

Table of Contents

1. UserController	2
1.1. 用户注册	2
1.2. 用户登录	3
1.3. 用户注销	4
1.4. 用户修改密码	5
1.5. 用户找回密码	6
1.6. 用户评价商品	9
1.7. 用户评价店铺	11
1.8. 用户评价商品评价	13
1.9. 用户评价商品评价	14
1.10. 用户评价商品评价	15
1.11. 用户评价	16
2. ReviewController	17
2.1. 商品评价	17
2.2. 店铺评价	21
2.3. 商品评价	22
2.4. 商品评价	24
2.5. 商品评价	24
2.6. 商品评价	26
2.7. 商品评价	27
2.8. 商品评价	28
3. StoreController	30
3.1. 商品上架	30
3.2. 商品下架	34
3.3. 商品上架	35
4. AdminController	36
4.1. 商品上架	36
4.2. 商品上架	37
4.3. 商品上架	39
4.4. 商品上架	40
5. ReviewAdminController	41
5.1. 商品评价	41
5.2. 商品评价	43
5.3. 商品评价	44

API 仕様 書

この仕様書は、API の仕様を定義する。~~~~~

CAUTION | 重要な事項を記載する

1. UserController

1.1. 新規登録

http-request

```
POST /api/users/normal HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 138
Host: localhost:8080

{
  "name" : "太郎",
  "email" : "zzanghd@gmail.com",
  "password" : "abc123!",
  "address" : "東京都 港区 港区"
}
```

Table 1. request-fields

Path	Type	Description
name	String	名前
email	String	Eメール
address	String	住所
password	String	パスワード

http-response

```
HTTP/1.1 201 Created
Location: /api/users/1
Content-Type: application/json
Content-Length: 90
```

```
{
  "response" : null,
  "message" : "aaaa aaaaa.",
  "httpCode" : 200
}
```

Table 2. response-fields

Path	Type	Description
httpCode	Number	aa aa
message	String	aa aaa
response	Object	aa aaa

Table 3. response-headers

Name	Description
Location	Location header. aaa aaaa URI

1.2. aa aa aa

http-request

```
POST /api/users/store HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 138
Host: localhost:8080
```

```
{
  "name" : "aaa",
  "email" : "zzanghd@gmail.com",
  "password" : "abc123!",
  "address" : "aaaaaa aa aaa"
}
```

request-fields

```
{
  "name" : "aaa",
  "email" : "zzanghd@gmail.com",
  "password" : "abc123!",
  "address" : "aaaaaa aa aaa"
}
```

http-response

```
HTTP/1.1 201 Created
```

```
Location: /api/users/1
Content-Type: application/json
Content-Length: 90
```

```
{
  "response" : null,
  "message" : "아이디 비밀번호.",
  "httpCode" : 200
}
```

Table 4. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터

Table 5. response-headers

Name	Description
Location	Location header. 아이디 비밀번호 URI

1.3. 요청 데이터

http-request

```
PATCH /api/users/password/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 35
Host: localhost:8080
```

```
{
  "email" : "zzanghd@gmail.com"
}
```

Table 6. request-fields

Path	Type	Description
email	String	이메일

Table 7. /api/users/password/{userId}

Parameter	Description
userId	아이디

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "성공하였습니다.",
  "httpCode" : 200
}
```

Table 8. response-fields

Path	Type	Description
httpCode	Number	상태 코드
message	String	메시지
response	Object	응답 데이터

1.4. GET /api/users/{userId} 요청

http-request

```
GET /api/users/1 HTTP/1.1
Accept: application/json
Host: localhost:8080
```

Table 9. /api/users/{userId}

Parameter	Description
userId	사용자 ID

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 273

{
  "response" : {
    "userId" : 1,
    "createdAt" : "2023-03-14T11:40:17.046479",
    "name" : "홍길동",
    "email" : "zzanghd@gmail.com",
    "address" : "서울특별시 강남구 테헤란로"
  },
  "message" : "성공하였습니다.",
  "httpCode" : 200
}
```

```
}
```

Table 10. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터
response.userIdx	Number	유저 ID
response.email	String	이메일
response.name	String	이름
response.createdAt	String	생성일
response.address	String	주소

1.5. 테스트 방법

http-request

```
GET /api/users HTTP/1.1
Accept: application/json
Host: localhost:8080
```

Table 11. request-parameters

Parameter	Description
page	페이지 번호 (0부터 시작)
size	페이지당 데이터 개수 (1부터 시작)
sort	정렬 방법

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 3252

{
  "response" : {
    "users" : [ {
      "userIdx" : 1,
      "userType" : "일반",
      "name" : "김민준",
      "email" : "test1@gmail.com",
      "createdAt" : "2023-03-14T11:40:17.059039",
      "reviewCnt" : 1,
      "reportCnt" : 1,
```

```

    "userStatus" : "00 00",
    "recoverAt" : "2023-03-21T00:00:00"
  }, {
    "userIdx" : 2,
    "userType" : "00",
    "name" : "0002",
    "email" : "test2@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059118",
    "reviewCnt" : 2,
    "reportCnt" : 2,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-22T00:00:00"
  }, {
    "userIdx" : 3,
    "userType" : "00",
    "name" : "0003",
    "email" : "test3@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.05913",
    "reviewCnt" : 3,
    "reportCnt" : 3,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-23T00:00:00"
  }, {
    "userIdx" : 4,
    "userType" : "00",
    "name" : "0004",
    "email" : "test4@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059138",
    "reviewCnt" : 4,
    "reportCnt" : 4,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-24T00:00:00"
  }, {
    "userIdx" : 5,
    "userType" : "00",
    "name" : "0005",
    "email" : "test5@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059145",
    "reviewCnt" : 5,
    "reportCnt" : 5,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-25T00:00:00"
  }, {
    "userIdx" : 6,
    "userType" : "00",
    "name" : "0006",
    "email" : "test6@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059153",
    "reviewCnt" : 6,
    "reportCnt" : 6,
    "userStatus" : "00 00",

```

```

    "recoverAt" : "2023-03-26T00:00:00"
  }, {
    "userIdx" : 7,
    "userType" : "00",
    "name" : "0007",
    "email" : "test7@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059161",
    "reviewCnt" : 7,
    "reportCnt" : 7,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-27T00:00:00"
  }, {
    "userIdx" : 8,
    "userType" : "00",
    "name" : "0008",
    "email" : "test8@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059167",
    "reviewCnt" : 8,
    "reportCnt" : 8,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-28T00:00:00"
  }, {
    "userIdx" : 9,
    "userType" : "00",
    "name" : "0009",
    "email" : "test9@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059174",
    "reviewCnt" : 9,
    "reportCnt" : 9,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-29T00:00:00"
  }, {
    "userIdx" : 10,
    "userType" : "00",
    "name" : "0010",
    "email" : "test10@gmail.com",
    "createdAt" : "2023-03-14T11:40:17.059181",
    "reviewCnt" : 10,
    "reportCnt" : 10,
    "userStatus" : "00 00",
    "recoverAt" : "2023-03-30T00:00:00"
  } ]
},
"pageInfo" : {
  "size" : 10,
  "page" : 0,
  "totalElement" : 1000,
  "totalPage" : 100,
  "isFirst" : true,
  "isFinish" : false
},

```



```

"message" : "회원 정보 조회.",
"httpCode" : 200
}

```

Table 12. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터
response.users[].userIdx	Number	회원 ID
response.users[].userType	String	회원 타입
response.users[].name	String	회원명
response.users[].email	String	회원 이메일
response.users[].createdAt	String	회원 생성일
response.users[].reviewCnt	Number	회원 리뷰 개수
response.users[].reportCnt	Number	회원 신고 개수
response.users[].userStatus	String	회원 상태
response.users[].recoverAt	String	회원 복구일
pageInfo	Object	페이지 정보
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	전체 데이터 개수
pageInfo.totalPage	Number	전체 페이지 개수
pageInfo.isFirst	Boolean	첫 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

1.6. 회원 정보 조회_회원 정보 조회

http-request

```

GET /api/users/1/review HTTP/1.1
Accept: application/json
Host: localhost:8080

```

Table 13. /api/users/{userIdx}/review

Parameter	Description
userIdx	회원 ID

Table 14. request-parameters

Parameter	Description
page	현재 페이지 0-index
size	페이지 당 개수
sort	정렬

http-response

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 1010

```

```

{
  "response" : {
    "reviews" : [ {
      "reviewIdx" : 2,
      "storeIdx" : 1,
      "storeName" : "1호점",
      "content" : "1호점 리뷰",
      "rating" : 1
    }, {
      "reviewIdx" : 3,
      "storeIdx" : 2,
      "storeName" : "2호점",
      "content" : "2호점 리뷰",
      "rating" : 2
    }, {
      "reviewIdx" : 4,
      "storeIdx" : 3,
      "storeName" : "3호점",
      "content" : "3호점 리뷰",
      "rating" : 3
    }, {
      "reviewIdx" : 5,
      "storeIdx" : 4,
      "storeName" : "4호점",
      "content" : "4호점 리뷰",
      "rating" : 4
    }, {
      "reviewIdx" : 6,
      "storeIdx" : 5,
      "storeName" : "5호점",
      "content" : "5호점 리뷰",
      "rating" : 5
    } ]
  },
  "pageInfo" : {
    "size" : 10,
    "page" : 0,

```

```

    "totalElement" : 1000,
    "totalPage" : 100,
    "isFirst" : true,
    "isFinish" : false
  },
  "message" : "aaaa aaaa.",
  "httpCode" : 200
}

```

Table 15. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 객체
response.reviews[].reviewIdx	Number	리뷰 인덱스
response.reviews[].storeIdx	Number	스토어 인덱스
response.reviews[].storeName	String	스토어명
response.reviews[].content	String	리뷰 내용
response.reviews[].rating	Number	리뷰 평점
pageInfo	Object	페이지 정보 객체
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	전체 데이터 개수
pageInfo.totalPage	Number	전체 페이지 수
pageInfo.isFirst	Boolean	첫 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

1.7. API 호출 방법

http-request

```

GET /api/users/1/store HTTP/1.1
Accept: application/json
Host: localhost:8080

```

Table 16. /api/users/{userIdx}/store

Parameter	Description
userIdx	사용자 인덱스

Table 17. request-parameters

Parameter	Description
page	페이지 번호 0부터 시작
size	페이지 당 개수 1부터 시작
sort	정렬 방법

http-response

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 1074

{
  "response" : {
    "stores" : [ {
      "storeId" : 1,
      "storeName" : "1호점",
      "address" : "서울특별시 강남구 테헤란로 123",
      "tel" : "02-0000-1111"
    }, {
      "storeId" : 2,
      "storeName" : "2호점",
      "address" : "서울특별시 강남구 테헤란로 222",
      "tel" : "02-0000-2222"
    }, {
      "storeId" : 3,
      "storeName" : "3호점",
      "address" : "서울특별시 강남구 테헤란로 333",
      "tel" : "02-0000-3333"
    }, {
      "storeId" : 4,
      "storeName" : "4호점",
      "address" : "서울특별시 강남구 테헤란로 444",
      "tel" : "02-0000-4444"
    }, {
      "storeId" : 5,
      "storeName" : "5호점",
      "address" : "서울특별시 강남구 테헤란로 555",
      "tel" : "02-0000-5555"
    } ]
  },
  "pageInfo" : {
    "size" : 10,
    "page" : 0,
    "totalElement" : 1000,
    "totalPage" : 100,
    "isFirst" : true,
    "isFinish" : false
  },
  "message" : "성공적으로 조회되었습니다."
}
```

```
"httpCode" : 200
}
```

Table 18. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터
response.stores[].storeId	Number	상점 ID
response.stores[].storeName	String	상점명
response.stores[].address	String	상점 주소
response.stores[].tel	String	상점 전화번호
pageInfo	Object	페이지 정보
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	전체 데이터 개수
pageInfo.totalPage	Number	전체 페이지 수
pageInfo.isFirst	Boolean	첫 번째 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

1.8. 사용자 정보 등록 API

http-request

```
PATCH /api/users/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 105
Host: localhost:8080

{
  "name" : "홍길동",
  "address" : "서울특별시 강남구 테헤란로",
  "password" : "abc123!"
}
```

Table 19. /api/users/{userId}

Parameter	Description
userId	사용자 ID

Table 20. request-fields

Path	Type	Description
name	String	名前
address	String	住所
password	String	パスワード

http-response

```
HTTP/1.1 200 OK
Location: /api/users/1
Content-Type: application/json
Content-Length: 90
```

```
{
  "response" : null,
  "message" : "ユーザー 1 作成成功.",
  "httpCode" : 200
}
```

Table 21. response-fields

Path	Type	Description
httpCode	Number	HTTPステータスコード
message	String	メッセージ
response	Object	レスポンスオブジェクト

Table 22. response-headers

Name	Description
Location	Location header. ユーザー 1 の URI

1.9. ユーザー 1 のプロフィール画像をアップロードする

http-request

```
POST /api/users/1/image HTTP/1.1
Content-Type: multipart/form-data; charset=UTF-8;
boundary=6o2knFse3p53ty9dmcQvWAIx1zInP11uCfbm
Host: localhost:8080

--6o2knFse3p53ty9dmcQvWAIx1zInP11uCfbm
Content-Disposition: form-data; name=profileImage; filename=testBusinessCert.png
Content-Type: image/png

test
--6o2knFse3p53ty9dmcQvWAIx1zInP11uCfbm--
```

Table 23. /api/users/{userId}/image

Parameter	Description
userId	사용자 ID

Table 24. request-parts

Part	Description
profileImage	프로필 이미지

http-response

```

HTTP/1.1 200 OK
Location: /api/users/1
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "사용자 정보 조회 성공.",
  "httpCode" : 200
}
```

Table 25. response-fields

Path	Type	Description
httpCode	Number	상태 코드
message	String	메시지
response	Object	응답 데이터

Table 26. response-headers

Name	Description
Location	Location header. 사용자 ID URI

1.10. 사용자 정보 수정 API

http-request

```

PATCH /api/users/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 105
Host: localhost:8080

{
  "name" : "홍길동",
  "address" : "서울특별시 강남구 테헤란로",
}
```

```
"password" : "abc123!"
}
```

Table 27. /api/users/{userId}

Parameter	Description
userId	아이디

Table 28. request-fields

Path	Type	Description
name	String	이름
address	String	주소
password	String	비밀번호

http-response

```
HTTP/1.1 200 OK
Location: /api/users/1
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "회원 등록 성공.",
  "httpCode" : 200
}
```

Table 29. response-fields

Path	Type	Description
httpCode	Number	상태 코드
message	String	메시지
response	Object	응답 객체

Table 30. response-headers

Name	Description
Location	Location header. 리소스 URI

1.11. DELETE

http-request

```
DELETE /api/users/1 HTTP/1.1
Host: localhost:8080
```


Table 31. /api/users/{userId}

Parameter	Description
userId	사용자 ID

http-response

HTTP/1.1 204 No Content

2. ReviewController

2.1. 리뷰 목록 조회

http-request

```
GET /api/store/1/review?page=1&size=10 HTTP/1.1
Accept: application/json
Host: localhost:8080
```

Table 32. /api/store/{storeId}/review

Parameter	Description
storeId	스토어 ID

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 6510

{
  "response" : {
    "storeReview" : [ {
      "reviewIdx" : 1,
      "content" : "리뷰 001",
      "rating" : 4,
      "reviewImage" : "리뷰 이미지 000000.",
      "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
      "createdAt" : "2023-03-14T11:40:15.518506",
      "modifiedAt" : "2023-03-14T11:40:15.518489"
    }, {
      "reviewIdx" : 2,
      "content" : "리뷰 002",
      "rating" : 4,
      "reviewImage" : "리뷰 이미지 000000.",
      "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
      "createdAt" : "2023-03-14T11:40:15.518532",
      "modifiedAt" : "2023-03-14T11:40:15.518527"
```

```

}, {
  "reviewIdx" : 3,
  "content" : "리뷰 003",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.518543",
  "modifiedAt" : "2023-03-14T11:40:15.518539"
}, {
  "reviewIdx" : 4,
  "content" : "리뷰 004",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.518554",
  "modifiedAt" : "2023-03-14T11:40:15.51855"
}, {
  "reviewIdx" : 5,
  "content" : "리뷰 005",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.518564",
  "modifiedAt" : "2023-03-14T11:40:15.51856"
}, {
  "reviewIdx" : 6,
  "content" : "리뷰 006",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.518575",
  "modifiedAt" : "2023-03-14T11:40:15.518571"
}, {
  "reviewIdx" : 7,
  "content" : "리뷰 007",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.51859",
  "modifiedAt" : "2023-03-14T11:40:15.518581"
}, {
  "reviewIdx" : 8,
  "content" : "리뷰 008",
  "rating" : 4,
  "reviewImage" : "리뷰 이미지 000000.",
  "tags" : [ "리뷰", "리뷰 리뷰", "리뷰" ],
  "createdAt" : "2023-03-14T11:40:15.5186",
  "modifiedAt" : "2023-03-14T11:40:15.518597"
}, {
  "reviewIdx" : 9,
  "content" : "리뷰 009",

```

```

    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.51861",
    "modifiedAt" : "2023-03-14T11:40:15.518607"
  }, {
    "reviewIdx" : 10,
    "content" : "00 0010",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518622",
    "modifiedAt" : "2023-03-14T11:40:15.518618"
  }, {
    "reviewIdx" : 11,
    "content" : "00 0011",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518632",
    "modifiedAt" : "2023-03-14T11:40:15.518629"
  }, {
    "reviewIdx" : 12,
    "content" : "00 0012",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518642",
    "modifiedAt" : "2023-03-14T11:40:15.518639"
  }, {
    "reviewIdx" : 13,
    "content" : "00 0013",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518652",
    "modifiedAt" : "2023-03-14T11:40:15.518649"
  }, {
    "reviewIdx" : 14,
    "content" : "00 0014",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518662",
    "modifiedAt" : "2023-03-14T11:40:15.518659"
  }, {
    "reviewIdx" : 15,
    "content" : "00 0015",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],

```

```

    "createdAt" : "2023-03-14T11:40:15.518672",
    "modifiedAt" : "2023-03-14T11:40:15.518669"
  }, {
    "reviewIdx" : 16,
    "content" : "00 0016",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518682",
    "modifiedAt" : "2023-03-14T11:40:15.518679"
  }, {
    "reviewIdx" : 17,
    "content" : "00 0017",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518692",
    "modifiedAt" : "2023-03-14T11:40:15.518689"
  }, {
    "reviewIdx" : 18,
    "content" : "00 0018",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518702",
    "modifiedAt" : "2023-03-14T11:40:15.518699"
  }, {
    "reviewIdx" : 19,
    "content" : "00 0019",
    "rating" : 4,
    "reviewImage" : "00 000 000 00000.",
    "tags" : [ "000", "0000 000", "000" ],
    "createdAt" : "2023-03-14T11:40:15.518713",
    "modifiedAt" : "2023-03-14T11:40:15.518709"
  } ]
},
"pageInfo" : {
  "size" : 10,
  "page" : 0,
  "totalElement" : 1000,
  "totalPage" : 100,
  "isFirst" : true,
  "isFinish" : false
},
"message" : "0000 00000.",
"httpCode" : 200
}

```

Table 33. response-fields

Path	Type	Description
httpCode	Number	상태 코드
message	String	상태 메시지
response	Object	응답 객체
response.storeReview[]	Array	스토어 리뷰 배열
response.storeReview[].reviewIdx	Number	리뷰 ID
response.storeReview[].rating	Number	평점
response.storeReview[].content	String	리뷰 내용
response.storeReview[].reviewImage	String	리뷰 이미지
response.storeReview[].tags	Array	태그 배열
response.storeReview[].createdAt	String	리뷰 생성 시간
response.storeReview[].modifiedAt	String	리뷰 수정 시간
pageInfo	Object	페이지 정보
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	총 데이터 개수
pageInfo.totalPage	Number	총 페이지 수
pageInfo.isFirst	Boolean	첫 번째 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

2.2. 테스트

http-request

```
POST /api/store/1/review HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 225
Host: localhost:8080
```

```
{
  "userIdx" : 1,
  "tags" : [ {
    "tagIdx" : 1
  }, {
    "tagIdx" : 2
  } ],
  "content" : "스토어 리뷰 내용",
  "image" : "스토어 리뷰 이미지",
}
```

```
"rating" : 4
}
```

Table 34. `/api/store/{storeId}/review`

Parameter	Description
<code>storeId</code>	점점 점점 ID

http-response

```
HTTP/1.1 201 Created
Location: /api/store/1/review
Content-Type: application/json
Content-Length: 150

{
  "response" : {
    "storeId" : 1,
    "reviewIdx" : 1,
    "userIdx" : 1
  },
  "message" : "점점점 점점점.",
  "httpCode" : 200
}
```

Table 35. *response-fields*

Path	Type	Description
<code>httpCode</code>	Number	점점 점점
<code>message</code>	String	점점 점점
<code>response</code>	Object	점점 점점
<code>response.storeIdx</code>	Number	점점 점점 ID
<code>response.reviewIdx</code>	Number	점점 점점 ID
<code>response.userIdx</code>	Number	점점 점점 ID

Table 36. *response-headers*

Name	Description
<code>Location</code>	Location header. 점점 점점 URI

2.3. 점점 점점

http-request

```
PATCH /api/store/1/review/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
```

```
Content-Length: 225
Host: localhost:8080
```

```
{
  "userIdx" : 1,
  "tags" : [ {
    "tagIdx" : 1
  }, {
    "tagIdx" : 2
  } ],
  "content" : "이제 정말 재미있는 이야기입니다",
  "image" : "이제 정말 재미있는 이야기.",
  "rating" : 4
}
```

Table 37. `/api/store/{storeId}/review/{reviewIdx}`

Parameter	Description
<code>storeId</code>	스토어 ID
<code>reviewIdx</code>	리뷰 ID

Table 38. `request-fields`

Path	Type	Description
<code>userIdx</code>	Number	사용자 ID
<code>tags</code>	Array	태그
<code>tags[].tagIdx</code>	Number	태그 ID
<code>content</code>	String	내용
<code>image</code>	String	이미지
<code>rating</code>	Number	평점

http-response

```
HTTP/1.1 200 OK
Location: /api/store/1/review
Content-Type: application/json
Content-Length: 150
```

```
{
  "response" : {
    "storeId" : 1,
    "reviewIdx" : 1,
    "userIdx" : 1
  },
  "message" : "리뷰 등록 성공.",
  "httpCode" : 200
}
```

Table 39. response-fields

Path	Type	Description
statusCode	Number	상태 코드
message	String	상태 메시지
response	Object	응답 데이터
response.storeIdx	Number	상품 ID
response.reviewIdx	Number	리뷰 ID
response.userIdx	Number	유저 ID

Table 40. response-headers

Name	Description
Location	Location header. 리다이렉트 URI

2.4. DELETE

http-request

```
DELETE /api/store/1/review/1 HTTP/1.1
Host: localhost:8080
```

Table 41. /api/store/{storeIdx}/review/{reviewIdx}

Parameter	Description
storeIdx	상품 ID
reviewIdx	리뷰 ID

http-response

```
HTTP/1.1 204 No Content
Location: /api/store/1/review
```

Table 42. response-headers

Name	Description
Location	Location header. 리다이렉트 URI

2.5. POST

http-request

```
POST /api/store/1/review/1/like HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
```



```
Content-Length: 19
Host: localhost:8080
```

```
{
  "userIdx" : 1
}
```

Table 43. `/api/store/{storeId}/review/{reviewIdx}/like`

Parameter	Description
<code>storeId</code>	상점 ID
<code>reviewIdx</code>	리뷰 ID

Table 44. `request-fields`

Path	Type	Description
<code>userIdx</code>	Number	사용자 ID

`http-response`

```
HTTP/1.1 200 OK
Location: /api/store/1/review/1
Content-Type: application/json
Content-Length: 169
```

```
{
  "response" : {
    "reviewIdx" : 1,
    "storeId" : 1,
    "userIdx" : 1,
    "like" : true
  },
  "message" : "리뷰 좋아요.",
  "httpCode" : 200
}
```

Table 45. `response-fields`

Path	Type	Description
<code>httpCode</code>	Number	상태 코드
<code>message</code>	String	메시지
<code>response</code>	Object	응답 객체
<code>response.userIdx</code>	Number	사용자 ID
<code>response.storeIdx</code>	Number	상점 ID
<code>response.reviewIdx</code>	Number	리뷰 ID
<code>response.like</code>	Boolean	좋아요 여부

Table 46. response-headers

Name	Description
Location	Location header. 0000 00000 URI

2.6. 00 0000

http-request

```
POST /api/store/1/review/1/hate HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 19
Host: localhost:8080

{
  "userIdx" : 1
}
```

Table 47. /api/store/{storeId}/review/{reviewIdx}/hate

Parameter	Description
storeId	00 000 ID
reviewIdx	00 000 ID

Table 48. request-fields

Path	Type	Description
userIdx	Number	000 00 ID

http-response

```
HTTP/1.1 200 OK
Location: /api/store/1/review/1
Content-Type: application/json
Content-Length: 169

{
  "response" : {
    "reviewIdx" : 1,
    "storeId" : 1,
    "userIdx" : 1,
    "hate" : true
  },
  "message" : "0000 00000.",
  "httpCode" : 200
}
```

Table 49. response-fields

Path	Type	Description
httpCode	Number	HTTP 코드
message	String	응답 메시지
response	Object	응답 객체
response.userIdx	Number	사용자 ID
response.storeIdx	Number	점포 ID
response.reviewIdx	Number	리뷰 ID
response.hate	Boolean	싫어요

Table 50. response-headers

Name	Description
Location	Location header. 리다이렉트 URI

2.7. 테스트

http-request

```
POST /api/store/1/review/1/report HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 80
Host: localhost:8080
```

```
{
  "userIdx" : 1,
  "content" : "점포 1에 리뷰를 남겼습니다."
}
```

Table 51. /api/store/{storeIdx}/review/{reviewIdx}/report

Parameter	Description
storeIdx	점포 ID
reviewIdx	리뷰 ID

Table 52. request-fields

Path	Type	Description
userIdx	Number	사용자 ID
content	String	리뷰 내용

http-response

```
HTTP/1.1 200 OK
```

```

Location: /api/store/1/review/1
Content-Type: application/json
Content-Length: 267

{
  "response" : {
    "storeId" : 1,
    "userId" : 1,
    "reviewIdx" : 1,
    "content" : "맛있고 양도 많네요. ㅎㅎ",
    "reportCreatedAt" : "2023-03-14T11:40:15.479143"
  },
  "message" : "맛있는 음식입니다.",
  "httpCode" : 200
}

```

Table 53. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터
response.userId	Number	사용자 ID
response.storeIdx	Number	점포 ID
response.reviewIdx	Number	리뷰 ID
response.content	String	리뷰 내용
response.image	String	리뷰 이미지
response.reportCreatedAt	String	리뷰 신고 시간

Table 54. response-headers

Name	Description
Location	Location header. 리소스 URI

2.8. 클라이언트 요청

http-request

```

POST /api/store/1/review/1 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 81
Host: localhost:8080

{
  "userId" : 1,

```

```

"content" : "한글 한글 한 한글. 한"
}

```

Table 55. /api/store/{storeId}/review/{reviewIdx}

Parameter	Description
storeId	한글 한글 ID
reviewIdx	한글 한글 ID

Table 56. request-fields

Path	Type	Description
userIdx	Number	한글 한글 ID
content	String	한글 한글

http-response

```

HTTP/1.1 201 Created
Location: /api/store/1/review/1
Content-Type: application/json
Content-Length: 150

{
  "response" : {
    "storeId" : 1,
    "reviewIdx" : 1,
    "userIdx" : 1
  },
  "message" : "한글한글 한글한글.",
  "httpCode" : 200
}

```

Table 57. response-fields

Path	Type	Description
httpCode	Number	한글 한글
message	String	한글 한글
response	Object	한글 한글
response.storeIdx	Number	한글 한글 ID
response.reviewIdx	Number	한글 한글 ID
response.userIdx	Number	한글 한글 한글 ID

Table 58. response-headers

Name	Description
Location	Location header. 한글 한글 URI

3. StoreController

3.1. API 요청 응답

http-request

```
GET /api/store?page=1&size=10 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 64
Host: localhost:8080
```

```
{
  "filterCondition" : [ "주소", "주소 지역" ]
}
```

Table 59. request-fields

Path	Type	Description
filterCondition	Array	필터 조건

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 5652

{
  "response" : {
    "storeHome" : [ {
      "storeId" : 1,
      "name" : "가나다",
      "address" : "서울 강남구 테헤란로 301",
      "longitude" : 203.12312312,
      "latitude" : 111.22323311,
      "tel" : "010-1234-1234",
      "etc" : "가나다",
      "rating" : 4.2,
      "image" :
        "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/20230115501377.jpg",
      "createdAt" : "2023-03-14T11:40:15.962372",
      "modifiedAt" : "2023-03-14T11:40:15.962392",
      "tags" : [ "가나다", "가나다" ]
    }, {
      "storeId" : 2,
      "name" : "가나다",
      "address" : "서울 강남구 테헤란로 301",
      "longitude" : 203.12312312,
```

```

    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "한글 한글.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
    0115501377.jpg",
    "createdAt" : "2023-03-14T11:40:15.962431",
    "modifiedAt" : "2023-03-14T11:40:15.962435",
    "tags" : [ "한글 한글", "한글 한글" ]
  }, {
    "storeId" : 3,
    "name" : "한글한글",
    "address" : "한글 한글 한글 한글 30100",
    "longitude" : 203.12312312,
    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "한글 한글.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
    0115501377.jpg",
    "createdAt" : "2023-03-14T11:40:15.962441",
    "modifiedAt" : "2023-03-14T11:40:15.962444",
    "tags" : [ "한글 한글", "한글 한글" ]
  }, {
    "storeId" : 4,
    "name" : "한글한글",
    "address" : "한글 한글 한글 한글 30100",
    "longitude" : 203.12312312,
    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "한글 한글.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
    0115501377.jpg",
    "createdAt" : "2023-03-14T11:40:15.962451",
    "modifiedAt" : "2023-03-14T11:40:15.962455",
    "tags" : [ "한글 한글", "한글 한글" ]
  }, {
    "storeId" : 5,
    "name" : "한글한글",
    "address" : "한글 한글 한글 한글 30100",
    "longitude" : 203.12312312,
    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "한글 한글.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023

```

```

0115501377.jpg",
  "createdAt" : "2023-03-14T11:40:15.96246",
  "modifiedAt" : "2023-03-14T11:40:15.962464",
  "tags" : [ "코스 코스", "코스 코스" ]
}, {
  "storeId" : 6,
  "name" : "코스코스",
  "address" : "코스 코스 코스 코스 30100",
  "longitude" : 203.12312312,
  "latitude" : 111.22323311,
  "tel" : "010-1234-1234",
  "etc" : "코스 코스코스.",
  "rating" : 4.2,
  "image" :
"https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
0115501377.jpg",
  "createdAt" : "2023-03-14T11:40:15.962468",
  "modifiedAt" : "2023-03-14T11:40:15.962471",
  "tags" : [ "코스 코스", "코스 코스" ]
}, {
  "storeId" : 7,
  "name" : "코스코스",
  "address" : "코스 코스 코스 코스 30100",
  "longitude" : 203.12312312,
  "latitude" : 111.22323311,
  "tel" : "010-1234-1234",
  "etc" : "코스 코스코스.",
  "rating" : 4.2,
  "image" :
"https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
0115501377.jpg",
  "createdAt" : "2023-03-14T11:40:15.962476",
  "modifiedAt" : "2023-03-14T11:40:15.962479",
  "tags" : [ "코스 코스", "코스 코스" ]
}, {
  "storeId" : 8,
  "name" : "코스코스",
  "address" : "코스 코스 코스 코스 30100",
  "longitude" : 203.12312312,
  "latitude" : 111.22323311,
  "tel" : "010-1234-1234",
  "etc" : "코스 코스코스.",
  "rating" : 4.2,
  "image" :
"https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
0115501377.jpg",
  "createdAt" : "2023-03-14T11:40:15.962484",
  "modifiedAt" : "2023-03-14T11:40:15.962487",
  "tags" : [ "코스 코스", "코스 코스" ]
}, {
  "storeId" : 9,

```



```

    "name" : "ㅇㅇㅇㅇㅇㅇ",
    "address" : "ㅇㅇ ㅇㅇㅇ ㅇㅇㅇ ㅇㅇㅇ 301ㅇ",
    "longitude" : 203.12312312,
    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "ㅇㅇㅇ ㅇㅇㅇㅇㅇ.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/2023
    0115501377.jpg",
    "createdAt" : "2023-03-14T11:40:15.962492",
    "modifiedAt" : "2023-03-14T11:40:15.962495",
    "tags" : [ "ㅇㅇㅇ ㅇㅇㅇ", "ㅇㅇㅇㅇ ㅇㅇ" ]
  } ]
},
"pageInfo" : {
  "size" : 10,
  "page" : 0,
  "totalElement" : 1000,
  "totalPage" : 100,
  "isFirst" : true,
  "isFinish" : false
},
"message" : "ㅇㅇㅇㅇ ㅇㅇㅇㅇ.",
"httpCode" : 200
}

```

Table 60. response-fields

Path	Type	Description
httpCode	Number	ㅇㅇ ㅇㅇ
message	String	ㅇㅇ ㅇㅇㅇ
response	Object	ㅇㅇ ㅇㅇㅇ
response.storeHome	Array	ㅇㅇ ㅇㅇ
response.storeHome[].storeIdx	Number	ㅇㅇ ㅇㅇㅇ ID
response.storeHome[].name	String	ㅇㅇ ㅇㅇ
response.storeHome[].address	String	ㅇㅇ ㅇㅇ
response.storeHome[].longitude	Number	ㅇㅇ
response.storeHome[].latitude	Number	ㅇㅇ
response.storeHome[].tel	String	ㅇㅇ ㅇㅇㅇㅇ
response.storeHome[].etc	String	ㅇㅇ
response.storeHome[].rating	Number	ㅇㅇ
response.storeHome[].image	String	ㅇㅇ URL
response.storeHome[].createdAt	String	ㅇㅇ ㅇㅇ

Path	Type	Description
response.storeHome[].modifiedAt	String	날짜
response.storeHome[].tags	Array	태그
pageInfo	Object	페이지 정보
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	전체 개수
pageInfo.totalPage	Number	전체 페이지 수
pageInfo.isFirst	Boolean	첫 번째 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

3.2. API 호출 방법

http-request

```
GET /api/store/1 HTTP/1.1
Accept: application/json
Host: localhost:8080
```

Table 61. /api/store/{storeId}

Parameter	Description
storeId	상점 ID

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 657

{
  "response" : {
    "storeId" : 1,
    "name" : "한빛출판사",
    "address" : "서울특별시 강남구 테헤란로 301",
    "longitude" : 203.12312312,
    "latitude" : 111.22323311,
    "tel" : "010-1234-1234",
    "etc" : "한빛출판사.",
    "rating" : 4.2,
    "image" :
    "https://flexible.img.hani.co.kr/flexible/normal/658/479/imgdb/original/2023/0115/20230115501377.jpg",
    "createdAt" : "2023-03-14T11:40:15.996428",
```

```

    "modifiedAt" : "2023-03-14T11:40:15.996444",
    "tags" : [ "찐찐 찐찐", "찐찐 찐" ]
  },
  "message" : "찐찐 찐찐.",
  "httpCode" : 200
}

```

Table 62. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 객체
response.storeHome	Array	홈 이미지
response.storeIdx	Number	찐 찐 ID
response.name	String	찐 찐
response.address	String	찐 찐
response.longitude	Number	찐
response.latitude	Number	찐
response.tel	String	찐 찐 찐 찐
response.etc	String	찐
response.rating	Number	찐
response.image	String	찐 URL
response.createdAt	String	찐 찐
response.modifiedAt	String	찐 찐
response.tags	Array	찐

3.3. 찐 찐 찐

http-request

```

POST /api/store/1/pick HTTP/1.1
Accept: application/json
Host: localhost:8080

```

Table 63. /api/store/{storeIdx}/pick

Parameter	Description
storeIdx	찐 찐 ID

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 110

{
  "response" : {
    "storeIdx" : 1
  },
  "message" : "aaaaa bbbbbb.",
  "httpCode" : 200
}
```

Table 64. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 객체
response.storeIdx	Number	응답 객체 ID

4. AdminController

4.1. 사용자 목록 조회

http-request

```
POST /api/admin/access/success HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 269
Host: localhost:8080

{
  "users" : [ {
    "userIdx" : 1
  }, {
    "userIdx" : 2
  }, {
    "userIdx" : 3
  }, {
    "userIdx" : 4
  }, {
    "userIdx" : 5
  }, {
    "userIdx" : 6
  }
]
```

```

    }, {
      "userIdx" : 7
    }, {
      "userIdx" : 8
    }, {
      "userIdx" : 9
    }, {
      "userIdx" : 10
    } ]
  }
}

```

Table 65. request-fields

Path	Type	Description
users	Array	목록
users[].userIdx	Number	목록의 인덱스

http-response

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "목록 조회 성공.",
  "httpCode" : 200
}

```

Table 66. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터

4.2. API 호출 방법

http-request

```

POST /api/admin/access/failure HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 269
Host: localhost:8080

{
  "users" : [ {

```

```

    "userIdx" : 1
  }, {
    "userIdx" : 2
  }, {
    "userIdx" : 3
  }, {
    "userIdx" : 4
  }, {
    "userIdx" : 5
  }, {
    "userIdx" : 6
  }, {
    "userIdx" : 7
  }, {
    "userIdx" : 8
  }, {
    "userIdx" : 9
  }, {
    "userIdx" : 10
  } ]
}

```

Table 67. request-fields

Path	Type	Description
<code>users</code>	Array	사용자 목록
<code>users[].userIdx</code>	Number	사용자 인덱스

http-response

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "사용자 목록 조회 성공.",
  "httpCode" : 200
}

```

Table 68. response-fields

Path	Type	Description
<code>httpCode</code>	Number	응답 코드
<code>message</code>	String	응답 메시지
<code>response</code>	Object	응답 데이터

4.3. □□ □□

http-request

```
POST /api/admin/block?period=3 HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 269
Host: localhost:8080
```

```
{
  "users" : [ {
    "userIdx" : 1
  }, {
    "userIdx" : 2
  }, {
    "userIdx" : 3
  }, {
    "userIdx" : 4
  }, {
    "userIdx" : 5
  }, {
    "userIdx" : 6
  }, {
    "userIdx" : 7
  }, {
    "userIdx" : 8
  }, {
    "userIdx" : 9
  }, {
    "userIdx" : 10
  } ]
}
```

Table 69. request-fields

Path	Type	Description
<code>users</code>	Array	□□□□
<code>users[].userIdx</code>	Number	□□□□□

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 90
```

```
{
  "response" : null,
  "message" : "□□□□ □□□□.",
}
```

```
"httpCode" : 200
}
```

Table 70. response-fields

Path	Type	Description
httpCode	Number	□□ □□
message	String	□□ □□□
response	Object	□□ □□□

4.4. □□ □□

http-request

```
POST /api/admin/fired HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 269
Host: localhost:8080
```

```
{
  "users" : [ {
    "userIdx" : 1
  }, {
    "userIdx" : 2
  }, {
    "userIdx" : 3
  }, {
    "userIdx" : 4
  }, {
    "userIdx" : 5
  }, {
    "userIdx" : 6
  }, {
    "userIdx" : 7
  }, {
    "userIdx" : 8
  }, {
    "userIdx" : 9
  }, {
    "userIdx" : 10
  } ]
}
```

Table 71. request-fields

Path	Type	Description
users	Array	□□□□

Path	Type	Description
<code>users[].userIdx</code>	Number	□□□□

http-response

HTTP/1.1 204 No Content

5. ReviewAdminController

5.1. □□□ □□ □□

http-request

GET /api/admin/reports?page=1&size=10 HTTP/1.1
 Accept: application/json
 Host: localhost:8080

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 2635

{
  "response" : {
    "reportedReview" : [ {
      "reviewIdx" : 1,
      "reviewContent" : "□□ □□ 1",
      "reportedCount" : 1,
      "reviewCreatedAt" : "2023-03-14T11:40:14.6862"
    }, {
      "reviewIdx" : 2,
      "reviewContent" : "□□ □□ 2",
      "reportedCount" : 2,
      "reviewCreatedAt" : "2023-03-14T11:40:14.686294"
    }, {
      "reviewIdx" : 3,
      "reviewContent" : "□□ □□ 3",
      "reportedCount" : 3,
      "reviewCreatedAt" : "2023-03-14T11:40:14.686307"
    }, {
      "reviewIdx" : 4,
      "reviewContent" : "□□ □□ 4",
      "reportedCount" : 4,
      "reviewCreatedAt" : "2023-03-14T11:40:14.686314"
    }, {
      "reviewIdx" : 5,
```

```

    "reviewContent" : "00 00 5",
    "reportedCount" : 5,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686321"
  }, {
    "reviewIdx" : 6,
    "reviewContent" : "00 00 6",
    "reportedCount" : 6,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686333"
  }, {
    "reviewIdx" : 7,
    "reviewContent" : "00 00 7",
    "reportedCount" : 7,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686339"
  }, {
    "reviewIdx" : 8,
    "reviewContent" : "00 00 8",
    "reportedCount" : 8,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686346"
  }, {
    "reviewIdx" : 9,
    "reviewContent" : "00 00 9",
    "reportedCount" : 9,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686353"
  }, {
    "reviewIdx" : 10,
    "reviewContent" : "00 00 10",
    "reportedCount" : 10,
    "reviewCreatedAt" : "2023-03-14T11:40:14.68636"
  }, {
    "reviewIdx" : 11,
    "reviewContent" : "00 00 11",
    "reportedCount" : 11,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686367"
  }, {
    "reviewIdx" : 12,
    "reviewContent" : "00 00 12",
    "reportedCount" : 12,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686374"
  }, {
    "reviewIdx" : 13,
    "reviewContent" : "00 00 13",
    "reportedCount" : 13,
    "reviewCreatedAt" : "2023-03-14T11:40:14.686381"
  }, {
    "reviewIdx" : 14,
    "reviewContent" : "00 00 14",
    "reportedCount" : 14,
    "reviewCreatedAt" : "2023-03-14T11:40:14.68641"
  }, {
    "reviewIdx" : 15,
    "reviewContent" : "00 00 15",

```

```

        "reportedCount" : 15,
        "reviewCreatedAt" : "2023-03-14T11:40:14.68643"
    } ]
},
"pageInfo" : {
    "size" : 10,
    "page" : 0,
    "totalElement" : 1000,
    "totalPage" : 100,
    "isFirst" : true,
    "isFinish" : false
},
"message" : "aaaaa aaaaaa.",
"httpCode" : 200
}

```

Table 72. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터
response.reportedReview[].	Array	리뷰
response.reportedReview[].reviewIdx	Number	리뷰 ID
response.reportedReview[].reviewContent	String	리뷰 내용
response.reportedReview[].reportedCount	Number	리뷰 수
response.reportedReview[].reviewCreatedAt	String	리뷰 생성 시간
pageInfo	Object	페이지 정보
pageInfo.page	Number	현재 페이지
pageInfo.size	Number	페이지 크기
pageInfo.totalElement	Number	전체 데이터 수
pageInfo.totalPage	Number	전체 페이지 수
pageInfo.isFirst	Boolean	첫 페이지 여부
pageInfo.isFinish	Boolean	마지막 페이지 여부

5.2. 관리자 권한 삭제

http-request

```
DELETE /api/admin/banned HTTP/1.1
```

```
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 291
Host: localhost:8080
```

```
{
  "reviews" : [ {
    "reviewIdx" : 1
  }, {
    "reviewIdx" : 2
  }, {
    "reviewIdx" : 3
  }, {
    "reviewIdx" : 4
  }, {
    "reviewIdx" : 5
  }, {
    "reviewIdx" : 6
  }, {
    "reviewIdx" : 7
  }, {
    "reviewIdx" : 8
  }, {
    "reviewIdx" : 9
  }, {
    "reviewIdx" : 10
  } ]
}
```

Table 73. request-fields

Path	Type	Description
reviews	Array	리뷰의 목록
reviews[].reviewIdx	Number	리뷰의 인덱스

http-response

```
HTTP/1.1 204 No Content
```

5.3. 리뷰 삭제

http-request

```
POST /api/admin/banned HTTP/1.1
Content-Type: application/json;charset=UTF-8
Accept: application/json
Content-Length: 291
Host: localhost:8080
```

```
{
  "reviews" : [ {
    "reviewIdx" : 1
  }, {
    "reviewIdx" : 2
  }, {
    "reviewIdx" : 3
  }, {
    "reviewIdx" : 4
  }, {
    "reviewIdx" : 5
  }, {
    "reviewIdx" : 6
  }, {
    "reviewIdx" : 7
  }, {
    "reviewIdx" : 8
  }, {
    "reviewIdx" : 9
  }, {
    "reviewIdx" : 10
  } ]
}
```

Table 74. request-fields

Path	Type	Description
reviews	Array	리뷰의 목록
reviews[].reviewIdx	Number	리뷰의 인덱스

http-response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 90

{
  "response" : null,
  "message" : "리뷰가 등록되었습니다.",
  "httpCode" : 200
}
```

Table 75. response-fields

Path	Type	Description
httpCode	Number	응답 코드
message	String	응답 메시지
response	Object	응답 데이터