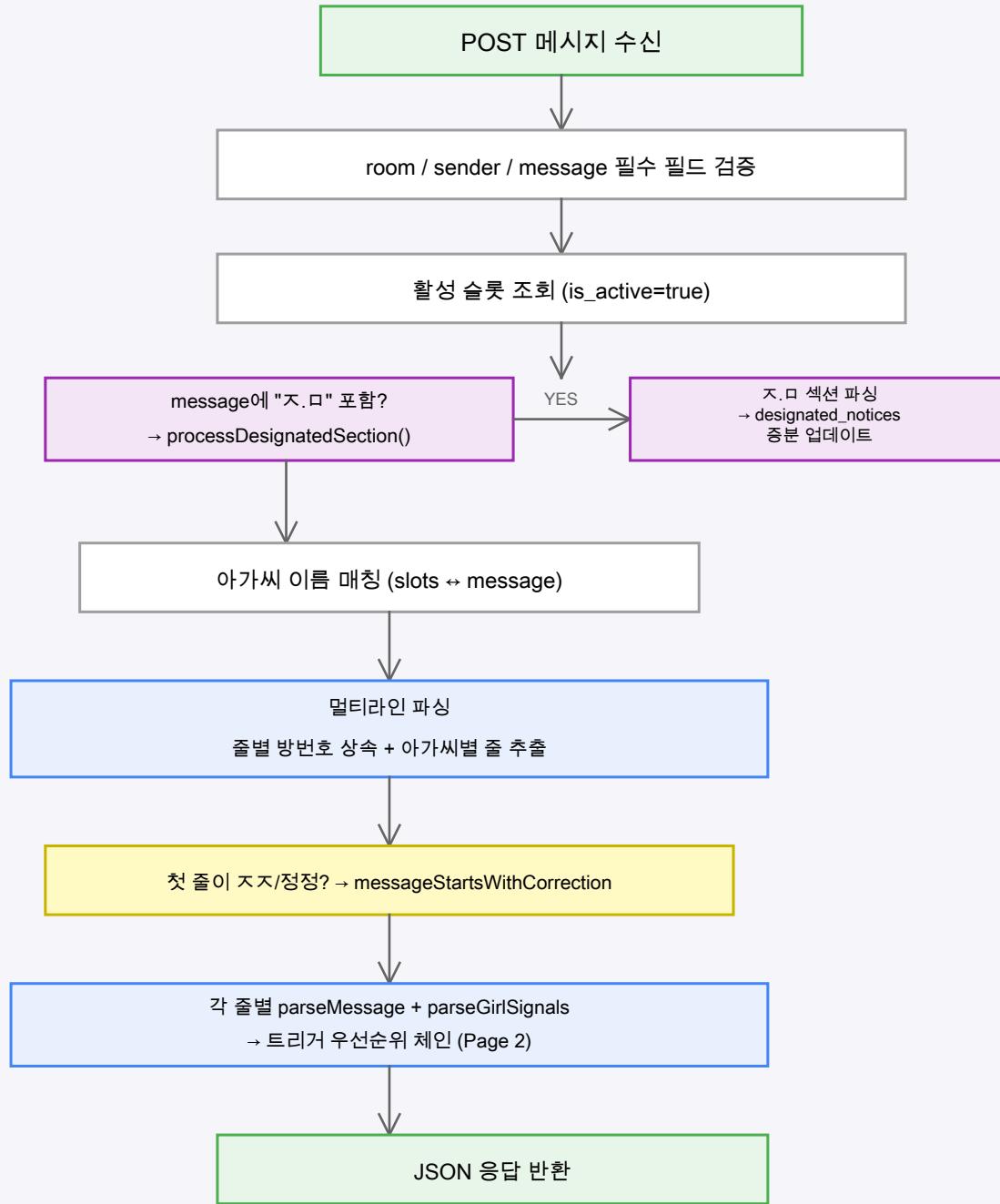


# 1. 전체 메시지 수신 흐름 (POST /api/bot/message)



범례:

- 시작/종료
- 일반 처리
- 핵심 로직
- ㅈ.ㅁ 지명 처리
- 조건 분기

## 2. 트리거 우선순위 체인 (각 아가씨 줄별)

순위	신호	조건	핸들러	DB 결과
1	ㄱㅌ (취소)	isCancel	handleCancel	trigger_type → canceled
2	ㅎㅅㄱㅈㅈㅎ (현시간재진행)	isNewSession && roomNumber	handleNewSession	새 세션 INSERT (기존 무시)
3	ㅈㅈㅎ (재진행)	isResume	handleResume	최근 종료 레코드 → start로 UPDATE
4	ㄱㄱ / 끝 (종료)	isEnd && roomNumber	handleSessionEnd	is_in_progress=false trigger_type=end
5	ㅈㅈ + 시간패턴 (ㄱ 없음)	isCorrection && roomNumber && manualTime	직접 UPDATE	start_time 수정 data_changed=true
6	방번호 (일반 시작)	roomNumber && !isEnd && !isDesignatedFee 등	handleSessionStart	새 세션 INSERT (증복시 무시)
7	ㅈㅈ + 방번호 (DB 세션 없음)	messageStartsWithCorrection && roomNumber && DB 진행중 세션 없음	handleSessionStart	신규 세션 생성
8	ㅈㅈ + 방번호 (DB 세션 있음)	messageStartsWithCorrection && roomNumber && DB 진행중 세션 있음	직접 UPDATE	시간패턴 있으면 start_time 정정
9	기타	위 조건 모두 불일치	없음	type: message (로그만 저장)

### 핵심 포인트:

- 순위 1~6: 위에서 아래로 else if 체인 (먼저 매칭되면 이후 무시)
- 순위 4: ㅈㅈ 메시지에서 ㄱ 있는 줄 → isCorrection=true 자동 설정
- 순위 6: ㅈㅁㅅㅅ/ㅈㅁㅂㅅㅅ/ㅇㅈ 있으면 일반 시작 차단됨
- 순위 7~8: ㅈㅈ 메시지 catch-all → DB에서 진행중 세션 직접 조회  
→ 세션 없음: 신규 생성 / 세션 있음: 시간패턴으로 start\_time 정정
- handleSessionStart 내부: 이므로 is\_in\_progress=true 세션 있으면 자동 무시

### 3. 핸들러 상세 동작

#### handleCancel (ㅌ ㅌ)

is\_in\_progress=true인 레코드 검색  
방번호 일치 확인 (있으면)  
trigger\_type → "canceled", is\_in\_progress → false

#### handleNewSession (ㅅ ㅅ ㅅ ㅅ)

기존 레코드 무시, 무조건 새 INSERT  
getOrCreateRoom() → 방 조회/생성  
trigger\_type="start", is\_in\_progress=true

#### handleResume (ㅈ ㅈ ㅎ)

가장 최근 레코드 1개 조회 (created\_at DESC)  
trigger\_type="end"가 아니면 재진행 불가  
is\_in\_progress=true, end\_time=null, trigger\_type="start"  
data\_changed=true, end\_sent\_at=null (재발송)

#### handleSessionEnd (ㄱ/끝)

기존 start\_time 조회  
is\_in\_progress=false, end\_time=현재시간  
usage\_duration 저장, trigger\_type="end"  
checkAndCloseRoom() → 모든 아가씨 종료시 방 닫기

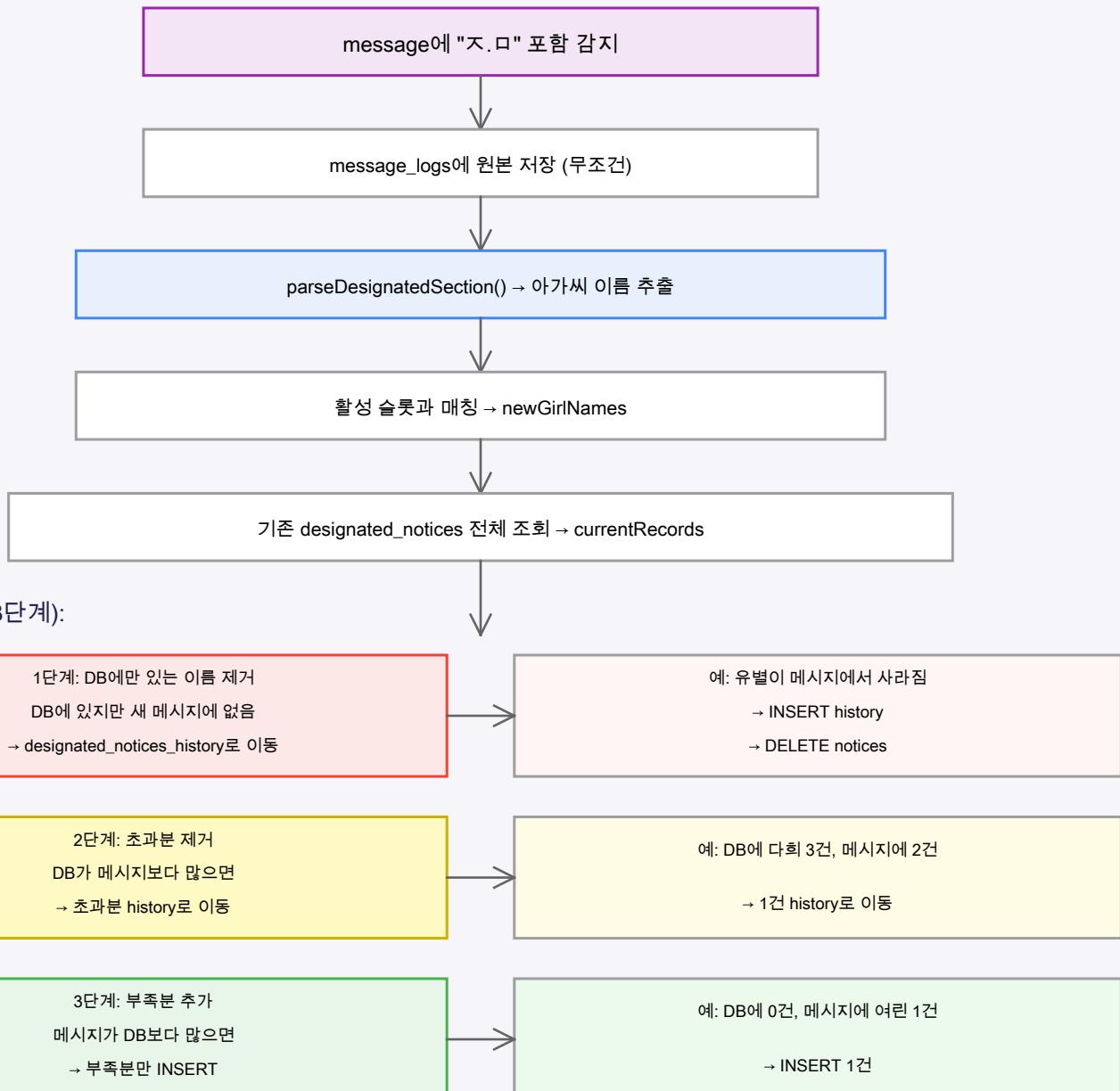
#### handleSessionStart (방번호)

이미 is\_in\_progress=true 세션 있으면 → 무시 (중복 방지)  
extractManualTime() → 수동 시간 있으면 사용  
getOrCreateRoom() → 방 조회/생성  
status\_board INSERT, trigger\_type="start"  
is\_designated 플래그 설정 (ㅈ ㅁ 포함시)

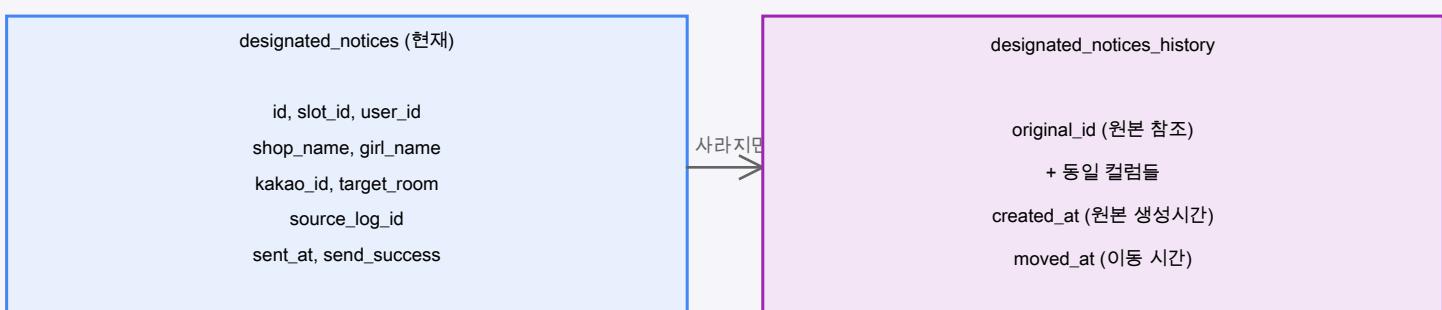
#### updateStatusBoard (공통)

isCorrection=true → 같은 slot+room 최근 레코드 UPDATE  
일반 종료 → in-progress 레코드 UPDATE  
usageDuration만 있고 세션 없음 → 종료된 레코드에 duration 추가  
새 시작 → INSERT

## 4. 지명(designated) 처리 흐름



### DB 테이블 구조:



## 5. 신호 코드 참조표

코드	별칭	의미	동작
ㄱ	끝	세션 종료	is_in_progress=false, end_time 기록, usage_duration 저장
ㅈㅈ	정정	수정/정정	메시지 첫줄: 전체 정정 모드. ㄱ 있으면 정정종료, 없으면 DB 체크
ㅈㅈㅎ	재진행	재진행	최근 종료 레코드를 다시 시작(start)으로 되돌림
ㅎㅅ ㄱㅈㅈㅎ	현시간재진행	새 세션	기존 무시, 무조건 새 세션 INSERT
ㄱㅌ	-	취소	trigger_type=canceled, is_in_progress=false
ㅈㅁ	-	지명	is_designated=true 플래그 설정
ㅈㅁㅅㅅ	-	지명순번삭제	별도 요금 신호 (세션 시작 차단)
ㅈㅁㅂㅅㅅ	-	지명반순번삭제	별도 요금 신호 (세션 시작 차단)
ㅇㅈ	-	연장	별도 연장 신호 (세션 시작 차단)

### 방번호 규칙:

- 3자리 숫자만 인식 (예: 802, 506, 707)
- 메시지 시작 또는 중간에서 추출
- 멀티라인: 이전 줄의 방번호를 상속 (가장 가까운 위쪽 줄)

### ㅈㅈ(정정) 메시지 특수 동작:

- 첫 줄이 ㅈㅈ/정정으로 시작 → messageStartsWithCorrection=true
- ㄱ 있는 줄 → isCorrection 자동 설정 → handleSessionEnd에서 정정 처리
- ㄱ 없는 줄 + 시간패턴 → start\_time만 수정
- 그 외 → DB에서 진행중 세션 조회 (무조건 정정):
  - 세션 없음 → 신규 세션 생성 (handleSessionStart)
  - 세션 있음 + 시간패턴 → start\_time 정정 (예: 1030 → 10:30)
  - 세션 있음 + 시간패턴 없음 → 무시

### ㅈ.ㅁ 섹션 파싱 규칙:

- 구분선: ㅈ.ㅁ 패턴 감지
- 구분선 아래 줄: "손님이름 — 아가씨이름" 형식
- — (U+3161) 또는 - 로 분리, 오른쪽이 아가씨 이름
- 점(.) 자동 제거: 다.희 → 다희
- 증분 동기화: 새 메시지와 DB 비교 → 추가/제거만 수행