5주차 진행 보고서 세부 사항

2021.03.18~ 2021.03.31 B71116장혜리

코드는 아래 gitnub에서 확인 가능합니다.

[**[github]MBTI-Lovers 안드로이드**](https://github.com/hyerijang/MBTI-Lovers)

[**[github]MBTI-Lovers-webServer**](https://github.com/hyerijang/MBTI-Lovers-webServer)

# 서버 구축 및 DB 추가

## 안드로이드

### WebViewActivity 생성. 웹뷰로 웹 서버와 연결

웹뷰를 통해 웹서버에 접근하기 전 저희 앱의 용도에 맞게 세부 설정을 하였습니다. 관련 소스코드는 다음과 같습니다.

### LoginActivity에서 WebViewActivity 이동시 자동 로그인 기능 추가

이 기능의 작동 방식은 다음과 같습니다.

1. LoginActivity에서 구글/페이스북 로그인 진행 시 firbaseAuth에서 해당 유저를 식별하기 위한 uid를 발급해 줌
2. uid를 이용하여 웹 서버 자동로그인 url 생성
3. 자동로그인 url을 WebViewActivity에 전송
4. 로그인 된 화면이 유저에게 보여짐

## 웹 서버

### FirebaseAuth 연결

FirebaseAuth를 웹서버에 추가하였습니다. Firebase는 회원가입 및 채팅기록 저장에 이용됩니다.

### 세션 기반 로그인 처리, 인증/로그인을 위한 인터셉터 추가

**HttpSession**을 이용하여 세션 기반 로그인 처리를 구현하였습니다.

### 인증 인터셉터

1. 로그인 url에 접근 시 이미 로그인 된 유저인지 아닌지 인증
2. 로그인 세션이 없다면 로그인 인터셉터 실행

### 로그인 인터셉터

1. uid를 이용하여 로그인
2. 로그인 성공 시 세션 정보를 서버에 보관
3. 로그인 접근 이전의 페이지로 이동

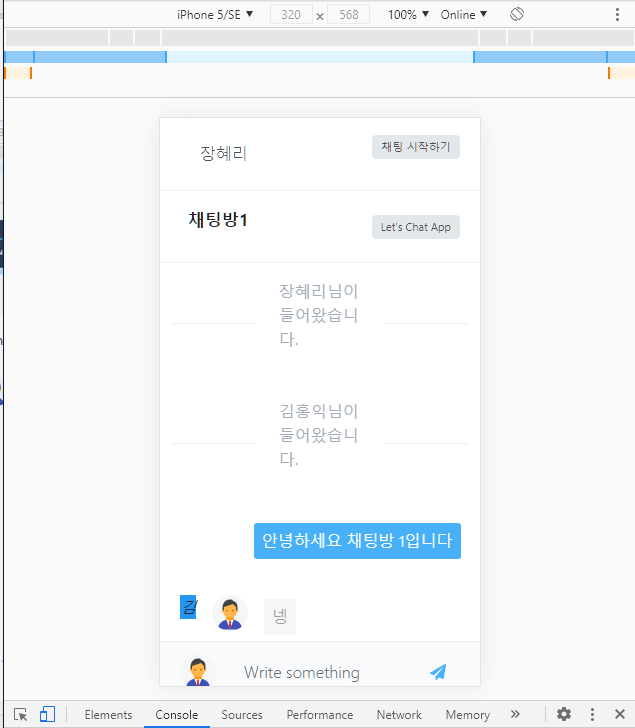
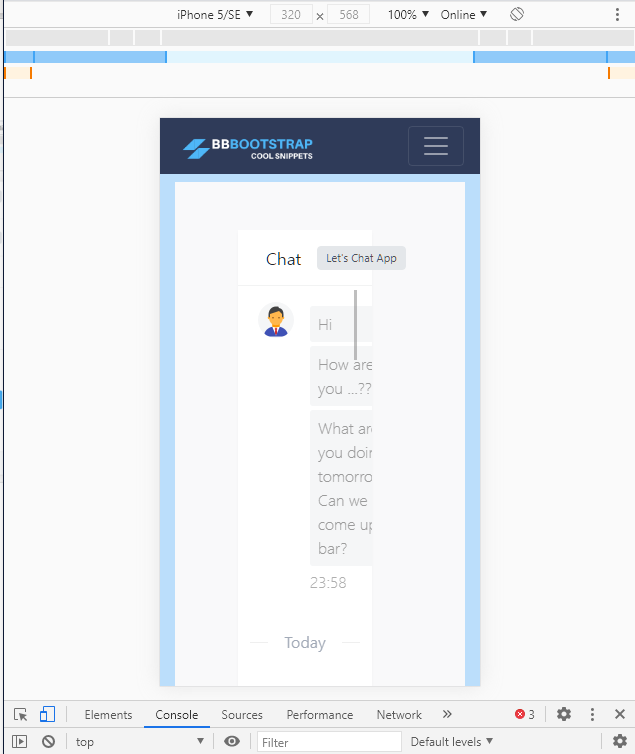
### DB 접근 방식 JDBC Template에서 SpringDataJpa로 변경

기존에는 DB접근시 JDBC Template를 사용하였으나 SpringDataJpa를 사용하도록 변경하였습니다.

# 채팅 구현

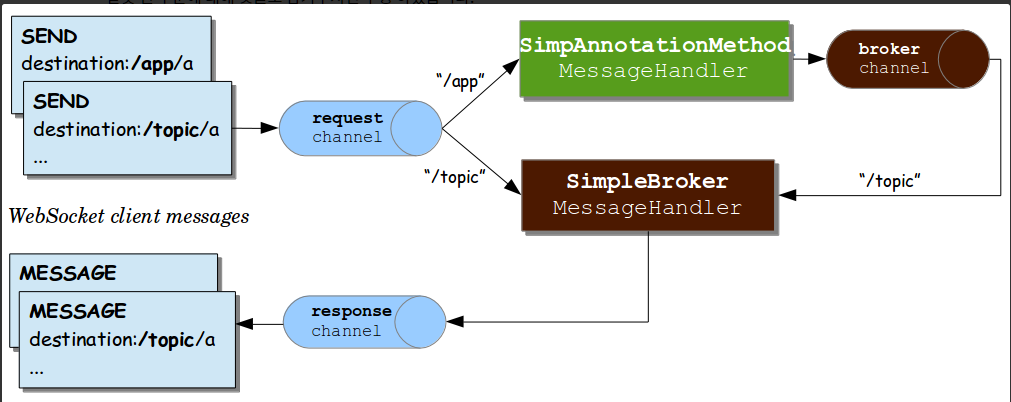
### 반응형 웹페이지로 변경

기존에 사용하던 부트스트랩 오픈소스는 모든 요소의 크기가 고정되어 있어 모바일 환경에서 사용시 채팅 화면이 깨지는 문제점이 있었습니다. 이를 해결하기 위해 element들의 width와 heigth를 브라우저 창의 가로 또는 세로 크기에 따라 변하게끔 변경하였습니다. 아래는 수정 전 후 레이아웃 차이 입니다.



### 채팅방 분리

채팅 시스템은 텍스트 기반 메세징 프로토콜인 STOMP를 사용합니다. 기존에 구현된 채팅은 STOMP의 기본 구조를 사용하며, 방의 구분 없이 모든 유저가 하나의 방에서 채팅 하는 방식으로 구현되어 있었습니다. 기존의 작동 방식은 아래의 그림과 같습니다.

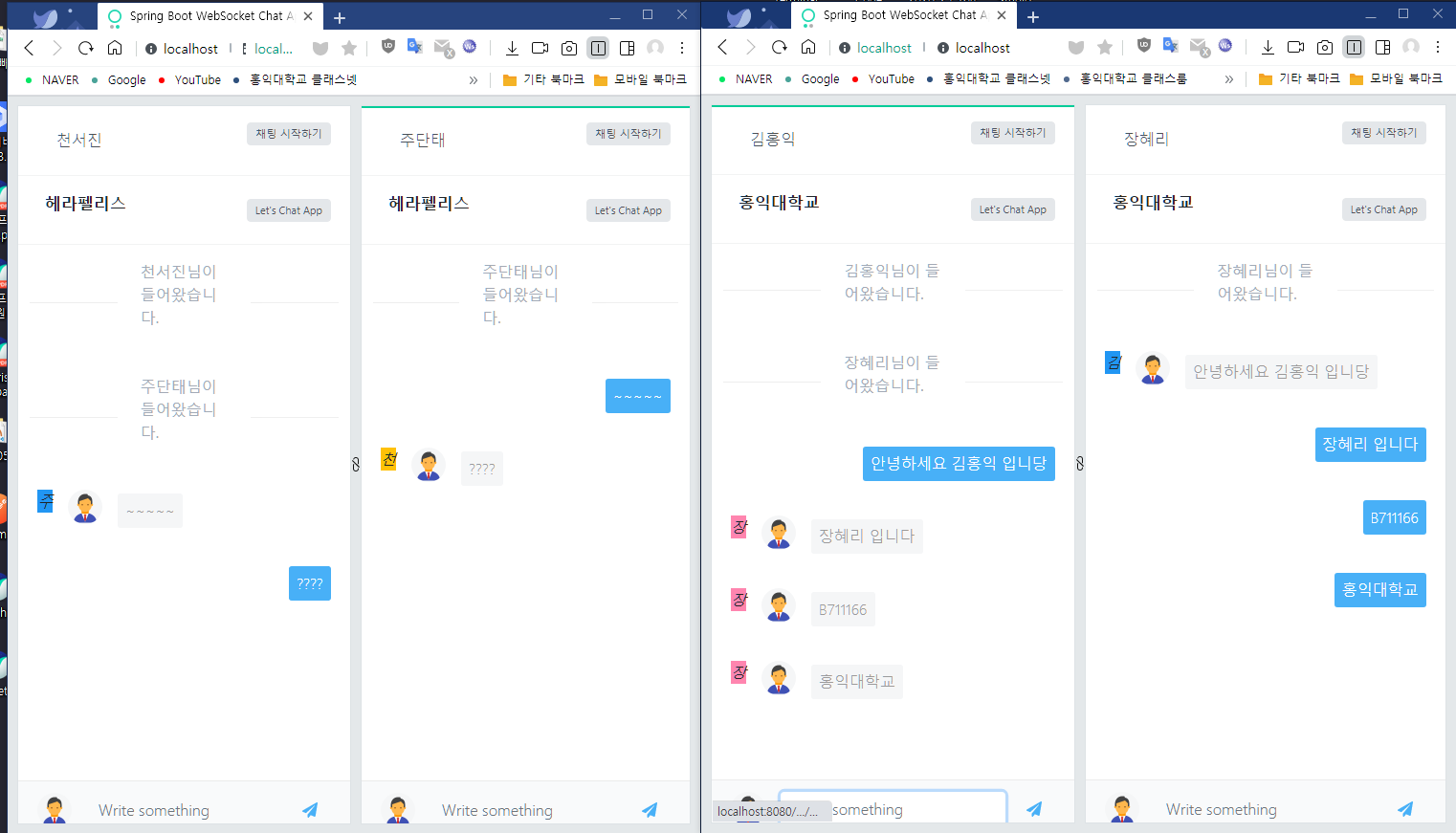


기본 STOMP구조

출처: 큐우의 개발 이야기 :: STOMP 맛보기

1. 모든 클라이언트는 채팅입장시 “/app”과 “/topic”을 웹소켓 서버에 전송
2. “/app”으로는 channel broker을 등록
3. “/topic”을 통하여 통신
   1. 어떤 클라이언트가 메시지를 전송하면 채널 브로커가 이를 받아 해당 “/topic”을 구독한 모든 클라이언트들에게 메시지를 전송한다.

중요한 점은 특정 토픽을 구독한 모든 유저에게 메시지가 전송된다는 것으로, “/topic”을 “/sub/chat/room/{roomId}”와 같이 세분화하면 {roomId}를 이용하여 채팅 방별로 채팅이 가능하게 됩니다. 구현 결과는 다음과 같습니다.



## 참조

[안드로이드 웹뷰에서 로그인 세션 유지하기 :: 미래의 나에게 과거의 나로부터](https://dailylonnie0125.tistory.com/11)

[[Android]웹뷰 자동로그인을 위한 여정](https://dbcp.tistory.com/42)

[# Spring MVC 게시판 예제 16 - HttpSession을 이용하는 로그인 처리](https://doublesprogramming.tistory.com/211)

[Moomini's Devlog :: 스프링(Spring) 세션(Session), 쿠키(Cookie)](https://devmoomini.tistory.com/92)

[[ CSS ] 오버플로우(overflow) 속.. : 네이버블로그](https://blog.naver.com/dsz08082/222214442168)

[큐우의 개발 이야기 :: STOMP 맛보기](https://faith-developer.tistory.com/194)

[Spring websocket chatting server(2) - Stomp로 채팅서버 고도화하기](https://daddyprogrammer.org/post/4691/spring-websocket-chatting-server-stomp-server/)

[[Spring Boot - WebSocket, STOMP을 연동한 채팅 서비스 구현] : 네이버 블로그](https://m.blog.naver.com/PostView.nhn?blogId=scw0531&logNo=221097188275&proxyReferer=https:%2F%2Fwww.google.co.kr%2F)

[스프링부트에서 STOMP 웹소켓 사용하기 | hwiVeloper](https://hwiveloper.dev/2019/01/10/spring-boot-stomp-websocket/)