

## 상세설계서

# 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## Beat Make & Crush

VR rhythm game with Beat Note automatic  
generation service using deep learning

Ver. 1.1

2019. 11. 05

한국외국어대학교

정보통신공학과

Team VRR

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 문서 정보

구 분	소 속	성 명	날 짜	서 명
작성자	한국외국어대학교	박영준	2019. 11. 04	
	한국외국어대학교	문명기	2019. 11. 04	
	한국외국어대학교	김세진	2019. 11. 04	
	한국외국어대학교	조동철	2019. 11. 04	
	한국외국어대학교	이호찬	2019. 11. 04	
검토자	한국외국어대학교	박영준	2019. 11. 04	
	한국외국어대학교	문명기	2019. 11. 04	
	한국외국어대학교	김세진	2019. 11. 04	
	한국외국어대학교	조동철	2019. 11. 04	
	한국외국어대학교	이호찬	2019. 11. 04	
사용자				
승인자	한국외국어대학교	홍진표		

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 개정 이력

버전	작성자	개정일자	개정 내역	승인자
1.0	박영준	2019. 11. 05.	초안 작성	
	문명기			
	김세진			
	조동철			
	이호찬			
	검토자	박영준		
1.1	박영준	2019. 12. 19.	요구사항 재 반영 및 구성도 수정	
	검토자	박영준, 문명기, 김세진, 조동철, 이호찬		

## 목 차

<b>1</b>	<b>개요</b>	<b>5</b>
1.1	서비스 정의	5
1.2	문서의 목적	5
1.3	용어 정리	5
<b>2</b>	<b>시스템 구성요소 및 요구사항 점검</b>	<b>6</b>
2.1	시스템 구성요소	6
2.2	요구사항 점검	7
2.3	요구사항 상세유형	8
<b>3</b>	<b>시스템 기능 상세 설계</b>	<b>10</b>
3.1	인터페이스 어플리케이션 부 시스템 설계 (SFR-I)	10
3.2	인터페이스 게임 어플리케이션 부 시스템 기능 요구사항 (SFR-GI)	16
3.3	서비스 어플리케이션 부 시스템 기능 요구사항 (SFR-V)	19
3.4	서버 부 시스템 기능 요구사항 (SFR-S)	31
3.5	ORM 서버 부 요구사항 (SFR-O)	38
<b>4</b>	<b>상세설계 산출물</b>	<b>41</b>
4.1	Use Case Diagram	41
4.2	VRR Deep Learning Model 구조	42
4.3	Web 플랫폼	45
4.3.1	Use Case Diagram	45
4.3.2	Sequence Diagram	46
4.3.2.1	LogIn	46
4.3.2.2	사용자 정보	47
4.3.2.3	Play 랭킹	49
4.3.2.4	음악 노트 & 사용자 검색	50
4.3.2.5	피드 보기	52
4.3.2.6	노트 변환 기능	53
4.3.2.7	미들 웨어 권한 확인	54
4.3.3	Class Diagram	54
4.3.3.1	Web Server Client	54
4.3.3.2	Web Server	55
4.3.3.3	ORM Server	56
<b>5</b>	<b>팀원 담당업무</b>	<b>59</b>
<b>6</b>	<b>프로젝트 세부일정</b>	<b>59</b>
<b>7</b>	<b>참고문헌</b>	<b>60</b>

# 1 개요

## 1.1 서비스 정의

자사의 서비스 'Beat make & Crush'는, VR(Virtual Reality)기기와 키넥트의 모션인식 기술을 이용하여 **Player의 Action으로 날아오는 Beat Note를 부수는 게임 서비스**이다.

**2인 play 모드**는 경쟁하며 콤보에 따른 패널티를 부과해 점수를 높이는 있는 새로운 방법의 VR리듬게임 서비스이며 키넥트의 모션인식 기능을 활용해 단순히 Beat를 치는 것에 국한되지 않고 벽을 피하는 등 움직이는 요소로 재미를 극대화한다.

deep learning 기술을 활용해 어떤 음악이던 재미있는 리듬 게임으로 Play할 수 있도록 **Beat Note를 자동으로 생성**해 주는 서비스가 있다. 매일 똑 같은 Beat Note를 Play하는 것은 금방 질리기 쉽고, 흥미가 떨어지기 쉽지만 자신이 Play하고 싶었던 음악을 Beat Note로 만들 수 있기 때문에 끊임없이 새로운 리듬게임을 Play할 수 있다

생성된 나만의 Beat Note는 자사의 VRR Server를 통해 다른 Player들과 공유할 수 있고 기록, 정보 등을 확인할 수 있는 웹 플랫폼을 제공하며, 웹 플랫폼을 통해 게임 기록, 사용자 랭킹, 생성한 Beat Note 목록 등을 확인할 수 있는 **편리한 웹 서비스**도 제공한다.

## 1.2 문서의 목적

본 문서는 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임(Beat make & Crush)사업에 관련된 요구사항을 토대로 설계 시 필요한 상세설계 부분에 대하여 중점적으로 설명한다.

## 1.3 용어 정리

구성 요소	설명
VRR	VR 리듬게임의 약어로 팀명이다.
Beat make & Crush	자사의 게임 이름으로 비트 노트를 자동 생성하는 기능이 포함된 'Beat make' 와 키넥트를 이용해 몸으로 게임하며 Beat Note를 치는 형식의 'Crush'를 의미한다.
비트 (Beat)	음악의 박자, 패턴과 같은 리듬을 의미한다.
비트 노트 맵 (Beat Note map)	음원을 기반으로 만들어진 게임 플레이에 필요한 리듬게임 악보이다.
비트 노트 (Beat Note)	만들어진 Beat Note map에 있는 하나 하나의 박자를 의미

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 2 시스템 구성요소 및 요구사항 점검

### 2.1 시스템 구성요소

본 시스템의 구성요소에는 User, VRR Software, VRR Web App, VRR Server가 포함된다.

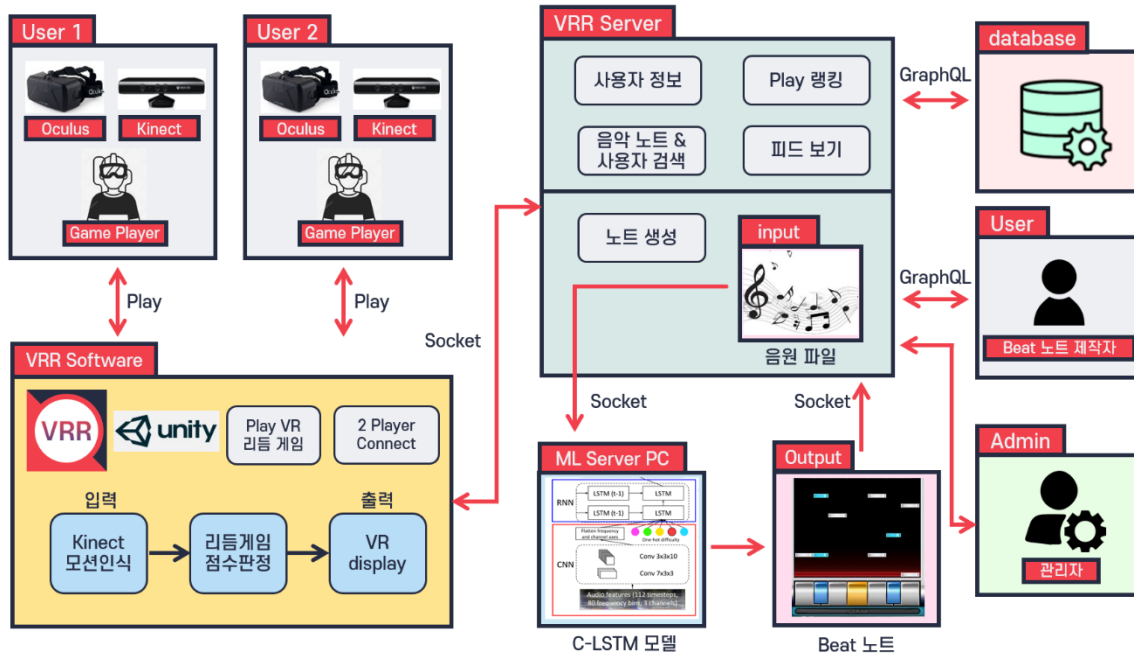


그림 1 VRR 구성도

구성 요소	설명
User	User는 VRR Game Service를 받는 대상이다.
VRR Software	VRR Software는 VRR Game 프로그램이다.
VRR Server	Web App 과 Software 사이에 필요한 Server이다.
ML Server	비트 노트 자동 생성에 필요한 CNN 모델의 머신러닝 Server이다.
Database	Database는 VRR System과 정보를 주고받는다.

언급한 구성요소에 대한 대략적인 설명을 위와 같다.

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 2.2 요구사항 점검

아래 테이블은 System의 상세설계 전 명세 된 요구사항을 검토하고자 유형을 기호화 Mapping한 자료이다.

유형		요구사항 유형(구성 요소)	요구사항 기호
시스템	기능 요구사항	인터페이스 어플리케이션 부 요구사항	SFR-I
		인터페이스 게임 어플리케이션 부 요구사항	SFR-GI
		서비스 어플리케이션 부 요구사항	SFR-V
		Server 요구사항	SFR-S
		ORM Server 요구사항	SFR-O

요구사항 기호가 의미하는 바에 대해 알기 위해 다음 Table을 참조할 수 있다.

요구사항 기호	의 미
SFR-I	'Interface application'에게 요구되는 기능을 의미하고, 크게 14개의 요구 사항으로 구분된다.
SFR-GI	'Interface Game application'에게 요구되는 기능을 의미하고, 9개의 요구 사항으로 구분된다.
SFR-V	'Service application'에게 요구되는 기능을 의미하고, 크게 10개의 요구 사항으로 구분된다.
SFR-S	'Server'에게 요구되는 기능을 의미하고, 크게 29개의 요구 사항으로 구분된다.
SFR-O	'ORM Server'에게 요구되는 기능을 의미하고, 크게 7개의 요구 사항으로 구분된다.



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 2.3 요구사항 상세유형

유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	인터페이스 어플리케이션 부 요구사항 (SFR-I)	로그인 기능 제공	SFR-I-01	14
		로그아웃 기능 제공	SFR-I-02	
		회원가입 기능 제공	SFR-I-03	
		전체 음악 노트 게시물 확인 기능	SFR-I-04	
		음악 노트 게시물 댓글, 좋아요, 파일, 이미지 보기 기능 제공	SFR-I-05	
		음악 노트 게시물 및 사용자 검색 기능 제공	SFR-I-06	
		개인정보 확인 기능 제공	SFR-I-07	
		탈퇴 기능 제공	SFR-I-08	
		개인정보 변경 기능 제공	SFR-I-09	
		좋아요 개수 높은 음악 노트 게시물 정보 제공	SFR-I-10	
		자신이 좋아하는 사용자의 음악 노트 게시물 확인 기능 제공	SFR-I-11	
		음악 노트 공유 기능 제공	SFR-I-12	
		음악 노트로 변환 기능 제공	SFR-I-13	
		페이지 렌더링 대기상태 시 보여지는 애니메이션 제공	SFR-I-14	
	인터페이스 게임 어플리케이션 부 요구사항 (SFR-GI)	키넥트 모션 인식을 이용한 인터페이스조작	SFR-GI-01	4
		플레이 모드선택(커스텀, 2인)	SFR-GI-02	
		BeatNote들 생성과 이동	SFR-GI-03	
		플레이한 결과 종합 정보 표시	SFR-GI-04	
	서비스 어플리케이션 부 요구사항 (SFR-V)	전처리1 - 음악파일 변환	SFR-V-01	11
		전처리2 - STFT적용 및 스케일링	SFR-V-02	
		파형 feature 추출	SFR-V-03	
		손실 함수 정의_ CNN, RNN	SFR-V-04	
		활성화 함수 정의	SFR-V-05	
		최적화 알고리즘	SFR-V-06	
		Dataset mirroring	SFR-V-07	
		CNN 모델	SFR-V-08	
		LSTM(RNN) 모델	SFR-V-09	
		LSTM input 및 output	SFR-V-10	
		Beat Note 점수 판정 범위	SFR-V-11	
	서버 부 요구사항 (SFR-S)	댓글 생성 기능 제공	SFR-S-01	29
		댓글 삭제 기능 제공	SFR-S-02	
		로그인 시도 시 정보 일치 여부 반환 기능 제공	SFR-S-03	





상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

		좋아요 상태 변경 기능 제공	SFR-S-04	7	
		게시글 생성 기능 제공	SFR-S-05		
		게시글 수정 기능 제공	SFR-S-06		
		게시글 삭제 기능 제공	SFR-S-07		
		게시글 검색 기능 제공	SFR-S-08		
		전체 게시글 보기 기능 제공	SFR-S-09		
		내 게시글 보기 기능 제공	SFR-S-10		
		사용자 생성 기능 제공	SFR-S-11		
		모든 사용자 보기 기능 제공	SFR-S-12		
		사용자 정보 수정 기능 제공	SFR-S-13		
		사용자 팔로우 기능 제공	SFR-S-14		
		사용자 팔로우 취소 기능 제공	SFR-S-15		
		자기 자신 정보 보기 기능 제공	SFR-S-16		
		비밀 키 송신 기능 제공	SFR-S-17		
		사용자 검색 기능 제공	SFR-S-18		
		댓글 수 확인 기능 제공	SFR-S-19		
		게시글 수 확인 기능 제공	SFR-S-20		
		자신이 한 팔로우 수 확인 기능 제공	SFR-S-21		
		자신에게 팔로우 한 수 확인 기능 제공	SFR-S-22		
		대화 방 생성 기능 제공	SFR-S-23		
		대화 방 보기 기능 제공	SFR-S-24		
		대화 메시지 실시간 수신/송신 기능 제공	SFR-S-25		
		유효 JWT (JSON Web Token) 확인 기능 제공	SFR-S-26		
		Secret Key 메일 보내기 기능 제공	SFR-S-27		
		JWT(JSON Web Token) 생성 기능 제공	SFR-S-28		
		광고 기능 제공	SFR-S-29		
		ORM 서버 부 요구사항 (SFR-O)	User Type 정의		SFR-O-01
			Post Type 정의		SFR-O-02
			Like Type 정의		SFR-O-03
	Comment Type 정의		SFR-O-04		
	File Type 정의		SFR-O-05		
	Room Type 정의		SFR-O-06		
	Message Type 정의		SFR-O-07		

총 항목 수: 65



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 3 시스템 기능 상세 설계

#### 3.1 인터페이스 어플리케이션 부 시스템 설계 (SFR-I)

유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	인터페이스 어플리케이션 부 요구사항 (SFR-I)	로그인 기능 제공	SFR-I-01	14
		로그아웃 기능 제공	SFR-I-02	
		회원가입 기능 제공	SFR-I-03	
		전체 음악 노트 게시물 확인 기능	SFR-I-04	
		음악 노트 게시물 댓글, 좋아요, 파일, 이미지 보기 기능 제공	SFR-I-05	
		음악 노트 게시물 및 사용자 검색 기능 제공	SFR-I-06	
		개인정보 확인 기능 제공	SFR-I-07	
		탈퇴 기능 제공	SFR-I-08	
		개인정보 변경 기능 제공	SFR-I-09	
		좋아요 개수 높은 음악 노트 게시물 정보 제공	SFR-I-10	
		자신이 좋아하는 사용자의 음악 노트 게시물 확인 기능 제공	SFR-I-11	
		음악 노트 공유 기능 제공	SFR-I-12	
		음악 노트로 변환 기능 제공	SFR-I-13	
		페이지 렌더링 대기상태 시 보여지는 애니메이션 제공	SFR-I-14	

#### <SFR-I 요약표>

ID	SFR-I-01	중요도	중	담당자	이호찬
요구사항 명	로그인 기능 제공				
상세설명	<p>사용자는 로그인 인증에 성공하여 VRR 웹 플랫폼에 접속했을 때 로그인을 해서 음악 노트 순위를 볼 수 있는 메인 페이지로 넘어 갈 수 있으며 회원이 아닌 경우 하단에 따로 회원가입 버튼을 눌러 회원 가입 창으로 전환할 수 있다.</p> <p>▶ Email 입력시 server에 query를 요청하여 Email의 존재를 확인한다.</p> <p>▶ Email이 있을 경우: Secret Key 입력 창으로 Route한다.</p> <p>- Secret Key 인증에 성공할 경우 메인 창으로 넘어간다.</p> <p>- Secret Key 인증에 실패할 경우 Toast(알림 팝업)을 띄운다.</p> <p>Email이 없을 경우: Toast(알림 팝업)을 띄운다.</p> <p>사용자는 페이스북, github, google 인증 정보를 통해 로그인 할 수 있다.</p> <p>▶ Server에 request 보낸다.</p> <p>인증 완료될 경우: 메인 창으로 넘어간다.</p> <p>인증 완료되지 않을 경우: React-toastify[SSR-14]을 띄운다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-I-02	중요도	중	담당자	이호찬
요구사항 명	로그아웃 기능 제공				
상세설명	<p>사용자는 로그아웃 버튼을 누르면 로그아웃을 할 수 있다.</p> <ul style="list-style-type: none"> <li>▶ 사용자는 로그아웃 버튼을 누른다.</li> <li>▶ Local Storage의 저장돼있는 JWT(JSON Web Token)을 삭제한다.</li> <li>▶ 사용자를 "/"로 redirection 시킨다.</li> </ul>				

ID	SFR-I-03	중요도	중	담당자	이호찬
요구사항 명	회원가입 기능 제공				
상세설명	<p>사용자는 회원가입 창을 통해 회원 가입을 할 수 있으며 이메일, 사용자 별칭이 중복되지 않아야 한다. 하단의 로그인 버튼을 눌러 로그인 화면으로 전환할 수 있다.</p> <ul style="list-style-type: none"> <li>▶ 이메일, 사용자 이름(성, 이름), 별명, 비밀번호를 입력할 수 있는 input form이 있어야 한다.</li> <li>▶ 버튼을 누르면 서버에 회원 생성을 요청한다.</li> </ul> <p>요청 성공할 경우: 사용자 가입 완료 React-toastify[SSR-14]를 띄운다.</p> <p>요청 실패할 경우: 오류내용 React-toastify[SSR-14]를 띄운다.</p>				

ID	SFR-I-04	중요도	상	담당자	이호찬
요구사항 명	전체 음악 노트 게시물 확인 기능				
상세설명	<p>로그인 후 첫 사용자 메인 화면이며 전체 음악 노트 게시물 확인이 가능하다.</p> <ul style="list-style-type: none"> <li>▶ 전체 DB에 저장되어있는 음악 노트들이 등록 순으로 상위 10개만 보여준다.</li> <li>▶ 이후 사용자가 원하는 액션을 취할 경우 5개씩 더 가져온다.</li> </ul> <p>보여주는 최소 요구 정보:</p> <pre>seeFeed 정보를 가져올 keyword: {   id   location   caption   files {     id     url   }   likeCount   createdAt }</pre>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-I-05	중요도	상	담당자	이호찬
요구사항 명	음악 노트 게시물 댓글, 좋아요, 파일, 이미지 보기 기능 제공				
상세설명	<p>[SFR-I-004] 에서 게시물 하나를 선택했을 때, 상세적으로 보여준다.</p> <p>▶ 게시물 관련 영역을 표시하면 한 게시물에 대한 상세 정보를 볼 수 있다.</p> <p>보여주는 최소 요구 정보:</p> <pre>seeFeed 정보를 가져올 keyword: {   id   location   caption   user {     id     avatar     username   }   files {     id     url   }   isLiked   comments {     id     text     user {       id       username     }   }   likeCount   createdAt</pre>				

ID	SFR-I-06	중요도	상	담당자	이호찬
요구사항 명	음악 노트 게시물 및 사용자 검색 기능 제공				
상세설명	<p>Search input form을 통해 음악 노트 게시물 정보나 사용자를 검색할 수 있다.</p> <p>▶ 검색을 하면 UI Grid를 두부분으로 나눠서 검색에 대한 상단에는 노트 게시물 정보, 하단에는 사용자 정보를 보여준다.</p> <p>▶ 검색 결과가 없을 경우 '검색 결과 없음' 을 출력한다.</p> <p>보여주는 최소 요구 정보:</p> <pre>searchPost(요청할 keyword: 검색내용) 정보를 가져올 keyword: {   id   files {     url</pre>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

	<pre>         }         likeCount         commentCount       }       searchUser(요청할 keyword: 검색내용)         정보를 가져올 keyword: {           id           avatar           username           isFollowing           isSelf         }       }     }   } } </pre>
--	--

ID	SFR-I-07	중요도	중	담당자	이호찬
요구사항 명	개인정보 확인 기능 제공				
상세설명	<p>사용자의 개인 정보를 볼 수 있다. 보여주는 최소 요구 정보:</p> <pre> seeUser(요청할 keyword: username)   정보를 가져올 keyword: {     id     avatar     username     fullName     isFollowing     isSelf     bio     followingCount     followersCount     postsCount     posts {       id       files {         url       }       likeCount       commentCount     }   } } </pre>				

ID	SFR-I-08	중요도	중	담당자	이호찬
요구사항 명	탈퇴 기능 제공				
상세설명	[SFR-I-07]에서 탈퇴 버튼을 통해서 탈퇴 요청을 보낼 수 있다.				

ID	SFR-I-09	중요도	중	담당자	이호찬
요구사항 명	개인정보 변경 기능 제공				
상세설명	[SFR-I-07]에서 수정 버튼을 누르면 사용자 정보들이 Input form 작성하여 Query를				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

요청할 수 있다.

▶ Input placeholder는 사용자 정보로 미리 채워져 있다.

보여주는 최소 요구 정보:

```
seeUser( 변환 요청할 keyword:
  id
  avatar
  username
  fullName
  isFollowing
  isSelf
  bio
  followingCount
  followersCount
  postsCount
  posts {
    id
    files {
      url
    }
    likeCount
    commentCount
  }
) 정보를 가져올 keyword: {
  id
  avatar
  username
  fullName
  isFollowing
  isSelf
  bio
  followingCount
  followersCount
  postsCount
  posts {
    id
    files {
      url
    }
    likeCount
    commentCount
  }
}
```

ID	SFR-I-10	중요도	상	담당자	이호찬
요구사항 명	좋아요 개수 높은 음악 노트 게시물 정보 제공				
상세설명	[SFR-I-04]에서 좋아요 개수로 정렬된 랭킹 순서를 보여준다.				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-I-11	중요도	상	담당자	이호찬
요구사항 명	자신이 좋아하는 사용자의 음악 노트 게시물 확인 기능 제공				
상세설명	<p>자신이 follow한 다른 사용자의 음악 노트를 볼 수 있다.</p> <p>보여주는 최소 요구 정보:</p> <pre> 정보를 가져올 keyword: {   isFollowing {     id     avatar     username     fullName     isFollowing     isSelf     bio     followingCount     followersCount     postsCount     posts {       id       files {         url       }       likeCount       commentCount     }   } } </pre>				

ID	SFR-I-12	중요도	상	담당자	이호찬
요구사항 명	음악 노트 공유 기능 제공				
상세설명	<p>자신의 게시물(음악 노트)을 공유/비공유 설정을 할 수 있다.</p> <p>▶ input form을 통해 공유/비공유 정보에 대한 변경을 요청한다.</p>				

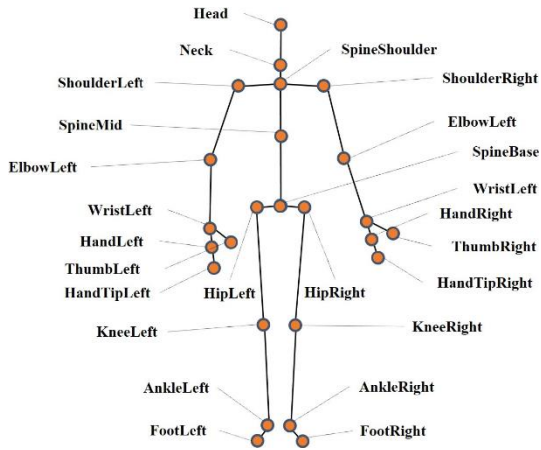
ID	SFR-I-13	중요도	상	담당자	이호찬
요구사항 명	음악 노트로 변환 기능 제공				
상세설명	<p>사용자가 음악을 업로드하면 완성된 음악 노트를 DownLoad할 수 있다.</p> <p>▶ Multer[SSR-22]을 사용하여 파일을 업로드 다운로드한다.</p>				

ID	SFR-I-14	중요도	중	담당자	이호찬
요구사항 명	페이지 렌더링 대기상태 시 보여지는 애니메이션 제공				
상세설명	페이지 렌더링 대기 상태 시 보여지는 애니메이션 제공한다.				

### 3.2 인터페이스 게임 어플리케이션 부 시스템 기능 요구사항 (SFR-GI)

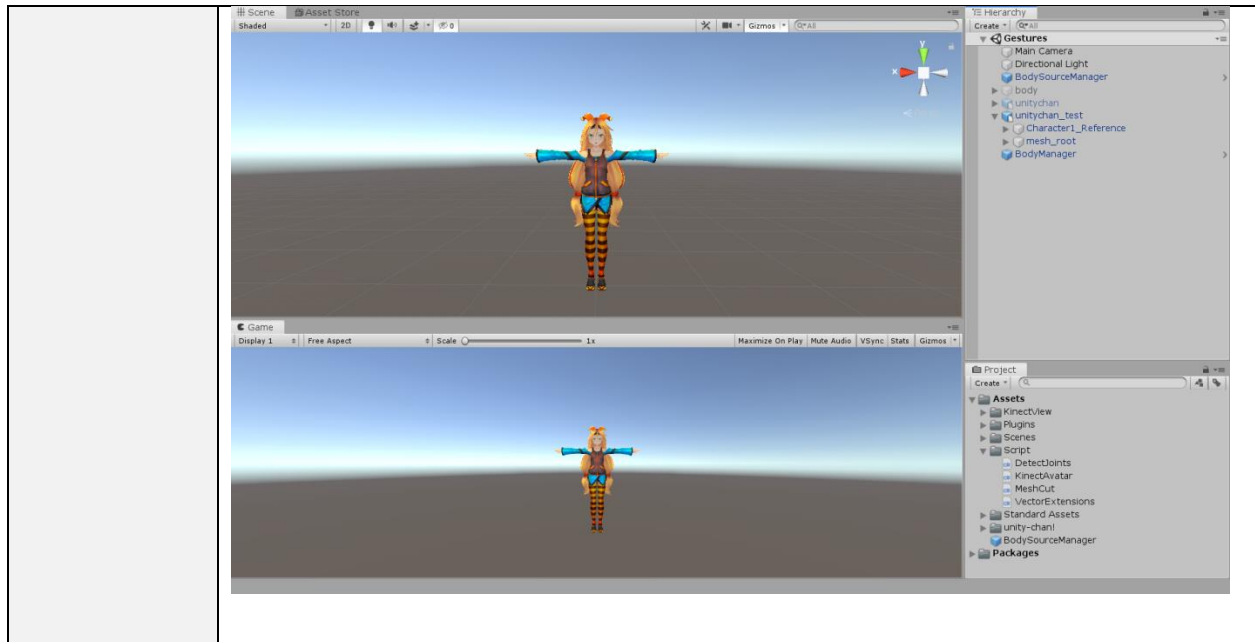
유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	인터페이스 게임 어플리케이션 부 요구사항 (SFR-GI)	키넥트 모션 인식을 이용한 인터페이스조작	SFR-GI-01	4
		플레이 모드선택(커스텀, 2인)	SFR-GI-02	
		BeatNote들 생성과 이동	SFR-GI-03	
		플레이한 결과 종합 정보 표시	SFR-GI-04	

#### <SFR-GI 요약표>

ID	SFR-GI-01	중요도	중	담당자	김세진
요구사항 명	Kinect 모션 인식을 이용한 인터페이스 조작				
상세설명	 <p>Kinect 인식 골격</p> <p>kinect asset을 사용하여 인체의 25개의 골격 구조를 받아온다. 받아온 골격 구조와 해당 골격의 위치정보를 받아와서 Unity 가상공간에 배치한다. 25개의 골격 중에서 hand-tip-right와 hand-tip-left를 이용하여 가상공간의 인터페이스를 조작할 수 있다. mesh collider를 이용하면 두 object가 충돌한 지점은 contactPoint함수로 위치 좌표를 받아와서 사용할 수 있다.</p>				

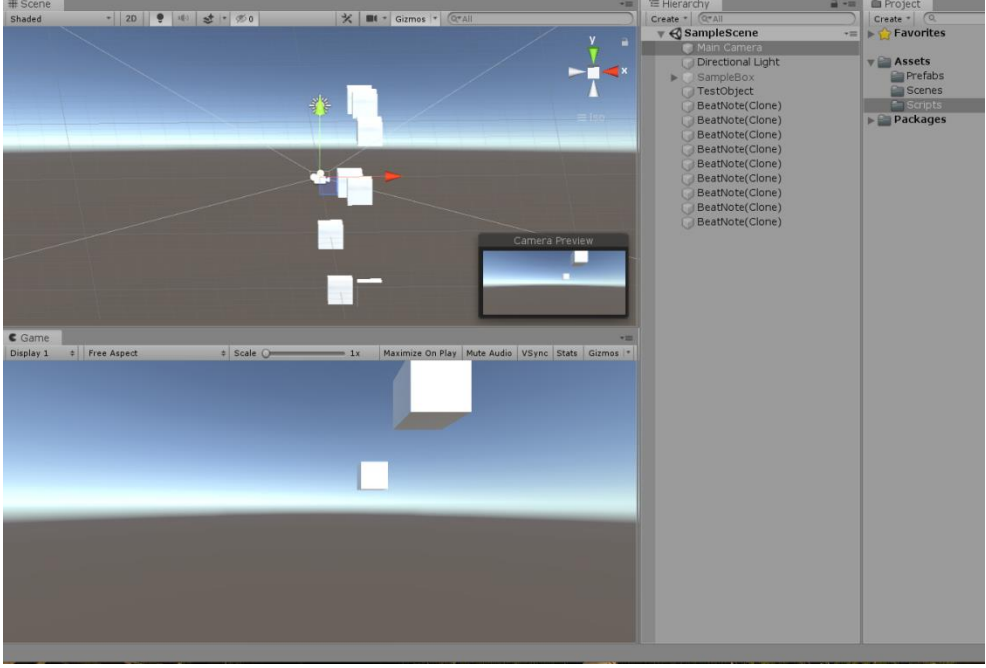


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



ID	SFR-GI-02	중요도	중	담당자	김세진
요구사항 명	플레이 모드선택(커스텀, 2인)				
상세설명	<p>게임에는 커스텀모드와 멀티플레이 모드 두 개의 모드가 있다. 처음 커스텀 모드를 선택할 경우 자신이 받은 노래와 웹에서 변환하여 만든 BeatNoteMap들을 적용하여 게임을 플레이할 수 있다. 반면에 멀티플레이 모드는 두명의 게임 플레이어가 동시에 게임을 진행하며 서로 경쟁하면서 플레이할 수 있다.</p> <p>Unity에서 MMORPG나 Survival 장르의 게임을 구현할 때 필요한 멀티 플레이어 기능을 Unity Asset 중 Mirror Asset을 활용해 2인이 동시에 같은 게임을 진행하며 경쟁할 수 있도록 설계할 수 있다.</p> <div style="display: flex; align-items: center;">   </div>  <p style="text-align: right;">유니티 mirror test</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-GI-03	중요도	중	담당자	김세진
요구사항 명	BeatNote들 생성과 이동				
상세설명	<p>BeatNoteMap 데이터를 통해 받은 BeatNote들의 정보를 List에 저장한다. 그리고 update()함수(매 frame마다 실행되는 함수)에서 각 BeatNote들의 첫 번째 인자 값인 time을 time.time(프로그램 시작부터 현재까지의 시간)과 비교하면서 BeatNote를 생성하고 생성위치는 BeatNote들의 두 번째, 세 번째인자인 lineIndex와 lineLayer를 통해 Transform.Translate(x,y,z) 함수를 통하여 배치한다. 그리고 생성된 GameObject들을 새로운 List에 저장하고 반복문을 통하여 List의 모든 GameObject들의 위치를 변화시켜주면 BeatNote들이 Player방향으로 이동한다.</p> 				

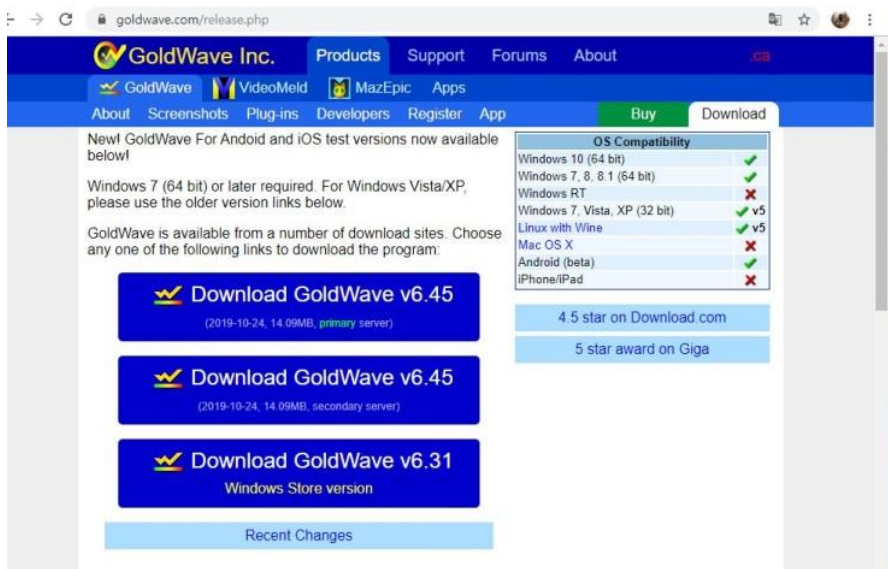
ID	SFR-GI-04	중요도	중	담당자	김세진
요구사항 명	플레이한 결과 종합 정보 표시				
상세설명	<p>게임을 플레이한 이후에 플레이어의 정확성 및 콤보 등 결과들을 표시해 준다. 점수 판정 범위와 콤보를 적용한 점수를 측정해 Unity UI를 통해 VR 화면에 출력해주며 2인 플레이를 한 경우 상대방과 점수를 비교해 경쟁할 수 있도록 설계한다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

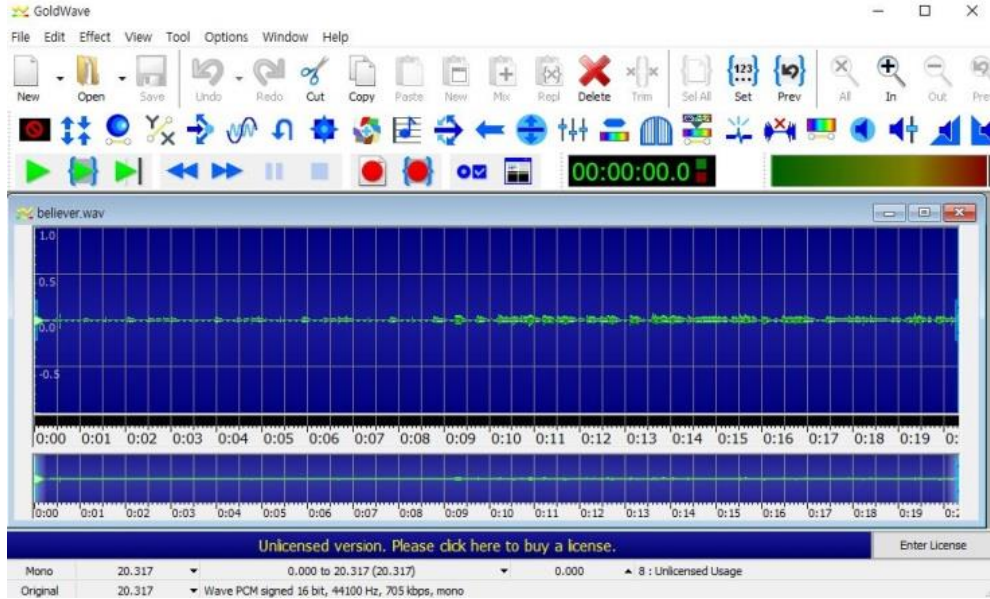
### 3.3 서비스 어플리케이션 부 시스템 기능 요구사항 (SFR-V)

유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	서비스 어플리케이션 부 요구사항 (SFR-V)	전처리1 – 음악파일 변환	SFR-V-01	11
		전처리2 – STFT적용 및 스케일링	SFR-V-02	
		파형 feature 추출	SFR-V-03	
		손실 함수 정의_ CNN, RNN	SFR-V-04	
		활성화 함수 정의	SFR-V-05	
		최적화 알고리즘	SFR-V-06	
		Dataset mirroring	SFR-V-07	
		CNN 모델	SFR-V-08	
		LSTM(RNN) 모델	SFR-V-09	
		LSTM input 및 output	SFR-V-10	
		Beat Note 점수 판정 범위	SFR-V-11	

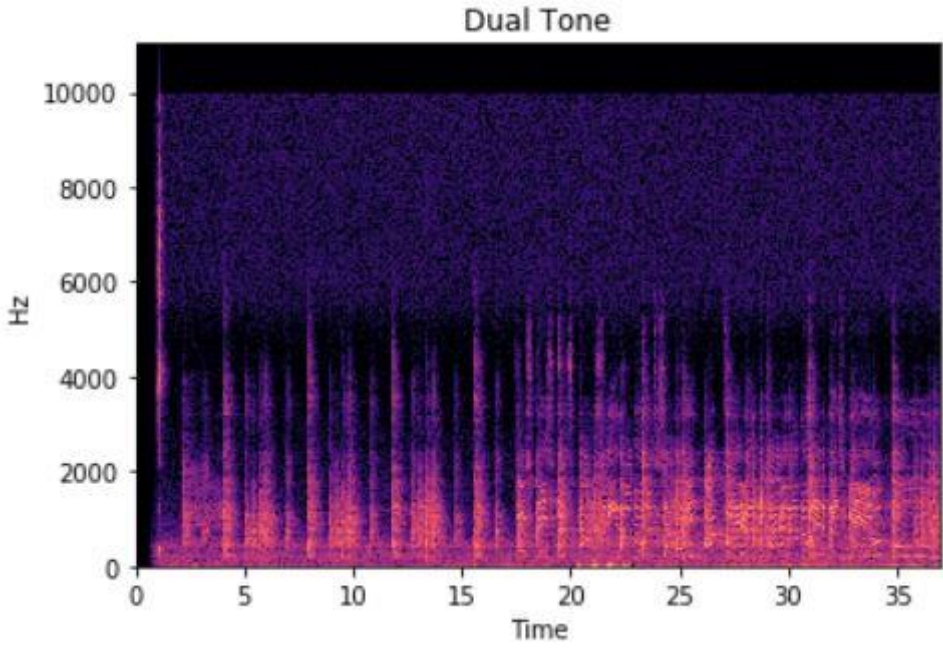
#### <SFR-V 요약표>

ID	SFR-V-01	중요도	중	담당자	조동철
요구사항 명	전처리1 – 음악파일 변환				
상세설명	<p>음악파일을 스테레오 pcm 오디오로 디코딩 하고 두 채널을 평균화하여 단조로운 표현을 만든다. Pcm을 지원하는 Wav 파일형태로 인코딩, 디코딩한 후 이 때 2라인(채널)로 나오는 파형의 평균화 하여 매끄러운 파형을 획득한다.</p> <p>Gold wave 프로그램을 이용하여 44.1khz의 wav음악의 인코딩 및 디코딩에 이용한다.</p> 				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

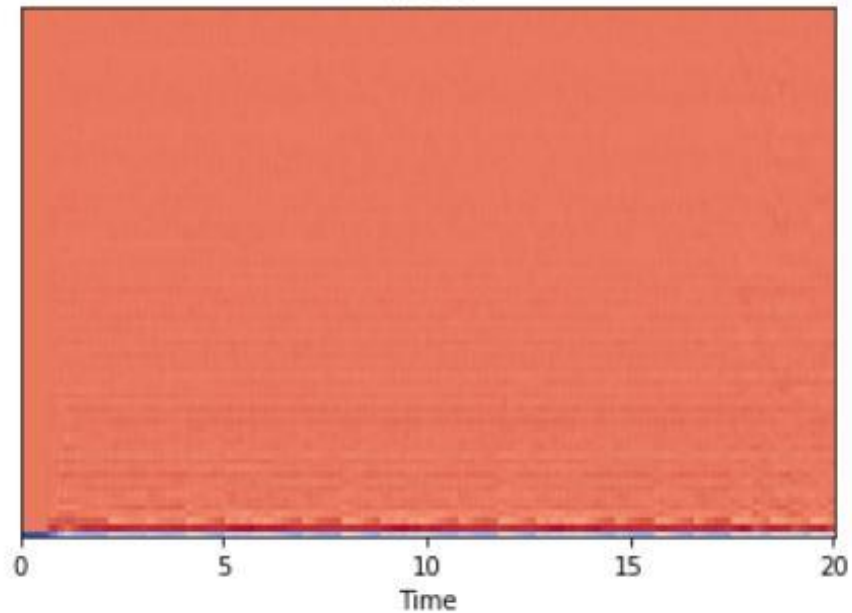


받은 음악파일(mono)

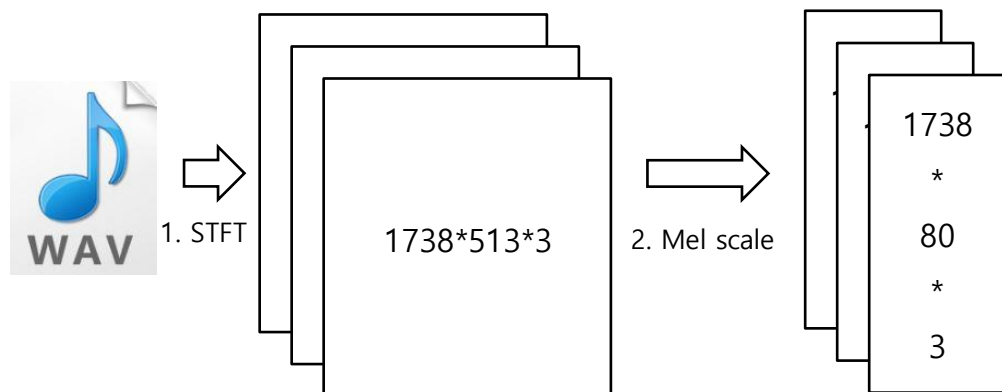
ID	SFR-V-02	중요도	상	담당자	조동철
요구사항 명	전처리2 - STFT적용 및 스케일링				
상세설명	<p>23, 46, 93ms의 윈도우 길이와 10ms 스텝을 이용하여 다중시간, 단기간 푸리에 변환(STFT)를 계산한다. 단기간 윈도우는 음정 간의 관계 맥락을 제공하고 큰 윈도우는 멜로디와 리듬 같은 높은 수준의 맥락을 제공한다.</p> <p>STFT를 적용한 후 2또는 3차원의 스펙토그램을 획득한다.</p> <div>  <p>stft로 얻은 스펙토그램</p> </div>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

얻게 된 스펙트로그램의 데이터의 양이 많기 때문에 mel scale을 적용해 80 주파수 대역으로 줄이고 로그 스케일링 해준다.



mel scale 후 그래프



1. 1738(time)\*513(feature)\*3(윈도우개수) (stft적용)
2. 1738(time)\*80(feature)\*3(윈도우개수) (mel scale적용)

-시간을 20초로 했을 때 1738개의 행이 생성됩니다



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-V-03	중요도	중	담당자	조동철
요구사항 명	파형 feature 추출				
상세설명	<p>CNN에서 이용할 수 있도록 파형에서 특색 값을 추출한다.</p> <p>구간 길이와 파형의 크기에 대한 feature를 추출하고</p> <p>시간, 좌우 위치, 상하 위치, 노트 색깔, 자르는 방향의 정보를 가진 노트와 매칭하여 데이터를 얻는다. (이 때 사용하는 노트는 one-hot 인코딩을 하여 사용한다.)</p> <p>이 후 CNN에서 feature 추출을 잘하는 layer를 테스트한 후 적용하고 리듬과 음정의 전후 관계를 파악하기 위해 RNN을 적용한다.</p>				

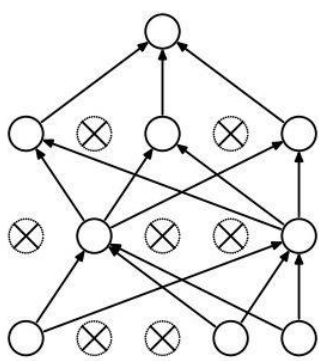
ID	SFR-V-04	중요도	중	담당자	조동철
요구사항 명	손실 함수 정의_ CNN, RNN				
상세설명	<p>전체 유닛에서 다중분류, 정확도 향상을 위해 크로스 엔트로피 손실함수를 이용한다.</p> $E = - \sum_k t_k \log y_k$				

ID	SFR-V-05	중요도	중	담당자	조동철
요구사항 명	활성화 함수 정의				
상세설명	<p>CNN에서 렐루 함수, 소프트맥스 함수를 이용하여 3차원 커널 값을 채널과 주파수 축의 2차원 값으로 바꾼다. 그 후 RNN에서 활성화 함수로 하이퍼볼릭 탄젠트를 이용하여 출력층에 들어갈 값을 구한다. 비선형함수를 활성화 함수로 이용하여 층을 구성한다.</p> $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ $\tanh'(x) = 1 - \tanh^2(x)$ <p>하이퍼 볼릭 탄젠트 공식</p> $f(x) = \max(0, x)$ <p>렐루 함수 공식</p> $f(\vec{x})_i = \frac{e^{x_i}}{\sum_{k=1}^K e^{x_k}} \quad \text{for } i = 1, \dots, K$ <p>소프트 맥스 함수 공식</p>				



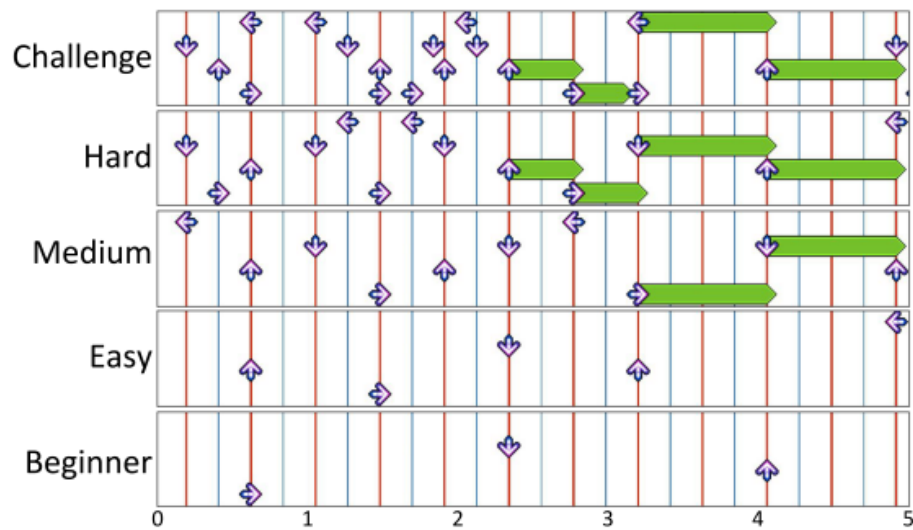


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-V-06	중요도	상	담당자	조동철
요구사항 명	최적화 알고리즘				
상세설명	<p>Overfitting을 해결하기 위해 dropout을 사용한다. 기본적으로 드롭아웃의 비율만큼 유닛 및 뉴런에서의 출력을 높이는 방법을 이용하거나 가중치 규제를 이용하여 해결할 것이다</p> <p>Underfitting을 해결하기 위해 튜닝과정을 거칠 것이고 여기서 적절한 함수를 찾고, 손실값 그래프를 시뮬레이션하여 최대 퍼포먼스 지점의 에포크 지점을 찾고 훈련의 조기 종료 등의 방법을 적용한다.</p>  <p>드롭아웃 방식</p> $\sum_{i=1}^n (y_i - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p \beta_j^2$ <p>L2 규제(Lidge regression) 공식</p>				

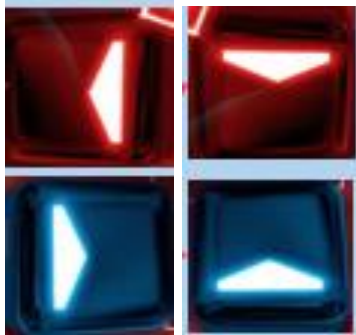
ID	SFR-V-07	중요도	중	담당자	문명기
요구사항 명	Dataset mirroring				
상세설명	<p>Chris Donahue, Zachary C. Lipton ,Julian McAuley Dance Dance Convolution. In ICML, 2017.</p> <p>에 따르면, 학습에 필요한 데이터양을 늘리기 위해 왼쪽/오른쪽, 위쪽/아래쪽을 미러링하여 데이터를 2배로 늘렸다. 이러한 dataset mirroring으로 성능이 상당히 향상되었다.</p> <p>예를 들어, 왼쪽은 -&gt; 오른쪽으로 바꾸고, 위쪽은 아래쪽으로 바꾸는 식으로 dataset mirroring 하여 같은 step 위치에 방향만 거꾸로 하여 새로운 데이터를 만들었다. 이는 level별로 (Beginner,Easy, Medium,Hard,Challenge) 각각 2배로 증가시켰다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



– Chris Donahue, Zachary C. Lipton, Julian McAuley(2017)

따라서, 본 프로젝트에서도 부족한 dataset을 늘리기 위해 동일한 왼쪽, 오른쪽, 위쪽, 아래쪽을 가진 beat note 에도 mirroring을 적용하여 dataset을 2배로 늘릴 것이다.



<왼쪽, 오른쪽, 위쪽, 아래쪽 예시>



Dataset mirroring의 예시: <아래쪽, 아래쪽> 이 <위쪽, 위쪽>으로 변환된다.

구현 방법:

(색깔, 행 위치, 열 위치, 자르는 방향)으로 구성된 beat map의 txt파일에서



아래 예시와 같이 마지막 인덱스인 '자르는 방향' 인덱스 값만 0에서 1로 바꿔준다.

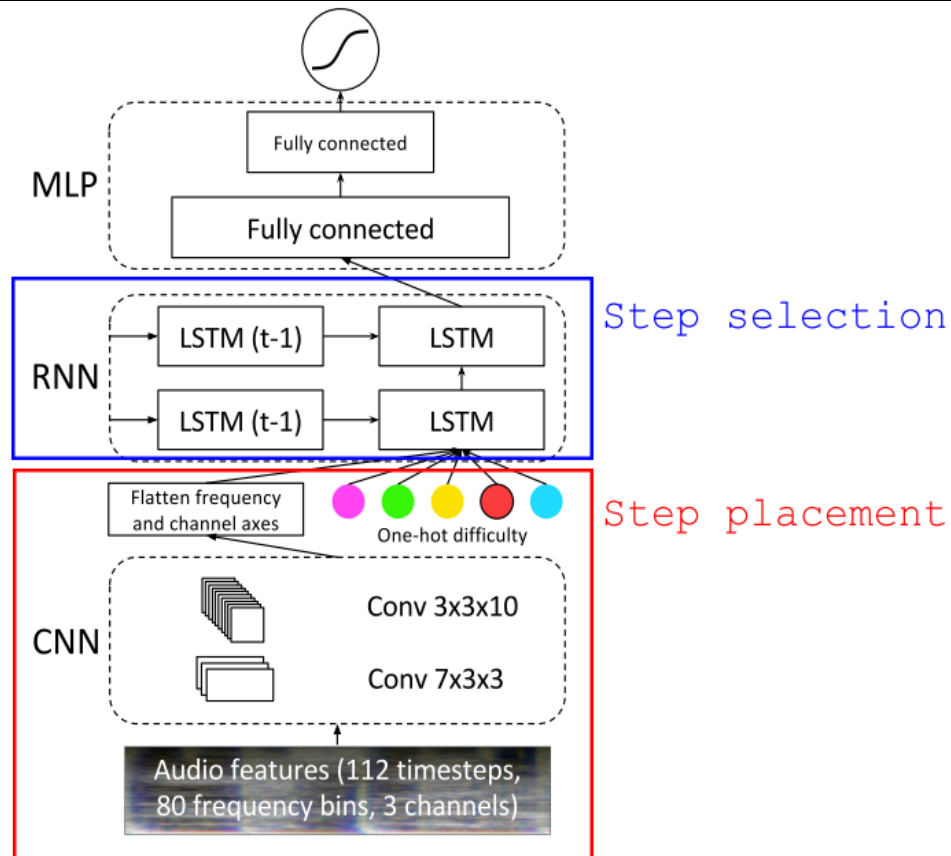




상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-V-08	중요도	중	담당자	문명기																												
요구사항 명	CNN 모델																																
상세설명	<p>&lt;CNN 모델 선택 이유&gt;</p> <p>Schlüter, Jan and Böck, Sebastian. Improved musical on-set detection with convolutional neural networks. In ICASSP, 2014.</p> <p>에 따르면, On-set detection은 음이 변하는 특징구간을 감지하는 것으로, 오디오파일에 맞춰 비트 매핑을 하기 위해 필요하다. 전처리 된 오디오파일에서 on-set detection을 하는데, CNN을 사용한 모델이 가장 높은 성능을 보였다.</p>																																
	<table><thead><tr><th></th><th>Precision</th><th>Recall</th><th>F-score</th></tr></thead><tbody><tr><td>RNN [10, 5]</td><td>0.892</td><td>0.855</td><td>0.873</td></tr><tr><td>CNN [1]</td><td>0.905</td><td>0.866</td><td>0.885</td></tr><tr><td>+ Dropout</td><td>0.909</td><td>0.871</td><td>0.890</td></tr><tr><td>+ Fuzziness</td><td>0.914</td><td>0.885</td><td>0.899</td></tr><tr><td>+ ReLU</td><td>0.917</td><td>0.889</td><td>0.903</td></tr><tr><td>SuperFlux [5]</td><td>0.883</td><td>0.793</td><td>0.836</td></tr></tbody></table>						Precision	Recall	F-score	RNN [10, 5]	0.892	0.855	0.873	CNN [1]	0.905	0.866	0.885	+ Dropout	0.909	0.871	0.890	+ Fuzziness	0.914	0.885	0.899	+ ReLU	0.917	0.889	0.903	SuperFlux [5]	0.883	0.793	0.836
		Precision	Recall	F-score																													
	RNN [10, 5]	0.892	0.855	0.873																													
	CNN [1]	0.905	0.866	0.885																													
	+ Dropout	0.909	0.871	0.890																													
	+ Fuzziness	0.914	0.885	0.899																													
	+ ReLU	0.917	0.889	0.903																													
	SuperFlux [5]	0.883	0.793	0.836																													
	<p>- Schlüter &amp; Böck(2014)</p>																																
<p>리듬에 따라 배치된 박스를 'beat note'라 하는데, 음악마다 'beat note'의 지도를 만들 때 필요한 기능을 2가지로 나눴다.</p> <p>Step placement 와 step selection 이다.</p> <p>Step placement 는 언제 beat note 가 올지 시간을 정하는 것이고,</p> <p>Step selection은 어떤 방향의 beat note 가 올지 방향을 결정하는 것이다.</p>																																	
<p>아래 그림 &lt;C-LSTM의 전체 구조&gt;과 같이</p> <p>Step placement 는 CNN 으로 구성하고, step selection 은 LSTM 모델로 구현한다.</p> <p>CNN 과 LSTM은 서로 연결된 형태인데, CNN의 output이 LSTM의 input으로 구성된다.</p>																																	

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



#### <C-LSTM의 전체 구조>

Step placement을 위한 CNN의 input, layer 구성, output 은 다음과 같이 구성된다.

Input: n개의 timesteps x 80개의 주파수 x 3개의 채널 100 개의 timestep 만큼 한꺼번에 CNN에 집어넣는다.

(-Timesteps는 음원 파일의 길이에 따라 다르다. 예를 들어, 200초의 음원파일을 20초단위로 쪼개서 10개의 토큰으로 나눈다. 10개의 토큰은 순차적으로 1개씩 들어간다. 1개의 토큰을 timestep으로 나타내면, 대략, 1738개가 된다. n=1738 timesteps가 된다.

- 3개의 채널은 학습단위가 된다. 23 은 20, 3 / 46은 20,20,6 / 93 은 20,20,20,20,13 단위로 쪼개져서 학습하려 했지만, 단위가 20이므로 그 안에서 나머지 값인 3,6,13 을 쪼개기 어려우므로, 23 단위로 쪼갬다.

23은 23, 46 은 23,23 93 은 23 23 23 23 단위로 쪼개어 학습하도록 한다.

따라서, 23은 23 토큰이 1번 ,46은 2번 , 93은 4번 연속으로 학습한 뒤

lstm의 parameter값을 업데이트한다.)

layer1은 7 wide의 시간 x 3 wide의 주파수x3개의 channel = 7x3의 filter kernel 10개로 구성되고,

layer2는 3x3x10의 filter kernel 20개로 구성된다.

매 filter kernel 이후 3x3 max pooling을 거친다.

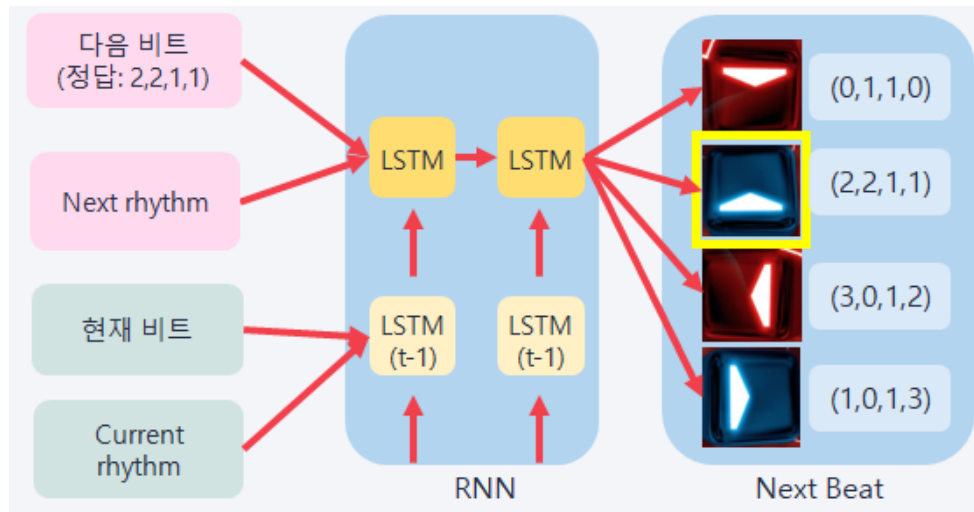


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

	<pre> # input place holders X = tf.placeholder(tf.float32, [None, 100, 80, 3]) Y = tf.placeholder(tf.float32, [None, 100])  # Layer 1 input shape=(?, 100, 80, 3) W1 = tf.Variable(tf.random_normal([7, 3, 3, 10], stddev=0.01)) L1 = tf.nn.conv2d(X, W1, strides=[1, 1, 1, 1], padding='SAME') # Conv (?, 100, 78, 10) L1 = tf.nn.relu(L1) L1 = tf.nn.max_pool(L1, ksize=[1, 1, 3, 1], strides=[1, 1, 3, 1], padding='VALID') # Pool (?, 100, 26, 10)  # Layer 2 input shape=(?, 100, 26, 10) W2 = tf.Variable(tf.random_normal([3, 3, 10, 20], stddev=0.01)) L2 = tf.nn.conv2d(L1, W2, strides=[1, 1, 1, 1], padding='SAME') # Conv (?, 100, 24, 20) L2 = tf.nn.relu(L2) L2 = tf.nn.max_pool(L2, ksize=[1, 1, 3, 1], strides=[1, 1, 3, 1], padding='VALID') # Pool (?, 100, 8, 20)  # Flattening (except for time-axis) L2_flat = tf.reshape(L2, [-1, 100, 8 * 20]) print('L2_flat after flattening: {}'.format(L2_flat)) </pre> <p>Output: 하나의 sigmoid function으로 구성되어 구간에 beat note가 들어가는지 아닌지를 판별한 결과값이 산출된다.</p> <p>이 결과값은 LSTM에 넣기 위해 시간 축은 보존한 상태에서 flattening을 한다.</p> <p>역할: max pooling 은 frequency 에서만 실행한다.</p>
--	--

ID	SFR-V-09	중요도	중	담당자	문명기
요구사항 명	LSTM(RNN) 모델				
상세설명	<p>&lt;RNN 모델 선택이유&gt;</p> <p>Chris Donahue, Zachary C. Lipton, Julian McAuley Dance Dance Convolution. In ICML, 2017.</p> <p>에 따르면, CNN으로만 구현한 모델은 오디오파일의 시간에 따른 리듬의 변화 패턴인 rhythmic feats를 반영하는데 한계가 있음을 지적하여 CNN과 LSTM(RNN) 을 섞은 C-LSTM 모델을 제안한다.</p> <p>-Chris Donahue, Zachary C. Lipton, Julian McAuley (2017)</p> <p>LSTM은 전처리 과정에서 얻은 93ms(Rhythmic feats), 46ms(Rhythmic feats &amp; Step feats), 23(Step feats)의 특징들을 전부 고려해서 다음 비트를 매핑하기 위해 필요하다.</p> <p>Beat note mapping 의 2가지 과정 Step placement 와 Step selection 중 LSTM 은 Step Selection을 위해 사용한다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



LSTM의 역할,구성, input, output 의 개념적 설계는 다음과 같다. 상세설계는 SFR-V-08-2에 기술하였다.

역할: LSTM은 다음 노트가 어떤 방향(상하좌우)의 노트가 올지를 정하는 것이다.

구성: 128개의 node 로 이뤄진 2 layer LSTM

Input : CNN의 output인 100개 의 time axis output을 LSTM 의 각 time step input 으로 넣는다.

학습 방식: 매 time step 마다 다음 step이 어떤 방향의 노트인지를 예측한다.

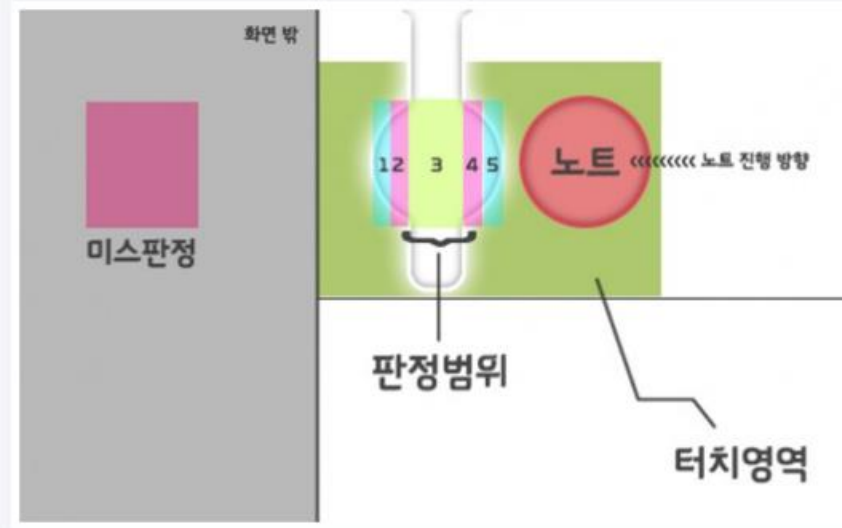
LSTM 의 학습방식은 Supervised learning 방식으로 학습 시에는 미리 매핑 되어있는 beat saber의 beat note map을 답안지로 하여 답을 예측하고 예측 값을 조절해 나간다.

Output: 예측한 방향(왼쪽,오른쪽,아래쪽,위쪽 중 하나)의 노트

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-V-10	중요도	중	담당자	문명기
요구사항 명	LSTM input 및 output				
상세설명	<p>LSTM 모델의 추가적인 musical context 을 주기 위해 rhythmic feature를 추가적인 input으로 넣어준다. Step placement 사이의 간격이 균일(uniform)하지 않기 때문에 다음 3가지 요소를 input에 추가해준다.</p> <p><math>\Delta</math>-time : 지난번 note 으로부터의 시간 / 다음 note 까지의 시간 -&gt; 2 개의 feature  <math>\Delta</math>-beat : 지난번 note 으로부터의 비트 갯수 / 다음 note 까지의 비트 갯수 -&gt; 2 개의 feature  beat phase : 이 note 에서 가장 가까운 note (각 beat 를 16 분 음표단위로 나눈 note subdivision 들 중)  input 은 총 길이 56 벡터가 들어가게 된다.</p> <p>이번 노트의 12 position * 4 direction 의 multi class encoding (48 개)</p> <div data-bbox="395 891 1252 1176"> </div> <p>&lt;4x3 의 12 개 position&gt; 과 &lt;왼,오른,위,아래 의 4 direction&gt;</p> <p>전 이전 노트와의 시간차 <math>\Delta</math>-time (1 개) + 다음 노트와의 시간차 <math>\Delta</math>-time (1 개)  지난번 노트로부터의 비트 갯수 <math>\Delta</math>-beat (1 개) + 다음 노트 까지의 비트 갯수 <math>\Delta</math>-beat (1 개)  노트가 속한 4 분음표 비트내에 있는 4 개의 16 분음표 중 가장 가까운 비트 phase 표시 (4 개)</p> <p>따라서, <math>48 \times 4 \times 4 = 768</math> 개의 가능한 노트 조합이 나온다.</p> <p>결과적으로, Output 은 길이 768 의 벡터가 된다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-V-11	중요도	상	담당자	박영준
요구사항 명	Beat Note 점수 판정 범위				
상세설명	 <p>리듬게임에서 중요하게 작용하는 note의 판정범위 알고리즘을 설계해 Unity 환경으로 VRR System에서 잘 구현하여 사용자가 VRR Game 서비스를 play하는데 재미를 느낄 수 있어야 한다.</p>				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 3.4 서버 부 시스템 기능 요구사항 (SFR-S)

유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	서버 부 요구사항 (SFR-S)	댓글 생성 기능 제공	SFR-S-01	29
		댓글 삭제 기능 제공	SFR-S-02	
		로그인 시도 시 정보 일치 여부 반환 기능 제공	SFR-S-03	
		좋아요 상태 변경 기능 제공	SFR-S-04	
		게시글 생성 기능 제공	SFR-S-05	
		게시글 수정 기능 제공	SFR-S-06	
		게시글 삭제 기능 제공	SFR-S-07	
		게시글 검색 기능 제공	SFR-S-08	
		전체 게시글 보기 기능 제공	SFR-S-09	
		내 게시글 보기 기능 제공	SFR-S-10	
		사용자 생성 기능 제공	SFR-S-11	
		모든 사용자 보기 기능 제공	SFR-S-12	
		사용자 정보 수정 기능 제공	SFR-S-13	
		사용자 팔로우 기능 제공	SFR-S-14	
		사용자 팔로우 취소 기능 제공	SFR-S-15	
		자기 자신 정보 보기 기능 제공	SFR-S-16	
		비밀 키 송신 기능 제공	SFR-S-17	
		사용자 검색 기능 제공	SFR-S-18	
		댓글 수 확인 기능 제공	SFR-S-19	
		게시글 수 확인 기능 제공	SFR-S-20	
		자신이 한 팔로우 수 확인 기능 제공	SFR-S-21	
		자신에게 팔로우 한 수 확인 기능 제공	SFR-S-22	
		대화 방 생성 기능 제공	SFR-S-23	
		대화 방 보기 기능 제공	SFR-S-24	
		대화 메시지 실시간 수신/송신 기능 제공	SFR-S-25	
		유효 JWT (JSON Web Token) 확인 기능 제공	SFR-S-26	
		Secret Key 메일 보내기 기능 제공	SFR-S-27	
		JWT(JSON Web Token) 생성 기능 제공	SFR-S-28	
		광고 기능 제공	SFR-S-29	

#### <SFR-S 요약표>



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-S-01	중요도	상	담당자	이호찬
요구사항 명	댓글 생성 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   addComment(text: String!, postId: String!): Comment! }</pre>				

ID	SFR-S-02	중요도	상	담당자	이호찬
요구사항 명	댓글 삭제 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   deleteComment(postId: String!): Boolean! }</pre>				

ID	SFR-S-03	중요도	중	담당자	이호찬
요구사항 명	로그인 시도 시 정보 일치 여부 변환 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   confirmSecret(secret: String!, email: String!): String! }</pre>				

ID	SFR-S-04	중요도	상	담당자	이호찬
요구사항 명	좋아요 상태 변경 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   toggleLike(postId: String!): Boolean! }</pre>				

ID	SFR-S-05	중요도	상	담당자	이호찬
요구사항 명	게시글 생성 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   upload(caption: String! Files: [String!]!): Post! }</pre>				

ID	SFR-S-06	중요도	상	담당자	이호찬
요구사항 명	게시글 수정 기능 제공				
상세설명	GraphQL Schema: <pre> enum ACTIONS {</pre>				





상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

	<pre> EDIT DELETE }  type Mutation{   editPost(id: String, caption:String, location: String,     action: ACTIONS!): Post! } </pre>
--	--

ID	SFR-S-07	중요도	상	담당자	이호찬
요구사항 명	게시글 삭제 기능 제공				
상세설명	GraphQL Schema: <pre> enum ACTIONS {   EDIT   DELETE }  type Mutation{   editPost(id: String, caption:String, location: String,     action: ACTIONS!): Post! } </pre>				

ID	SFR-S-08	중요도	상	담당자	이호찬
요구사항 명	게시글 검색 기능 제공				
상세설명	GraphQL Schema: <pre> type Query {   searchPost(term: String!): [Post!]! } </pre>				

ID	SFR-S-09	중요도	상	담당자	이호찬
요구사항 명	전체 게시글 보기 기능 제공				
상세설명	GraphQL Schema: <pre> type Query{   seeFeed: [Post!]! } </pre>				

ID	SFR-S-10	중요도	상	담당자	이호찬
요구사항 명	내 게시글 보기 기능 제공				
상세설명	GraphQL Schema:				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

	<pre> type Query {   seeFullPost(id: String!): Post! } </pre>
--	---

ID	SFR-S-11	중요도	중	담당자	이호찬
요구사항 명	사용자 생성 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   createAccount(     username: String!,     email: String!,     firstName: String,     lastName: String,     bio: String   ): Boolean! } </pre>				

ID	SFR-S-12	중요도	중	담당자	이호찬
요구사항 명	모든 사용자 보기 기능 제공				
상세설명	GraphQL Schema: <pre> type Query {   allUsers: [User!]! } </pre>				

ID	SFR-S-13	중요도	중	담당자	이호찬
요구사항 명	사용자 정보 수정 기능 제공				
상세설명	GraphQL Schema: <pre> type Mutation {   editUser(     username: String     email: String     firstName: String     lastName: String     bio: String     avatar: String   ): User! } </pre>				

ID	SFR-S-14	중요도	중	담당자	이호찬
----	----------	-----	---	-----	-----



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

요구사항 명	사용자 팔로우 기능 제공
상세설명	<p>GraphQL Schema:</p> <pre> type Mutation {   followUser(id: String!): Boolean }</pre>

ID	SFR-S-15	중요도	중	담당자	이호찬
요구사항 명	사용자 팔로우 취소 기능 제공				
상세설명	<p>GraphQL Schema:</p> <pre> type Mutation {   unfollowUser(id: String!): Boolean! }</pre>				

ID	SFR-S-16	중요도	중	담당자	이호찬
요구사항 명	자기 자신 정보 보기 기능 제공				
상세설명	<p>GraphQL Schema:</p> <pre> type Query {   me: User! }</pre>				

ID	SFR-S-17	중요도	중	담당자	이호찬
요구사항 명	비밀 키 송신 기능 제공				
상세설명	<p>GraphQL Schema:</p> <pre> type Mutation {   requestSecret(email: String!): Boolean! }</pre>				

ID	SFR-S-18	중요도	중	담당자	이호찬
요구사항 명	사용자 검색 기능 제공				
상세설명	<p>GraphQL Schema:</p> <pre> type Query {   searchUser(term: String!): [User!]! }</pre>				

ID	SFR-S-19	중요도	상	담당자	이호찬
요구사항 명	댓글 수 확인 기능 제공				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

상세설명	GraphQL Schema: <pre>type Post {   commentCount: Int! }</pre>
------	--

ID	SFR-S-20	중요도	상	담당자	이호찬
요구사항 명	게시글 수 확인 기능 제공				
상세설명	GraphQL Schema: <pre>type User {   postsCount: Int! }</pre>				

ID	SFR-S-21	중요도	하	담당자	이호찬
요구사항 명	자신이 한 팔로우 수 확인 기능 제공				
상세설명	GraphQL Schema <pre>type User {   followingCount: Int! }</pre>				

ID	SFR-S-22	중요도	하	담당자	이호찬
요구사항 명	자신에게 팔로우 한 수 확인 기능 제공				
상세설명	GraphQL Schema <pre>type User {   followersCount: Int! }</pre>				

ID	SFR-S-23	중요도	상	담당자	이호찬
요구사항 명	대화 방 생성 기능 제공				
상세설명	GraphQL Schema <pre>type Mutation {   createRooms(     participants: [User!]!     messages: [Message!]!     createdAt: String     updatedAt: String   ): Boolean! }</pre>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-S-24	중요도	상	담당자	이호찬
요구사항 명	대화 방 보기 기능 제공				
상세설명	GraphQL Schema <pre> type Query {   seeRoom(id: String!): Room! }</pre>				

ID	SFR-S-25	중요도	상	담당자	이호찬
요구사항 명	대화 메시지 실시간 수신/송신 기능 제공				
상세설명	GraphQL Schema <pre> type Subscription {   newMessage(roomId: String!): Message } type Mutation {   sendMessage(roomId: String, message: String!, toId: String): Message }</pre>				

ID	SFR-S-26	중요도	상	담당자	이호찬
요구사항 명	유효 JWT(JSON Web Token) 확인 기능 제공				
상세설명	JWT를 받으면 decode 하여 유효성을 확인한다.				

ID	SFR-S-27	중요도	상	담당자	이호찬
요구사항 명	Secret Key 메일 보내기 기능 제공				
상세설명	▶ 로그인 요청을 받는다. 로그인에 성공을 할 경우 Secret Key를 보낸다. 로그인에 실패할 경우 실패 응답을 한다. ▶ Secret Key는 랜덤 단어이다.				

ID	SFR-S-28	중요도	상	담당자	이호찬
요구사항 명	JWT(JSON Web Token) 생성 기능 제공				
상세설명	▶ JsonWebToken[SSR-021]을 통해 JWT를 생성한다.				

ID	SFR-S-29	중요도	상	담당자	이호찬
요구사항 명	광고 기능 제공				
상세설명	Google adsense로 광고를 제공한다. ▶ HTML 파일에 스크립트 파일만 포함하면 된다.				

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 3.5 ORM 서버 부 요구사항 (SFR-O)

유형	상세유형	요구사항 명	요구사항 ID	항목수
시스템 기능 요구사항	ORM 서버 부 요구사항 (SFR-O)	User Type 정의	SFR-O-01	7
		Post Type 정의	SFR-O-02	
		Like Type 정의	SFR-O-03	
		Comment Type 정의	SFR-O-04	
		File Type 정의	SFR-O-05	
		Room Type 정의	SFR-O-06	
		Message Type 정의	SFR-O-07	

#### <SFR-O 요약표>

ID	SFR-O-01	중요도	상	담당자	이호찬
요구사항 명	User Type 정의				
상세설명	<p>id, avatar, username, email, firstName, lastName, bio, followers, following, posts, likes, comments, rooms, loginSecret, createdAt, updatedAt의 Data들을 갖고며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>Avatar</b>의 Type은 String이며 default 값은 "<a href="https://icon-library.net/images/default-user-icon/default-user-icon-4.jpg">https://icon-library.net/images/default-user-icon/default-user-icon-4.jpg</a>"이다.</p> <p><b>Username</b>의 Type은 String이며 유일한 값을 갖고며 반드시 존재해야 한다.</p> <p><b>Email</b>의 Type은 String이며 유일한 값을 갖고며 반드시 존재해야 한다.</p> <p><b>firstName</b>의 Type은 String이며 default값은 ""이다.</p> <p><b>lastName</b>의 Type은 String이다.</p> <p><b>Bio</b>의 Type은 String이다.</p> <p><b>Followers</b>의 Type은 User 자체의 집합이다.</p> <p><b>Following</b>의 Type은 User 자체의 집합이다.</p> <p><b>Posts</b>의 Type은 Post 자체의 집합이다.</p> <p><b>Likes</b>의 Type은 like 자체의 집합이다.</p> <p><b>Comments</b>의 Type은 comment 자체의 집합이다.</p> <p><b>Rooms</b>의 Type은 Room 자체의 집합이다.</p> <p><b>loginSecret</b>의 Type은 String이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-O-02	중요도	상	담당자	이호찬
요구사항 명	Post Type 정의				
상세설명	<p>id, location, caption, user, files, likes, comments, createdAt, updatedAt의 Data들을 갖고며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>Location</b>의 Type은 String이다.</p> <p><b>Caption</b>의 Type은 String이며 반드시 존재해야 한다.</p> <p><b>User</b>의 Type은 User 자체의 집합이다.</p> <p><b>Files</b>의 Type은 file 자체의 집합이다.</p> <p><b>Likes</b>의 Type은 like 자체의 집합이다.</p> <p><b>Comments</b>의 Type은 comment 자체의 집합이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				

ID	SFR-O-03	중요도	상	담당자	이호찬
요구사항 명	Like Type 정의				
상세설명	<p>id, user, post의 Data들을 갖고며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>User</b>의 Type은 User 자체의 집합이다.</p> <p><b>Posts</b>의 Type은 Post 자체의 집합이다.</p>				

ID	SFR-O-04	중요도	상	담당자	이호찬
요구사항 명	Comment Type 정의				
상세설명	<p>id, text, user, post, createdAt, updatedAt의 Data들을 갖고며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>Text</b>의 Type은 String이며 반드시 존재해야 한다.</p> <p><b>User</b>의 Type은 User 자체의 집합이다.</p> <p><b>Posts</b>의 Type은 Post 자체의 집합이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ID	SFR-O-05	중요도	상	담당자	이호찬
요구사항 명	File Type 정의				
상세설명	<p>id, url, post, createdAt, updatedAt의 Data들을 갖으며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>url</b>의 Type은 String이며 반드시 존재해야 한다.</p> <p><b>Posts</b>의 Type은 Post 자체의 집합이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				

ID	SFR-O-06	중요도	상	담당자	이호찬
요구사항 명	Room Type 정의				
상세설명	<p>id, participants, messages, createdAt, updatedAt의 Data들을 갖으며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>Participants</b>의 Type은 User 자체의 집합이다.</p> <p><b>Messages</b>의 Type은 Message 자체의 집합이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				

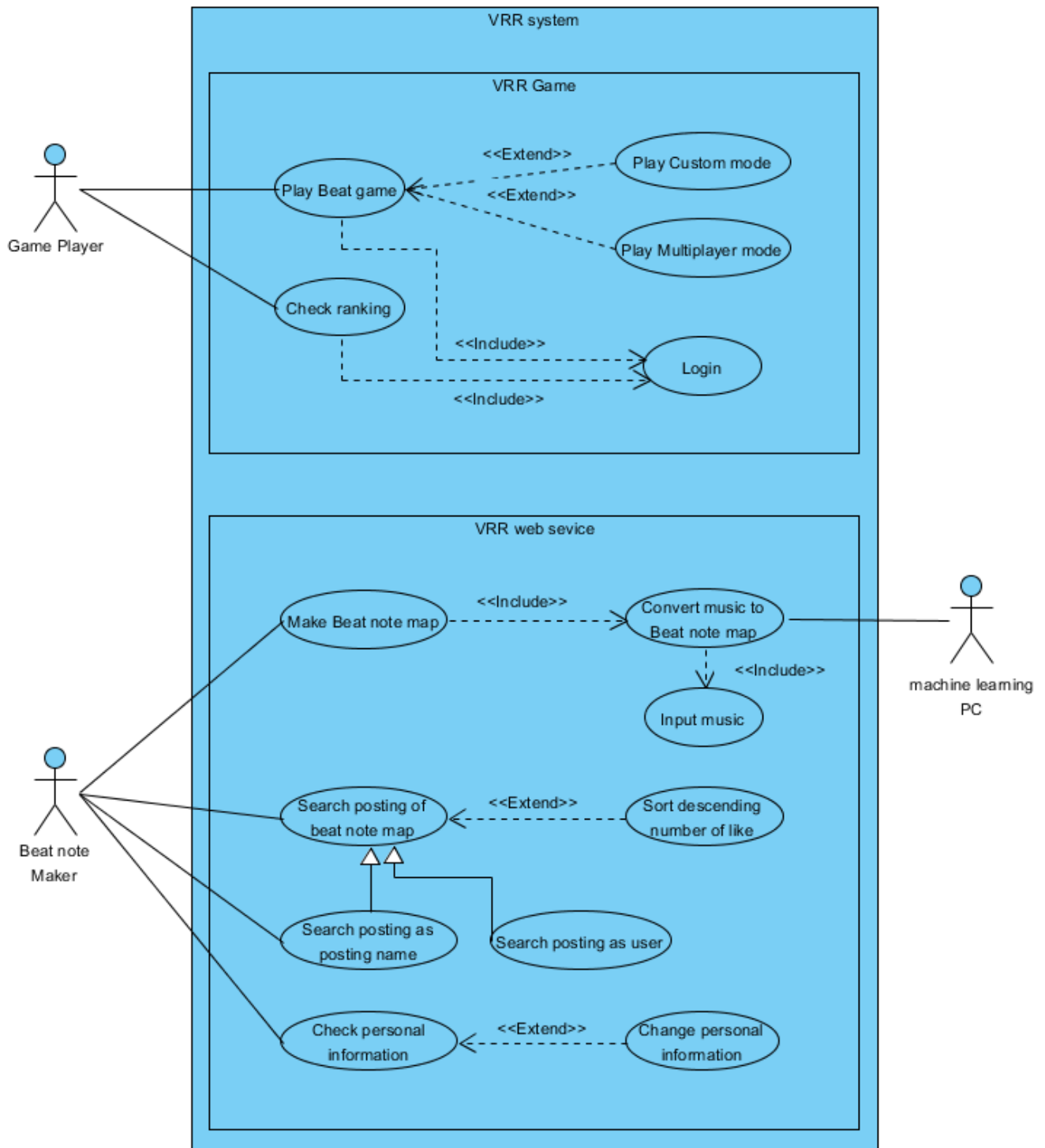
ID	SFR-O-07	중요도	상	담당자	이호찬
요구사항 명	Message Type 정의				
상세설명	<p>id, text, from, to, room, createdAt, updatedAt의 Data들을 갖으며 각각의 특성은 다음과 같다.</p> <p><b>Id</b>는 고유의 유일한 값이며 Database에서 정의되며 반드시 존재해야 한다.</p> <p><b>Text</b>의 Type은 String이며 반드시 존재해야 한다.</p> <p><b>From</b>의 Type은 User 자체의 집합이다.</p> <p><b>To</b>의 Type은 User 자체의 집합이다.</p> <p><b>Room</b>의 Type의 Room 자체의 집합이다.</p> <p><b>createdAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p> <p><b>updateAt</b>의 Type은 DateTime이고 반드시 존재하며 생성될 때 자동으로 생성된다.</p>				



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 4 상세설계 산출물

### 4.1 Use Case Diagram



## 4.2 VRR Deep Learning Model 구조

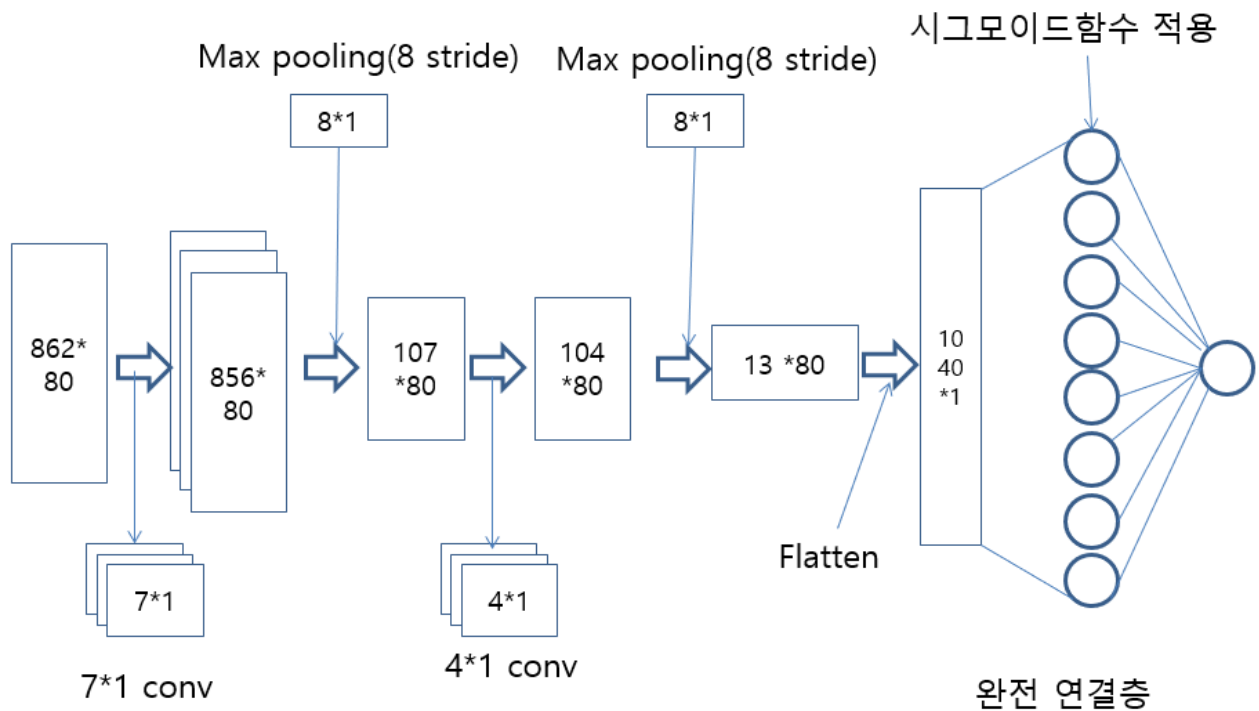
### □ 반영된 요구사항

[SFR-V-02] 전처리2 – STFT적용 및 스케일링

[SFR-V-06] 최적화 알고리즘

입력 데이터 량 -> mel scale이용(513개의 feature 값들 중에 특색을 가장 잘 보여주는 값들로 추출한다.)\_실습은 80개 feature로 추출 또한 시간 간격을 절반으로 줄여 데이터 량을 줄임과 동시에 정확도도 높인다. (기존 20초에서 10초로 변경)

타깃 값의 시간 간격이 너무 길어 delay가 생길 수 있다. -> 기존 0.5초 간격의 타깃 데이터 간격을 10ms으로 줄였다





상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

➤ Model: "sequential\_17"

Layer (type)	Output Shape	Param #
conv2d_21 (Conv2D)	(None, 80, 1, 1)	3
max_pooling2d_17 (MaxPooling)	(None, 10, 1, 1)	0
flatten_12 (Flatten)	(None, 10)	0
dense_24 (Dense)	(None, 30)	330
dense_25 (Dense)	(None, 3)	93
Total params: 426		
Trainable params: 426		
Non-trainable params: 0		

689 689

Train on 689 samples, validate on 173 samples

Epoch 1/10

689/689 [=====] - 1s 952us/sample - loss: 29.2064 - acc: 0.0943 -

Epoch 2/10

689/689 [=====] - 0s 114us/sample - loss: 6.8176 - acc: 0.4819 - v

Epoch 3/10

689/689 [=====] - 0s 91us/sample - loss: 0.2587 - acc: 0.9594 - va

Epoch 4/10

689/689 [=====] - 0s 88us/sample - loss: 0.1491 - acc: 0.9898 - va

Epoch 5/10

689/689 [=====] - 0s 84us/sample - loss: 0.1455 - acc: 0.9898 - va

Epoch 6/10

689/689 [=====] - 0s 96us/sample - loss: 0.1410 - acc: 0.9898 - va

Epoch 7/10

689/689 [=====] - 0s 88us/sample - loss: 0.1399 - acc: 0.9898 - va

Epoch 8/10

689/689 [=====] - 0s 89us/sample - loss: 0.1389 - acc: 0.9898 - va

Epoch 9/10

689/689 [=====] - 0s 85us/sample - loss: 0.1360 - acc: 0.9898 - va

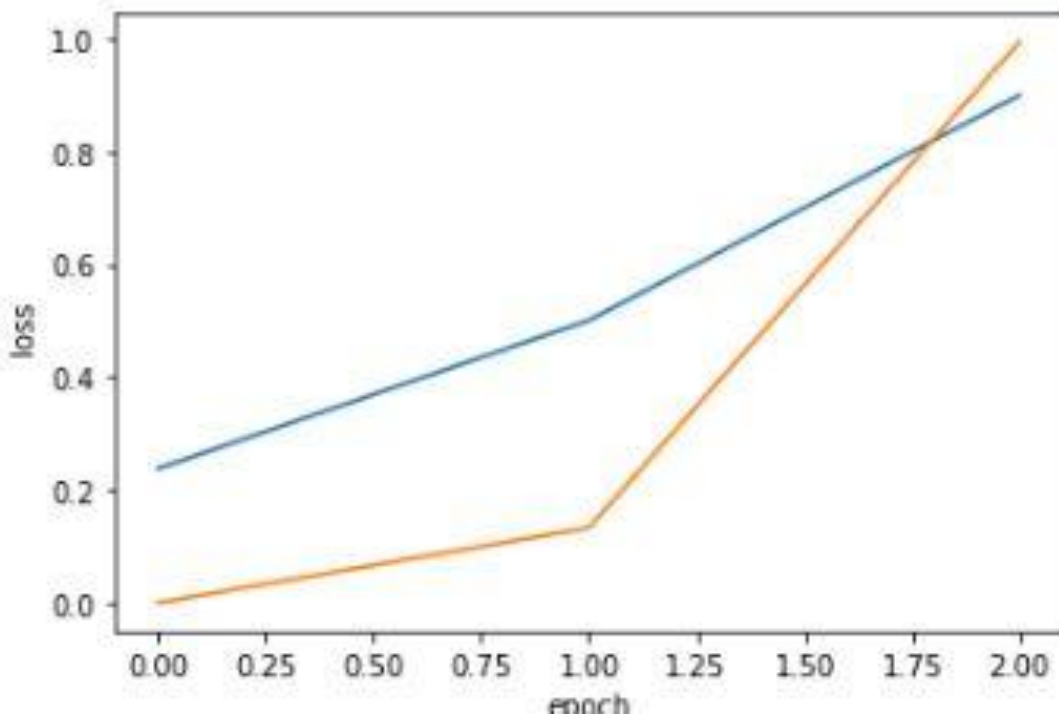
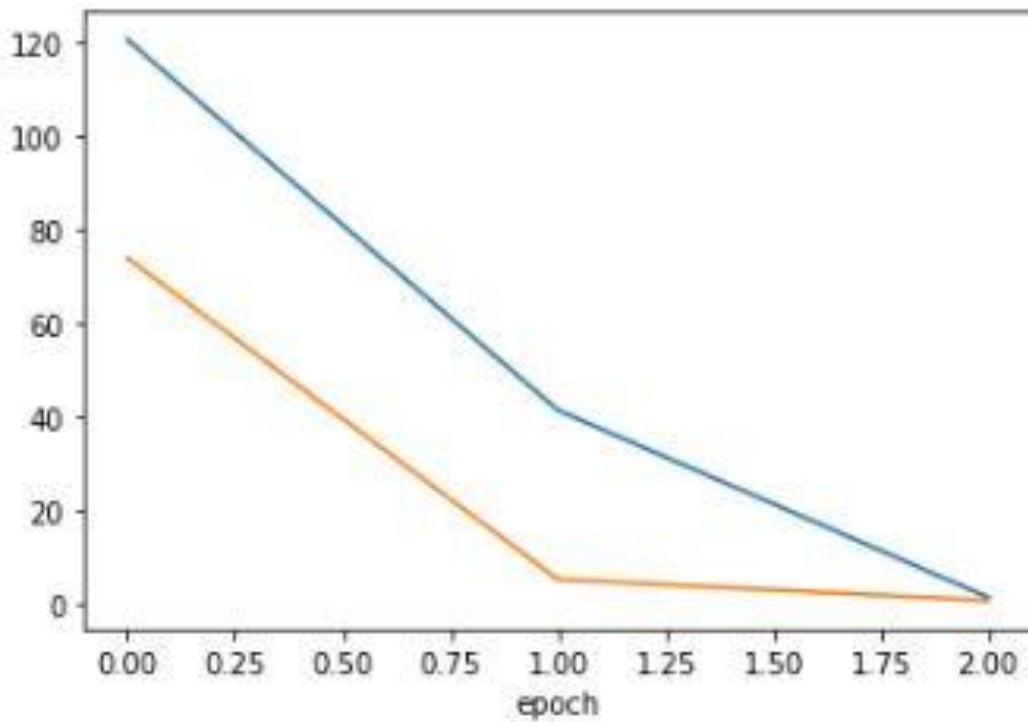
Epoch 10/10

689/689 [=====] - 0s 90us/sample - loss: 0.1350 - acc: 0.9898 - va



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

53]



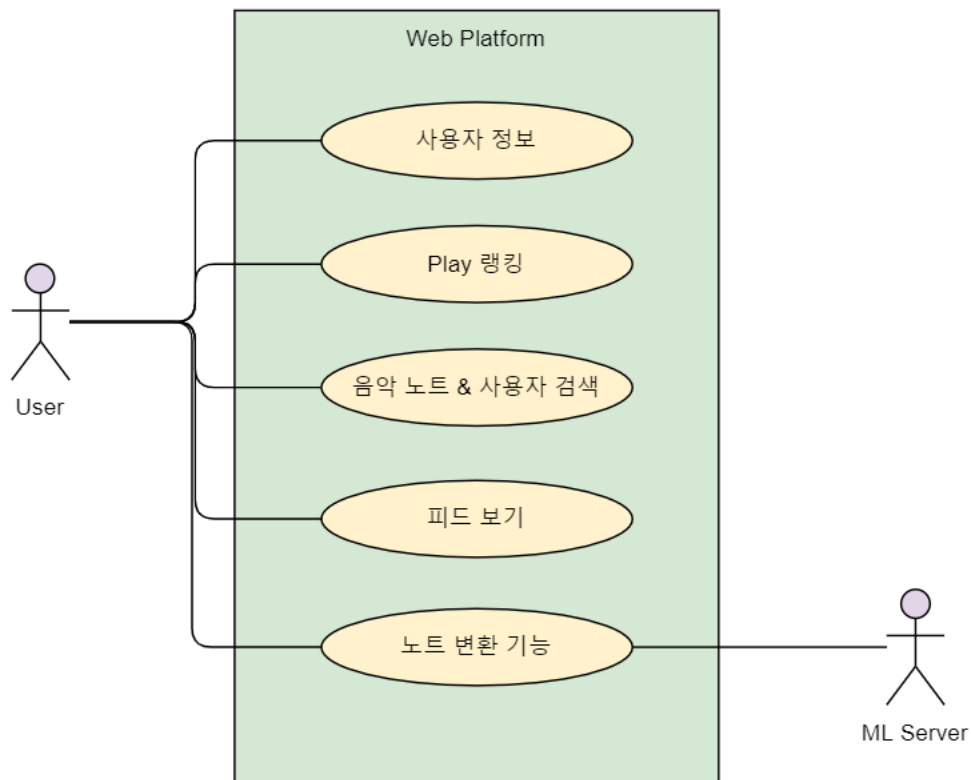
상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 4.3 Web 플랫폼

### 4.3.1 Use Case Diagram

#### □ 반영된 요구사항

[SFR-I-01~14] 인터페이스 어플리케이션 부 요구사항



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 4.3.2 Sequence Diagram

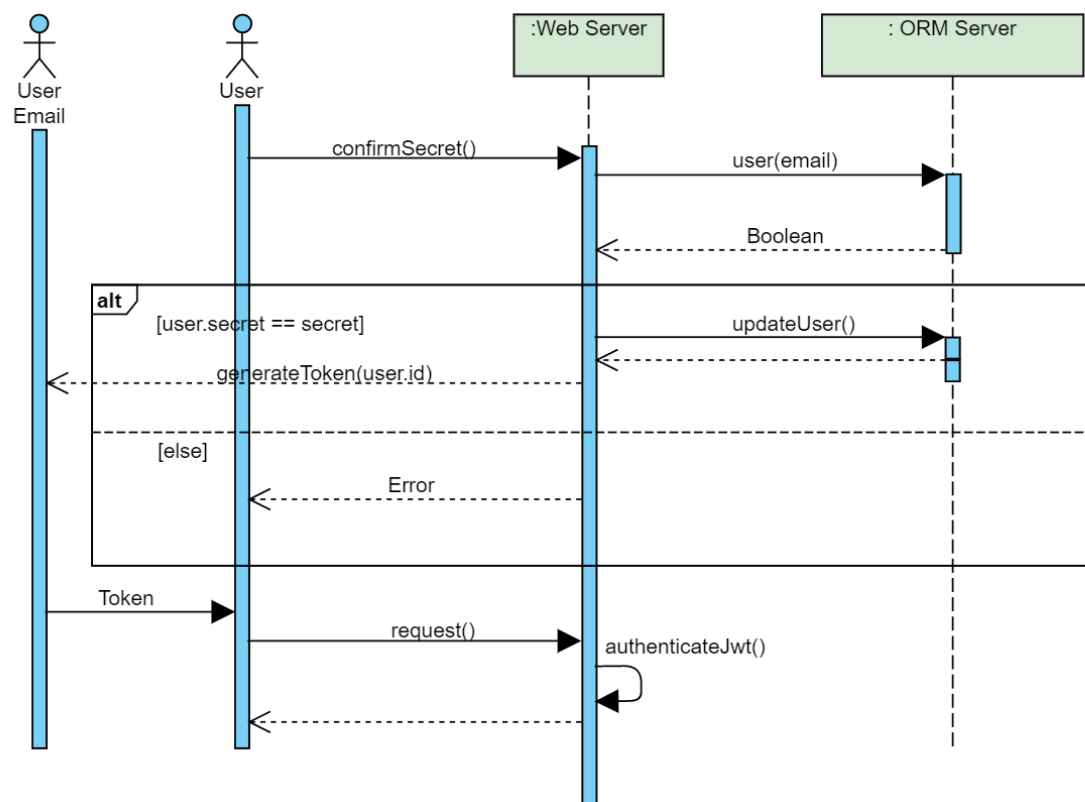
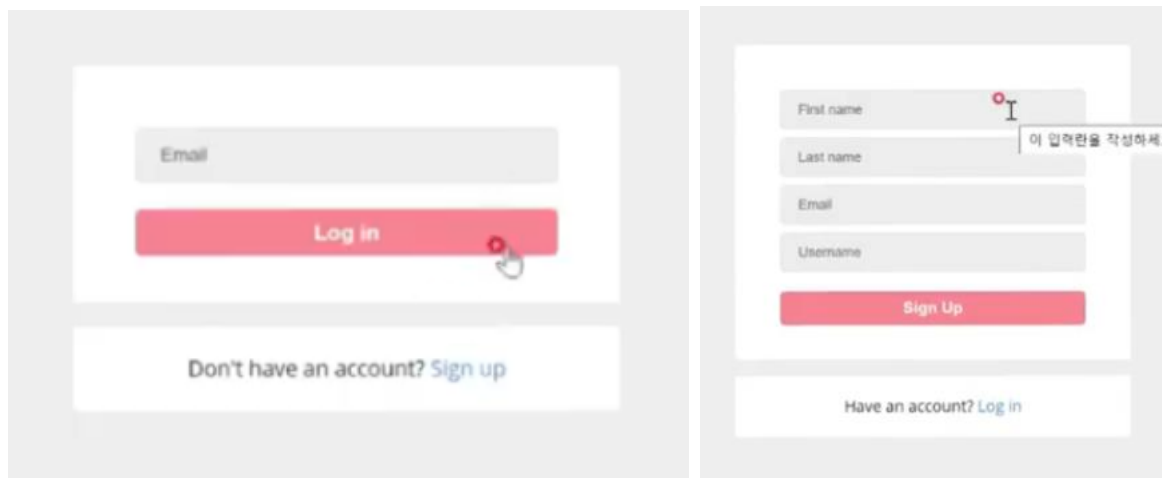
### 4.3.2.1 Login

#### □ 반영된 요구사항

SFR-I-01 로그인 기능 제공

SFR-I-02 로그아웃 기능 제공

SFR-I-03 회원가입 기능 제공

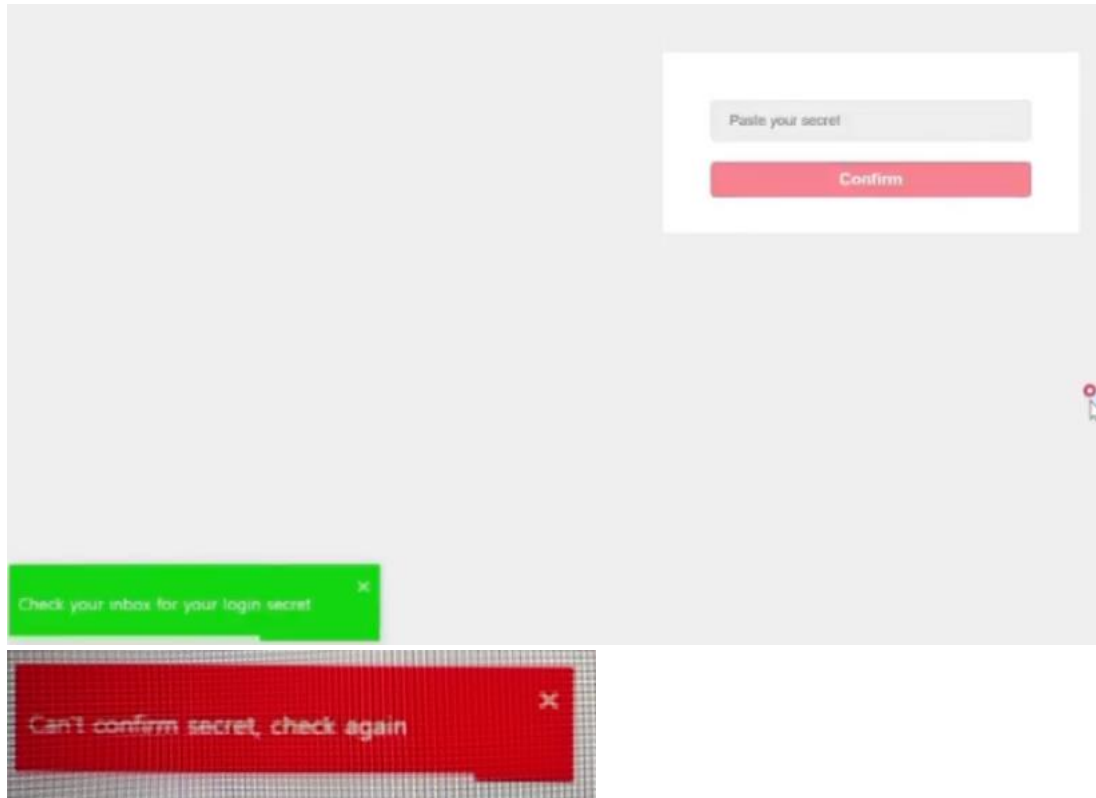


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

받는 사람: hochan049@gmail.com

Hello! Your login secret is **best mom**.

Copy paste on the app/website login



#### 4.3.2.2 사용자 정보

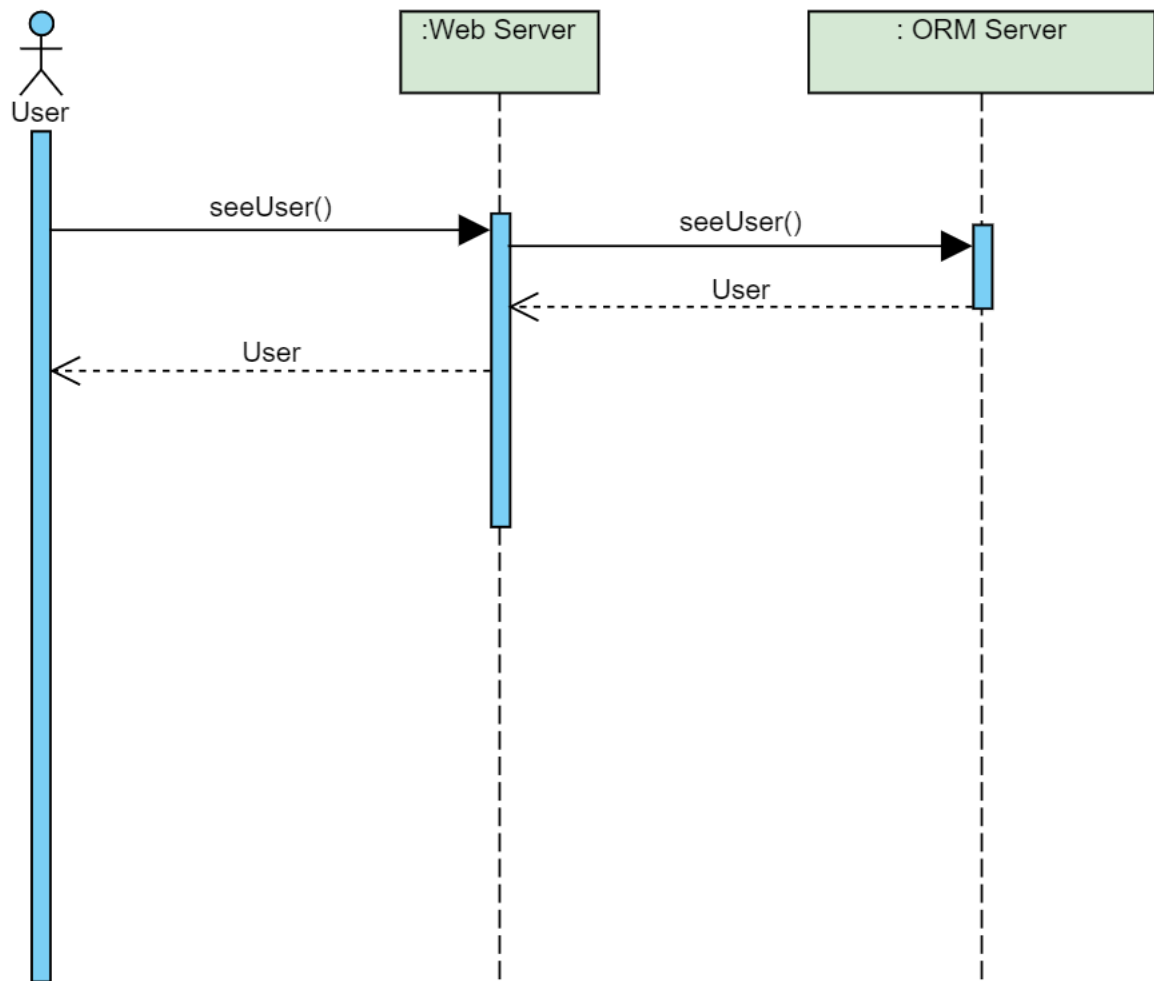
##### □ 반영된 요구사항

SFR-I-07 개인정보 확인 기능 제공

SFR-I-08 탈퇴 기능 제공

SFR-I-09 개인정보 변경 기능 제공

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



**hochan** Log Out

posts 3 followers 3 following 0

Lee Hochan

**KOROS SC 8 (공공기관) 본과위원회 회의록**

~ 18. 2. 15(일) University of Melbourne Building 140A, room 8.403

**이 목 록**

- 1. 제정된 KOROS SC 8 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403)
- 2. 제정된 KOROS SC 8 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403)

**이 목 록**

- 1. 제정된 KOROS SC 8 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403)
- 2. 제정된 KOROS SC 8 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403) 18. 2. 8. 403 (공공기관) 본과위원회 회의록(18. 2. 8. 403)

구분	성명	직위	소속
주최위원장	김영준	교수	한국과학기술원

```

classDiagram
    class User {
        id Integer
        reserveDate Date
        reserveTime Date
        reserveTime Integer
    }
    class Reserve {
        id Integer
        reserveDate Date
        reserveTime Date
        reserveTime Integer
    }
    class Local {
        id Integer
        reserveDate Date
        reserveTime Date
        reserveTime Integer
    }
    class Reservation {
        id Integer
        reserveDate Date
        reserveTime Date
        reserveTime Integer
    }
    User "1" -- "1" Reserve
    Reserve "1" -- "1" Local
    Local "1" -- "1" Reservation
    Reservation "1" -- "1" Reserve
  
```

문서이름	상태	날짜
문서이름	완료	2019-11-05
문서이름	완료	2019-11-05
문서이름	완료	2019-11-05
문서이름	완료	2019-11-05

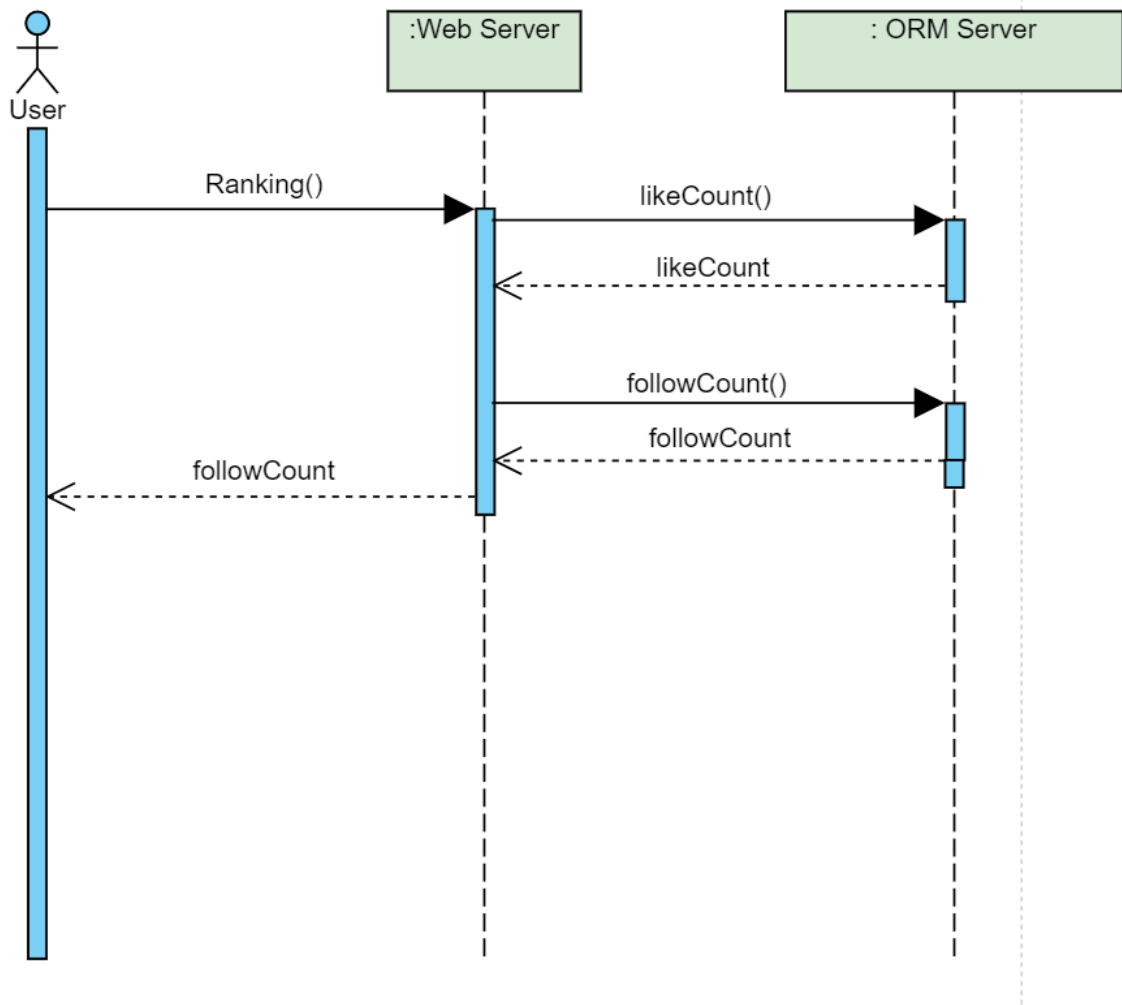


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 4.3.2.3 Play 랭킹

#### □ 반영된 요구사항

SFR-I-05 음악 노트 게시물 댓글, 좋아요, 파일, 이미지 보기 기능 제공



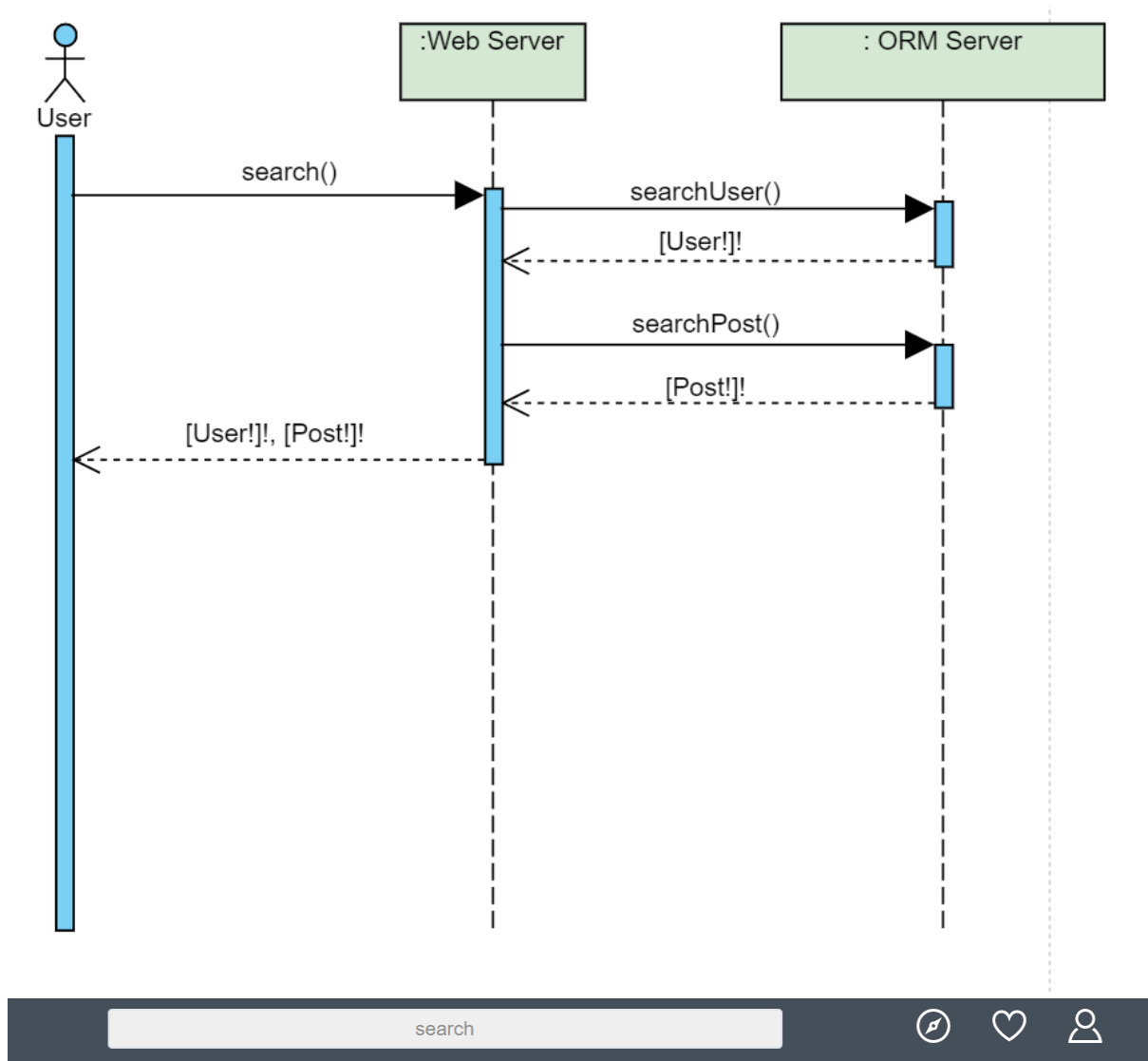
#### 4.3.2.4 음악 노트 & 사용자 검색

##### □ 반영된 요구사항

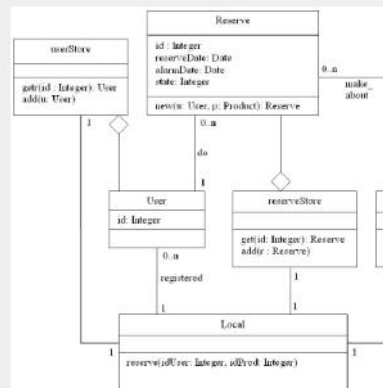
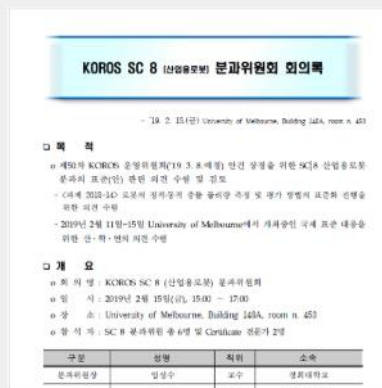
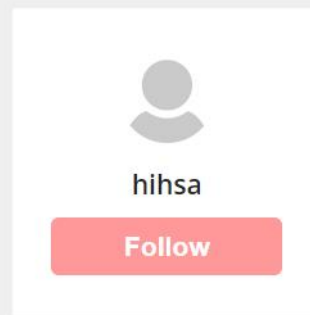
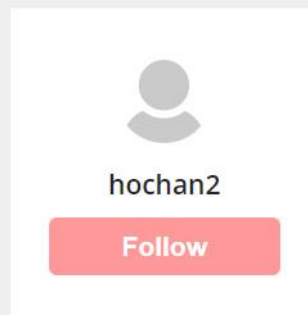
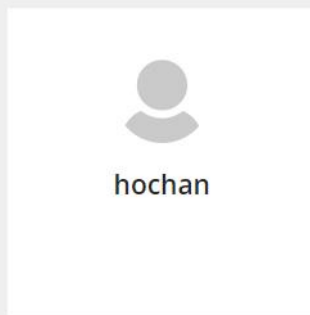
SFR-I-04 전체 음악 노트 게시물 확인 기능

SFR-I-06 음악 노트 게시물 및 사용자 검색 기능 제공

SFR-I-12 음악 노트 공유 기능 제공



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

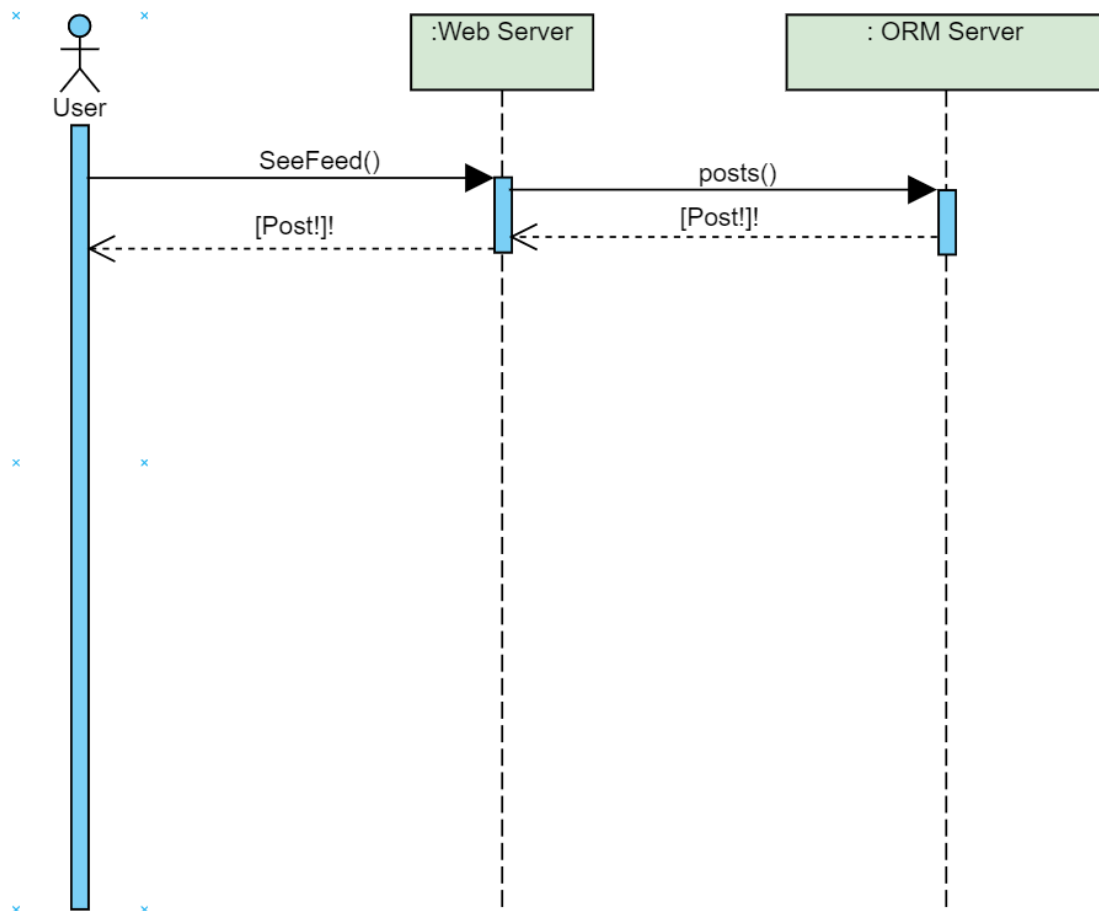


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

#### 4.3.2.5 피드 보기

##### □ 반영된 요구사항

SFR-I-14 페이지 렌더링 대기상태 시 보여지는 애니메이션 제공

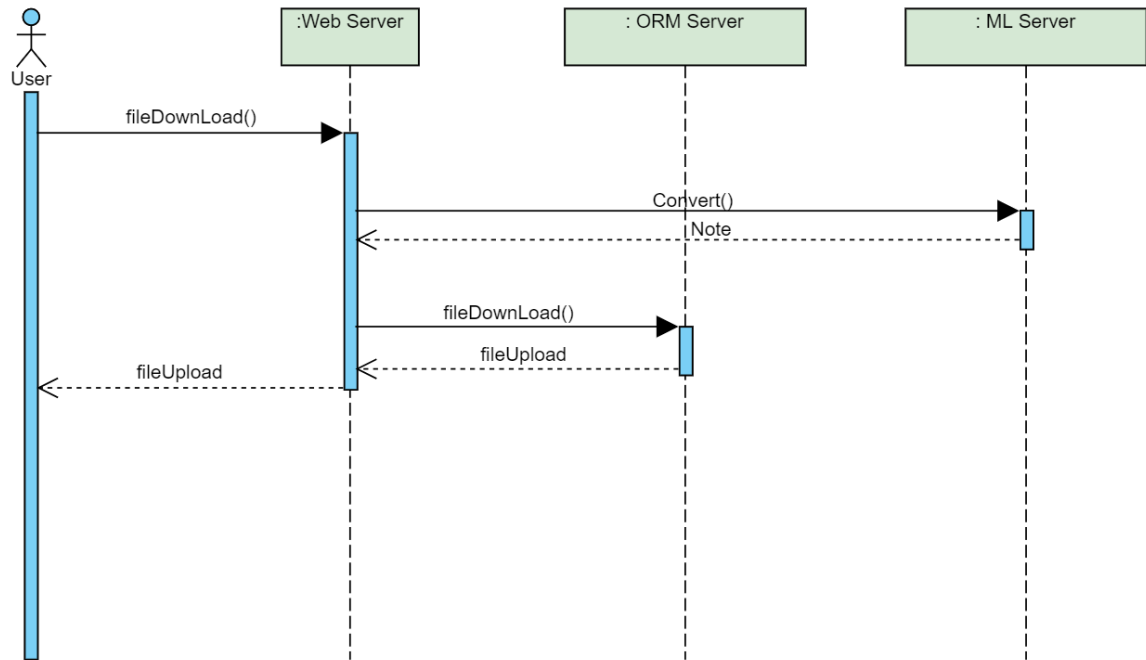


상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

#### 4.3.2.6 노트 변환 기능

##### □ 반영된 요구사항

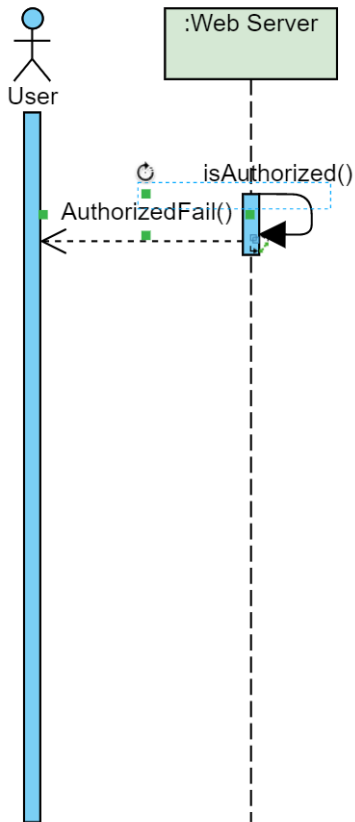
SFR-I-13 음악 노트로 변환 기능 제공



연월일: 2019-11-05	문서번호:	변경코드: 1.1	수정회수: 2	페이지: <b>54</b>
--------------------	-------	--------------	------------	-------------------

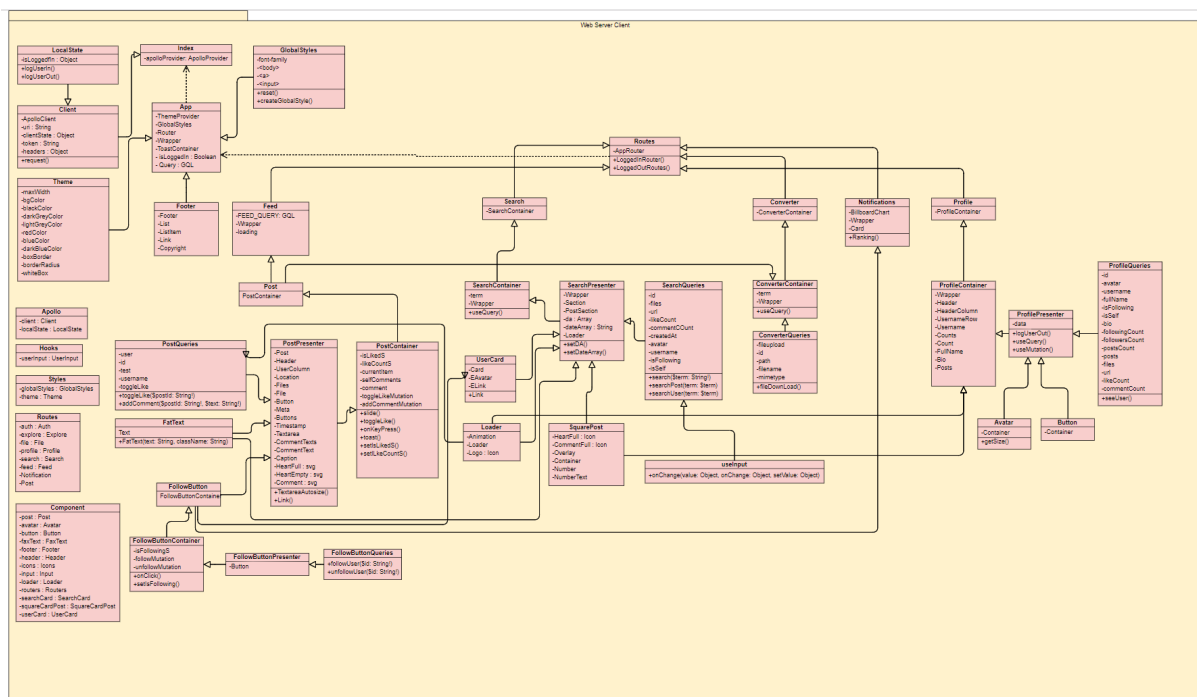
상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

#### 4.3.2.7 미들웨어 권한 확인



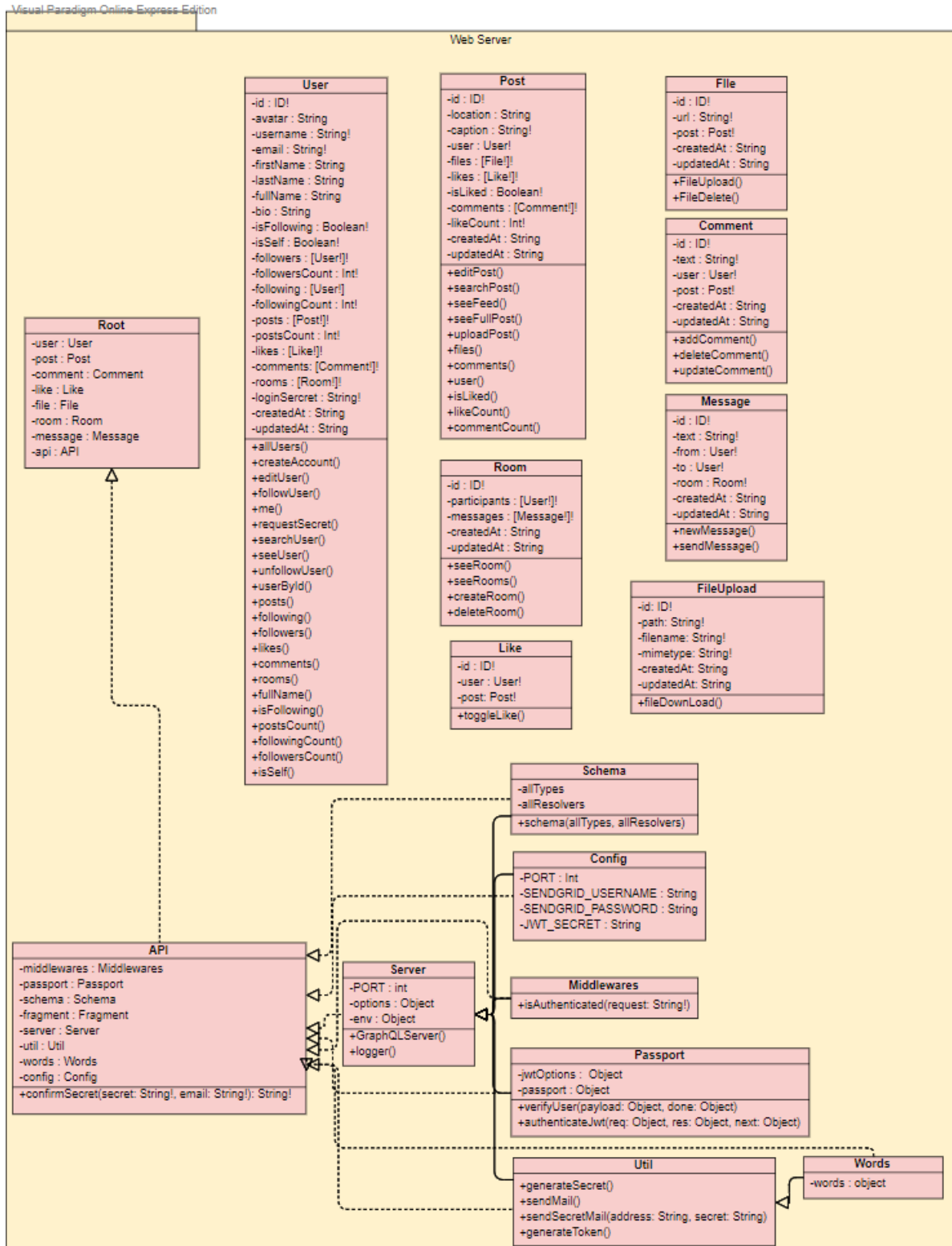
### 4.3.3 Class Diagram

#### 4.3.3.1 Web Server Client



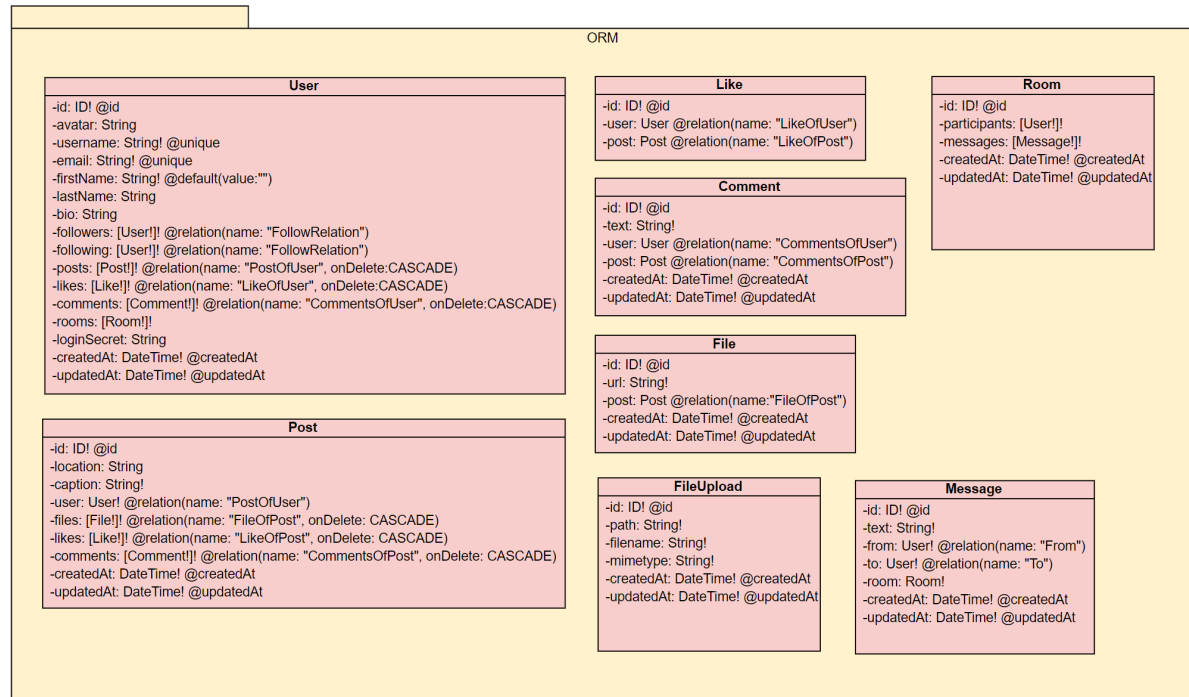
상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 4.3.3.2 Web Server



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

### 4.3.3.3 ORM Server







상세설계서: 딴 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

ORM

User

-id: ID! @id  
-avatar: String  
-username: String! @unique  
-email: String! @unique  
-firstName: String! @default(value: "")  
-lastName: String  
-bio: String  
-followers: [User!]! @relation(name: "FollowRelation")  
-following: [User!]! @relation(name: "FollowRelation")  
-posts: [Post!]! @relation(name: "PostOfUser", onDelete: CASCADE)  
-likes: [Like!]! @relation(name: "LikeOfUser", onDelete: CASCADE)  
-comments: [Comment!]! @relation(name: "CommentsOfUser", onDelete: CASCADE)  
-rooms: [Room]!  
-loginSecret: String  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

Post

-id: ID! @id  
-location: String  
-caption: String!  
-user: User! @relation(name: "PostOfUser")  
-files: [File!]! @relation(name: "FileOfPost", onDelete: CASCADE)  
-likes: [Like!]! @relation(name: "LikeOfPost", onDelete: CASCADE)  
-comments: [Comment!]! @relation(name: "CommentsOfPost", onDelete: CASCADE)  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

Like

-id: ID! @id  
-user: User @relation(name: "LikeOfUser")  
-post: Post @relation(name: "LikeOfPost")

Comment

-id: ID! @id  
-text: String!  
-user: User @relation(name: "CommentsOfUser")  
-post: Post @relation(name: "CommentsOfPost")  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

Room

-id: ID! @id  
-participants: [User]!  
-messages: [Message]!  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

File

-id: ID! @id  
-uri: String!  
-post: Post @relation(name: "FileOfPost")  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

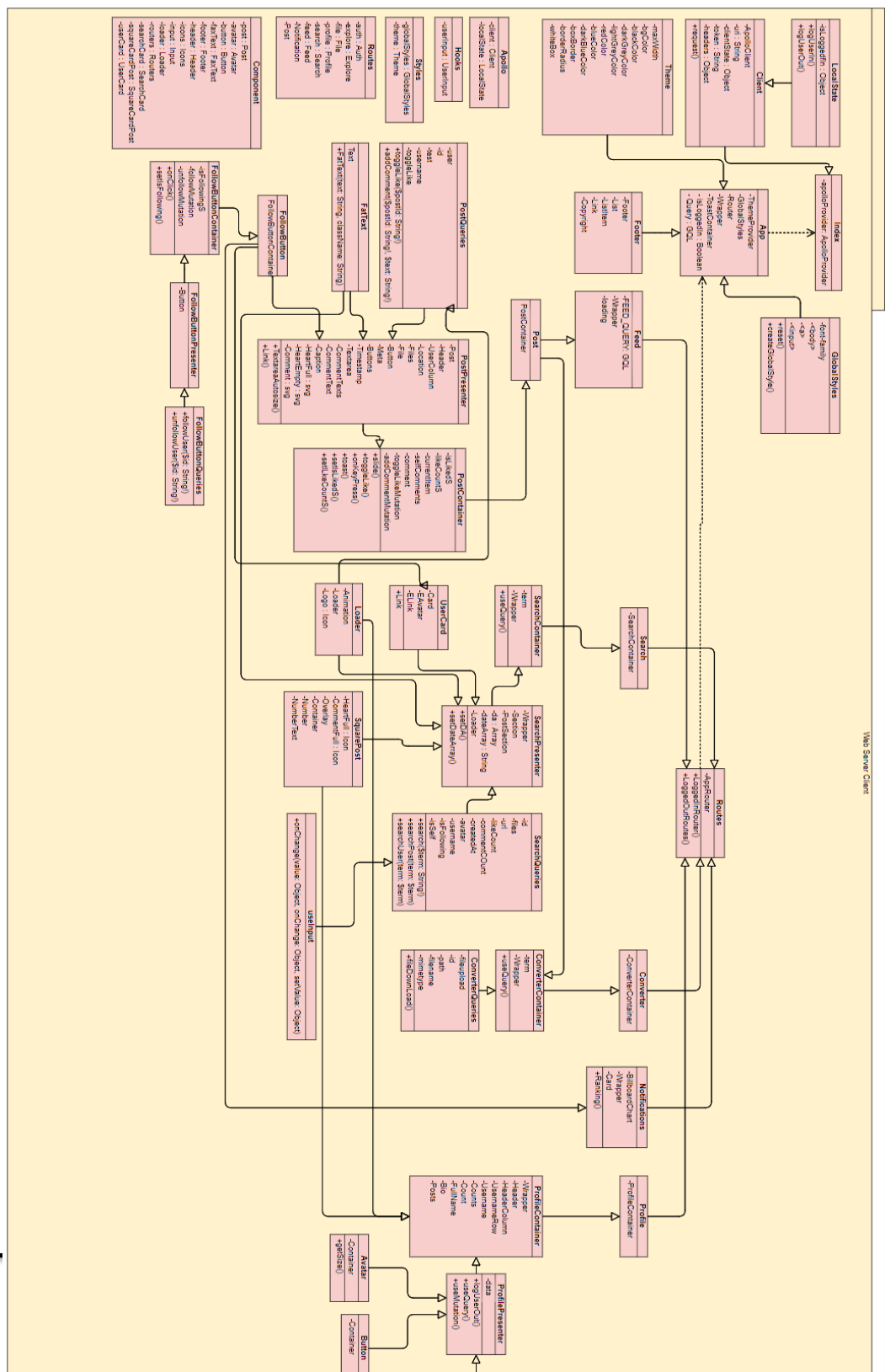
FileUpload

-id: ID! @id  
-path: String!  
-filename: String!  
-mimeType: String!  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

Message

-id: ID! @id  
-text: String!  
-from: User! @relation(name: "From")  
-to: User! @relation(name: "To")  
-room: Room!  
-createdAt: DateTime! @createdAt  
-updatedAt: DateTime! @updatedAt

상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임



상세설계서: 딥 러닝을 활용한 음악 Beat Note 자동 생성 서비스가 있는 VR 리듬게임

## 5 팀원 담당업무

이름	업무	세부사항
박영준(팀장)	게임 개발 및 모델 설계 총괄	Unity, Deep learning Model 설계 총괄
문명기	Deep Learning Model 설계, 구현	attention mechanism과 seq2seq를 통한 모델 설계 및 서버 구축
김세진	게임 개발	Unity
이호찬	Web Server	웹 플랫폼 구축
조동철	Deep Learning Model 전처리 설계	전처리 및 설계 구현

## 6 프로젝트 세부일정

구분	추진 내용	3 주차	4 주차	5 주차	6 주차	7 주차	8 주차	9 주차	10주차	11주차	12주차	13주차	14주차
계획	사업 아이디어 회의												
	제안서 작성												
분석	기술조사 및 시장 조사												
	요구사항 수집 및 분석												
설계	게임 환경 설계												
	머신러닝 모델 설계												
	Web Platform 설계												
개발	Unity 게임환경 구현												
	CRNN 설계 및 학습												
	Web Server 및 Platform												
테스트	CRNN 적용 output 확인												
	게임 오류 수정 및 디버깅												
종료	최종 발표 및 시연 준비												

## 7 참고문헌

- [1] 박한솔.(2019) 리듬 게임 노트 자동 생성에 적합한 합성곱 신경망 설계 및 구현에 관한 연구.  
가천대학교 게임대학원 석사논문
- [2] Gamemeca. (2007). 리듬액션 게임의 역사 – 탄생에서 현재까지  
<https://www.gamemeca.com/view.php?gid=124557>
- [3] DEV KOREA. (2019.01). 리듬게임 노트 생성 알고리즘  
[http://www.devkorea.co.kr/bbs/board.php?bo\\_table=m03\\_gna&wr\\_id=95493](http://www.devkorea.co.kr/bbs/board.php?bo_table=m03_gna&wr_id=95493)
- [4] Keunwoo Choi. (2016). Convolutional Recurrent Neural Networks for Music Classification  
<http://keunwoochoi.wordpress.com/tag/crnn/>
- [5] 브라우저는 어떻게 동작하는가  
<https://www.html5rocks.com/en/tutorials/internals/howbrowserswork/>
- [6] About React TimeSlicing and Suspense  
<https://www.youtube.com/watch?v=v6iR3Zk4oDY&feature=youtu.be&t=135>
- [7] graphql  
<https://tech.kakao.com/2019/08/01/graphql-basic/>
- [8] restful api 단점  
<https://www.slideshare.net/devview/112rest-graph-ql-relay> -