. Dla testu, można utworzyć konto użytkownika w oknie rejestracji, zalogować się i utworzyć zgłoszenia.

## 5. Opis struktury projektu

- **Controllers** – Zawiera kontrolery odpowiedzialne za obsługę żądań HTTP.
- **Data** – Zawiera pliki pomocnicze oraz kontekst bazy danych.
- **Migrations** – Zawiera migracje Entity Framework.
- **Models** – Zawiera modele danych używane przez widoki.
- **Properties** – Zawiera konfiguracje uruchomieniowe.
- **Views** – Zawiera widoki Razor do renderowania stron HTML.
- **Wwwroot** – Zawiera pliki statyczne (CSS, JavaScript).

## 6. Wylistowane wszystkie modele

- **ApplicationDbContext.cs** – Kontekst bazy danych.
- **Ticket.cs** – Model reprezentujący parametry zgłoszeń.
- **ApplicationUser.cs** – Model reprezentujący użytkowników.

## 7. Wylistowane kontrolery wraz z metodami

### Controllers/HomeController.cs

**Index()** - Wyświetla stronę główną (dla adminów pokazuje statystyki zgłoszeń i użytkowników)

**Error()** - Wyświetla stronę błędu

### Controllers/TicketsController.cs

**Index(string sortOrder, string priorityFilter)** - Wyświetla listę aktywnych zgłoszeń z sortowaniem i filtrowaniem

- Parametry:
  - **sortOrder**: Kierunek sortowania (date\_asc/date\_desc)
  - **priorityFilter**: Filtrowanie po priorytecie (P1-P5)

**Edit(int? id) (GET)** - Wyświetla formularz edycji zgłoszenia

- Parametr:
  - ID zgłoszenia

**Edit(int id, [Bind(...)] Ticket ticket, string action) (POST)** - Aktualizuje zgłoszenie w bazie danych

- Obsługuje dwie akcje:
  - Zmiana statusu/priorytetu
  - Archiwizacja (action=archive)

**Create() (GET)** - Wyświetla formularz tworzenia nowego zgłoszenia

**Archive()** - Wyświetla listę archiwalnych zgłoszeń (status Closed)

**Delete(int? id) (GET)** - Wyświetla potwierdzenie usunięcia zgłoszenia

**DeleteConfirmed(int id) (POST)** - Usuwa zgłoszenie z bazy danych

**Details(int? id)** - Wyświetla szczegóły zgłoszenia z czasem pozostałym do terminu

Controllers/UserTicketsController.cs

**Index(string sortOrder, string priorityFilter)** - Wyświetla listę zgłoszeń zalogowanego użytkownika

- Parametry:
  - tak jak w TicketsController

**Create() (GET)** - Wyświetla formularz zgłoszeniowy dla użytkownika

**Create([Bind(...)] Ticket ticket) (POST)** - Zapisuje nowe zgłoszenie użytkownika z automatycznymi danymi:

- **Autor** = bieżący użytkownik
- **Status** = New
- **Priorytet** = P5

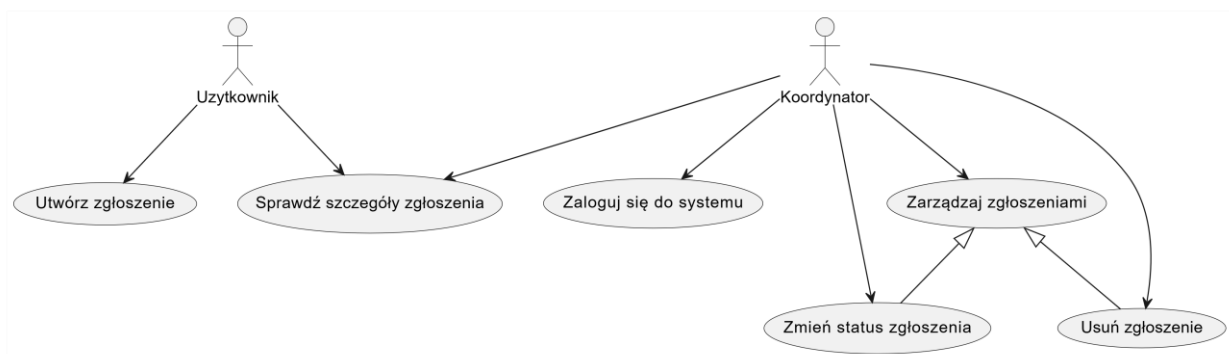
## 8. Opis systemu użytkowników

- **Logowanie do panelu administratora/użytkownika** – W bazie danych mamy zadeklarowane dane logowania dla koordynatorów, za pomocą których koordynator może zalogować się do panelu stworzonego do edycji statusu zgłoszeń.
- **Wylogowanie z panelu administratora/użytkownika** – Po zakończonej sesji użytkownik może wylogować się z panelu. Wylogowanie użytkownika z bieżącej sesji również działa automatycznie po 20 minutach bezczynności.
- **Panel administratora** - Zarządzanie zgłoszeniami (odczyt, edycja, usuwanie, filtrowanie, sortowanie). Przeglądanie zamkniętych zgłoszeń w archiwum.
- **Panel użytkownika** - Tworzenie zgłoszeń, przeglądanie swoich zgłoszeń.

## 9. Krótka charakterystyka najciekawszych funkcjonalności

- Tworzenie zgłoszeń – Pracownicy mają możliwość utworzenia zgłoszenia, w którym uzupełniają Tytuł, Opis zgłoszenia. Po utworzeniu zgłoszenia pojawia się w zakładce Moje zgłoszenia oraz w panelu administratora.
- Dostęp do panelu administratora ma wyłącznie administrator.
- W panelu administratora osoba uprawniona może usunąć zgłoszenie, wyświetlić jego szczegóły oraz zmienić status zgłoszenia. Zgłoszenia można filtrować, wybierając priorytet oraz sortować wg daty utworzenia.
  - Administrator widzi w tabeli pozostały czas na realizację zgłoszenia.
  - Gdy administrator zmieni status zgłoszenia na Zamknięte, zgłoszenie pojawia się w archiwum.

## 10. Diagram przypadków użycia.



## 11. Diagram klas

