# 1.Пользователь.

### 1.1. Регистрация.

POST /api/v1/register

Headers:

Content-Type: application/json

Body

{  
 "username": "Al'bert", // Логин  
 "firstName": "Альберт", // Имя  
 "lastName": "Гафаров", // Фамилия  
 "email": "danilek2.gafarov1@gmail.com", // Адрес электронной почты  
 "password":"123" // Пароль  
}

### 1.2. Авторизация.

POST /api/v1/auth/login

Headers:

Content-Type: application/json

Body

{  
 "username":"danilek", // Логин  
 "password":"1234" // Пароль  
}

### 1.3. Получение информации о пользователе.

GET /api/v1/users/userinfo

Headers:

Authorization: Bearer\_{token}

Response body:

{  
 "id": 20, // Идентификатор  
 "username": "danilek", // Логин  
 "firstName": "Даниэль", // Имя  
 "lastName": "Гафаров", // Фамилия  
 "email": "danilek.gafarov@gmail.com" // Адрес электронной почты  
}

### 1.4. Изменение данных пользователя.

PUT /api/v1/users/change\_info

Headers:

Content-Type: application/json

Authorization: Bearer\_{token}

Body:

{  
 "username": "Al'bert", // Логин  
 "firstName": "Альберт", // Имя  
 "lastName": "Гафаров", // Фамилия  
 "email": "danilek2.gafarov1@gmail.com", // Адрес электронной почты  
 "password":"123" // Пароль  
}

### 1.5. Поиск пользователя по части username, имени, фамилии, e-mail

GET /api/v1/users/search\_people/{часть имени} // от 3 до 15 символов

Headers:

Authorization: Bearer\_{token}

Response body:

[  
 {  
 "id": 1,  
 "username": "albert",  
 "firstName": "Альберт",  
 "lastName": "Гафаров",  
 "email": "ag3d@mail.ru"  
 },  
 {  
 "id": 2,  
 "username": "elena",  
 "firstName": "Елена",  
 "lastName": "Гафарова",  
 "email": "elena.gafarova1@gmail.com"  
 }  
]

Поиск зарегистрированных пользователей по части имени, за исключением себя.

# 2. Сообщения.

### 2.1. Отправить сообщение.

POST /api/v1/messages

Headers:

Content-Type: application/json

Authorization: Bearer\_{token}

Body:

{  
 "destinationId": 1, // id получателя.  
 "textMessage": "Привет! Как дела?" // текст сообщения.  
}

Response body:

{  
 "id": 4, // Идентификатор сообщения  
 "sender": { // Отправитель  
 "id": 2, // Идентификатор отправителя  
 "username": "elena" // Логин отправителя  
 },  
 "destination": { // Получатель  
 "id": 1, // Идентификатор получателя  
 "username": "albert" // Логин получателя  
 },  
 "createdDate": "2021-02-06T16:25:03.070+00:00", // Дата отправки сообщения  
 "readDate": null, // Дата прочтения сообщения  
 "status": "NOT\_READ", // Статус сообщения  
 "textMessage": "У меня все хорошо!", // Текст сообщения  
 "bet": null // Спор, прикрепленный к сообщению  
}

### 2.2. Получить переписку.

GET /api/v1/messages/with/{id} // Уникальный идентификатор собеседника.

Headers:

Authorization: Bearer\_{token}

### 2.3. Прочитать сообщение. Пометить сообщение прочтенным.

GET /api/v1/messages/{id} // Уникальный идентификатор сообщения.

Response body:

{  
 "id": 4, // Идентификатор сообщения  
 "sender": { // Отправитель  
 "id": 2, // Идентификатор отправителя  
 "username": "elena" // Логин отправителя  
 },  
 "destination": { // Получатель  
 "id": 1, // Идентификатор получателя  
 "username": "albert" // Логин получателя  
 },  
 "createdDate": "2021-02-06T16:25:03.070+00:00", // Дата отправки сообщения  
 "readDate": null, // Дата прочтения сообщения  
 "status": "NOT\_READ", // Статус сообщения  
 "textMessage": "У меня все хорошо!", // Текст сообщения  
 "bet": null // Спор, прикрепленный к сообщению  
}

# 3.Споры

### 3.1. Предложить спор

POST /api/v1/bets

Headers:

Content-Type: application/json

Authorization: Bearer\_{token}

Body:

{  
 "reasonMessageId": 1,// Сообщение, которое стало причиной спора  
 "definition": "Я изучу Spring к марту очень хорошо",// Версия инициатора  
 "wager": "1000 р",// Ставка  
 "finishDate": "2021-02-28"// Дата завершения спора  
}

### 3.2. Получить спор

POST /api/v1/bets/{id} //Идентификатор спора

Headers:

Authorization: Bearer\_{token}

Response body:

{  
 "id": 13, // Идентификатор спора  
 "initiator": { // Инициатор спора  
 "id": 1,  
 "username": "albert"  
 },  
 "definition": "Я изучу Spring к марту очень хорошо",// Условие спора  
 "wager": "1000 р", // Ставка  
 "finishDate": "2021-02-28T00:00:00.000+00:00", // Дата окончания спора  
 "opponent": { // Оппонент  
 "id": 2,  
 "username": "elena"  
 },  
 "initiatorBetStatusEnum": "OFFERED", // Статус спора, установленный инициатором  
 "opponentBetStatusEnum": "OFFERED", // Статус спора, установленный оппонентом  
 "status": "ACTIVE", // Общий статус спора  
 "createdDate": "2021-02-07T09:05:43.894+00:00" // Дата предложения спора  
}

Возможные статусы спора:

*OFFERED* // Предложен  
,*CANCEL* // Спор отклонен  
,*ACCEPTED* // Принят  
,*WIN* // Победил  
,*LOSE* // Проиграл  
,*STANDOFF* // Ничья  
,*WAGERPAID* // Выигрыш оплатил  
,*WAGERRECIEVED* // Выигрыш получил