

1. 백엔드 API 흐름 및 모듈

4-1) 백엔드 API 흐름

- (1) API 요청 (포스트맨 또는 프론트엔드)
- (2) 컨트롤러의 요청 DTO에서 요청 값 유효성 검사
- (3) 인증이 필요한 API 는 AuthGuard에서 JWT 토큰 검증
- (4) 비즈니스 로직은 서비스에서 실행 (필요한 경우 모델 ORM를 활용)
- (5) 예외처리는 서비스에서 진행하며 500에러가 발생하는 경우 Slack 알람 발생
- (6) 성공은 컨트롤러에서 반환 응답 DTO에서 타입 검사
- (7) e2e 테스트로 API 테스트를 진행
- (8) Swagger API 문서를 생성

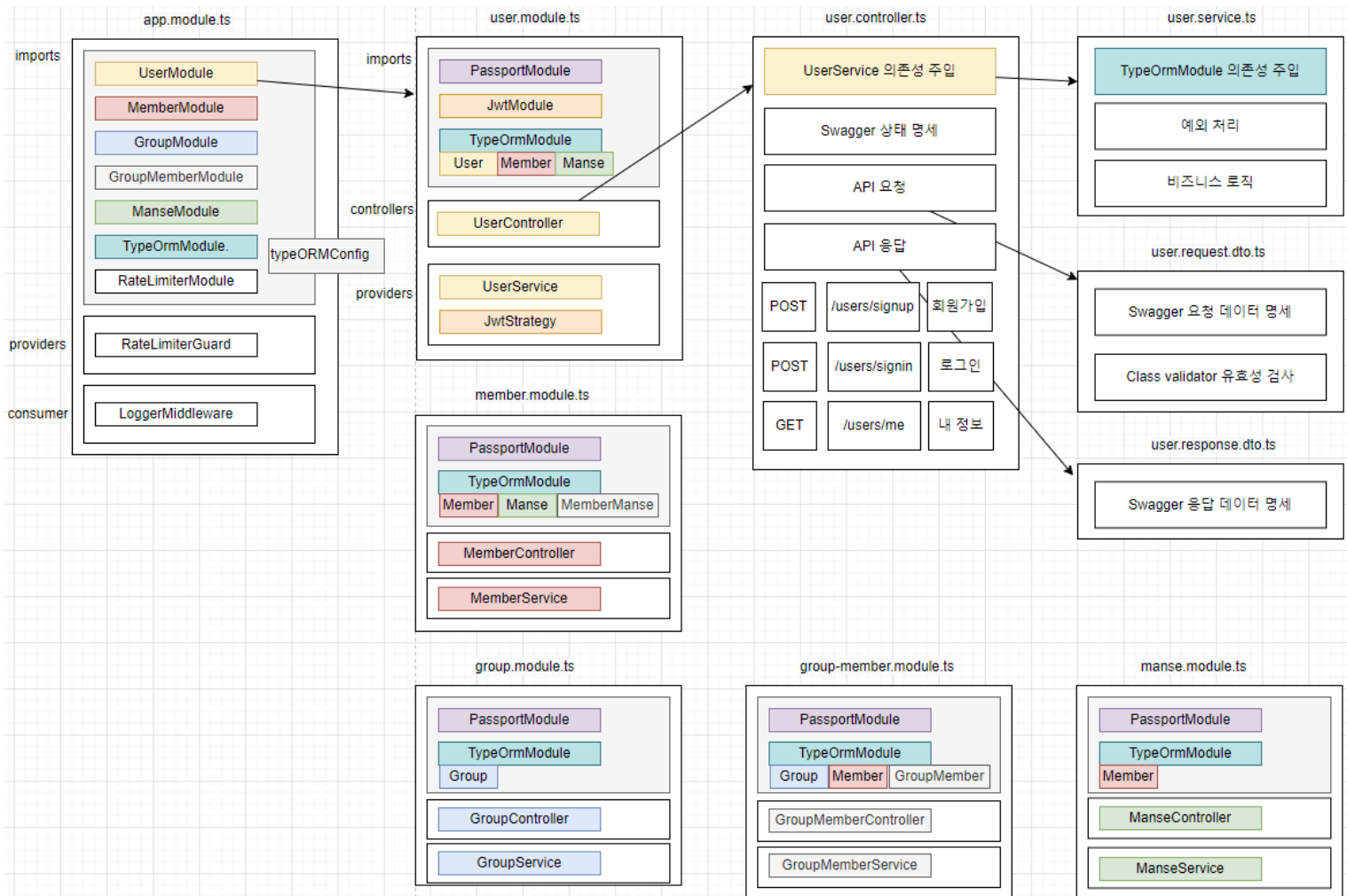
```
saju-backend-nestjs
├─ mysql
│   └─ sqls
│       ├── Dump20220620.sql    //모든 테이블 덤프
│       └─ manses.sql          //만세력 테이블 덤프
├─ src
│   ├── commons                //공통
│   │   ├── auth              //인증
│   │   │   ├── jwt.strategy.ts //JWT 토큰 검증
│   │   │   └─ token.decorator.ts
│   │   └─ dto
│   │       ├── exception.dto.ts //예외 DTO
│   │       ├── member.dto.ts    //멤버 DTO
│   │       ├── pagination.dto.ts //페이지네이션 DTO
│   │       └─ saju.dto.ts       //사주 DTO
│   ├── exceptions
│   │   └─ httpException.filter.ts //예외 필터
│   └─ middlewares
│       └─ logger.middleware.ts  //로그 미들웨어
```

```
| | └ saju
| |   └ birth-to-saju.ts      //생일을 사주로 변환
| |   └ format.ts            //사주 반환 포맷
| |   └ fortune.ts           //운세 변환
| |   └ saju-data.ts          //사주 데이터
| |   └ slack
| |     └ slack.ts            //Slack 500에러
| └ configs
|   └ typeorm.config.ts      //TypeORM 설정
| └ entities
|   └ group-member.entity.ts  //그룹 멤버 entity
|   └ group.entity.ts         //그룹 entity
|   └ manse.entity.ts         //만세력 entity
|   └ member-manse.entity.ts  //멤버 만세력 entity
|   └ member.entity.ts        //멤버 entity
|   └ user.entity.ts          //회원 entity
| └ group                     //그룹
|   └ dto
|     └ group.request.dto.ts  //그룹 request DTO
|     └ group.response.dto.ts //그룹 response DTO
|     └ group.controller.ts   //그룹 컨트롤러
|     └ group.module.ts       //그룹 모듈
|     └ group.service.ts      //그룹 서비스
| └ group-member
|   └ dto
|     └ group-member.response.dto.ts //그룹 멤버 response DTO
|     └ group-member.controller.ts   //그룹 멤버 컨트롤러
|     └ group-member.module.ts       //그룹 멤버 모듈
|     └ group-member.service.ts      //그룹 멤버 서비스
| └ manse
|   └ dto
|     └ manse.response.dto.ts //만세력 DTO
|     └ manse.controller.ts   //만세력 컨트롤러
|     └ manse.module.ts       //만세력 모듈
|     └ manse.service.ts      //만세력 서비스
```

```
| └ member
|   └ dto
|     └ member.request.dto.ts      //멤버 request DTO
|     └ member.response.dto.ts    //멤버 response DTO
|     └ member.controller.ts      //멤버 컨트롤러
|     └ member.module.ts          //멤버 모듈
|     └ member.service.ts         //멤버 서비스
| └ user
|   └ dto
|     └ user.request.dto.ts       //회원 request DTO
|     └ user.response.dto.ts     //회원 response DTO
|     └ user.controller.ts       //회원 컨트롤러
|     └ user.module.ts           //회원 모듈
|     └ user.service.ts          //회원 서비스
| └ app.controller.ts             //앱 컨트롤러
| └ app.module.ts                 //앱 모듈
| └ main.ts                       //main
└ test
  └ group-member.e2e-spec.ts      //그룹 멤버 e2e 테스트
  └ group.e2e-spec.ts            //그룹 e2e 테스트
  └ jest-e2e.json
  └ manse.e2e-spec.ts            //만세력 e2e 테스트
  └ member.e2e-spec.ts           //멤버 e2e 테스트
  └ user.e2e-spec.ts             //회원 e2e 테스트
└ .dockerignore
└ .env                          //환경변수 파일
└ .env.template
└ .eslintrc.js
└ .gitignore
└ .prettierrc
└ buildspec.yml                 //빌드스펙 - AWS Codebuild
└ docker-compose.yml            //도커 컴포즈 - MySQL 생성
└ Dockerfile                    //도커 파일
└ nest-cli.json
└ package-lock.json
```

└ package.json //package.json
 └ README.md
 └ tsconfig.build.json
 └ tsconfig.json
 └ webpack-hmr.config.js

NestJS 모듈



2. package.json

package.json에는 프로그램 실행 명령어 (scripts) 및 라이브러리 버전 관리 (dependencies)를 합니다.

(1) scripts 를 보시면 프로그램 실행 명령어가 있습니다. 로컬 서버 실행 명령어는 npm run start:dev 입니다.

(2) dependencies 에는 bcryptjs, cross-env, @nestjs/platform-express, passport, passport-jwt, @nestjs/passport, @nestjs/jwt, typeorm, @nestjs/typeorm, mysql2, moment, class-validator, @slack/client, nestjs-rate-limiter, pm2, nestjs/swagger, express-basic-auth, swagger-ui-express

bcryptjs	패스워드 암호화시 사용되는 라이브러리입니다. DB에 저장할 때 암호화 된 값이 저장됩니다.
cross-env	윈도우와 맥 계열에서 package.json의 scripts에 nodejs 관련 실행 명령어를 작성할 때 공통적으로 처리하기 위해 필요합니다.
@nestjs/platform-express	프론트엔드와 HTTP 통신을 쉽게 해주는 웹 서버 라이브러리입니다. 백엔드 API 서버를 쉽게 만들어 줍니다
passport, passport-jwt, @nestjs/passport, @nestjs/jwt	JWT 토큰 생성 및 검증에 필요한 라이브러리입니다.
typeorm, @nestjs/typeorm	DB 모델 생성과 관계를 통해 sql를 쉽게 사용할 수 있도록 해주는 라이브러리 입니다.

mysql2	typeorm에서 mysql 서버와 연동을 위한 라이브러리 입니다.
moment	시간 관련 라이브러리 입니다.
class-validator	유효성 검사 라이브러리 입니다.
@slack/client	slack 에 500에러 알림을 보내기 위해 사용합니다. 보통 에러가 발생해도 사용자가 개발자에게 전달하지 않으면 알 수 없습니다. 이에 대한 해결책으로 500에러가 발생하면 자동으로 500에러 알림과 URI와 body, 발생시간을 포함시켜서 개발자 slack에 메시자가 보내지도록 할 수 있습니다.
nestjs-rate-limiter	<p>사용자 API 횟수를 제한하기 위한 용도입니다. 1분 동안 30회 이상 API 요청을 하는 경우는 정상적인 동작방식이 아니라고 생각합니다. 서비스에 문제를 일으키는 DDoS라고 판단해서 1분 동안 30회 이상 API를 요청하면 429 Too many request 를 반환 합니다. 서비스 마다 기간(duration) 및 횟수(points) 는 적절히 설정해야 합니다.</p> <p>요청 값 유효성검사는 DTO 파일에서 class-validator를 사용하여 진행합니다.</p>

pm2	프로세스 매니저로 배포 환경 테스트를 위해서 필요합니다.
nestjs/swagger, express-basic-auth, swagger-ui-express	swagger API 문서 자동화와 관련있습니다.

(3) devDependencies에 supertest가 있는데 e2e API 테스트 자동화와 관련있습니다.

3. API 문서

6-2) API 문서

프론트엔드 또는 앱 개발자와 공유하는 문서입니다.

공유하는 형태는 Google docs, Notion, Swagger 등 다양합니다.

회원 API		
회원 가입	POST	/users/signup
Request Header		

Request Body	<div>email : 이메일</div> <div>gender : MALE (남성) FEMALE (여성)</div> <div>birthdayType : SOLAR (양력) LUNAR (음력)</div> <div>예시</div> <div>{ "email" : "test@test.com", "password" : "1234", "nickname" : "koo", "gender" : "MALE", "birthdayType" : "SOLAR", "birthday" : "19870213", "time" : "0710" }</div>	
Response Body	<div>예시 : 성공 201</div> <div>{ "statusCode": 201, "message": "회원가입 성공", "data": { "accessToken": "*****" } }</div> <div>예시 : 실패 409</div> <div>{ "statusCode": 409, "message": "이미 사용중인 이메일 입니다." }</div>	
로그인	POST	/users/signin

Request Header		
Request Body	<p>email : 이메일</p> <p>예시</p> <pre>{ "email" : "test@test.com", "password" : "1234" }</pre>	
Response Body	<p>예시 : 성공 200</p> <pre>{ "statusCode": 200, "message": "로그인 성공", "data": { "accessToken": "***" } }</pre> <p>예시 : 실패 401</p> <pre>{ "statusCode": 401, "message": "로그인 실패" }</pre>	
내정보 가져오기	GET	/users/me
Request Header	Authorization : bearer {{access_token}}	

Request Body	
-----------------	--

Response Body	<p>age : 숫자</p> <p>gender : MALE (남성) FEMALE (여성)</p> <p>birthdayType : SOLAR (양력) LUNAR (음력)</p> <p>예시</p> <pre>{ "statusCode": 200, "message": "내정보보기 성공", "data": { "member": { "id": 1, "nickname": "test", "age": 36, "birthday": "1987-02-13", "time": null, "birthdayType": "SOLAR", "gender": "MALE", "createdAt": "2022-05-05 21:08:15" }, "saju": { "bigFortuneNumber": 3, "bigFortuneStartYear": 1990, "seasonStartTime": null, "yearSky": { "chinese": "丁", "korean": "정", "fiveCircle": "화", "fiveCircleColor": "#F44336", "tenStar": "편재",</pre>
---------------	--

```
    "minusPlus": "음",
    "jijangGan": null
  },
  "yearGround": {
    "chinese": "卯",
    "korean": "묘",
    "fiveCircle": "목",
    "fiveCircleColor": "#4CAF50",
    "tenStar": "식신",
    "minusPlus": "음",
    "jijangGan": {
      "first": {
        "chinese": "甲",
        "korean": "갑",
        "fiveCircle": "목",
        "fiveCircleColor": "#4CAF50",
        "minusPlus": "양",
        "rate": 10
      },
      "second": null,
      "third": {
        "chinese": "乙",
        "korean": "을",
        "fiveCircle": "목",
        "fiveCircleColor": "#4CAF50",
        "minusPlus": "음",
        "rate": 20
      }
    }
  }
}
```

```
},
"monthSky": {
  "chinese": "戌",
  "korean": "임",
  "fiveCircle": "수",
  "fiveCircleColor": "#039BE5",
  "tenStar": "겁재",
  "minusPlus": "양",
  "jijangGan": null
},
"monthGround": {
  "chinese": "寅",
  "korean": "인",
  "fiveCircle": "목",
  "fiveCircleColor": "#4CAF50",
  "tenStar": "상관",
  "minusPlus": "양",
  "jijangGan": {
    "first": {
      "chinese": "戊",
      "korean": "무",
      "fiveCircle": "토",
      "fiveCircleColor": "#FFD600",
      "minusPlus": "양",
      "rate": 7
    },
    "second": {
      "chinese": "丙",
      "korean": "병",
```

```
        "fiveCircle": "화",
        "fiveCircleColor": "#F44336",
        "minusPlus": "양",
        "rate": 7
    },
    "third": {
        "chinese": "甲",
        "korean": "갑",
        "fiveCircle": "목",
        "fiveCircleColor": "#4CAF50",
        "minusPlus": "양",
        "rate": 16
    }
},
"daySky": {
    "chinese": "쫘",
    "korean": "계",
    "fiveCircle": "수",
    "fiveCircleColor": "#039BE5",
    "tenStar": "비견",
    "minusPlus": "음",
    "jijangGan": null
},
"dayGround": {
    "chinese": "巳",
    "korean": "사",
    "fiveCircle": "화",
    "fiveCircleColor": "#F44336",
```

```
"tenStar": "정재",
"minusPlus": "음",
"jijangGan": {
  "first": {
    "chinese": "戊",
    "korean": "무",
    "fiveCircle": "토",
    "fiveCircleColor": "#FFD600",
    "minusPlus": "양",
    "rate": 7
  },
  "second": {
    "chinese": "庚",
    "korean": "경",
    "fiveCircle": "금",
    "fiveCircleColor": "#E0E0E0",
    "minusPlus": "양",
    "rate": 7
  },
  "third": {
    "chinese": "丙",
    "korean": "병",
    "fiveCircle": "화",
    "fiveCircleColor": "#F44336",
    "minusPlus": "양",
    "rate": 16
  }
}
},
```

```
"timeSky": null,
"timeGround": null
},
"fortune": {
  "bigFortune": {
    "number": 33,
    "sky": {
      "chinese": "戊",
      "korean": "무",
      "fiveCircle": "토",
      "fiveCircleColor": "#FFD600",
      "tenStar": "정관",
      "minusPlus": "양",
      "jijangGan": null
    },
    "ground": {
      "chinese": "戊",
      "korean": "술",
      "fiveCircle": "토",
      "fiveCircleColor": "#FFD600",
      "tenStar": "정관",
      "minusPlus": "양",
      "jijangGan": {
        "first": {
          "chinese": "辛",
          "korean": "신",
          "fiveCircle": "금",
          "fiveCircleColor": "#E0E0E0",
          "minusPlus": "음",
```



```
        "rate": 9
      },
      "second": {
        "chinese": "丁",
        "korean": "정",
        "fiveCircle": "화",
        "fiveCircleColor": "#F44336",
        "minusPlus": "음",
        "rate": 3
      },
      "third": {
        "chinese": "戊",
        "korean": "무",
        "fiveCircle": "토",
        "fiveCircleColor": "#FFD600",
        "minusPlus": "양",
        "rate": 18
      }
    }
  },
  "smallFortune": {
    "number": "2022",
    "sky": {
      "chinese": "壬",
      "korean": "임",
      "fiveCircle": "수",
      "fiveCircleColor": "#039BE5",
      "tenStar": "겁재",
```

```
    "minusPlus": "양",
    "jijangGan": null
  },
  "ground": {
    "chinese": "寅",
    "korean": "인",
    "fiveCircle": "목",
    "fiveCircleColor": "#4CAF50",
    "tenStar": "상관",
    "minusPlus": "양",
    "jijangGan": {
      "first": {
        "chinese": "戊",
        "korean": "무",
        "fiveCircle": "토",
        "fiveCircleColor": "#FFD600",
        "minusPlus": "양",
        "rate": 7
      },
      "second": {
        "chinese": "丙",
        "korean": "병",
        "fiveCircle": "화",
        "fiveCircleColor": "#F44336",
        "minusPlus": "양",
        "rate": 7
      },
      "third": {
        "chinese": "甲",
```

```
        "korean": "갑",
        "fiveCircle": "목",
        "fiveCircleColor": "#4CAF50",
        "minusPlus": "양",
        "rate": 16
    }
}
},
"monthFortune": {
    "number": 5,
    "sky": {
        "chinese": "乙",
        "korean": "을",
        "fiveCircle": "목",
        "fiveCircleColor": "#4CAF50",
        "tenStar": "식신",
        "minusPlus": "음",
        "jijangGan": null
    },
    "ground": {
        "chinese": "巳",
        "korean": "사",
        "fiveCircle": "화",
        "fiveCircleColor": "#F44336",
        "tenStar": "정재",
        "minusPlus": "음",
        "jijangGan": {
            "first": {
```

```
    "chinese": "戊",
    "korean": "무",
    "fiveCircle": "토",
    "fiveCircleColor": "#FFD600",
    "minusPlus": "양",
    "rate": 7
  },
  "second": {
    "chinese": "庚",
    "korean": "경",
    "fiveCircle": "금",
    "fiveCircleColor": "#E0E0E0",
    "minusPlus": "양",
    "rate": 7
  },
  "third": {
    "chinese": "丙",
    "korean": "병",
    "fiveCircle": "화",
    "fiveCircleColor": "#F44336",
    "minusPlus": "양",
    "rate": 16
  }
}
},
"dayFortune": {
  "number": 5,
  "sky": {
```

```
"chinese": "戊",
"korean": "무",
"fiveCircle": "토",
"fiveCircleColor": "#FFD600",
"tenStar": "정관",
"minusPlus": "양",
"jijangGan": null
},
"ground": {
  "chinese": "午",
  "korean": "오",
  "fiveCircle": "화",
  "fiveCircleColor": "#F44336",
  "tenStar": "편재",
  "minusPlus": "양",
  "jijangGan": {
    "first": {
      "chinese": "丙",
      "korean": "병",
      "fiveCircle": "화",
      "fiveCircleColor": "#F44336",
      "minusPlus": "양",
      "rate": 10
    },
    "second": {
      "chinese": "己",
      "korean": "기",
      "fiveCircle": "토",
      "fiveCircleColor": "#FFD600",
```

	<pre> "minusPlus": "음", "rate": 10 }, "third": { "chinese": "丁", "korean": "정", "fiveCircle": "화", "fiveCircleColor": "#F44336", "minusPlus": "음", "rate": 10 } } } } } } } } }</pre>	
멤버 API		
멤버 추가	POST	/members
Request Header	Authorization : bearer {{access_token}}	

Request Body	gender : MALE (남성) FEMALE (여성) birthdayType : SOLAR (양력) LUNAR (음력) 예시 { "nickname" : "koo11", "gender" : "MALE", "birthdayType" : "SOLAR", "birthday" : "19870213", "time" : "0710" }	
Response Body	예시 : 성공 201 { "statusCode": 201, "message": "멤버 추가 성공" }	
멤버 리스트	GET	/members
Request Header	Authorization : bearer {{access_token}}	
Request Params	page : 0, size : 10	
Request Body		

Response Body	<p>type : MEMBER (멤버) USER (회원 – 본인)</p> <p>gender : MALE (남성) FEMALE (여성)</p> <p>birthdayType : SOLAR (양력) LUNAR (음력)</p> <p>age : 숫자</p> <p>예시</p> <pre>{ "statusCode": 200, "message": "멤버 리스트 성공", "data": { "totalItems": 2, "totalPages": 1, "currentPage": 0, "memberList": [{ "id": 2, "type": "MEMBER", "nickname": "koo11", "birthday": "1987-02-13", "birthdayType": "SOLAR", "gender": "MALE", "time": "07:10", "age": 36, "createdAt": "2022-05-01 22:28:09" }, { "id": 1, "type": "USER", "nickname": "koo",</pre>
---------------	---

	<pre>"birthday": "1987-02-13", "birthdayType": "SOLAR", "gender": "MALE", "time": "07:10", "age": 36, "createdAt": "2022-05-01 22:27:57" }] } }</pre>	
멤버 삭제	DELETE	/memembers/{id}
Request Header	Authorization : bearer {{access_token}}	
Request Body		
Response Body	<p>예시 : 성공 200</p> <pre>{ "statusCode": 200, "message": "멤버 삭제 성공" }</pre> <p>예시 : 실패 403</p> <pre>{ "statusCode": 403, "message": "해당 멤버의 삭제 권한이 없습니다." }</pre>	

그룹 API		
그룹 추가	POST	/groups
Request Header	Authorization : bearer {{access_token}}	
Request Body	{ "name" : "친구233233" }	
Response Body	{ "statusCode": 201, "message": "그룹 추가 성공" }	
그룹 리스트	GET	/groups
Request Header	Authorization : bearer {{access_token}}	
Request Params	page : 0, size : 10	
Request Body		

Response Body	<pre>{ "statusCode": 200, "message": "그룹 리스트 성공", "data": { "totalItems": 1, "totalPages": 1, "currentPage": 0, "groupList": [{ "id": 2, "name": "친구233233", "createdAt": "2022-05-03 12:31:14", "memberCount": 2 }] } }</pre>	
그룹 수정	PATCH	/groups/{id}
Request Header	Authorization : bearer {{access_token}}	
Request Body	<pre>{ "name" : "친구22222222" }</pre>	

Response Body	<pre>{ "statusCode": 200, "message": "그룹 수정 성공" }</pre> <p>예시 : 실패 403</p> <pre>{ "statusCode": 403, "message": "해당 그룹의 수정 권한이 없습니다." }</pre>	
그룹 삭제	DELETE	/groups/{id}
Request Header	Authorization : bearer {{access_token}}	
Request Body		
Response Body	<pre>{ "statusCode": 200, "message": "그룹 삭제 성공" }</pre> <p>예시 : 실패 403</p> <pre>{ "statusCode": 403, "message": "해당 그룹의 삭제 권한이 없습니다." }</pre>	
그룹 명 리스트	GET	/groups/names

Request Header	Authorization : bearer {{access_token}}	
Request Body		
Response Body	<pre>{ "statusCode": 200, "message": "그룹명 리스트 성공", "data": [{ "id": 1, "userId": 5, "name": "친구22", "createdAt": "2022-05-01T12:32:45.858Z", "updatedAt": "2022-05-01T12:32:45.858Z" }] }</pre>	
그룹 - 멤버 API		
그룹에 멤버 추가	POST	/groups/{groupId}/members/{memberId}
Request Header	Authorization : bearer {{access_token}}	
Request Body		

Response Body	<pre>{ "statusCode": 201, "message": "그룹에 멤버 추가 성공" } { "statusCode": 403, "message": "해당 그룹에 멤버 추가 권한이 없습니다." } { "statusCode": 409, "message": "해당 그룹에 같은 멤버가 존재합니다." }</pre>	
그룹 별 멤버 리스트	GET	/groups/{groupId}/members
Request Header	Authorization : bearer {{access_token}}	
Request Params	page : 0, size : 10	
Request Body		

Response Body	<p>type : MEMBER (멤버) USER (회원 - 본인)</p> <p>gender : MALE (남성) FEMALE (여성)</p> <p>birthdayType : SOLAR (양력) LUNAR (음력)</p> <p>age : 숫자</p> <p>{</p> <p> "statusCode": 200,</p> <p> "message": "그룹별 멤버 리스트 성공",</p> <p> "data": {</p> <p> "group": {</p> <p> "id": 2,</p> <p> "userId": 5,</p> <p> "name": "친구233233",</p> <p> "createdAt": "2022-05-03 12:31:14",</p> <p> "updatedAt": "2022-05-03 12:31:14"</p> <p> },</p> <p> "members": {</p> <p> "totalItems": 2,</p> <p> "totalPages": 1,</p> <p> "currentPage": 0,</p> <p> "memberList": [</p> <p> {</p> <p> "id": 6,</p> <p> "type": "USER",</p> <p> "nickname": "koo",</p> <p> "gender": "MALE",</p> <p> "birthdayType": "SOLAR",</p> <p> "birthday": "1987-02-13",</p> <p> "time": "07:10",</p> <p> }</p> <p>]</p> <p> }</p> <p> }</p>
---------------	---

	<pre> "age": 36, "createdAt": "2022-05-03 12:30:14" }, { "id": 7, "type": "MEMBER", "nickname": "koo11", "gender": "MALE", "birthdayType": "SOLAR", "birthday": "1987-02-13", "time": "07:10", "age": 36, "createdAt": "2022-05-03 12:31:08" }] } { "statusCode": 403, "message": "해당 그룹에 대한 권한이 없습니다." } </pre>	
그룹에서 멤버 제거	DELETE	/groups/{groupId}/members/{memberId}
Request Header	Authorization : bearer {{access_token}}	

Request Body		
Response Body	<pre>{ "statusCode": 200, "message": "그룹에서 멤버 제거 성공" } { "statusCode": 403, "message": "해당 그룹에서 멤버 제거 권한이 없습니다." }</pre>	
만세력 API		
만세력 가져오기	GET	/manse/members/{memberId}/fortune/:bigNum?/:smallNum?
Request Header	Authorization : bearer {{access_token}}	
Request Body		
Response Body	<pre>{ "statusCode": 200, "message": "만세력 가져오기 성공" } { "statusCode": 403, "message": "해당 멤버에 대한 접근 권한이 없습니다." }</pre>	

공통		
유효성 검증 에러	400	{ "statusCode": 400, "message": "잘못된 요청값 입니다.", "error": [{ "value": "SOLAR1", "msg": "Invalid value", "param": "birthdayType", "location": "body" }] }
인증 에러	401	{ "statusCode": 401, "message": "Unathorized" }
서버 에러	500	{ "statusCode": 500, "message": "서버 에러" }