1. Użytkownik może obejrzeć plan konferencji.

**GET:**

<http://localhost:8080/api/conference/>

1. Użytkownik po podaniu swojego loginu może obejrzeć prelekcje, na które się zapisał.

**POST:**

<http://localhost:8080/api/user/schedule>

@RequestBody:

{

    "username": "jannowak"

}

1. Jeżeli prelekcja ma jeszcze wolne miejsca, użytkownik ma możliwość dokonania rezerwacji. Podczas dokonywania rezerwacji powinien podać swój login oraz adres e-mail;

Jeżeli w systemie istnieje już użytkownik z danym loginem, ale z innym adresem e-mail, system powinien zaprezentować komunikat „Podany login jest już zajęty”;

Poprawne dokonanie rezerwacji skutkuje wysłaniem powiadomienia użytkownikowi na podany przez niego adres (wysyłka wiadomości - patrz wskazówki):

**POST:**

<http://localhost:8080/api/conference/lectures/reserve>

@RequestBody:

{

    "blockId" : "2",

    "lectureId": "5",

    "username": "jannowak",

    "email": "jannowak@email.com"

}

1. Użytkownik może anulować rezerwację.

**DELETE:**

<http://localhost:8080/api/user/schedule/cancel>

@RequestBody:

{

    "reservationId": "1"

}

1. Użytkownik może zaktualizować swój adres e-mail.

**PATCH:**

<http://localhost:8080/api/user/edit-form>

@RequestBody:

{

    "username": "jannowak",

    "email": "jannowak11111@email.com"

}

1. System umożliwia wyświetlenie listy zarejestrowanych użytkowników wraz z ich adresami e-mail.

**GET:**

<http://localhost:8080/api/conference/users>