**Shared Lecture Note 프로토콜 정의서**

B011004 경도준

B011224 최병광

최종 수정일 2016. 6. 1.

목차

1. [회원가입](#_1._회원가입_1)
2. [로그인](#_2._로그인)
3. [방 목록 요청](#_3._방목록_요청)
4. [방 검색](#_8._방_검색)
5. [방 생성 요청](#_4._방_생성)
6. [방 입장](#_5._방_입장)
7. [그리기 정보 전송 및 수신](#_5._그리기_정보)
8. [방 나가기](#_9._방_나가기)

|  |
| --- |
| **1. 회원가입** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 사용자 회원가입 | **인증** | X | | **요청 URL** | /register | **메소드** | POST | | **매개변수** | ID : 사용자 ID (필수)  PW : 로그인 비밀번호 (필수) | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “register”,  “result” : “ok”  } | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “register”,  “result” : “실패이유”  } | | | | | |

|  |
| --- |
| **2. 로그인** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 사용자 회원가입 | **인증** | X | | **요청 URL** | /login | **메소드** | POST | | **매개변수** | ID : 사용자 ID (필수)  PW : 로그인 비밀번호 (필수) | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “login”,  “result” : “ok”  } | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “login”,  “result” : “실패이유”  } | | | | | |

|  |
| --- |
| **3. 방목록 요청** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 현재 개설된 방목록을 요청 | **인증** | X | | **요청 URL** | /roomlist | **메소드** | GET | | **매개변수** |  | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “roomlist”,  “result” : [  {  “roomNum” : “1”,  “masterId” : “abdc”  },  {  …  },  …  ]  현재 개설된 방의 목록을 전달하는데 방 번호와 마스터의 ID를 JASON객체로 보낸다.  방은 최대 5개라고 가정한다. | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “roomlist”,  “result” : “실패이유”  } | | | | | |

|  |
| --- |
| **4. 방 검색** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 그림 정보 전송 | **인증** | X | | **요청 URL** | /search | **메소드** | POST | | **매개변수** | searchId : 찾고자 하는 masterid | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “search”,  “result” : {  “roomNum” : “1”,  “masterId : “master1”  } | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “search”,  “result” : “실패이유”  } | | | | | |
| **5. 방 생성 요청** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 방 개설을 요청 | **인증** | X | | **Event** | makeRoom | **메소드** | Socket | | **매개변수** | ID : 사용자 ID | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  "success" : 1,  "work" : "makeroom",  "result" : 생성된 방번호  }; | | | | | |
| **6. 방 입장** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 방 입장시 해당 방의 정보 전송 | **인증** | X | | **Event** | enterRoom | **메소드** | Socket | | **매개변수** | ID : 사용자 ID  num : roomnum | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  "success" : 1,  "userList" : [“user1”, “user2”, “user3”, …],  "result" : [{  “x” : 1241.123,  “y” : 2312.123,  “str” : “Shared Lecture Note”,  ….  },  {  “x” : 1241.123,  “y” : 2312.123,  “str” : “Shared Lecture Note”,  ….  },  ….  ….  }]  DB에 저장된 해당 방의 현재 그림정보와 현재 입력된 그리기 정보를 전달한다.  그리기 객체는 정보가 다양하므로 해당 정보를 모두 배열로 만들어 전달하도록 한다.  여기서 JASON객체를 배열의 원소로 한다. | | | | | |

|  |
| --- |
| **7. 그리기 정보 전송 및 수신** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 그리기 정보 전달 | **인증** | X | | **요청 URL** | /room | **메소드** | POST | | **매개변수** | ID : 사용자 ID  num : roomnum  penStyle : pen정보(색연필, 형광펜 등)  penSize : 펜의 굵기  x : x좌표  y : y좌표  RGB : RGB색상값  string : 문자열이 있을 경우 문자열  strType : 문자열의 특징(굵게, 기울기, 밑줄 등)  strSzie 문자열의 크기 | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “room”,  “result” : [  {  “penStlye” : “1”  “penSize” : “1.0”,  “x” : “15”,  “y” : “60”,  “RGB” : “FF9A44”  },  {  …  },  …  ]  서버에서 DB에 저장된 해당 방의 그리기 정보를 클라이언트로 전송하거나  클라이언트에서 현재 입력한 그리기 정보를 서버로 전송한다..  그리기 객체는 정보가 다양하므로 해당 정보를 모두 배열로 만들어 전달하도록 한다.  여기서는 그림 정보가 아닌 그리기 정보만을 전달한다.  마찬가지로 JASON객체가 배열의 원소가 된다. | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “room”,  “result” : “실패이유”  } | | | | | |

|  |
| --- |
| **8. 방 나가기** |
| |  |  |  |  | | --- | --- | --- | --- | | **기능** | 그림 정보 전송 | **인증** | X | | **요청 URL** | /exitRoom | **메소드** | POST | | **매개변수** | ID : 사용자 ID  num : roomnum | | | | **응답메시지** | | | | | | 성공시 | | | | | | {  “success” : 1,  “work” : “exitRoom”,  “result” : {  “roomNum” : “1”,  “action” : “master가 방을 나갔습니다.”  }  방을 나가는 경우는 master인 경우와 아닌 경우 두가지로 나뉘는데  master인 경우 방을 나가면 해당 방의 참가자 모두에게 위와 같은 응답을 보내며  방을 나가고 RoomList Activity로 이동하게 된다.  master가 아닌 경우에는 방을 나간 사용자만 RoomList Activity로 이동한다.  동시에 해당 방에 참여한 정보도 Database에서 삭제한다. | | | | | | 실패시 | | | | | | {  “success” : 0,  “work” : “exitRoom”,  “result” : “실패이유”  } | | | | | |