

API 명세서

<div>≡</div> 구분	Aa 기능	<div>⌵</div> METHOD	<div>≡</div> 주소	<div>≡</div> request	<div>≡</div> response	<div>≡</div> 오류
사용 자 관리	<u>회원</u> <u>가입</u>	POST	/api/signup	{ "email": "string", "nickname": "string", "password": "string" }	{ "success": true, "status": 201, "message": "회원가입 성공", "data": null, "timestamp": "2025-01-16T18:00:00", "path": "/api/signup" }	{ "success": false, "status": 409, "message": "이미 사 용 중인 이메일입니다.", "timestamp": "2025-01-16T18:00:00", "path": "/api/signup" }
사용 자 관리	<u>로그</u> <u>인</u>	POST	/api/user/login	{ "email": "string", "password": "string" }	{ "success": true, "status": 200, "message": "로그인 성공", "data": { "nickname": "string" }, "timestamp": "2025-01-16T18:00:00", "path": "/api/user/login" }	{ "success": false, "status": 401, "message": "이메일 또는 비밀번호가 잘못되 었습니다.", "timestamp": "2025-01-16T18:00:00", "path": "/api/user/login" }
방 관리	<u>방</u> <u>생성</u>	POST	/api/room/create	{ "roomName": "string", "maxCapacity": 5, "password": "string" }	{ "success": true, "status": 201, "message": "방이 생성되었습니다.", "data": { "roomId": 1, "roomName": "string", "roomStatus": "waiting" }, "timestamp": "2025-01-16T18:00:00", "path": "/api/room/create" }	
방 관리	<u>방</u> <u>입장</u>	POST	/api/room/join	{ "roomId": 1, "password": "string" }	{ "success": true, "status": 200, "message": "방에 입장했습니다.", "data": { "roomId": 1, "roomName": "string", "currentCapacity": 3 }, "timestamp": "2025-01-16T18:00:00", "path": "/api/room/join" }	{ "success": false, "status": 403, "message": "방 비밀 번호가 잘못되었습니 다.", "timestamp": "2025-01-16T18:00:00", "path": "/api/room/join" }
게임 관리	<u>카드</u> <u>선택</u>	POST	/api/game/card	{ "gameId": 1, "cardId": 10, "playerId": 5 }	{ "success": true, "status": 200, "message": "카드가 선택되었습니다.", "data": { "selectedCard": "키워드: 바다" }, "timestamp": "2025-01-16T18:00:00", }	

구분	Aa 기능	METHOD	주소	request	response	오류
					<pre>"path": "/api/game/card" }</pre>	
게임 관리	<u>이야 기 생성</u>	POST	<code>/api/game/story</code>	<pre>{ "gameId": 1, "playerId": 5, "story": "옛날 옛날에 바다를 건너는 모험이 시작되었습니다." }</pre>	<pre>{ "success": true, "status": 200, "message": "이야기가 생성되었습니다.", "data": { "imageUrl": "http://example.com/generated_image.jpg", "currentStory": "옛날 옛날에 바다를 건너는 모험 이 시작되었습니다." }, "timestamp": "2025-01-16T18:00:00", "path": "/api/game/story" }</pre>	
책 관리	<u>책 목록 조회</u>	GET	<code>/api/book/list</code>	<pre>{ "success": true, "status": 200, "message": "책 목록 을 가져왔습니다.", "data": [{ "bookId": 1, "title": "바다 모험 이야 기", "likeCount": 15 }, { "bookId": 2, "title": "산속의 비밀", "likeCount": 12 }], "timestamp": "2025-01-16T18:00:00", "path": "/api/book/list" }</pre>		
책 관리	<u>책 좋아 요</u>	POST	<code>/api/book/like</code>	<pre>{ "bookId": 1, "userId": 5 }</pre>	<pre>{ "success": true, "status": 200, "message": "좋아요가 반영되었습니다.", "data": { "bookId": 1, "likeCount": 16 }, "timestamp": "2025-01-16T18:00:00", "path": "/api/book/like" }</pre>	

1. 사용자 관리 (User Management)

1.1 회원가입

- **URL:** `/api/signup`
- **Method:** POST
- **Request Body:**

```
{
  "email": "string",
  "nickname": "string",
  "password": "string"
}
```

- **Response:**

- **201 Created** (성공):

```
{
  "success": true,
  "status": 201,
  "message": "회원가입 성공",
  "data": null,
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/signup"
}
```

- **409 Conflict** (중복된 이메일/닉네임):

```
{
  "success": false,
  "status": 409,
  "message": "이미 사용 중인 이메일입니다.",
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/signup"
}
```

1.2 로그인

- **URL:** `/api/user/login`
- **Method:** POST
- **Request Body:**

```
{
  "email": "string",
  "password": "string"
}
```

- **Response:**

- **200 OK** (성공):

```
{
  "success": true,
  "status": 200,
  "message": "로그인 성공",
}
```

```
"data": {
  "nickname": "string"
},
"timestamp": "2025-01-16T18:00:00",
"path": "/api/user/login"
}
```

- **401 Unauthorized** (실패):

```
{
  "success": false,
  "status": 401,
  "message": "이메일 또는 비밀번호가 잘못되었습니다.",
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/user/login"
}
```

2. 방 관리 (Room Management)

2.1 방 생성

- **URL:** `/api/room/create`
- **Method:** POST
- **Request Body:**

```
{
  "roomName": "string",
  "maxCapacity": 5,
  "password": "string"
}
```

- **Response:**
 - **201 Created:**

```
{
  "success": true,
  "status": 201,
  "message": "방이 생성되었습니다.",
  "data": {
    "roomId": 1,
    "roomName": "string",
    "roomStatus": "waiting"
  },
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/room/create"
}
```

2.2 방 입장

- **URL:** `/api/room/join`
- **Method:** POST
- **Request Body:**

```
{
  "roomId": 1,
  "password": "string"
}
```

- **Response:**
 - **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "방에 입장했습니다.",
  "data": {
    "roomId": 1,
    "roomName": "string",
    "currentCapacity": 3
  },
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/room/join"
}
```

- **403 Forbidden:**

```
{
  "success": false,
  "status": 403,
  "message": "방 비밀번호가 잘못되었습니다.",
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/room/join"
}
```

3. 게임 관리 (Game Management)

3.1 카드 선택

- **URL:** `/api/game/card`
- **Method:** POST
- **Request Body:**

```
{
  "gameId": 1,
  "cardId": 10,
  "playerId": 5
}
```

```
}
```

- **Response:**
 - **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "카드가 선택되었습니다.",
  "data": {
    "selectedCard": "키워드: 바다"
  },
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/game/card"
}
```

3.2 이야기 생성

- **URL:** `/api/game/story`
- **Method:** POST
- **Request Body:**

```
{
  "gameId": 1,
  "playerId": 5,
  "story": "옛날 옛날에 바다를 건너는 모험이 시작되었습니다."
}
```

- **Response:**
 - **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "이야기가 생성되었습니다.",
  "data": {
    "imageUrl": "http://example.com/generated_image.jpg",
    "currentStory": "옛날 옛날에 바다를 건너는 모험이 시작되었습니다."
  },
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/game/story"
}
```

4. 책 관리 (Book Management)

4.1 책 목록 조회

- **URL:** `/api/book/list`

- **Method:** GET
- **Response:**
 - **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "책 목록을 가져왔습니다.",
  "data": [
    {
      "bookId": 1,
      "title": "바다 모험 이야기",
      "likeCount": 15
    },
    {
      "bookId": 2,
      "title": "산속의 비밀",
      "likeCount": 12
    }
  ],
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/book/list"
}
```

4.2 책 좋아요

- **URL:** `/api/book/like`
- **Method:** POST
- **Request Body:**

```
{
  "bookId": 1,
  "userId": 5
}
```

- **Response:**
 - **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "좋아요가 반영되었습니다.",
  "data": {
    "bookId": 1,
    "likeCount": 16
  },
  "timestamp": "2025-01-16T18:00:00",
  "path": "/api/book/like"
}
```

