**API 스펙 문서**

1. 설계 사상
   1. 프로그래밍 과제임을 고려해 생략되거나 간소화된 내용:

* State-less하게 동작해야 하는 API 서버의 특성을 고려해, 로그인은 세션, 쿠키등을 이용하지 않고 매 요청에 Basic Auth token을 보내는 방식으로 구현한다.
* Post가 존재하는 업무의 경우, 목록을 보여주는 별도의 Get은 구현하지 않는다. 이는 추후 개선 가능하다.
* 기본적으로 필수 파라미터는 프로그램 내부에서 확인하지만, 이에 대한 별도 메시지 처리는 하지 않는다. 이는 추후 개선 가능하다.
* Method Not Allowed에 대한 별도 메시지 처리는 하지 않는다. 이는 추후 개선 가능하다.
* Csrf에 대한 처리는 하지 않는다. 이는 추후 개선 가능하다.
* 공통코드 변환은 사용하지 않는다. 이는 추후 개선 가능하다.
  1. 의도된 내용
* Timezone 처리는 UTC time 기준으로 한다. 이는 서버에서 UTC로 시간을 저장하고, 클라이언트에서 Timezone 처리를 하는 것이 용이하다고 판단하기 때문이다.

1. 스펙 개요

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **순번** | **Endpoints** | **Allowed methods** | **Auth** | **비고** |
| 1 | http://localhost:8000/api/signup/ | POST | - | 회원가입 |
| 2 | http://localhost:8000/api/list/ | GET | - | 목록조회 |
| 3 | http://localhost:8000/api/request/allocation/ | POST | Basic | 배차요청 |
| 4 | http://localhost:8000/api/request/assignment/ | POST | Basic | 기사배차 |

1. 스펙 상세

2.1 회원가입

* 1. **ENDPOINT**: http://localhost:8000/api/signup/
  2. **METHOD**: Post
  3. **Authorization**: 없음
  4. **Body Parameters**: - 모든 http 파라미터는 문자열 유형이다.

|  |  |  |
| --- | --- | --- |
| 순번 | 변수명 | 값 예시 |
| 1 | email | “somebody@somewhere.com” |
| 2 | pw | “somepassword” |
| 3 | type | {“user” | “driver”} |

* 1. **Response**: 성공적 생성시 200 OK, 필수값 부재 시 400 오류를 반환한다.
  2. **Description**: 회원가입을 수행한다.

2.2 목록조회

1. **ENDPOINT**: http://localhost:8000/api/list/
2. **METHOD**: GET
3. **Authorization**: 없음
4. **Body Parameters**: 없음
5. **Response**: 항상 200 OK와 함께 Json 결과를 반환한다.

|  |
| --- |
| 응답 예시 |
| {      "data": [          {              "id": 4,              "status": "ASSG",              "from\_addr": "dd",              "to\_addr": "aaa",              "user\_id": 2,              "driver\_id": 4,              "requested": "2020-06-05T01:13:57.862Z",              "assigned": "2020-06-05T01:14:17.195Z"          },          {              "id": 3,              "status": "INIT",              "from\_addr": "dd",              "to\_addr": "aaa",              "user\_id": 2,              "driver\_id": null,              "requested": "2020-06-05T01:13:56.215Z",              "assigned": null          }      ]  } |

1. **Description**: 현재 기준 모든 배차요청, 할당결과를 최근에 추가된 요청 순으로 정렬하여 반환합니다. 요청 상태의 경우 상태값이 “INIT”, 배차가 된 상태의 경우 “ASSG”으로 표기되어 나타납니다.

2.3 배차요청

1. **ENDPOINT**: http://localhost:8000/api/request/allocation/
2. **METHOD**: Post
3. **Authorization**: Basic – username과 password를 header로 함께 전송해야 함
4. **Body Parameters**: - 모든 http 파라미터는 문자열 유형이다.

|  |  |  |
| --- | --- | --- |
| 순번 | 변수명 | 값 예시 |
| 1 | from\_addr | “서울시 종로구” |
| 2 | to\_addr | “경기도 성남시” |

1. **Response**: 성공적 생성시 200 OK, 필수값 부재 시 400 오류를 반환한다.
2. **Description**: 출발지와 목적지를 토대로 배차 요청을 생성한다. 생성된 배차 요청은 목록보기에서 조회가 가능하다.

2.4 기사배차

1. **ENDPOINT**: http://localhost:8000/api/request/assignment/
2. **METHOD**: Post
3. **Authorization**: Basic – username과 password를 header로 함께 전송해야 함
4. **Body Parameters**: - 모든 http 파라미터는 문자열 유형이다.

|  |  |  |
| --- | --- | --- |
| 순번 | 변수명 | 값 예시 |
| 1 | request\_id | “1” |

1. **Response**: 성공적 생성시 200 OK, 필수값 부재 이거나 이미 할당된 요청건을 다시 할당하고자 할 때 400 오류를 반환한다.
2. **Description**: 배차 요청 식별자를 기준으로 배차를 할당한다. 할당은 상태값이 “INIT”인 경우에만 할당이 이루어질 수 있으며 할당이 완료되면 상태값이 “ASSG”로 변경되게 된다. 이미 할당된 요청건은 다시 할당할 수 없다.