



API 명세서

📶 통신 방향	📡 통신 프로토콜	📋 관련 기능	📋 세부 기능	📄 Description	📄 Requet	📄 Response	Aa 기타 관련 사항
Pi → Backend	HTTP	웹소켓 연결		Spring의 WebSocket에 연결	[Headers] GET /ws/device HTTP/1.1 Host: <백엔드주소:8080> Upgrade: websocket Connection: Upgrade Sec-WebSocket-Key: <16바이트 숫자를 base64로 인코딩> Sec-WebSocket-Version: 13 Authorization: Bearer <serial을 SHA 256 암호화>	200, Ok (시리얼이 DB에 있음) HTTP/1.1 101 Switching Protocols Upgrade: websocket Connection: Upgrade Sec-WebSocket-Accept: <Base64-encoded value> 403, Forbidden	제목 없음
Pi → Backend	WebSocket	디바이스 정보	디바이스 ID 요청		[destination] "/app/DeviceInfo/Id" { "token": "시리얼 jwt" }	[구독 경로] "/topic/DeviceInfo/Id" + 시리얼 { " id " : "기기 id" }	제목 없음
Pi → Backend	WebSocket	캡슐 정보 요청			[destination] "/app/DeviceStatus/Capsule/Info" { "token": "id jwt" }	[구독 경로] "/topic/DeviceStatus/Capsule/Info" + deviceId { " slot1 ":"캡슐숫자", " slot2 ":"캡슐숫자", " slot3 ":"캡슐숫자", " slot4 ":"캡슐숫자" }	제목 없음
Pi → Backend	WebSocket	모니터링	온습도 센서 결과 전송	15분에 한번씩 온습도 센서 결과를 전송	[destination] "/app/DeviceStatus/Sensor/TempHum" { "token": "id jwt", " temperature ": 22.5 (float), " humidity ": 60 (int) }		제목 없음
Pi → Backend	WebSocket	모니터링	디퓨저 잔여량 전송	스케줄에 맞춰서 혹은 자동화 모드에 따라 분사한 뒤 잔여량 서버에 전송	[destination] "/app/DeviceStatus/Capsule/Remainder" { "token": "아이디 jwt", " slot1RemainingRatio " : "90", " slot2RemainingRatio " : "100", " slot3RemainingRatio " : "80", "		제목 없음

📶 통신 방향	⚙ 통신 프로토콜	≡ 관련 기능	≡ 세부 기능	≡ Description	≡ Request	≡ Response	Aa 기타 관련 사항
					<code>slot4RemainingRatio</code> " : "20"		
Pi → Backend	WebSocket	모드 정보	자동화 혹은 스케줄 정보를 얻어옴	자동화 혹은 스케줄 모드 정보 요청	[destination] "/app/Mode" { "token": "아이디 jwt" }	[구독 경로] "/topic/Mode" + 디바이스 id 스케줄 { " <code>mode</code> " : "0" } 자동화 { " <code>mode</code> " : "1" }	제목 없음
Pi → Backend	WebSocket	스케줄 모드	당일 예약 정보를 요청	모드를 확인하였을 때 예약 모드면 실행	[destination] "/app/Schedule/Initial" { "token": "아이디 jwt" }	[구독 경로] "/topic/Schedule/Initial" + deviceId { "schedules": [{ " <code>id</code> " : "스케줄id" " <code>deviceId</code> " : "기기 id" "combination" : { " <code>choice1</code> " : 1, " <code>choice1Count</code> " : 3, " <code>choice2</code> " : 2, " <code>choice2Count</code> " : 2, " <code>choice3</code> " : null, " <code>choice3Count</code> " : null, " <code>choice4</code> " : null, " <code>choice4Count</code> " : null }, " <code>startTime</code> " : "09:00:00", " <code>endTime</code> " : "18:00:00", " <code>interval</code> " : 15 " <code>modeOn</code> " : true }] }	제목 없음
Pi → Backend	WebSocket	자동화 모드	설정된 자동화 정보 요청	모드를 확인하였을 때 자동화 모드면 실행	[destination] "/app/Auto/Schedule/Initial" { "token": "아이디 jwt" }	[구독 경로] "/topic/Auto/Schedule/Initial" + device Id { "schedules" : [{ " <code>id</code> " : 2, " <code>combinationId</code> " : 1,	제목 없음

🕒 통신 방향	⚙ 통신 프로토콜	≡ 관련 기능	≡ 세부 기능	≡ Description	≡ Request	≡ Response	Aa 기타 관련 사항
						<pre>" subMode " : 0, " type " : null, " interval ": 15 " modeOn " : true }] }</pre>	
Pi → Backend	WebSocket	자동화 모드	자동화 모드 항기 정보 요청		<p>[destination] "/app/DeviceStatus/Sensor"</p> <pre>{ "token": "시리얼 번호 jwt", " combinationId " : int }</pre>	<p>[구독 경로] "/topic/Auto/Operation" + 기기id</p> <pre>{ "combination" : { " choice1 " : 1, " choice1Count ": 3, " choice2 " : 2, " choice2Count ": 2, " choice3 " : null, " choice3Count ": null, " choice4 " : null, " choice4Count ": null, } }</pre>	제목 없음
Backend → Pi	WebSocket	모드 변경	모드가 변경되었을 때 정보가 전송	스케줄 모드로 모드를 변경할 때, 현재로부터 이후 예약 정보를 받아와야 함		<p>[구독 경로] "/topic/Mode/Change" + deviceId</p> <p>스케줄</p> <pre>{ " mode " : "1" }</pre> <p>자동화</p> <pre>{ " mode " : "0" }</pre>	제목 없음
Backend → Pi	WebSocket	스케줄 모드	데일리 스케줄 제공	매일 자정에 하루에 예약된 스케줄을 모두 수신		<p>[구독 경로] "/topic/Schedule/Initial" + deviceId</p> <pre>{ "schedules": [{ " id " : "스케줄id" " deviceId " : "기기 id" "combination" : { " choice1 " : 1, " choice1Count ": 3, " choice2 " : 2, " } }] }</pre>	제목 없음

🕒 통신 방향	⚙ 통신 프로토콜	≡ 관련 기능	≡ 세부 기능	≡ Description	≡ Requet	≡ Response	Aa 기타 관련 사항
						<pre>choice2Count ": 2, " choice3 ": null, " choice3Count ": null, " choice4 ": null, " choice4Count ": null }, " startTime ": "09:00:00", " endTime ": "18:00:00", " interval ": 15 " modeOn " : true }] }</pre>	
Backend → Pi	WebSocket	스케줄 모드	스케줄 추가	당일에 해당하는 스케줄이 추가되었을 때 전송		<p>[구독 경로] "/topic/Schedule/Add" + deviceId</p> <pre>{ "schedules": [{ " id " : "스케줄id" " deviceId " : "기기 id" "combination" : { " choice1 ": 1, " choice1Count ": 3, " choice2 ": 2, " choice2Count ": 2, " choice3 ": null, " choice3Count ": null, " choice4 ": null, " choice4Count ": null }, " startTime ": "09:00:00", " endTime ": "18:00:00", " interval ": 15 " modeOn " : true }] }</pre>	제목 없음
Backend → Pi	WebSocket	스케줄 모드	스케줄 변경	1. 당일에 설정된 스케줄 설정을 변경 2. 다른 날짜		<p>[구독 경로] "/topic/Schedule/Change/" + id</p> <pre>{ "schedule": {</pre>	제목 없음

🕒 통신 방향	⚙ 통신 프로토콜	≡ 관련 기능	≡ 세부 기능	≡ Description	≡ Requet	≡ Response	Aa 기타 관련 사항
				의 스케줄을 당일로 변경		<pre>" id " : "수정 스케줄id" "combination" : { " choice1 ": 1, " choice1Count ": 3, " choice2 ": 2, " choice2Count ": 2, " choice3 ": null, " choice3Count ": null, " choice4 ": null, " choice4Count ": null, }, " startTime ": "10:00:00", " endTime ": "18:00:00", " interval ": 15 " modeOn " : true }</pre>	
Backend → Pi	WebSocket	스케줄 모드	스케줄 삭제	당일에 설정된 스케줄을 삭제		[구독 경로] "/topic/Schedule/Delete" + deviceId { "scheduleId" : "삭제 스케줄id" }	<u>삭제할 id가 없을 때 status도 추가 고려</u>
Backend → Pi	WebSocket	자동화 모드	항기 조합 수정			[구독 경로] "/topic/Combination/Change" + 기기id { " id " : 스케줄 id (int), " combinationId " : int }	<u>제목 없음</u>
Backend → Pi	WebSocket	자동화 모드	인터벌 정보 수정			[구독 경로] "/topic/Interval/Change" + 기기id { " id " : 스케줄 id (int), " interval " : int }	<u>제목 없음</u>
Backend → Pi	WebSocket	자동화 모드	세부 모드 정보			[구독 경로] "/topic/Auto/Mode/Change" + 기기id { " "	<u>제목 없음</u>

📍 통신 방향	⚙ 통신 프로토콜	≡ 관련 기능	≡ 세부 기능 수정	≡ Description	≡ Request	≡ Response	Aa 기타 관련 사항
						<code>id</code> : 스케줄 id, "modeOn" : true/ false, }	
Backend → Pi	WebSocket	즉시 분사	즉시 분사	사용자가 웹 페이지에서 즉시 분사 버튼을 누르면 그 시각에 바로 디폴트 향 조합으로 분사한다.		[구독 경로] "/topic/Remote/Operation" + 기기 id { " <code>id</code> " : combination id " <code>name</code> ": 시원한 향, " <code>choice1</code> ": 1, " <code>choice1Count</code> ": 3, " <code>choice2</code> ": 2, " <code>choice2Count</code> ": 2, " <code>choice3</code> ": null, " <code>choice3Count</code> ": null, " <code>choice4</code> ": null, " <code>choice4Count</code> ": null }	제목 없음
Backend → Pi	WebSocket	기기 삭제	웹소켓 통신 종료			[구독 경로] "/topic/Connection/Close/" + deviceId { " Close" }	제목 없음
Pi → Jetson	MQTT	분사 동작	분사를 위한 모터 제어	분사를 위해서 각 모터의 제어 여부 그리고 횟수를 Jetson으로 넘겨준다	topic : "{device_id}/Operation", payload : { " <code>choice1</code> ": 1, " <code>choice1Count</code> ": 3, " <code>choice2</code> ": 2, " <code>choice2Count</code> ": 2, " <code>choice3</code> ": null, " <code>choice3Count</code> ": null, " <code>choice4</code> ": null, " <code>choice4Count</code> ": null }		제목 없음

📍 통신 방향	📡 통신 프로토콜	📖 관련 기능	📖 세부 기능	📖 Description	📖 Request	📖 Response	Aa 기타 관련 사항
Pi → Jetson	MQTT	동작 모드 설정		기기가 자동화 모드로 동작할 것인지, 스케줄 모드로 동작할 것인지 모드 정보를 제공	topic : "{device_id}/SetOperationMode" 스케줄 { " mode " : "0" } 자동화 { " mode " : "1" }		제목 없음
Pi → Jetson	MQTT	디퓨저 정보	캡슐 정보 제공	클라이언트에서 설정한 디퓨저 슬롯 - 캡슐 맵핑 정보 제공	topic : "{device_id}/CapsuleInfo", payload : { " slot1 ":"캡슐숫자", " slot2 ":"캡슐숫자", " slot3 ":"캡슐숫자", " slot4 ":"캡슐숫자" }		제목 없음
Pi → Jetson	MQTT	디퓨저 설정	자동화 모드 설정 내 보내기	현재 디퓨저에 설정된 자동화 모드 관련 정보들을 제공	topic : "{device_id}/AutoModelnit", payload : { " automative_active " : "", " simple_detection " : "", " stink_detection " : "", " action_detection " : "", }		아마 Backend에서 처리할 듯?
Pi → Jetson	MQTT	디퓨저 설정	자동화 모드 설정 변경하기	현재 디퓨저에 설정된 자동화 모드 정보들을 변경	topic: "{device_id}/AutoModeChange" { "schedules" : [{ " id " : 2, " combinationId " : 1, " subMode " : 0, " type " : null, " interval ": 15 " modeOn " : true }] }		제목 없음
Pi → Jetson	MQTT	연결 상태 관리	웹소켓 종료 정보 제공		topic :"{device_id}/Websocket/Disconnect" payload : 0		제목 없음
Jetson → Pi	MQTT	디퓨저	디퓨저		topic :"{device_id}/Request/AutoModelInfo", payload : 0		제목 없음

☯ 통신 방향	≡ 통신 프로토콜	≡ 관련 기능 상태	≡ 세부 기능 자동화 정보 요청	≡ Description	≡ Requet	≡ Response	Aa 기타 관련 사항
Jetson → Pi	MQTT	디퓨저 상태	디퓨저 동작 정보 요청		topic : "{device_id}/Request/OperationModeInfo" payload : 0		제목 없음
Jetson → Pi	MQTT	디퓨저 상태	디퓨저 캡슐 정보 요청		topic : "{device_id}/Request/Capsule/Info" payload : 0		제목 없음
Jetson → Pi	MQTT	디퓨저 상태	잔여량 반환		topic : "{device_id}/Status/Remainder" payload: { " " slot1RemainingRatio " : 100, " slot2RemainingRatio " : 100, " slot3RemainingRatio " : 100, " slot4RemainingRatio " : 100, }		제목 없음
Jetson → Pi	MQTT	항기 조합	항기 조합 요청		topic : "{device_id}/Request/Combination" payload: { "combinationId" : Combination Id }		제목 없음