

Dokumentacja projektu z BDBT
Etap 2: Stworzenie aplikacji webowej
Dom Kultury

Sofia Levchenko 308996, Mateusz Izbicki 303957

Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych

12 lipca 2022

Spis treści

1 Wstęp	2
2 Treść	2
2.1 Zakres i cel projektu	2
2.2 Specyfikacja bazy danych	2
2.3 Zrealizowane perspektywy użytkowników	3
2.4 Zrealizowane transakcje	3
3 Model relacyjny wykorzystywanej bazy danych	3
4 Użyta technologia	4
4.1 Wybór narzędzi	4
4.2 Backend - Node.js	4
4.3 Komunikacja Bakendu z Bazą Danych Oracle	5
4.4 Frontend - React.js	5
5 Zrzuty funkcjonalne programu	6
5.1 Podstawowe widoki	6
5.2 Widoki developera	8
5.3 Widoki organizatora	10
5.4 Widoki administratora	11
5.5 Widoki uczestnika	13

1 Wstęp

2 Treść

W podanej dokumentacji zostaną krótko przedstawione zagadnienia dotyczące tematu naszego projektu, model wykorzystywanej bazy danych, opis słowny zrealizowanych transakcji oraz perspektyw użytkowników. Również dokumentacja zawiera opis wykorzystanych technologii (miedzy innymi wykorzystany język programowania, biblioteki, menadżer zarządzania pakietami itp) oraz rzuty funkcjonalne zrealizowanej aplikacji.

2.1 Zakres i cel projektu

Celem tego etapu naszego projektu jest zbudowanie aplikacji webowej do zarządzania systemem bazy danych dla Domu Kultury, stworzonym na poprzednim etapie.

Podstawowymi funkcjonalnościami naszego systemu są:

- Zarządzanie wydarzeniami organizowanych w ośrodku
- Zapisywanie się uczestników na organizowane wydarzenia

2.2 Specyfikacja bazy danych

Funkcjonalność którą posługuje się zaimplementowana aplikacja (oraz, oczywiście, obsługiwana przez nasz system bazodanowy).

- Przyjęcie przez system danych osobowych użytkowników chcących zarejestrować się na zajęcia

Funkcja ta umożliwia sprawdzenie poprawności wpisania wszystkich podanych informacji dotyczących danych osobowych uczestnika chcącego zarejestrować się w systemie i jego bezpośrednia rejestracja - dodanie podanych informacji do odpowiedniej tablicy naszej bazy danych (żeby móc zarejestrować się na wydarzenia organizowane przez ośrodek).

- Lista osób zapisanych na zajęcia

Pracownicy o określonych typach stanowisk mają dostęp do danych wszystkich pracowników ośrodków i mogą ich modyfikować.

- Lista pracowników ośrodka

Pracownicy ośrodku będą mieli dostęp do informacji osobowej uczestników konkretnego wydarzenia.

- Zapisanie się na zajęcia i wypisywanie się z nich

Zarejestrowane uczestnicy mogą zapisywać się (oraz wypisywać się) na wszelkie organizowane przez ośrodek wydarzenia.

- Zarządzanie wydarzeniami

Pracownicy o określonych typach stanowisk mogą zarządzać wydarzeniami, czyli usuwać uczestników, modyfikować dane dotyczące pewnego wydarzenia, dodawać nowe wydarzenia.

- Zarządzanie personalną informacją

Każdy zalogowany (i wcześniej, oczywiście, zarejestrowany) użytkownik może modyfikować swoje dane.

- Dodatkowa informacja dotycząca organizowanych wydarzeń

Każda chętna osoba (pracownicy ośrodka, prowadzący zajęć, osoba wystawiająca, wszyscy chętni do zapisania się na zajęcia, uczestnicy wydarzenia) ma dostęp do wszelkich informacji dotyczących bezpośrednio wydarzenia, czyli godziny w których odbywają się, numer sali itp.

2.3 Zrealizowane perspektywy użytkowników

- **Developer**

Ma dostęp do wszystkich zaimplementowanych funkcjonalności - tworzenie i modyfikowanie wydarzeń, edytowanie danych osobowych pracowników, przegląd list obecności wszystkich wydarzeń, wyrzucenie uczestników z listy obecności oraz zarządzanie własną informacją osobową.

- **Administrator**

Ma dostęp do wszystkich wydarzeń i ich uczestników, znaczy również może modyfikować wydarzenia i ich listy uczestników), może dodawać nowe wydarzenia, posiada możliwości modyfikowania swoich danych osobowych.

- **Organizator**

Ma dostęp do listy uczestników wydarzeń które organizuje oraz posiada możliwości edytowania tych wydarzeń (włącznie z usuwaniem uczestników), posiada możliwości modyfikowania swoich danych osobowych.

- **Uczestnik wydarzenia**

Posiada dostęp do informacji dotyczącej wydarzeń organizowanych w ośrodku, ma możliwość zapisywania się na zajęcia oraz również modyfikowania swoich danych osobowych.

- **Gość**

Posiada dostęp do informacji dotyczącej wydarzeń organizowanych w ośrodku (jako i wszystkie inne rodzaje użytkowników, o różnych perspektywach).

2.4 Zrealizowane transakcje

Według określonych wcześniej funkcjonalności zatem wydzieliliśmy realizowane przez naszą aplikację transakcje:

- w obszarze obsługa pracowników

– modyfikacja danych osobowych i stanowisk

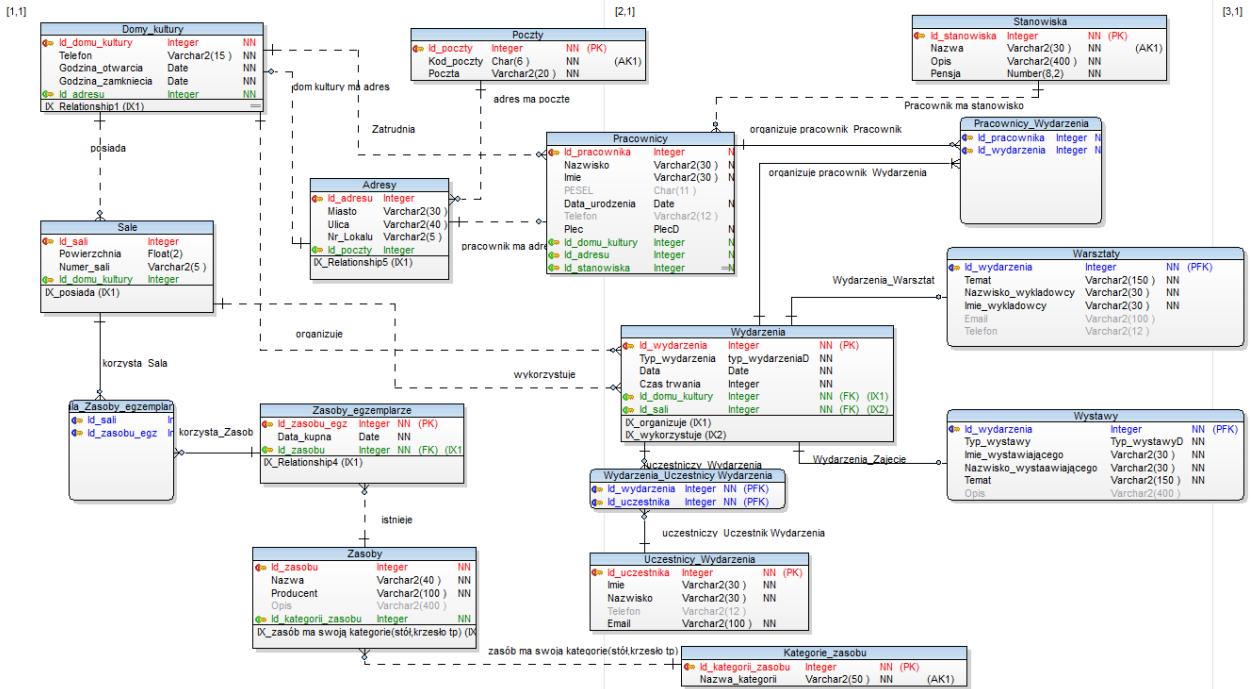
- w obszarze obsługa wydarzeń

– tworzenie wydarzeń
– zarządzanie uczestnikami który zapisali się na dane wydarzenie
– modyfikacja informacji o wydarzeniu

- w obszarze uczestnik wydarzenia

– dodanie (przez samego użytkownika) i usuwanie uczestników wydarzeń
– modyfikacja danych osobowych (przez samego uczestnika)

3 Model relacyjny wykorzystywanej bazy danych



Rys. 1: Schemat modelu relacyjnego

4 Użyta technologia

4.1 Wybór narzędzi

Aby nasza aplikacja posiadała podstawowe założone funkcje, potrzebowaliśmy narzędzi programistycznych, które pomagają tworzenie aplikacji webowych.

Językiem, w którym napisaliśmy aplikacje jest JavaScript. Jest to w tym momencie najpopularniejszy język programowania, który umożliwia budowanie stron internetowych, ale także umożliwia zbudowanie obsługi bazy danych albo API.

Wyбралиśmy go dlatego, że jest w tym momencie najpopularniejszy, ale także dlatego że każdy z nas ma już większe doświadczenie w pisaniu aplikacji webowych w tym języku.

4.2 Backend - Node.js

Aby porozumiewać się z bazą danych potrzebne było stworzenie API. Do tego celu wykorzystaliśmy środowisko uruchomieniowe Node.js. Jest to program do tworzenia aplikacji serwerowych, które wykorzystuje JavaScript. Jest to bardzo przydatne narzędzie ponieważ kilka lat temu aby właśnie napisać część łączącą się z bazą danych w aplikacji było trzeba wykorzystywać inne języki typu PHP, Ruby. Było spowodowane to tym że wtedy JavaScript nie był w stanie wykonywać operacji.

Wraz z Node.js instalowany jest npm. Jest to manager pakietów domyślnie instalowany wraz z Node.js (do wyboru również jest yarn). Npm pozwala nam dodawać biblioteki które rozszerzają nasze możliwości. Dodatkowo konfiguracja instalowanych pakietów powstaje automatycznie dzięki npm. Działa to tak, że npm sam tworzy plik package.json, który zawiera w sobie informacje na temat naszego projektu ale także paczek które są wymagane w naszym projekcie. Dzięki temu sama konfiguracja projektu na innym komputerze jest bardzo prosta i sprowadza się do napisania jednej linijki w terminalu. W pliku package.json możemy znaleźć informacje tj.: nazwy projektu, wersji projektu, autora, licencji, wcześniej wspomnianej listy potrzebnych paczek, ale także skrypty które pomagają np. uruchamiać cały program albo wykonać test

Oczywiście nasz projekt posiada kilka bibliotek które pozwoliły nam stworzyć własne API. Biblioteki których użyliśmy:

- Express.js - biblioteka do tworzenia szkieletu aplikacji serwerowej, API
- oracledb - paczka zawierająca potrzebne biblioteki pozwalające połączyć się z bazą danych Oracle
- node-oracledb - umożliwia połączenie się z bazą danych za pomocą node.js (współpracuje z paczką oracledb)
- dotenv - paczka umożliwiająca zapisanie zmiennych środowiskowych projektu
- morgan - jest to middleware do Node.js która loguje w terminalu zapytania http w przyjazny sposób (głównie do celów poprawy jakości pracy)

4.3 Komunikacja Bakendu z Bazą Danych Oracle

Aby porozumieć się z bazą danych Oracle potrzebowaliśmy znaleźć paczkę która umożliwiała nam w zrozumiałym sposób komunikować z bazą danych. Paczką którą wybraliśmy są oracledb oraz node-oracledb.

Również, aby umożliwić bezpieczeństwo nawiązywanego połączenia (i właśnie zestawiać to połączenia), poprawną komunikację i wykonanie komend z bazą danych, musieliśmy dodatkowo zainstalować pewnie pakiety, dystrybuowane przez Oracle - instantclient.

Paczka oracledb obsługuje całe połączenie z bazą danych Oracle (również korzystając w odpowiedni sposób z instantclient) i wykorzystuje biblioteki do tego potrzebne, zaś paczka node-oracledb pozwala nam na użycie tego połączenia oraz porozumienia się z bazą poprzez właśnie Node.js.

Nasza baza danych została stworzona na serwerze wydziałowym ora3.elka.edu.pl i aby się do niej połączyć należało podać login i hasło. Te informacje zostały ukryte dzięki paczce dotenv która pozwala na stworzenie pliku .env który jest zbiorem zmiennych środowiskowych które są przypisane tylko temu projektowi.

```
1 < module.exports = {
2 <   hrPool: {
3 <     user: process.env.HR_USER,
4 <     password: process.env.HR_PASSWORD,
5 <     connectString: '(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = ora3.elka.pw.edu.pl)(PORT = 1521))(CONNECT_DATA =(SID= ora3inf)))',
6 <     poolMax: 10,
7 <     poolMin: 10,
8 <     poolIncrement: 0
9 <   }
0 > }
```

Rys. 2: Moduł konfiguracyjny bazy danych

4.4 Frontend - React.js

Posiadając już połączenie z bazą danych oraz działające API, to należało wszystkie te dane i całe funkcjonalności wyświetlić użytkownikowi. Do tego użyliśmy biblioteki React.js. Służy ona do tworzenia interfejsów użytkownika. Dany framework jest bardzo wygodny w kontekście zarządzania stanami, czego właśnie najbardziej potrzebowaliśmy w naszym projekcie. Dodatkowo użyliśmy kilku paczek usprawniających nam pracę:

- react.js - dodaje kilka swoich paczek służących m.in. do tworzenie komponentów, obsługi routingu,
- axios - biblioteka pomagająca wykonywanie zapytań http
- http-proxy-middleware - wspieranie obsługi proxy
- moment - paczka pomagająca w przetwarzaniu dat
- @react-google-maps/api - paczka umożliwiająca korzystanie z map Google'a

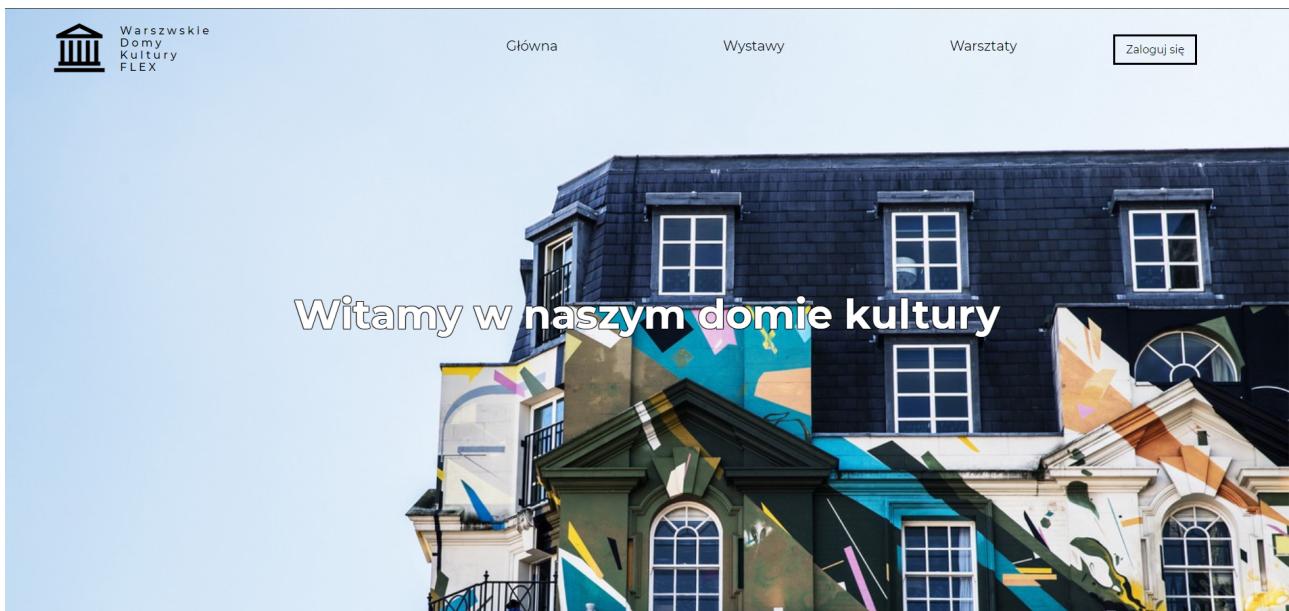
- react-loader-spinner - prosta paczka tworząca loader

React.js pozwala na renderowanie stworzonych wcześniej przez nas komponentów, które są przetwarzane wcześniej przez przeglądarkę. Komponent jest funkcją która zwraca kod jsx który po przetworzeniu przez przeglądarkę wyświetla kod html. Specjalny rodzaj kodu pozwala nam na pisanie zwykłych tagów html oraz wypełnianie ich dynamicznie poprzez użycie stanów. W stanie może zostać zapisana jakakolwiek informacja np. dane o domu kultury, albo to co użytkownik wpisał w dane pole danych.

Dodatkowo, żeby móc rozdzielić posiadaną funkcjonalność przez użytkowników różnego typu, po zalogowaniu się dostajemy od bazy danych obiekt reprezentujący danego użytkownika, który jest przechowywany w globalnym stanie. Na podstawie pewnych pól tego obiektu są ograniczane pewne możliwości funkcjonalne.

5 Zrzuty funkcjonalne programu

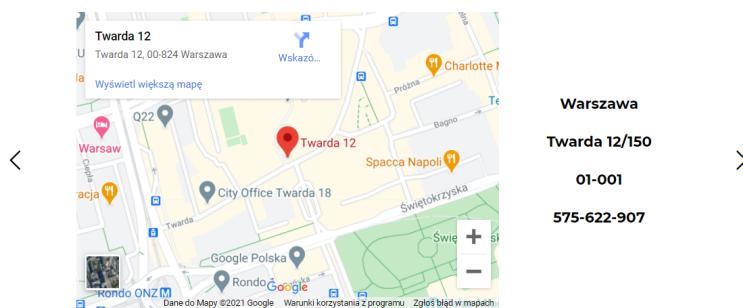
5.1 Podstawowe widoki



Rys. 3: Homescreen



NASZE LOKALIZACJE



Rys. 4: Część dalsza strony home

Rys. 3 oraz Rys. 4 przedstawia stronę główną naszej aplikacji. Możemy tak znaleźć informacje o lokalizacji wszystkich domów kultury. Wykorzystaliśmy tutaj GET request w celu pobrania danych głównych o domach kultury, które następnie są używane w pokazaniu ma mapach Google

 Warszawskie
Domy
Kultury
FLEX
[Główna](#)
[Wystawy](#)
[Warsztaty](#)
[Zaloguj się](#)

Logowanie

Zaloguj się jako:

Pracownik Uczestnik

Email:

Hasło:

Nie masz konta uczestnika [Zarejestruj się](#)

[Zaloguj się](#)

Rys. 5: Widok do zalogowania się jako uczestnik

Na Rys. 5 został pokazany widok do zalogowania się jako uczestnik pewnych wydarzeń. Dany ekran umożliwia wpisanie danych uczestnika, które potem odsypane są do bazy danych za pomocą metody POST, uczestnik jest wyszukiwany w bazie danych na podstawie wpisanych danych i, jeżeli taki uczestnik istnieje na liście, to odpowiednio, jest zwracany obiekt reprezentujący tego uczestnika (który jest zapisywany do stanu).



Rejestracja uczestnika:

Imię: _____

Nazwisko: _____

Telefon: _____

Email: _____

Hasło: _____

Powtórz hasło: _____

Zarejestruj się

Rys. 6: Widok do zarejestrowania się jako uczestnik

Na Rys. 6 jest pokazany formularz zarejestrowania się do strony jako uczestnik. Wpisane dane w pola są przekazywana metodą POST oraz dane zostają dodane do bazy danych zarejestrowanych uczestników. Pozwala to na późniejsze zalogowanie uczestnika



Logowanie

Zaloguj się jako:

Pracownik Uczestnik

Imię: _____

Nazwisko: _____

Hasło: _____

Zaloguj się

Rys. 7: Widok do zalogowania się jako pracownik

Na Rys. 7 został pokazany widok do zalogowania się jako pracownik (dowolnego stanowiska). Dany ekran umożliwia wpisanie danych pracownika, które potem odsyłane są do bazy danych za pomocą metody POST, pracownik jest wyszukiwany w bazie danych na podstawie wpisanych danych i, jeżeli taki pracownik istnieje na liście, to odpowiednio, jest zwracany obiekt reprezentujący tego pracownika (który jest zapisywany do stanu i, na podstawie jego stanowiska, są oddzielane między sobą poszczególne perspektywy użytkowników).

5.2 Widoki developera

Zalogowany jako: Developer

Twoje dane:

Sofia Levchenko Płeć: Kobieta Data ur.: 31.05.2002 Pesel: null
 Telefon: 347-583-481 Pensja: 4000

Panel sterowania:

Zmiana danych pracownika
 Zmiana danych / utworzenie wydarzenia

[Zmień dane](#)

Rys. 8: Mój profil jako developera

Na Rys. 8 został pokazany widok My Profile developera, tutaj developer może zobaczyć (metoda GET, czyli wyciągnięcia danych znajdujących się w tabeli) i zmienić swoje dane osobowe (metoda PUT, czyli modyfikacja danych znajdujących się w tabeli) oraz przejść do dostępnych mu stron zarządzania danym Domem Kultury.

Uczestnicy wydarzeń:

[Dodaj wydarzenie](#)

Rodzaj: wystawa

Violin

Ola Babicz

Numer sali: 102

nie jeszcze ma uczestników

18 stycznia 2021

02:45 - 6:45

[Zmień dane](#)

Contemporary art in us

James Bond

Numer sali: 102

Katarzyna Zyn

Tel: 247-341-249

Email: kzyn@email.com

17 stycznia 2021

10:00 - 13:00

[Zmień dane](#)

Rys. 9: Zarządzanie jako developer

Na Rys. 9 został pokazany widok Zarządzanie developera, tutaj developer może przejrzeć wszystkie istniejące wydarzenia i ich uczestników (metoda GET, czyli oglądania danych znajdujących się w tabeli), edytować wydarzenia (metoda PUT, czyli modyfikacja danych znajdujących się w tabeli), dodawać nowe wydarzenia do pewnego Domu Kultury (metoda POST, czyli dodanie danych do tabeli) oraz wyrzucać uczestników z listy obecności pewnego wydarzenia (metoda DELETE, czyli wyrzucenia danych znajdujących się w tabeli).

Wybierz Dom Kultury:

Dom Kultury numer 1

City: Warszawa
Street: Twarda 12/150
Postal code: 01-001
Phone: 575-622-907
Opens: 07:00
Closes: 19:00

Pracownicy w domie Kultury numer 1:

Developerzy:

Mateusz Izbicki	Plec: Mężczyzna	Data ur.: 23.03.2000	Pesel: 3940284932
Telefon: 495-395-325	Pensja: 4000	Zmień dane	

Sofia Levchenko	Plec: Kobieta	Data ur.: 31.05.2002	Pesel: null
Telefon: 347-583-481	Pensja: 4000	Zmień dane	

Organizatorzy:

Aleksandra Glos	Plec: Kobieta	Data ur.: 01.01.2000	Pesel: 20457274916
Telefon: 934-134-246	Pensja: 2100	Zmień dane	

Michał Papiernicki	Plec: Mężczyzna	Data ur.: 12.02.2000	Pesel: 84596481631
Telefon: 749-136-425	Pensja: 2100	Zmień dane	

Rys. 10: Podstrona pracownicy jako developer

Zmień dane Mateusz Izbicki

nazwisko	Izbicki
imie	Mateusz
pesel	3940284932
haslo	...
data_urodzenia	23.03.2000
telefon	495-395-325
plec	Mężczyzna
stanowisko	Developer
pensja	4000
miasto	Suwałki
ulica	Związków Walki
nr_lokalu	93c/8
kod_poczty	16-401

Telefon: 934-134-246 Pensja: 2100

Zmień dane

Rys. 11: Podstrona pracownicy jako developer

Na podstronie, dzięki której możemy zarządzać pracownikami [Rys. 10] (dostęp ma tylko developer), możemy zobaczyć dane pracowników oraz je zmieniać. Kliknięciu pokazuje się popup (Rys. 11), dzięki któremu możemy dane te zmienić. Zmienione dane są aktualizowane metodą PUT.

5.3 Widoki organizatora

Uczestnicy wydarzeń:

Rodzaj: wystawa

Warszawa i my
Tadeusz Babicz
Numer sali: 203

Maciej Nikonski Tel: 247-374-573 Email: mnikonski@email.com

16 stycznia 2021
15:19 - 19:19

Zmień dane

^

Contemporary art in us
James Smith
Numer sali: 203

30 stycznia 2021
14:00 - 0:00

Zmień dane

▼

Rys. 12: Zarządzanie jako organizator (wgłód na tylko te wydarzenia, które organizuje)

Na Rys. 12 został pokazany widok Zarządzanie organizatora, ten widok się różni od widoku zarządzania Developerem tym, że organizator ma podgląd tylko to organizowanych przez niego wydarzeń, posiada możliwości ich zamiany, ale nie może dodawać nowych wydarzeń.

Zalogowany jako: Organizator

Twoje dane:

Aleksandra Glos	Plec: Kobieta	Data ur.: 01.01.2000	Pesel: 20457274916
Telefon: 934-134-246	Pensja: 2100		

Panel sterowania:

Organizowane przeze mnie wydarzenia

Zmień dane

Rys. 13: Mój profil jako organizator

Na Rys. 13 został pokazany widok My Profile organizatora, tutaj organizator może zmienić swoje dane osobowe (metoda PUT, czyli modyfikacja danych znajdujących się w tabeli) oraz przejść do dostępnych mu stron zarządzania danym Domem Kultury (w danym przypadku do zarządzania organizowanych przez niego wydarzeń).

5.4 Widoki administratora

Zalogowany jako: Administrator

Twoje dane:

Jakub Matysiak Płeć: Mężczyzna Data ur.: 12.04.2000 Pesel: 38591602835
Telefon: 924-854-249 Pensja: 2100

[Zmień dane](#)

Panel sterowania:

Zmiana danych / utworzenie wydarzenia / uczestników

Rys. 14: Mój profil jako administrator

Na Rys. 14 został pokazany widok My Profile administratora, który prawie się nie różni od widoku organizatora - może tutaj zmieniać swoje dane osobowe oraz przejść do zarządzania wydarzeniami (ale już w bardziej rozszerzony sposób).

Uczestnicy wydarzeń:

[Dodaj wydarzenie](#)

Rodzaj: wystawa

Violin

Ola Babicz

Numer sali: 102

18 stycznia 2021

01:45 - 5:45

[Zmień dane](#)



Contemporary art in us

James Bond

Numer sali: 102

17 stycznia 2021

09:00 - 12:00

[Zmień dane](#)



Rys. 15: Widok zarządzania wydarzeniami dla administratora (widzi wszystkie)

Na Rys. 15 został pokazany widok Zarządzania administratora, który się nie różni od widoku i posiadanych funkcjonalności zarządzania wydarzeniami przez Developera.

5.5 Widoki uczestnika



Warszawskie
Domy
Kultury
FLEX

Główna Wystawy Warsztaty Moj profil Witaj,
Katarzyna! Exit

Wybierz Dom Kultury:

Dom Kultury numer 1 ▾

City: Warszawa
Street: Twarda 12/150
Postal code: 01-001
Phone: 575-622-907
Opens: 07:00
Closes: 19:00

Wystawy w domie Kultury numer 1:

Violin

Wystawa muzyczna Ola Babicz Numer sali: 102

18 stycznia 2021
02:45 - 6:45

Zapisz się

Contemporary art in us

Wystawa malarska James Bond Numer sali: 102

17 stycznia 2021
10:00 - 13:00

Wypisz się

Rys. 16: Widok do wyświetlenia wystaw w pewnym Domu Kultury z perspektywy użytkownika

Na Rys. 16 został przedstawiony widok na wszystkie wystawy które są w danym domu kultury. Wybór domu kultury można zmienić po lewej stronie strony (wykorzystanie metody GET aby wyszukać każdy dom kultury dostępny w bazie). następnie jako uczestnik możemy zapisać się na dowolną wystawę (pobranie wszystkich wystaw metoda GET, zapisanie się na wystawę metoda POST) oraz z nią wypisać się (jeżeli jest już zapisany, za pomocą metody DELETE).



Warszawskie
Domy
Kultury
FLEX

Główna Wystawy Warsztaty Moj profil Witaj,
Katarzyna! Exit

Wybierz Dom Kultury:

Dom Kultury numer 1 ▾

City: Warszawa
Street: Twarda 12/150
Postal code: 01-001
Phone: 575-622-907
Opens: 07:00
Closes: 19:00

Warsztaty w domie Kultury numer 1:

Patrz i rysuj

Wykładowca: Piotr Jabłonski Numer sali: 102

28 stycznia 2021
09:00 - 11:00

Zapisz się

AAAAA

Wykładowca: AAAA AAAA Numer sali: 103

31 stycznia 2021
11:00 - 13:00

Matematyka dla wszystkich

Zapisz się

Rys. 17: Widok do wyświetlenia warsztatów w pewnym Domu Kultury z perspektywy użytkownika

Rys. 17 jest można powiedzieć identyczny jak Rys. 16, ale różni się tym że na Rys. 17 widzimy dostępne warsztaty w danym domu kultury.



Zalogowany jako: Uczestnik

Twoje dane:

Katarzyna Zyn

Email: kzyn@email.com Telefon: 247-341-249

Telefon: 247-341-249

Zmień dane

Jesteś zapisany na wydarzenia:

Rodzaj: wystawa

Contemporary art in us

James Bond

Numer sali: 102

17 stycznia 2021

10:00 - 13:00

[Wypisz się](#)

Rys. 18: Widok My Profile z perspektywy użytkownika

Na Rys. 18 został pokazany widok My Profile uczestnika, tutaj uczestnik, tak samo jak i pracownicy, może zmieniać swoje dane osobowe. Tutaj uczestnik może przeglądać wydarzenia na które zapisał się oraz z nich wypisywać się. Uczestnik nie posiada możliwości zarządzania czymkolwiek prócz listy wydarzeń, w których uczestniczy.