


# 입 사 지 원 서

입사 후 포부 및 각오	커뮤니케이션이 되는, 성장하는 신입사원이 되겠습니다!				
-----------------	-------------------------------	--	--	--	--

응시부문	프론트엔드 개발자	희망연봉	회사내규에 따름	입사구분	신입
------	-----------	------	----------	------	----

휴대폰	010-8378-0494	이메일	gyeongmilee23@gmail.com
-----	---------------	-----	-------------------------

	성 명	한글) 이경미                      영문) Lee Gyeong Mi		
	생 년 월 일	1997년 10월 23일	성 별	여
	주 소	서울특별시 영등포구 신길동 1515-8(신풍로 105-5) 세원연립 202호		

학 력 사 항	입학년월	졸업년월	출신학교 및 전공	구분(졸업여부)
	2016년 03월	2022년 02월	전주대학교 상담심리학과	학사 졸업
	2013년 03월	2016년 02월	전주유일여자고등학교	졸업

자 격 증	자격면허	취득년월	발행처
	지텔프 level 2	2021년 12월	지텔프 코리아

사 회 경 험	회사명	근무기간	업무내용	직 급
	Cj 올리브영	2022년 03월~2022년 11월	셀링	파트타이머
	EIE 어학원	2021년 03월~2022년 01월	초,중등 영어 교육	강사

어 학 연 수	국가	활동 기간	활동내용
	호주	2019년 01월 ~ 2021년 02월	워킹홀리데이 / 유학(academic class 수료)

교육사항	교육과목	(스마트웹&콘텐츠개발)반응형 UI/UX 웹콘텐츠 개발자 양성과정A5		
	교육기관	KH정보교육원	출석률	100%
	기간	2022 년 05 월 30 일 ~ 2023 년 01 월 31 일		
	교육내용	습득기술		
<ul style="list-style-type: none"><li>FRONTEND - HTML, CSS, JS , JQUERY, BOOTSTRAP, MYSQL, PHP</li><li>VERSION CONTROL - GIT, GITHUB, GITKRAKEN, GITDESKTOP</li></ul>				

# 자 기 소 개 서

자기소개	<b>해당업무에 지원한 본인만의 역량과 강점, 지원동기 등을 소개해주세요.</b>
	<p>[같이의 가치를 알고 있는 개발자]</p> <p>학부시절 IT 교육 경험을 계기로 '프론트 엔드 개발자'로 지원하게 되었습니다.</p> <p>'몽골 해외 교육 봉사' 활동으로 IT 교육팀에 들어가게 되었습니다. 저는 'PPT'를 통해 전달하고 싶은 내용을 명확히 보여주는 것에 흥미를 느껴 PPT 교육을 맡게 되었습니다. 그 당시 컴퓨터공학과 출신으로 팀장이었던 언니에게 부족한 부분들을 배웠습니다. 그 과정에서 PPT의 짜여져 있는 기능을 쓰는 것과는 달리, 직접 코드를 통해 하나하나 구현하는 '웹 페이지'에도 관심이 생겼습니다. 직접적으로 사용자의 최전방에서 페이지가 어떤 기능을 구현했는지, 얼마나 편리 했는지에 따라 실제 사용에 영향을 끼치는 '프론트 엔드 개발자'에 대한 꿈을 꾸게 되었습니다.</p> <p>개발자로 저의 가장 큰 강점은 단체 생활 경험이 많아 협업에 능숙하다는 점입니다. 개발 프로젝트 팀장, 대학교 부 과대, 고등학교 동아리 기장 경험을 하며 팀원들의 의견을 먼저 물어보고, 서로 다른 의견들을 조정했습니다. 단체 생활을 하면서 가장 도움이 되었던 습관은 '문서화'였습니다. 프로젝트 회의를 진행할 때마다 다같이 볼 수 있는 페이지를 만들면 좋겠다는 생각이 들어, '노션'을 공부해서 팀 페이지를 직접 제작했습니다. 또한 팀원들과 공유하면 좋은 개념에 대해 공부할 때는 따로 블로그에 정리해 '같이의 '가치'를 경험했습니다. 리더를 맡은 경험과 더불어, 팀원과 신입의 역할 로서도 누구보다 잘 협동하여 업무 할 자신이 있습니다.</p>
성격의 장/단점	<b>자신의 성격 중 장점과 단점은 무엇이며 단점의 경우 보완하기 위하여 어떤 노력을 했나요?</b>
	<p>[정리하며 알아내는 것을 좋아하는 개발자]</p> <p>'스스로의 문제에 대해 파악하고, 해결을 위해 노력하는 끈기'를 가지고 있습니다.</p> <p>국비 지원학원을 다니며 자바스크립트 수업 중에 배워서 익숙해진 코드 작성은 가능했지만, 이를 응용하는 것이 힘들었습니다. 미래의 제 코드 발전을 위해 '기초를 잡자'는 해결방안을 떠올렸습니다. '자바스크립트 딥다이브' 책 구매 후 스터디를 꾸렸습니다. 3명의 팀원들과 함께 확실한 이해와 책임감을 목표로 선생님처럼 서로 설명했고 이를 velog에 정리했습니다. 이를 통해 컴퓨터가 읽는 코드의 길과 var, let, const의 차이나 함수를 쓰는 이유 등 전에는 생각하지 못했던 것들을 스스로 깨닫게 되었습니다.</p> <p>저는 간혹 일정을 잊은 적이 있습니다. 이를 극복하기 위해 모든 일정을 정리해두고 복기합니다. 오히려 더 꼼꼼하게 일정을 체크한다는 칭찬을 들은 적이 있을 정도로 노력했습니다. 개발에 대해 배우면서 많은 내용들의 정리가 중요하다는 것을 더더욱 깨닫고 아이패드, 핸드폰, 노트북 등에 언제든지 잊기 쉬운 것들을 적어 두며 단점을 보완하고 있습니다.</p>

목표	입사 후 이루고 싶은 목표가 무엇이고 그 목표를 이루기 위해 어떠한 노력을 할 계획 인가요?
	[사용자 입장을 고려하는 역량을 갖추고 리드하는 개발자]
	3년동안 빠르게 성장해, 프로젝트 흐름을 완전히 이해하고, 리드할 수 있는 개발자가 되고 싶습니다.
	‘사용자 입장’에서 서비스를 구현하는 개발자가 되고 싶습니다. 세미프로젝트에서 ‘티켓판매’를 목적으로 하는 웹 서비스 기획을 할 때, 사용자입장에서 한눈에 보기 편하고, 쉽게 적용할 수 있는 서비스를 구현하고 싶었습니다. 그래서 직관적인 정보 전달로 접근하기 쉬운 UI요소를 넣었고, 회원가입페이지를 만들 때는 ‘tabindex’를 넣으면서 키보드접근성을 고려했습니다.
	세미 프로젝트에서 ‘팀장’을 맡게 되면서 일정을 관리하고, 강사님께 상황을 보고하며, 회의 주관, 회의록 정리 등의 역할을 맡게 되었습니다. 저희 팀은 백/프론트로 역할이 나뉘어 있었고, 각자 서비스를 구현해가는 과정에서 급하게 회의가 필요하거나 충돌이 있을 때 조정하는 역할 또한 수행하였습니다. ‘팀장’으로서의 역할이 얼마나 막중 한지 알게 되었고, 이를 바탕으로 회사에서도 팀원의 역할을 수행하면서 ‘팀 리더’로서의 자질을 높여 나갈 것입니다.
	입사 후에도 이러한 목표를 바탕으로 꾸준히 성장하는 신입사원이 되겠습니다.

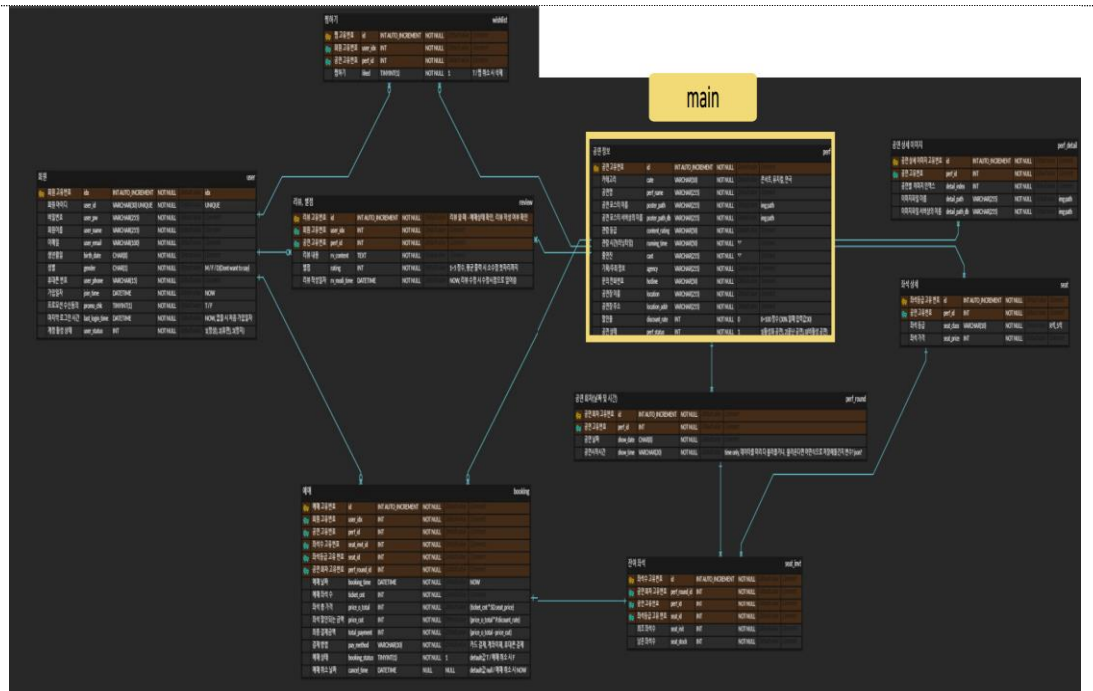
## 직 무 능 력 소 개 서

지식 /기술	해당분야에 지원하기 위해 어떤 지식과 기술을 습득하였으며, 어떻게 노력하였나요?
	[말보단 행동으로 실천하는 개발자]
	저는 프론트엔드 개발자로 적성을 정한 이후, 학원에서 배운 것과 더불어 개인 공부의 중요성을 깨닫게 되었습니다. 그에 대한 전략으로 ‘언어에 대한 이해’와 ‘리액트 독학’ 두가지 방안을 세웠습니다. ‘언어의 이해’를 위해서 프론트엔드 언어인 자바스크립트의 기반을 쌓으려는 목적으로 책을 읽으며 정리하였습니다. 이를 통해 스터디와 velog를 적극 활용해, 배운 내용을 정리하고 나누면서 혼자서는 생각하지 못한 내용들을 얻게 되었습니다. ‘리액트 독학’은 세미프로젝트가 끝난 이후, SPA페이지로 빠르게 동작해 사용성을 높이는 프로젝트를 구현을 목표로 인터넷 강의를 찾아보며 하나하나 직접 부딪혀 나갔습니다. 이처럼 저는 입사 후에도 획을 세워 행동으로 실천해나가는 신입사원이 될 것을 확신합니다.

태도	교육과정 혹은 프로젝트 수행 시 어려운 일에 직면했을 때 어떻게 문제를 해결 또는 극복하였나요?
	[공동의 목표를 이끌어가는 개발자]
	2022년 11월 29일부터 2022년 12월 08일까지 총 10일간의 세미프로젝트를 진행하였습니다. 초기에는 한 페이지씩 각자 프론트와 백을 맡기로 했지만, 각자 진로 방향이 뚜렷하게 나뉘어 있었기 때문에 프로젝트 기간 직전에 프론트와 백의 업무를 나누게 되었습니다. 이 과정에서 프론트 파트를 맡은 저와 팀원 2명은 평균 3페이지씩을 맡게 되었습니다. 백과 소통하면서 변동사항이 생겨, 한 사람이 급하게 업무가 많아지는 경우가 생길 때 마다 전체 일정에 문제가 되지 않도록 ‘zoom회의’를 사용해서 업무를 나누는 방안을 마련했습니다. 또한, 프론트 팀원들도 각자 그날 한 파트들을 보고하는 방식을 사용해서 서로의 진행상황을 파악했습니다. 부족한 부분은 도움을 주며 피드백 하는 방법으로 일정이 지연되지 않도록 문제를 극복했습니다. 향후 업무를 하는 과정에서도 공동의 발전을 위해 노력하고 맞춰가는 신입사원이 되겠습니다.

수 행 프 로 젝 트	구 분	항목	주요내용																																																		
	세 미 프 로 젝 트	프로젝트 명	아트온(세미 프로젝트)																																																		
		Github 및 포트폴리오 주소	<a href="https://gyeongmilee.github.io/ARTON_Gyenogmi/">https://gyeongmilee.github.io/ARTON_Gyenogmi/</a> <a href="https://harmless-patio-fe8.notion.site/ARToN-6b34bf3422d541c5aaa228600b7bb4aa">https://harmless-patio-fe8.notion.site/ARToN-6b34bf3422d541c5aaa228600b7bb4aa</a>																																																		
		수행기간	2022년 11월 29일 ~ 2022년 12월 08일																																																		
		개발 목표	직관적으로 공연 정보를 확인할 수 있고, 구매가 편리한 공연 티켓 판매 사이트 구현																																																		
		개발 환경	<div>운영체제 : Window</div> <div>개발도구 : Visual Studio Code, IntelliJ, Oven(카카오 Prototyping tool)</div> <div>DBMS : Mysql</div> <div>Server : Apache Tomcat 9.0.68</div> <div>Language : Html5, CSS3, JavaScript, JQuery, Jstl, JSP</div> <div>FrameWork : Spring</div> <div>버전관리 시스템 : git(Github Desktop)</div>																																																		
		프로젝트 시 활용해본 대표 기술	Javascript, JQuery																																																		
		구현 기능	<p>&lt;링크: 전체적인 기능구현 리스트, 사진은 개인 기능 구현 리스트&gt;  <a href="https://harmless-patio-fe8.notion.site/_-f26fb817fa04455294e98729a167dd9b">https://harmless-patio-fe8.notion.site/_-f26fb817fa04455294e98729a167dd9b</a></p> <table> <tr> <th>페이지 종류</th><th>회원가입/약관동의 페이지</th><th>로그인 페이지</th><th>카테고리 페이지</th></tr> <tr> <td>페이지 기능</td><td>약관동의</td><td></td><td>메인 슬라이드</td></tr> <tr> <td>(색칠한 섹션= 각 섹션의 제목)</td><td>필수 체크 항목이 체크되지 않았다면 메시지 띄우기</td><td>회원가입, 아이디 찾기, 비밀번호 찾기 마우스 hover 시 폰트 굵기 굵어짐</td><td>왼쪽 오른쪽 버튼 클릭시 방향 이동(버튼 클릭후 이벤트 실행중에는 다른 버튼 클릭해도 막혀있음)</td></tr> <tr> <td></td><td>취소시 새로고침 / 필수 입력 항목 체크 후 확인 버튼 누르면 회원가입 화면으로 넘어가기</td><td>아이디/ 패스워드 미 입력시 메시지 띄워주기</td><td>3초 지나면 자동으로 슬라이드 오른쪽에서 왼쪽으로 이동 / 중앙 이미지 scale</td></tr> <tr> <td></td><td>회원가입</td><td></td><td>What's hot</td></tr> <tr> <td></td><td>입력된 아이디가 조건에 맞지않으면 메시지 띄우기</td><td></td><td>각 포스터 hover시 이미지 scale</td></tr> <tr> <td></td><td>입력된 비밀번호가 조건에 맞지 않으면 메시지 띄우기</td><td></td><td>Weekly Ranking</td></tr> <tr> <td></td><td>비밀번호와 재확인 비밀번호가 일치하지 않으면 메시지 띄우기</td><td></td><td>순위 hover시 각 순위를 차지하는 공간 세로길이 커짐</td></tr> <tr> <td></td><td>비밀번호 재확인 항목이 키다운 또는, 키입 되서 한글자라도 바뀌었을때, 비밀번호 항목과 일치 하지 않으면 하단에 다시 경고창 띄워주기</td><td></td><td>What about this 슬라이드</td></tr> <tr> <td></td><td>공백이라면 해당항목 하단에 경고창 띄우기</td><td></td><td>공연제목 list 중 하나 hover시 밑줄 그어짐</td></tr> <tr> <td></td><td>클릭시 조건에 맞게 입력했다면, 회원가입 완료(alert)후 메인페이지 이동</td><td></td><td>hover 할때마다 공연제목, 포스터이미지, 배경색 바뀜</td></tr> <tr> <td></td><td></td><td></td><td>Aarton's pick</td></tr> <tr> <td></td><td></td><td></td><td>공연 포스터 hover시 아래에서 위로 홍보문구 올라옴</td></tr> </table>	페이지 종류	회원가입/약관동의 페이지	로그인 페이지	카테고리 페이지	페이지 기능	약관동의		메인 슬라이드	(색칠한 섹션= 각 섹션의 제목)	필수 체크 항목이 체크되지 않았다면 메시지 띄우기	회원가입, 아이디 찾기, 비밀번호 찾기 마우스 hover 시 폰트 굵기 굵어짐	왼쪽 오른쪽 버튼 클릭시 방향 이동(버튼 클릭후 이벤트 실행중에는 다른 버튼 클릭해도 막혀있음)		취소시 새로고침 / 필수 입력 항목 체크 후 확인 버튼 누르면 회원가입 화면으로 넘어가기	아이디/ 패스워드 미 입력시 메시지 띄워주기	3초 지나면 자동으로 슬라이드 오른쪽에서 왼쪽으로 이동 / 중앙 이미지 scale		회원가입		What's hot		입력된 아이디가 조건에 맞지않으면 메시지 띄우기		각 포스터 hover시 이미지 scale		입력된 비밀번호가 조건에 맞지 않으면 메시지 띄우기		Weekly Ranking		비밀번호와 재확인 비밀번호가 일치하지 않으면 메시지 띄우기		순위 hover시 각 순위를 차지하는 공간 세로길이 커짐		비밀번호 재확인 항목이 키다운 또는, 키입 되서 한글자라도 바뀌었을때, 비밀번호 항목과 일치 하지 않으면 하단에 다시 경고창 띄워주기		What about this 슬라이드		공백이라면 해당항목 하단에 경고창 띄우기		공연제목 list 중 하나 hover시 밑줄 그어짐		클릭시 조건에 맞게 입력했다면, 회원가입 완료(alert)후 메인페이지 이동		hover 할때마다 공연제목, 포스터이미지, 배경색 바뀜				Aarton's pick		
페이지 종류	회원가입/약관동의 페이지	로그인 페이지	카테고리 페이지																																																		
페이지 기능	약관동의		메인 슬라이드																																																		
(색칠한 섹션= 각 섹션의 제목)	필수 체크 항목이 체크되지 않았다면 메시지 띄우기	회원가입, 아이디 찾기, 비밀번호 찾기 마우스 hover 시 폰트 굵기 굵어짐	왼쪽 오른쪽 버튼 클릭시 방향 이동(버튼 클릭후 이벤트 실행중에는 다른 버튼 클릭해도 막혀있음)																																																		
	취소시 새로고침 / 필수 입력 항목 체크 후 확인 버튼 누르면 회원가입 화면으로 넘어가기	아이디/ 패스워드 미 입력시 메시지 띄워주기	3초 지나면 자동으로 슬라이드 오른쪽에서 왼쪽으로 이동 / 중앙 이미지 scale																																																		
	회원가입		What's hot																																																		
	입력된 아이디가 조건에 맞지않으면 메시지 띄우기		각 포스터 hover시 이미지 scale																																																		
	입력된 비밀번호가 조건에 맞지 않으면 메시지 띄우기		Weekly Ranking																																																		
	비밀번호와 재확인 비밀번호가 일치하지 않으면 메시지 띄우기		순위 hover시 각 순위를 차지하는 공간 세로길이 커짐																																																		
	비밀번호 재확인 항목이 키다운 또는, 키입 되서 한글자라도 바뀌었을때, 비밀번호 항목과 일치 하지 않으면 하단에 다시 경고창 띄워주기		What about this 슬라이드																																																		
	공백이라면 해당항목 하단에 경고창 띄우기		공연제목 list 중 하나 hover시 밑줄 그어짐																																																		
	클릭시 조건에 맞게 입력했다면, 회원가입 완료(alert)후 메인페이지 이동		hover 할때마다 공연제목, 포스터이미지, 배경색 바뀜																																																		
			Aarton's pick																																																		
			공연 포스터 hover시 아래에서 위로 홍보문구 올라옴																																																		

## DB 설계



ERD URL: <https://www.erdcloud.com/d/gwDQEk7hpAbojeb7p>

담당 역할/  
기여도

담당역할: 프론트엔드, 팀장  
기여도: 20% (팀 구성원: 5명)

## 프로젝트 참여소감

처음으로 진행했던 팀 프로젝트인 데다가 팀장을 맡게 되어서 책임감이 막중 했습니다. 일정에 치여, 책임감이 불안감으로 바뀌려고 할 때마다 소중한 팀원들이 도와주고 저 또한 서로를 돕고 배려하며 무사히 끝낼 수 있었습니다. 소통의 중요성을 알게 되었고, 팀원 한 명, 한 명의 역할이 중요하다는 것을 깨달았습니다.

# 세미 프로젝트 내용

## 구현기능설명

1. 버튼 클릭 시 가운데 위치한 공연 포스터가 scale(1.2)로 커지는 클래스 적용
2. 포스터가 이동하는 동안 버튼 막기 기능, 자동슬라이드 기능(2초)
3. 모든 공연포스터 클릭 시 공연 상세페이지로 이동

ARTON

콘서트 뮤지컬 연극 클래식 전시 아동

0 1 2 3 4 5 6 7 8 9 10



```
//오른쪽 버튼 클릭시
$('.slide_next').click(function () {
  btn_stop() //클릭시 원/오 버튼 기능 잠시 멈춤

  $('.banner > a> img').removeClass();
  $('.banner > a> img').eq((idx + 2) % b_count).prop('class', 'on');

  // 버튼 클릭시 -> 2번째 있는 배너가 가운데로 오므로!
  ban.animate({ left: '-=${b_width}' }, timer, 'linear');

  // 배너색선택범위로 나간배너를 끝으로 이동
  ban.eq(idx % b_count).animate({
    left: mb_width + b_width
  }, 0);
  idx += 1;
});
```

화면설명: 카테고리페이지의 메인슬라이드(새로고침 할 때 마다 랜덤으로 이미지 불러옴)

## 구현기능설명

1. 각 랭킹 hover 시 height커지면서 이미지, 상세정보 보임(클릭시 공연 상세정보 이동)
2. 더보기 버튼(클릭시 공연상세정보 이동)
3. (이미지에 정보 포함)

### WEEKLY RANKING

#### Hover시 예시

- 1위 2022.11.20 ~ 2022.11.23 코엑스 컨퍼런스룸 401호
- 2위 MOONCHILD Live In Seoul w/ Special Gu...
- 3위 프렙 내한공연 [PREP LIVE IN SEOUL]
- 4위 2022 신지훈 단독 콘서트 'Winter Neverla...
- 5위 2022 Wave to Earth 단독공연 [floral field ...]

### WHAT ABOUT THIS?

3

[화원]

[난쟁이들]

[Earmuffs]

[줄리엣과 줄리엣]

[줄리엣과 줄리엣]

2022.06.14 ~ 2022.07.07

3) What about this 섹션

[공연 제목] 클릭 시 배경색, 공연제목, 공연 날짜

바뀜

(포스터, 공연 정보 클릭 시 공연상세정보 이동)

function color\_chg(color) {

/\*무조건 true\*/

\$('.what\_about\_body').css({

zIndex: 99,

background-color: color

}).fadeIn(f\_timer);

화면설명: 카테고리페이지의 ranking / 서브슬라이드(fadein/out)



## 회원가입

아이디

5~20자의 영소문자,숫자,특수문자(.\_) 조합

비밀번호

8~16자 영문 대소문자,숫자,특수문자 조합  
특수문자는 @!#\$%&\*~ 중 1개 이상 사용가능합니다.

비밀번호 재확인

동일한 비밀번호를 입력하세요

이름

생년월일

YYYYMMDD(8자리)

성별

성별

이메일

이메일 입력

휴대전화

휴대전화번호 입력

약관동의

이제부터는 회원으로서의 권리와 의무를 동시에 부담할 수 있습니다. 이에 동의하는 회원은 회원가입 시 제공되는 개인정보 처리방침에 동의하는 것으로 동의합니다. (동의하지 않는 회원은 회원가입을 진행할 수 없습니다.)

약관동의는 회원가입 시점에 동의하는 것으로, 약관변경 시에는 별도로 동의할 수 있습니다. 약관변경 시에는 홈페이지에 게시된 약관변경 사항을 반드시 확인하시기 바랍니다.

가입하기

## 1) 정규식 사용으로 onsubmit / onfocus 마다 체크 함수를 걸어줌

```
const reg_id = /^[a-z0-9-]{5,20}$/; // 소문자, _ - 허용(영어만도 가능) / 5~20자리 //
const reg_pw = /^(?=.*[a-zA-Z])(?=.*[!@#$%^&*~]).{8,16}$/;
// .* => 0회 이상 / 숫자8~16자 / 영문 대소문자 / 최소 1개의 숫자 혹은 특수 문자를 포함해야 함.
const reg_name = /^[가-힣a-zA-Z]{2,}$/; // 한글, 영문만
// 만약 한글만으로 고려할 경우, /^[가-힣]+$/
const reg_birth = /^(19|20)[0-9]{2}([0-1][0-2])([0-1][0-1])$/;
const reg_email = /^[a-zA-Z0-9-]{1,30}@([a-zA-Z0-9-]{1,30})$/;
const reg_phone = /^[0-9]{11}$/; // 11자리 이상 숫자만
```

```
<div class="reg_msg" id="reg_msg"></div>
</div>
<div class="form_sec">
<div class="form_text">비밀번호 재확인</div>
<input type="password" class="reg_text" id="pw_ag" onfocus="focus_chk('pw_ag')">
<div class="reg_msg" id="reg_msg_pw"></div>
</div>
```

```
else if (type == 'pw_ag') {
  remove_msg('#pw_ag');
  // id 조건이 일치하지 않을때 -> 메시지 띄우게
  if (reg_id.test(id.val())) {
    set_message(msg, '5~20자의 영문 소문자, 숫자 또는 특수문자(?!@#$%^&*~)를 사용하세요.', '#id');
  }
  blank_chk($('#reg_msg_id'), $('#id'))
  if (reg_pw.test(pw.val())) {
    set_message(msg, '8~16자 영문 대문자 또는 소문자, 숫자 또는 특수문자(?!@#$%^&*~)를 사용하세요.', '#pw');
  }
  else if (pw.val().search(/[a-zA-Z0-9-]/) != -1) {
    set_message(msg, '비밀번호에는 공백이 들어갈 수 없습니다.', '#pw');
  }
  blank_chk($('#reg_msg_pw'), $('#pw'))
}
```

## 구현기능설명

1. (이미지에 정보 포함)
2. 항목 중 '비밀번호 재확인(onfocus)'의 예시
3. 비밀번호 재확인 type 확인
4. (비밀번호 재확인)의 앞인 비밀번호의 조건이 일치하면 remove\_msg 함수로 메시지 지움
5. (비밀번호 재확인) 앞 모두의 정규식이 일치하지 않을 경우 set\_message 함수로 메시지 띄움
6. 공백 체크 후 공백일 경우 메시지 띄움

## 화면설명: 회원가입 페이지

## 구현기능설명

- 1,2,3. 키프레임을 사용해 반응형 화면을 구현함(이미지에 정보 포함)

로그인

회원가입

ARTON

WHAT'S HOT

1) 메인슬라이드

기존페이지에서 전체적인 크기 감소

2) what's hot섹션

기존 : 5개씩 2줄 -> 3개씩 3줄 -> 2개씩 5줄 -> 1개씩 10줄

3) 기존 레이아웃

WHAT ABOUT THIS?

반응형 레이아웃

WHAT ABOUT THIS?

## 화면설명: 반응형 페이지

수 행 프 로 젝 트	구 분	항목	주요내용
	파 이 널 프 로 젝 트	프로젝트 명	Dear me,
		Github 및 포트폴리오 주소	<a href="https://project-dearme-gyeongmi.web.app/">https://project-dearme-gyeongmi.web.app/</a> <a href="https://github.com/GyeongmiLee/derame.git">https://github.com/GyeongmiLee/derame.git</a> <a href="https://harmless-patio-fe8.notion.site/Dear-me-33fbc63ed718428d95881151816653b3">https://harmless-patio-fe8.notion.site/Dear-me-33fbc63ed718428d95881151816653b3</a>
		수행기간	2023년 01월 16일 ~ 2023년 01월 27일
		개발 목표	지친 현대인들의 오늘을 기록할 수 있는 나만의 일기장"을 컨셉으로 기획부터 구현까지 완성했습니다.
		개발 환경	<b>운영체제</b> : Window <b>개발도구</b> : Visual Studio Code, Oven(카카오 Prototyping tool) <b>Language</b> : Html5, CSS3, JavaScript <b>Library</b> : React <b>버전관리 시스템</b> : GitDesktop, Git
		프로젝트 시 활용해본 대표 기술	React, Javascript
		구현 기능	<ul style="list-style-type: none"> <li>- 메인 페이지(각 페이지에 대한 정보제공, 간단한 힐링 콘텐츠)</li> <li>- 센터 페이지(근처 상담센터에 대한 정보 제공)</li> <li>- 일기 페이지(일기 쓰기, 수정, 삭제, 상세페이지 구현)</li> <li>- 건강정보 페이지(정신 건강에 대한 정의와 관리방법 정보 제공)</li> </ul>
		담당 역할/기여도	기여도: 100% (개인프로젝트)



프로젝트  
참여소감

리액트 독학 후 처음으로 진행했던 프로젝트라 모든 것들이 새롭고 직접 부딪혀 나가야 하는 과제 같았습니다. 하지만, 혼자서 꼭 해내리라는 마음 하나로 어려운 부분들을 헤쳐 나갈 수 있었습니다. 혼자서 공부하는 즐거움을 알게 되었고, 새로운 라이브러리를 공부하는 방식을 알게 되어 뜻깊었던 프로젝트였습니다.

## 파이널 프로젝트 내용 (1)

### 메인 페이지



화면설명: 메인페이지의 간단한 힐링 콘텐츠페이지

## 메인 페이지 상세 코드

- 1) 총 4개의 핸드폰화면 페이지 넣는 use State 혹 사용
  - 첫번째 페이지는 무조건 보이게 true로 넣어둠

```
// 핸드폰 화면 안의 페이지 4개 useState t/f로 넣어두기
const [visible, setVisible] = useState([true, false, false, false]);
```

나의 하루는  
어땠나요?

```
2
</div>
<div className={styles.col_sec_wrapper}>
  <p onClick={() => chg(0)}>오늘 하루 힘들었어요</p>
  <p onClick={() => chg(1)}>오늘도 이런 게 없는 것 같아요</p>
</div>
```

2) 클릭 시 chg 함수 호출

오늘 하루 힘들었어요

오늘도 이룬 게 없는 것 같아요

```
3 // 클릭시 하단바 트루이게, num으로 화면 페이지 알아보게한 후, setVisible로 true인 state가 보이게
const chg = (num) => {
  let tmp = visible.map(() => false);
  tmp[num] = true;
  console.log(tmp);
  setVisible(tmp);
}
```

4) 페이지별로 true인 경우에만 보이도록 단축평가로 걸어둠

```

84 > <div class=theme<styles>contentes_wrapper</div>
85 > {visible[0] && (...
86 > )}
87 > {visible[1] && (...
88 > )}
89 > {visible[2] && (...
90 > )}
91 > {visible[3] && (...
92 > )}

```

5) 버튼 클릭 시 =>  
handleNextPage 함수 호출

```

    </div>
    <MyButton
      text={"입력했어요!"}
      onClick={() => handleNextPage(3, diary, diaryRef)}
    />

```

```
// button누르면 다. 페이지로 넘어가게
const handleNextPage = (num, state, ref) => {
  if (firstData.length < 1 || secondData.length < 1 || thirdData.length < 1) {
    window.confirm("모든 항목을 채워주세요!");
    return;
  }
  if (state.length < 5) {
    ref.current.focus();
    console.log(`${state}Ref`);
    window.confirm("5글자 이상 입력해주세요!");
    return;
  }
  return chg(num);
};
```

6) 페이지 수, 각 input의 조건을 위한 인자, focus위한 ref 인자 호출  
⇒ 조건 맞으면 chg함수 호출

화면설명: 메인 페이지의 상세 코드

## 일기 페이지

1) "마음일기" 페이지 들어오면, 사용자가 쓴 일기 열람 가능

1

새 알기쓰기

2

3

오늘의 감정

4

5

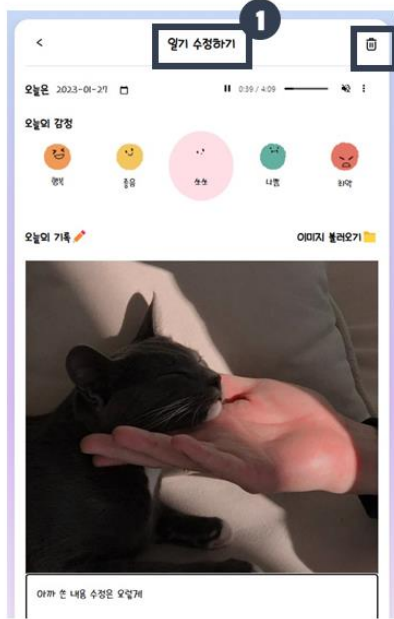
6

알기들 작성시 이전식으로요!

- 2) 날짜 선택 가능
- 3) 배경음악 들으며  
일기 작성 가능
- 4) 오늘의 감정 선택  
가능
- 5) 사용자의 로컬에  
서 원하는 이미지 볼  
러오기 구현
- 6) 일기 내용은 web  
storage API 중  
localStorage 이용  
해 브라우저 닫혀도  
데이터 저장 가능

화면설명: 일기 리스트를 볼수 있는 페이지 / 일기쓰기 페이지

## 일기 수정 / 상세 페이지



- 1) 일기 수정 클릭 시 => 일기쓰기 페이지와 같이 구성
- 2) 삭제 가능하게 구현

- 3) 일기 상세페이지
- 4) 이전으로 돌아가기
- 5) 수정페이지로 갈수있게 구현



화면설명: 일기 수정 / 상세 페이지

## 일기 페이지 상세 코드

```
function App() {
  // useReducer(데이터 관리)
  const [data, dispatch] = useReducer(reducer, []);
```

2

```
// CREATE
const onCreate = (date, content, emotion, img) => {
  dispatch({
    type: "CREATE",
    data: {
      id: dataId.current,
      emotion,
      content,
      date: new Date(date).getTime(), // new Date(date) 안넣어주면 현재 시간 들어감!
      img,
    },
  });
  dataId.current += 1;
};

// REMOVE
const onRemove = (targetId) => {
  dispatch({
    type: "REMOVE",
    targetId,
  });
};

// EDIT
const onEdit = (targetId, content, emotion, date, img) => {
  dispatch({
    type: "EDIT",
    id: targetId,
    date: new Date(date).getTime(),
    content,
    emotion,
    img,
  });
};
```

- 1) 일기 쓰기, 삭제, 수정 전체적으로 관리할 useReducer 혹 사용
- 2) 각 함수를 실행하면, dispatch 호출해서 상태변화일으키고 => reducer 함수 실행

3

```
return (
  <DiaryStateContext.Provider value={data}>
    <DiaryDispatchContext.Provider value={{ onCreate, onEdit, onRemove }}>
      <BrowserRouter>
        <div className="App">
          <showButton && {
            <img
              src={process.env.PUBLIC_URL + "/assets/letter.png"}
              className="btn_gotop"
              onClick={handleScroll}
              alt={"go_top"}
            />
          />
        </div>
        <CommonHeader />
        <Routes>
          <Route path="/" element={Main} />
          <Route path="/home" element={Home} />
          <Route path="/new" element={New} />
          <Route path="/diary/:id" element={Diary} />
          <Route path="/edit/:id" element={Edit} />
          <Route path="/info/:id" element={Info} />
          <Route path="/center" element={Center} />
        </Routes>
      </BrowserRouter>
    </DiaryDispatchContext.Provider>
  </DiaryStateContext.Provider>
);
```

- 3) useContext로 비효율적인 데이터 전달이 아닌(props drilling) 효율적인 데이터 공급이 가능하도록 구현

화면설명: 일기 페이지 상세 코드(useReducer/ useContext)

# 센터 페이지

## 위치 정보 이용 방법

- 1 지도를 확대/ 축소 하려면 ctrl을 누른 뒤 스크롤로 조정해보세요!
- 2 위치 정보를 확인 하려면, 마커를 클릭해보세요!
- 3 제공되는 정보는 홈페이지, 운영시간, 도로명 주소, 전화번호, 주요 프로그램이 있습니다
- 4 제공되는 주소는 서울, 부산, 대구, 인천, 광주, 대전, 울산, 세종, 경기 지역입니다



- 1) 마커 클릭시 infoWindow(2번) 창이 뜨게 구현
- 3) X 누르면 창 닫힘

화면설명: 센터페이지(근처 상담센터 정보 제공)

# 센터 페이지 상세 코드(카카오 맵 Api 사용)

```
import { CustomOverlayMap, MapMarker, Map } from "react-kakao-maps-sdk";
import React, { useState, useEffect } from "react";
import { ScrollView } from "react-native";
import Footer from "../components/Footer";
import styles from "../Center.module.css";

const Center = () => {
  const [isOpen, setIsOpen] = useState("");
  const [centerData, setCenterData] = useState({});

  1-2 url = "https://apis.data.go.kr/1383000/gmis/teenDscnSrcnService/getTeenDscnSrcn";

  const getData = async () => {
    const response = await fetch(url);
    const data = await response.json();
    const locations = data.response.body.items.item.map((spot) => {
      2
      spot.cntNm,
      spot.lat,
      spot.lot,
      spot.hmpgAddr,
      spot.opprHrcn,
      spot.roadNmAddr,
      spot.rprtTelno,
      spot.sprtCnt,
    });
    setCenterData(locations);
  };
};
```

2) 공공데이터 Api 데이터 받아오는 함수, 필요한 내용만 locations에 배열로 담음

1-1) InfoWindow 상태관리를 위한 useState  
1-2) 공공데이터 Api 상태관리 위한 useState

cnterNm: 센터이름  
hmpgAddr: 홈페이지 주소(url)  
위도(lat) 경도(lot)  
opperHrcn: 상담운영시간  
roadNmAddr: 도로명주소  
sprtCnt: 주요 프로그램

```
{centerData &&
centerData.map((it, idx) => {
  3
  let {
    cnterNm,
    lat,
    lot,
    hmpgAddr,
    opperHrcn,
    roadNmAddr,
    rprtTelno,
    sprtCnt,
  } = it;
  4
  return (
    <div key={idx}>
      <MapMarker
        position={{ lat: +lat, lng: +lot }}
        onClick={() => setIsOpen(idx)}
      />
      5
      {isOpen === idx && (
        <CustomOverlayMap position={{ lat: +lat, lng: +lot }}>
          <div className={styles.wrap}>
            <div className={styles.info}>
              <div className={styles.title}>
                {centerNm}
              </div>
              <div className={styles.close}>
                <img
                  className={styles.close}
                  src={process.env.PUBLIC_URL + "/assets/x.png"}
                  alt="닫기"
                />
              </div>
            </div>
          </div>
        </div>
      )}
    </div>
  );
});
}
```

- 3) centerData가 있을때만 실행하는 단축평가
- 4) 배열 이름으로 재할당하여 데이터 관리
- 5) isOpen이 현재 클릭된 idx와 같을때만 열리게 구현

화면설명: 센터페이지 상세 코드