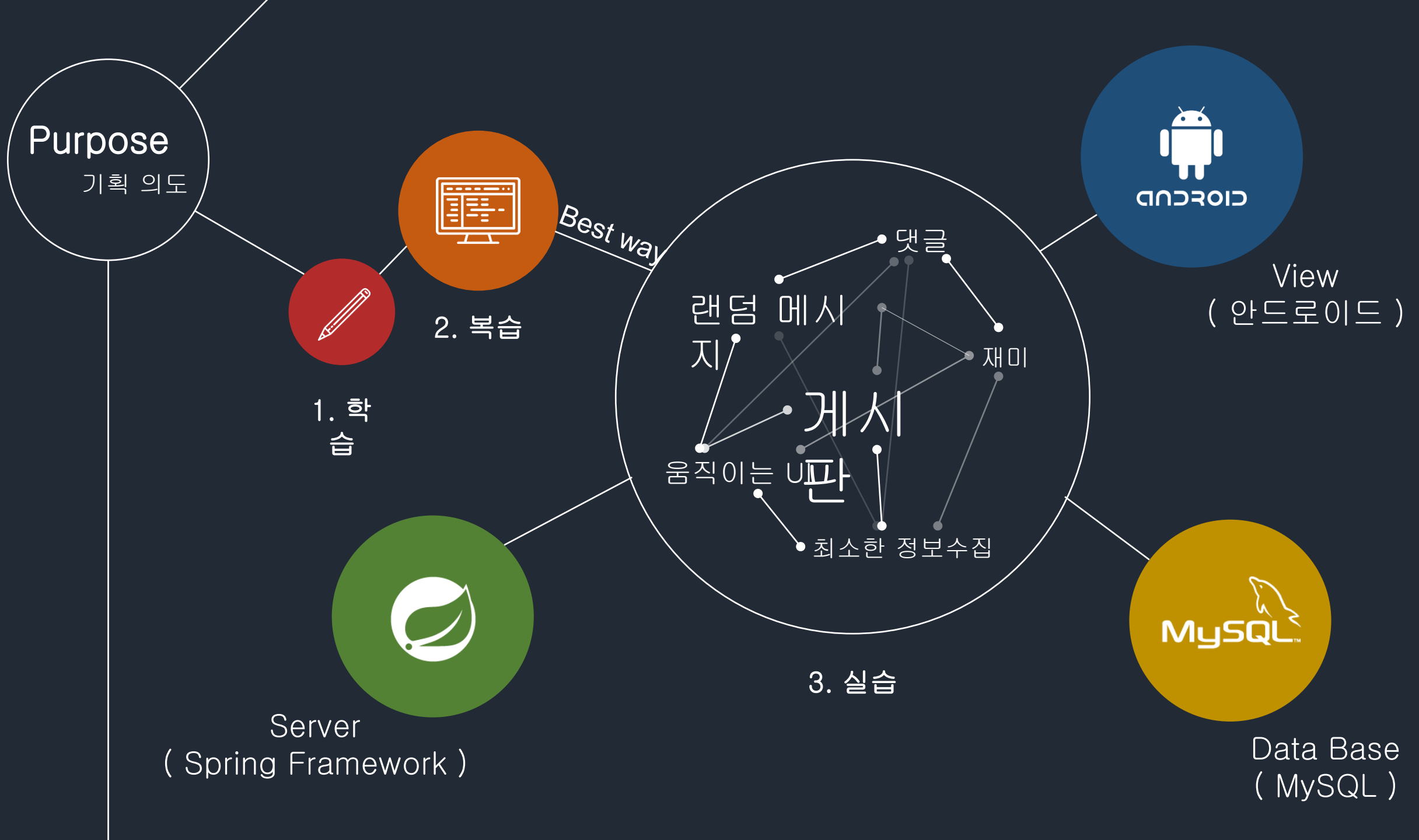




병 속에 담긴 메시지

안드로이드 애플리케이션





게시판

1. 실제로는 텍스트 게시판
2. 보여지는 UI는 물병에 담긴 게시물
3. 물병 안에 담긴 내용은 데이터베이스에 있는 게시물 중 랜덤으로 선택된 게시물.
4. 사용자가 능동적으로 선택하는 것이 아니라 수동적으로 받아 보는 형식
5. 게시물을 작성한 사용자도 자신이 쓴 글이 다시 물병 속에 담겨 오거나, 앱을 설치한 지 100일이 지나 글을 확인할 수 있는 **잠금이 해제될 때까지** 자신의 게시물을 다시 볼 수가 없다.
6. 게시물에 댓글을 달 수 있도록 해서 메시지를 보낸 사용자가 나중에 자신의 글에 달린 댓글을 확인할 수 있다.
7. 아이템 상자를 생성해서 글을 쓰려면 물병, 종이, 펜 세 종류가 각각 1개씩 있어야 한다.





서버 구성 (Spring Framework + MyBatis 사용)

MyBatis

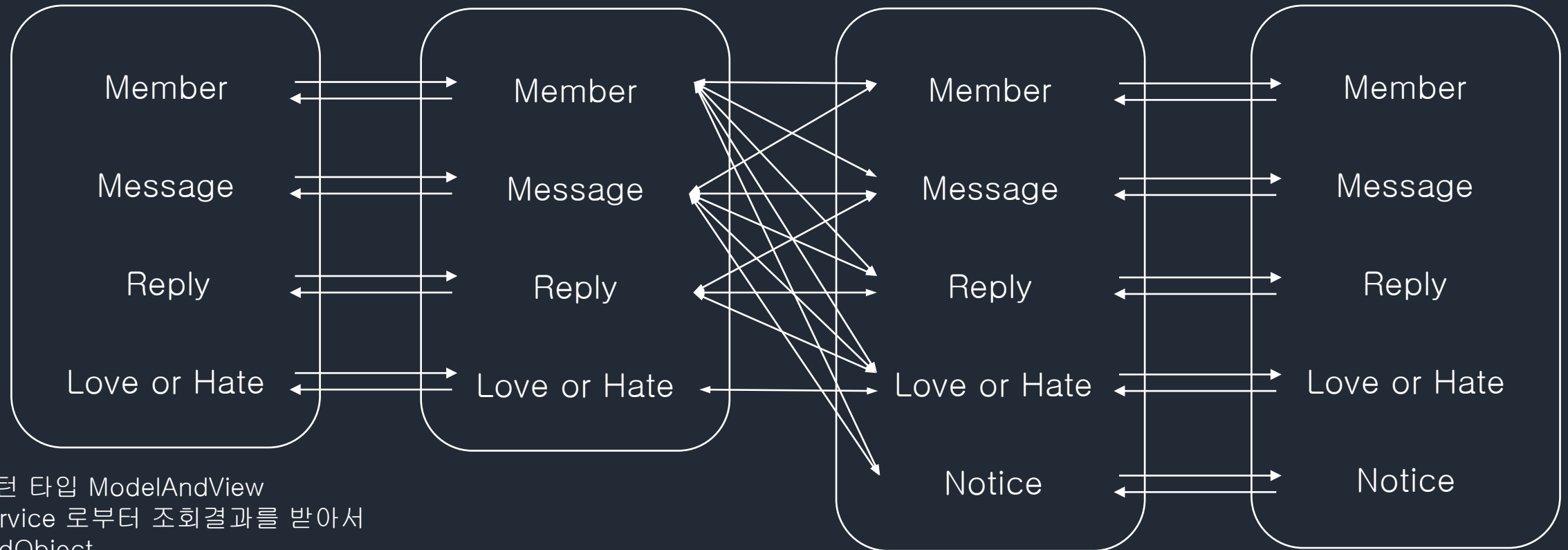
SqlSessionFactory
SqlSessionTemplate

@ Controller

@ Service

@ Repository

sql.xml



리턴 타입 ModelAndView
Service 로부터 조회결과를 받아서
addObject

BeanNameViewResolve로 만들어 놓은
JSON view로 setViewName



데이터 베이스 구성

사용자 정보 수집 최소화

IMEI : 단말기 식별(고유) – 중고 기기 등록 문제, 와이파이 전용 기기 없음

mac address : 단말기 마다 고유, 특정 단말 와이파이 off상태 값을 얻어오지 못함

Android id : OS 설치 시 부여 고유 값, OS초기화 하면 재생성 (채택)

TO_REACH_YOU_MEMBER 회원				
물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
ANDROID_ID	사용자	N/A	VARCHAR(20)	N-N
JOIN_DATE	가입일	N/A	DATETIME	N-N
BOTTLE	물병보유갯수	N/A	INT	N-N
PENCIL	연필보유갯수	N/A	INT	N-N
PAPER	종이보유갯수	N/A	INT	N-N
NOTICE_CHECK	공지확인여부	N/A	INT	N-N

TO_REACH_YOU_NOTIC 공지				
물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
NOTICE_NO	공지번호	N/A	INT	N-N
NOTICE_DATE	공지날짜	N/A	DATETIME	N-N
NOTICE_CONTENT	공지글	N/A	VARCHAR(1500)	N-N

TO_REACH_YOU_MESSAGE 메시지 리스트				
물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
SEQ_NO	게시물번호	N/A	INT	N-N
ANDROID_ID	사용자	N/A	VARCHAR(20)	N-N
CONTENT	내용	N/A	VARCHAR(1500)	N-N
CRE_DATE	작성일자	N/A	DATETIME	N-N
HIT	조회수	N/A	INT	N-N

TO_REACH_YOU_LOVE_OR_HATE 좋아싫어요				
물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
LOVE_OR_HATE_NO	순번	N/A	INT	N-N
ANDROID_ID	사용자	N/A	VARCHAR(20)	N-N
SEQ_NO	게시물번호	N/A	INT	NULL
REPLY_NO	댓글번호	N/A	INT	NULL
LOVE_OR_HATE	좋아요싫어요	N/A	INT	NULL

기본값0 / 좋아요1 / 싫어요2

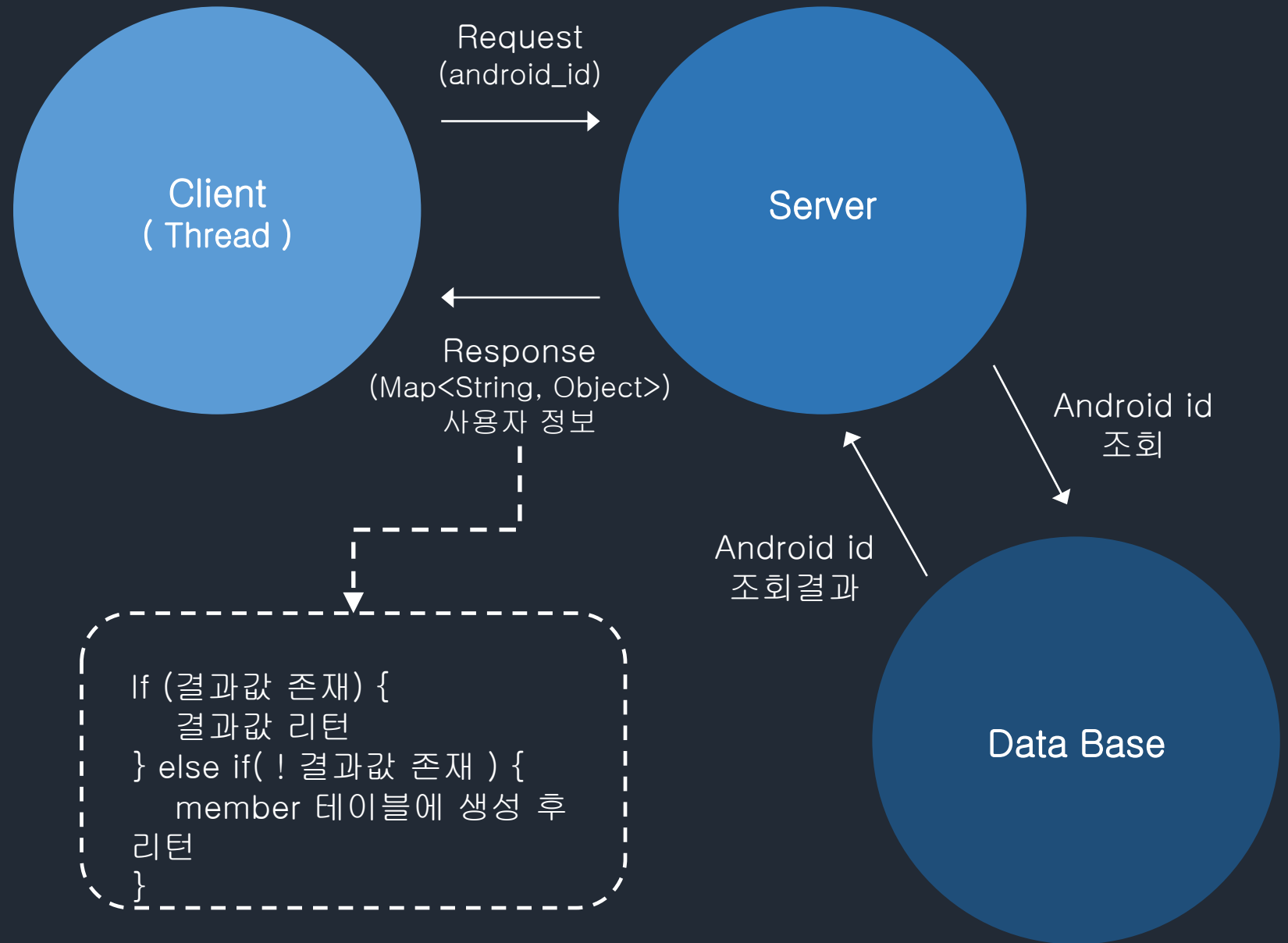
TO_REACH_YOU_REPL 댓글				
물리 이름*	논리 이름*	도메인	데이터 타입	널 허용
REPLY_NO	댓글번호	N/A	INT	N-N
SEQ_NO	게시물번호	N/A	INT	N-N
ANDROID_ID	사용자	N/A	VARCHAR(20)	N-N
REPLY	내용	N/A	VARCHAR(500)	N-N
REPLY_DATE	작성일자	N/A	DATETIME	N-N

공지 글이 등록 되면 사용자 접속 시 물방울 모양의 아이콘으로 확인가능

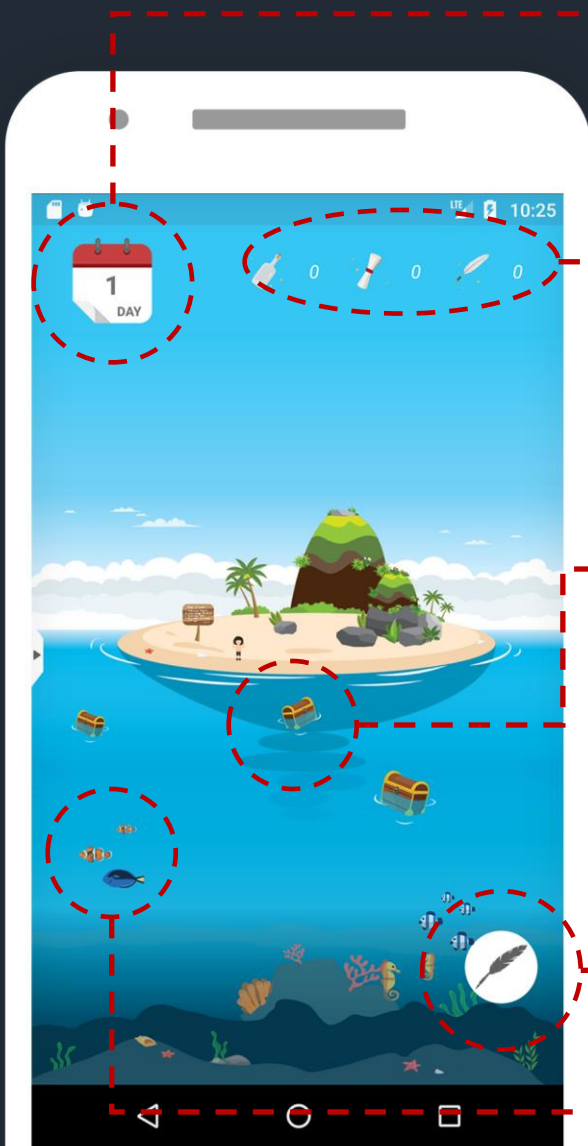
사용자가 읽고 버튼 클릭 시 member 테이블의 notice_check 컬럼에 공지 번호가 등록됨

앱 실행 시 서버에서 사용자 정보를 받아오고, 최신 공지 글 번호와 notice_check에 등록됨

1. Intro Activity



2. Main Activity



1. 경과한 기간(일)

입)
DataBase – member 테이블의 가입일(now()로 생성한 datetime타입)

메인액티비티에서 currentTimeMillis()로 현재 날짜를 생성한 후
인트로에서 넘겨받은 사용자 정보의 가입과 계산해서 적용.

2. 보유한 아이템 수

물병 메시지 클릭 & 아이템 상자에서 얻은 보유 아이템 누적 수
글을 작성하면 종류별로 1개씩 감소.

3. 물병 또는 아이템 상자

메인액티비티의 onCreate() 메소드가 실행될 때 물병과 아이템 상
자가
boolean값으로 번갈아 가며 생성(1~5개 랜덤). ImageView는 같이
사용

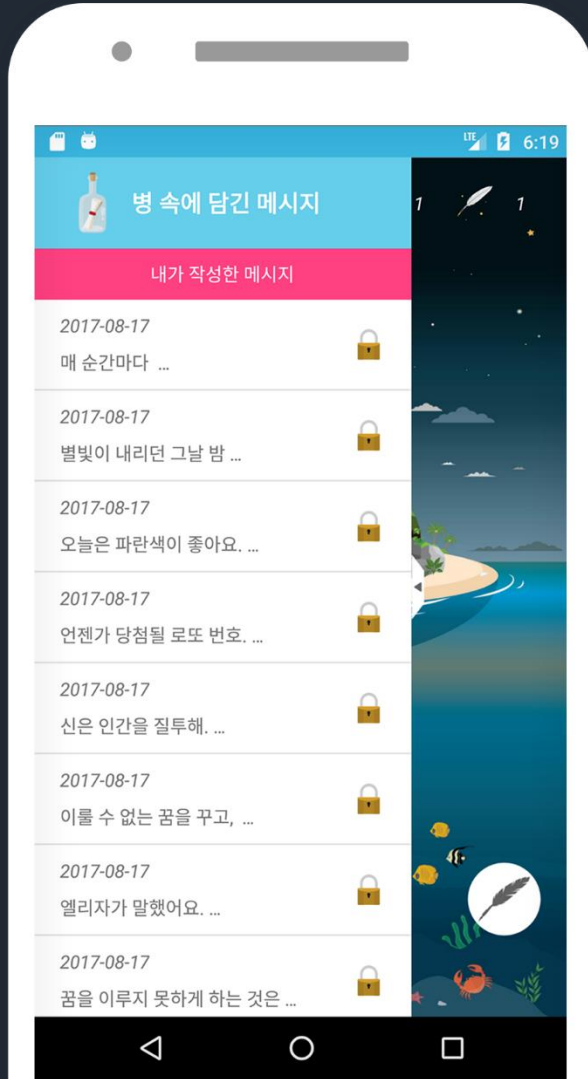
생성된 개수에 맞게 물병은 Thread로 서버에 게시물 조회요청.
아이템 상자는 세 종류의 아이템이 랜덤으로 생성되서 배치됨.

아이템 세 종류가 최소 1개 씩은 있어야 작성 가능함(부족 : 토스트
메시지)

5. 애니메이션 배경 이미지

정적인 배경은 심심해 보여서 움직이는 물고기를 배치(애니메이션
클래스)

3. Drawable menu(내가 작성한 메시지 리스트)



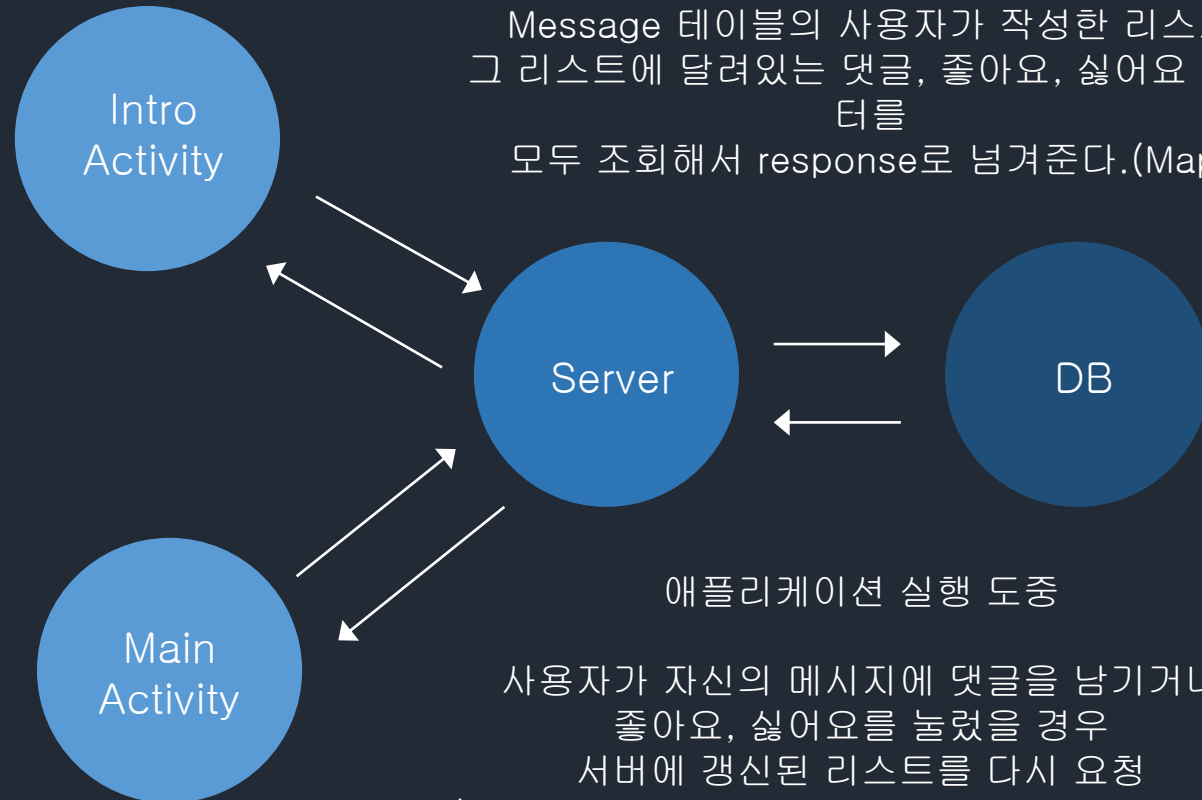
애플리케이션 실행

애플리케이션 실행 시 사용자 정보 조회 시에
Member 테이블의 사용자 정보

+

Message 테이블의 사용자가 작성한 리스트
그 리스트에 달려있는 댓글, 좋아요, 싫어요 데이
터를

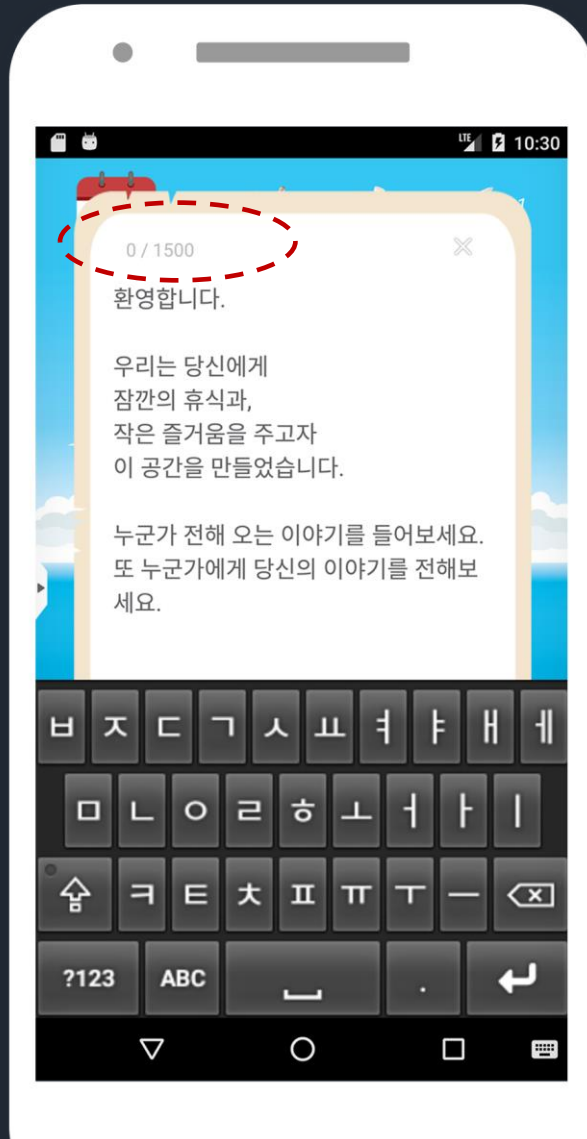
모두 조회해서 response로 넘겨준다.(Map)



애플리케이션 실행 도중

사용자가 자신의 메시지에 댓글을 남기거나
좋아요, 싫어요를 눌렀을 경우
서버에 갱신된 리스트를 다시 요청
(자신의 메시지가 잠금 해제 된 상태일 경우 갱신
하지 않으면 처음 인트로에서 받아온 데이터를 그
대로 사용하기 때문에 추가된 내용이 보이지 않기
때문)

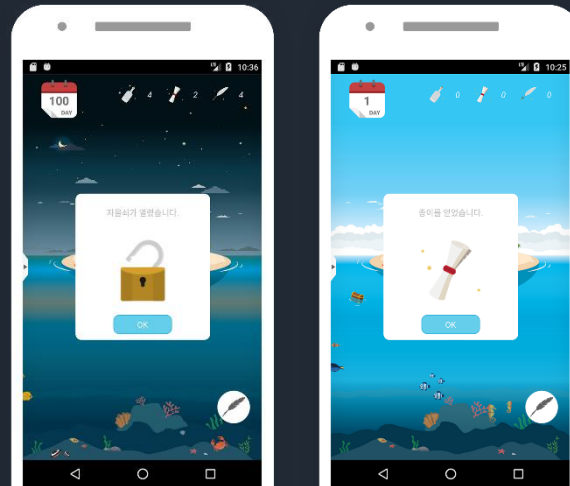
4. Dialog(메시지 작성 & 읽기 / 아이템 획득 / 메시지 잠김 / 공지 사항)



클래스 생성(Dialog 클래스 상속)

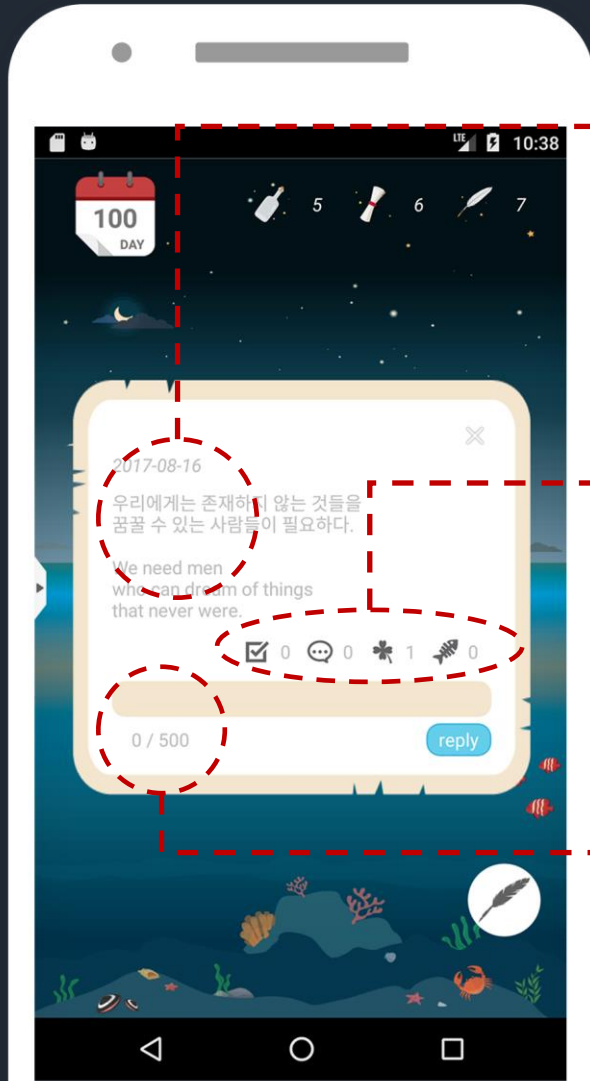
String.length() 메소드로 문자열 길이를 체크해서 실시간으로 작성되는 텍스트의 글자 수를 표시(Handler 사용)

메시지 작성 후 던지기 버튼 클릭 시에 "첨벙 하고 물에 빠지는 사운드"를 출력
(soundManager class 사용)



아이템 상자를 클릭하면 나오는 itemDialog는 메인액티비티에서 아이템상자를 생성할 때 무엇이 들어갈 지 랜덤으로 값을 부여하고, 사용자가 상자를 클릭하면 그때 dialog를 생성함. 생성자에 아이템 종류에 따른 분류 값을 넘겨 줘서 생성.

5. Dialog(메시지 읽기)



1. 메시지(게시물)

랜덤으로 생성된 물병 개수만큼 DB에 랜덤으로 게시물 조회 (Thread)

물병 클릭 시에 readDialog 클래스를 생성하면서 인자로 DB에서 조회해 온 게시물 리스트 중 마지막 인덱스의 객체를 넘겨 준다.

메인액티비티 전역 변수로 인덱스 값을 넣고 readDialog에 하나씩 넘겨준다.
2. 조회수 / 좋아요 / 싫어요 count
물병을 클릭해서 readDialog가 실행될 때 서버에 게시물 조회수 증가.

좋아요, 싫어요 버튼을 클릭할 때 마다 서버에 증감 전송 누르는 버튼에 따라 토스트 메시지로 알림(누름, 취소, 변경)

3. 댓글

댓글 버튼을 클릭하면 댓글 작성하는 editText가 등장하고, 게시물에 댓글이 있다면 최근 작성된 순으로 정렬되어 표시. 댓글도 실시간으로 글자 수 체크 적용(500자)
답변형 댓글 미적용 (메시지 특성상 여러 사람이 선택적으로 게시

물을

보는 것이 아니다 보니, 댓글 작성이 일회성으로 끝나기 때문)

6. 동적 UI / 이스터 에그

MainActivity에서
onResume() 메소드를 @Override.
onResume이 실행 중간의에도 체크할 수 있기
때문.

현재 시간을 체크 후 아침/낮, 저녁 일 때의
배경이미지를 다르게 보이도록 설정.

화면 이곳 저곳에 클릭 이벤트를 설정해서
일정 횟수가 클릭 되면 개발자를 표시하는
기능 추가.

