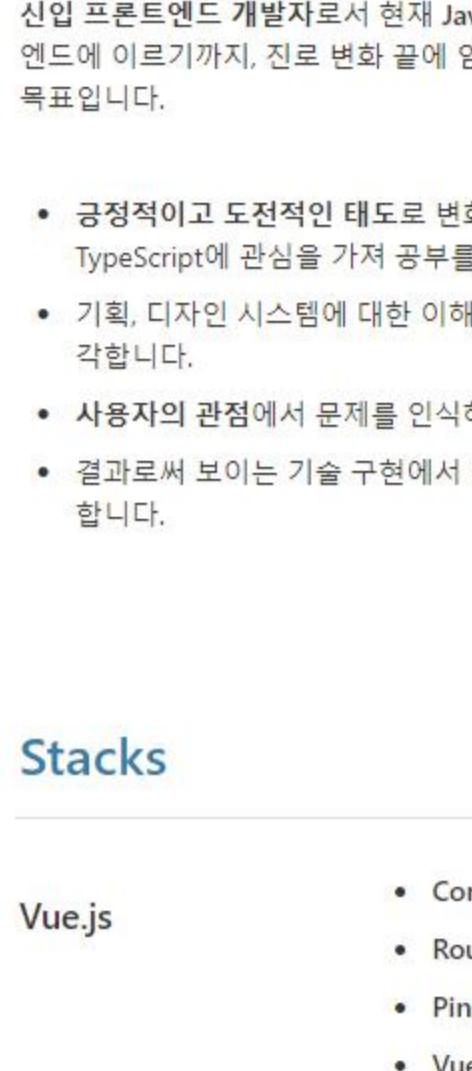




사용자와 서비스를 잇는 개발자

안녕하세요. 프론트엔드 개발자를 희망하는 장민영입니다.



Contact

- 📞 Phone | 010-XXXX-XXXX
- ✉️ E-Mail | myoungever@gmail.com

Channel

- 💻 Github | github.com/spaceyoung
- .padlet | padlet.com/myoungever

About Me

🌟 더 나은 사용자 경험

.." 적극적 소통

🔥 성장을 위한 학습 의지

신입 프론트엔드 개발자로서 현재 JavaScript와 Vue.js를 주로 다루고 있습니다. 디자인에서 시작한 관심이 프론트엔드에 이르기까지, 진로 변화 과정에 얻은 확신으로 사용자에게 좋은 경험과 편의를 제공하는 개발자가 되는 것이 목표입니다.

- 긍정적이고 도전적인 태도로 변화를 유연하게 받아들이며, 새로운 것에 대한 습득 의지가 강합니다. 최근에는 TypeScript에 관심을 가져 공부를 이어가고 있습니다.
- 기획, 디자인 시스템에 대한 이해를 바탕으로 협업과 소통의 중요성을 알고, 함께함으로써 성장할 수 있다고 생각합니다.
- 사용자의 관점에서 문제를 인식하고, 사용자의 다양한 행동 케이스에 대비하여 개발하기 위해 노력합니다.
- 결과로써 보이는 기술 구현에서 만족하지 않고, 유지 보수를 위해 더 효율적인 코드 및 구조 개선 방안을 고민합니다.

Stacks

Vue.js

- Composition API 문법에 익숙하며, 사용을 지향합니다.
- Router를 활용하여 SPA를 제작할 수 있습니다.
- Pinia를 활용하여 전역 상태 관리를 할 수 있습니다.
- Vuetify3의 컴포넌트를 사용 및 커스텀하여 UX/UI를 개선할 수 있습니다.

JavaScript

- ES6 이후 JavaScript 문법을 이해하고 활용할 수 있습니다.
- Vanilla JavaScript를 활용하여 DOM을 다룰 수 있습니다.
- fetch, async/await 문법을 사용해 비동기 처리를 할 수 있습니다.

HTML5, CSS3

- 시멘틱 태그를 사용하여 웹 접근성과 웹 표준을 고려한 마크업을 작성합니다.
- 반응형 웹에 대응할 수 있도록 마크업 구조를 설계합니다.
- flex, 반응형 단위, 미디어 쿼리를 활용하여 반응형 웹을 제작합니다.

Tools

- Figma : 개인 프로젝트의 초기 UI 디자인을 위해 사용한 경험이 있으며, 기본적인 기능을 숙지하고 있어 협업 툴로 사용 가능합니다.
- Github : Git, Github의 프로젝트 버전 관리를 이해하고 사용합니다.

Main Project

공책 Note, Book

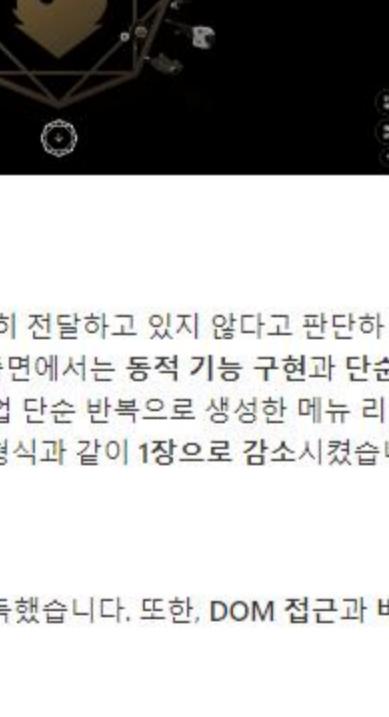
도서 검색 및 기록을 통해 독서 기록을 관리할 수 있는 웹 앱

개인 프로젝트 기여도 100%

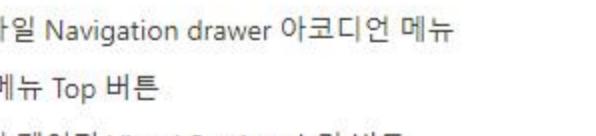
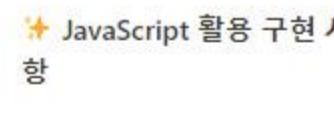
2024.01.10 ~ 2024.02.13 배포 완료
리팩토링 및 기능 보완 작업 꾸준히 진행 중

배포 링크 | notebook-service.vercel.app

Github | github.com/spaceyoung/notebook



모바일 대응 화면



모바일 대응 화면

PC 대응 화면

2024년 새해 목표 중 하나로 '독서 습관 기르기'를 계획하여, 이에 대한 기록 및 동기부여와 CRUD 기능을 직접 구현하고 Open API와 연계해 확장하기 위해서 작업한 프로젝트입니다. 기획부터 화면 구현, 배포까지 스스로 진행하는 걸 목표로 삼았습니다.

해당 프로젝트를 통해 데이터 처리 기능과 SPA 프레임워크의 이점을 알고 경험했으며, React와 같은 다른 프레임워크에도 관심을 두게 되었습니다. 또한, 기획부터 시작해 기본적인 개발 프로세스를 따르면서 서비스 흐름에 관해 구체적으로 고민해 보고, 협업의 필요성을 몸소 느낄 수 있었던 기회였습니다.

사용 기술

JavaScript, Vue.js, Vuetify3, Vue Router, Axios, Pinia, Firebase, Vercel

구현 사항

- Firebase를 활용한 회원 가입/로그인 기능(기능 보완 진행 중)
- 알라딘 Open API를 활용한 도서 검색 기능
- 독서 기록 정보 CRUD 기능
- 동적 라우트를 활용한 개별 도서 정보 및 독서 기록 페이지
- 사용자의 독서 기록 정보 입력 조건 설정 및 유효성 검사
- 웹 앱을 고려한 반응형 디자인

트러블 슈팅

- 도서 검색 결과 랜덤 출력

문제 배경 : 같은 검색어를 입력해도 검색 시마다 검색 결과 리스트 순서가 일정하지 않고 랜덤하게 뒤바뀌는 현상이 발생했습니다.

문제 원인 : 검색 결과 데이터를 받아와 저장하는 과정에서 순차적 처리가 이루어지지 않는 것이 원인이었습니다.

해결 과정 : async/await으로 불러온 검색 결과 순서가 일정했기에, 받아온 데이터를 forEach 문으로 배열에 저장하는 과정에 문제가 있다고 판단했습니다. 반복문의 비동기 처리에 관하여 찾아본 결과, forEach 문은 순차 처리를 하지 않는다는 것을 깨닫고 for...of 문으로 바꾸어 문제를 해결했습니다.

알게 된 점 : forEach 문의 동작 원리와 배열에 비동기 작업을 실행할 때의 순차 처리와 병렬 처리 방법을 알게 되었습니다.

- 도서 검색 결과, 독서 기록 입력/수정 페이지 새로고침 오류

문제 배경 : 도서 검색 결과, 독서 기록 입력/수정 페이지에서 새로고침을 누르면 404 페이지로 넘어가는 오류가 발생했습니다.

문제 원인 : Pinia를 사용해 저장한 state 객체 정보가 새로고침 시 초기화되는 것이 원인입니다.

해결 과정 : JavaScript로 To Do List를 만들어 볼 때 웹 스토리지를 사용하여 데이터를 저장했던 방법을 떠올리고, pinia-plugin-persistedstate 라이브러리를 추가 설치해서 데이터를 로컬 스토리지에 저장하여 문제를 해결했습니다.

알게 된 점 : 웹 스토리지 API를 통해 데이터의 지속성을 유지하는 방법을 다시 한번 상기할 수 있었습니다.

PURADAK

푸라닭 웹사이트 리뉴얼

개인 프로젝트 기여도 100%

1차(HTML&CSS) 2023.11.20 ~ 2023.12.01

2차(JavaScript) 2023.12.13 ~ 2024.01.20

배포 링크 | spaceyoung.github.io/puradak/

Github | github.com/spaceyoung/puradak



푸라닭의 공식 웹사이트 디자인으로 고급스러움을 강조하는 브랜드 이미지를 충분히 전달하고 있다고 판단하여 기획했습니다. JavaScript를 학습하며 직접 적용해 본 첫 프로젝트이며, 기능적 측면에서는 동적 기능 구현과 단순 반복 마크업 감소, 지도 API 활용을 중심으로 두고 작업을 진행했습니다. 특히 마크업 단순 반복으로 생성한 메뉴 리스트를 JSON으로 정리한 후, 반복문으로 자동화하여 맵 메뉴 페이지 5장을 SPA 형식과 같이 1장으로 감소시켰습니다.

해당 프로젝트를 통해 JavaScript 문법을 주도적으로 익히며 전반적인 지식을 습득했습니다. 또한, DOM 접근과 비동기 통신을 이해하고 활용할 수 있게 되었습니다.

사용 기술

HTML5, CSS3, JavaScript

구현 사항

- 모바일 Navigation drawer 아코디언 메뉴
- 퀵 메뉴 Top 버튼
- 메인 페이지 Visual Section 스kip 버튼
- JSON 비동기 처리를 통한 메뉴 안내 서브 페이지 및 하위 템 메뉴
- Kakao map API를 활용한 매장 지도

WHIPPED

WHIPPED 웹사이트 리디자인

개인 프로젝트 기여도 100%

2023.12.04 ~ 2023.12.08

배포 링크 | spaceyoung.github.io/whipped/

Github | github.com/spaceyoung/whipped

다양한 Viewport에 따른 화면 변화에 중점을 두고 리디자인을 진행했습니다. 해당 프로젝트를 통해 반응형 웹의 레이아웃 설계를 익힐 수 있었습니다.

사용 기술

HTML5, CSS3, JavaScript

구현 사항

- 총 7개의 Viewport에 대응하는 반응형 디자인

Edutations

충남대학교 천문우주과학과(정보통계학 복수전공)

2018.03 ~ 2023.02(학사 졸업)

(산대특) 뷰(Vue.js)를 활용한 프로그래시브 웹 앱(PWA) 개발자 양성 과정

2023.09.18 ~ 2024.02.27(진행 중)



민영 · 4일



공부 및 작업 기록

Vue.js를 활용한 PWA 개발자 양성 과정(산대특)

2023.09.18 ~ 2024.02.26(856시간)

▶ 프로젝트

모바일에서는 화면을 옆(오른쪽)으로 넘기면 다음 기록
이 이어집니다 😊

작업한 프로젝트 목록

Github | <https://github.com/spaceyoung>

[Vue.js]

- 공책 Note, Book
github.com/spaceyoung/notebook
- 오늘의 할 일 To Do List
spaceyoung.github.io/TodoList/

[JavaScript]

- 푸라닭 웹사이트(반응형)
spaceyoung.github.io/puradak/

[HTML & CSS]

- WHIPPED 웹사이트(반응형)
spaceyoung.github.io/whipped/
- U+ 유모바일 웹사이트 클론 코딩(적응형)
spaceyoung.github.io/uplus-umobile/

2024.01.22 자체평가 제출

JavaScript (2023.12.04~2023.12.29)

2023.12.26 ~ 2023.12.29

[작업 내용]

- 푸라닭 반응형 웹사이트 리뉴얼
 - 메뉴 안내 서브 페이지의 메뉴 리스트 자바스크립트로 구현 후 CSS 작업
 - 치킨 메뉴 페이지 하위의 탭 메뉴 자바스크립트 DOM 접근 방식으로 구현(반복되는 하위 페이지 HTML 줄일 수 있었음)
 - 매장 안내 지도 구현을 위한 Kakao map API 활용 방안 모색 중

[계획]

- 작업한 프로젝트들 꾸준히 보완
- 푸라닭 반응형 웹사이트 리뉴얼
 - Kakao map API로 매장 안내 지도 구현
- 추가 작업할 개인 프로젝트 구상하고 아이디어 정리

+)

자바스크립트를 배우고 직접 적용해보면서, HTML, CSS만 활용 가능했을 때 구상만 했던 동적인 기능들을 추가할 수 있었음. 그 과정에서 사용성도 디테일하게 다시 고려해 볼 수 있었던 듯! 앞으로 시작될 Vue 수업도 잘 따라갈 수 있도록 노력하기🔥

2023.12.18 ~ 2023.12.22

[작업 내용]

- 모던 자바스크립트 수업 및 교재 복습
- 지금까지 작업한 프로젝트 코드 리팩토링

Vue.js (2024.01.02~2024.01.17)

2024.01.08 ~ 2024.01.12

[작업 내용]



[작업 내용]

- Vue 수업 및 교재 복습
- To Do List
 - Vuetify3를 활용해 디자인 추가
 - 예제 수업 진도까지 기능 구현
 1. 이벤트 emitter를 사용하여 할 일 완료 토글, 남은/완료한 할 일 갯수 알림 기능
 2. 컴포넌트를 세분화하여 분리
- 독서 기록 달력 -> 독서 기록 서비스
 - 다양한 사용자를 고려해 서비스 제공 의미를 다시 생각하여 전반적인 기획 수정

원래는 위 사진처럼 Fullcalendar 라이브러리를 활용해 읽은 책을 추가하면 책 표지가 달력에 채워지는 독서 기록 달력을 기획했으나, 사용자에 따라 성취감의 시각화 측면에서 부정적 경험을 제공할 수 있다는 생각이 들어 달력 대신 리스트 형식으로 갈아엎는 것으로 결정 😂 (ex. 책을 가끔 썩 읽는 사용자는 달력이 많이 채워지지 않아 서비스 내용 이미가 전하지 않고 빠르게...)

개인 프로젝트 작업 1차 (2024.01.18~2024.02.13)

2024.02.09 ~ 2024.02.12

[작업 내용]

- 공책 Note, Book : 독서 기록 서비스
 - Github Actions를 활용한 연동 및 배포
 - 기록 삭제 재확인 모달창 추가
 - 데이터 반영을 고려한 지연 시간 설정(임시 조치)

[계획]

- 공책 Note, Book : 독서 기록 서비스
 - Github 정적 배포 시 proxy 설정 적용 방안 찾기
 - 세부적인 디자인 요소 수정
 - 컴포넌트 구조 재구성
- 이력서, 자기소개서, 포트폴리오 보완
- 프로그래머스 코딩테스트 공부

+)

연휴 기간 중 vite 정적 파일 배포 시 개발 환경에서의 proxy 설정이 적용되지 않는다는 걸 깨달았다

Github actions를 활용해서 해결하는 방안을 찾아보긴 했는데 아직 마땅한 해결책을 얻지 못했고 nginx같은 서버를 이용하는 방법도 있다는데 백엔드 쪽 지식은 많이 부족해서 더 알아보는 중

이전에 오류 때문에 실패했던 Github 연동은 성공했으니 이제 배포 환경에서의 CORS 오류만 어떻게 해결해보자!

2024.02.05 ~ 2024.02.08

[작업 내용]

개인 프로젝트 작업 (2024.02.13~2024.02.13)

2024.02.13 ~ 2024.02.13

[작업 내용]

- 공책 Note, Book : 독서 기록 서비스
 - Github, Vercel을 활용
 - 전반적인 프로젝트 커
 - Pinia 스토어 분리 및 :
 - 기록 삭제 시 재확인, :
 - router 지연 로딩 적용
- 이력서, 자기소개서, 포!

[계획]

- 공책 Note, Book : 독서
 - 파일베이스 활용하

+)
어떻게든 Github에서 프록시를 찾지 못해 결국 Vercel을 택한 이유는 가 일어날 때마다 자동으로 문에! 그렇게 며칠 고생하던 히 해결 😊

그리고 기존 프로젝트의 상상나에서 이루어지고 있고, 컴: 통일성도 없는 것 같아 구조: 금했는데..

새삼 설계의 중요성을 아주 를 염두에 두고 효율적인 코: 력하자!

notebook

Unpin Unwatch 1 Fork 0 Star 0

main	1 Branch	0 Tags	Go to file
Docs: update README.md	1685885 · 5 minutes ago	60 Commits	Add file
.github/workflows	Docs: create deploy.yml	last week	
.vscode	Initial commit	3 weeks ago	
public	Remove: remove pwa settings	last week	
src	Fix change route of record/modify form button	24 minutes ago	
.gitignore	Fix hide API&SDK	3 weeks ago	
README.md	Docs: update README.md	5 minutes ago	
index.html	Rename: rename modal file & class name	last week	
package-lock.json	Remove: remove pwa settings	last week	
package.json	Remove: remove pwa settings	last week	
vercel.json	Fix vercel settings	last week	
vite.config.js	Refactor: restructure project components	last week	
README			

공책 Note, Book

책으로 기록하는 나만의 공간

개인 프로젝트 기여도 100%

2024.01.10 ~ 2024.02.13 배포 완료
리팩토링 및 기능 보완 작업 꾸준히 진행 중

개요

공책(Note, Book)은 도서 검색 및 기록을 통해 독서 기록을 관리할 수 있는 웹 앱입니다.

2024년 새해 목표 중 하나로 '독서 습관 기르기'를 계획하여, 이에 대한 기록 및 동기부여와 CRUD 기능을 직접 구현하고 Open API와 연계해 확장하기 위해서 작업한 프로젝트입니다. 기획부터 화면 구현, 배포까지 스스로 진행하는 걸 목표로 삼았습니다.

해당 프로젝트를 통해 데이터 처리 기능과 SPA 프레임워크의 이점을 알고 경험했으며, React와 같은 다른 프레임워크에도 관심을 두게 되었습니다. 또한, 기획부터 시작해 기본적인 개발 프로세스를 따르면서 서비스 흐름에 관해 구체적으로 고민해 보고, 협업의 필요성을 몸소 느낄 수 있었던 기회였습니다.

배포 사이트 주소

<https://notebook-service.vercel.app>

사용 기술

JavaScript / Vue.js / Vuetify3 / Vue Router / Axios / Pinia / Firebase / Vercel

구현 기능

- Firebase를 활용한 회원 가입/로그인 기능(기능 보완 진행 중)
- 일라인 Open API를 활용한 도서 검색 기능
- 독서 기록 정보 CRUD 기능
- 동적 라우트를 활용한 개별 도서 정보 및 독서 기록 페이지
- 사용자의 독서 기록 정보 입력 조건 설정 및 유효성 검사
- 웹 앱을 고려한 반응형 디자인

트러블 슈팅

도서 검색 결과 랜덤 출력

문제 배경	같은 검색어를 입력해도 검색 시마다 검색 결과 리스트 순서가 일정하지 않고 랜덤하게 뒤바뀌는 현상 발생.
문제 원인	검색 결과 데이터를 받아와 저장하는 과정에서 순차적 처리가 이루어지지 않음.
해결 과정	async await으로 불러온 검색 결과 순서가 일정했기에, 받아온 데이터를 forEach 문으로 배열에 저장하는 과정에 문제가 있다고 판단. 반복문의 비동기 처리에 관하여 찾아본 결과, forEach 문은 순차 처리를 하지 않는다는 것을 깨닫고 for...of 문으로 바꾸어 문제 해결.
알게 된 점	forEach 문의 동작 원리와 배열에 비동기 작업을 실행할 때의 순차 처리와 병렬 처리 방법을 알게 됨.

도서 검색 결과, 독서 기록 입력/수정 페이지 새로고침 오류

문제 배경	도서 검색 결과, 독서 기록 입력/수정 페이지에서 새로고침을 누르면 404 페이지로 넘어가는 오류 발생.
문제 원인	Pinia를 사용해 저장한 state 객체 정보가 새로고침 시 초기화됨.
해결 과정	JavaScript로 To Do List를 만들어 볼 때 웹 스토리지 사용하여 데이터를 저장했던 방법을 떠올리고, pinia-plugin-persistedstate 라이브러리를 추가 설치해서 데이터를 로컬 스토리지에 저장해 문제 해결.
알게 된 점	웹 스토리지 API를 통해 데이터의 지속성을 유지하는 방법을 다시 한번 상기할 수 있었습니다.

구현 화면

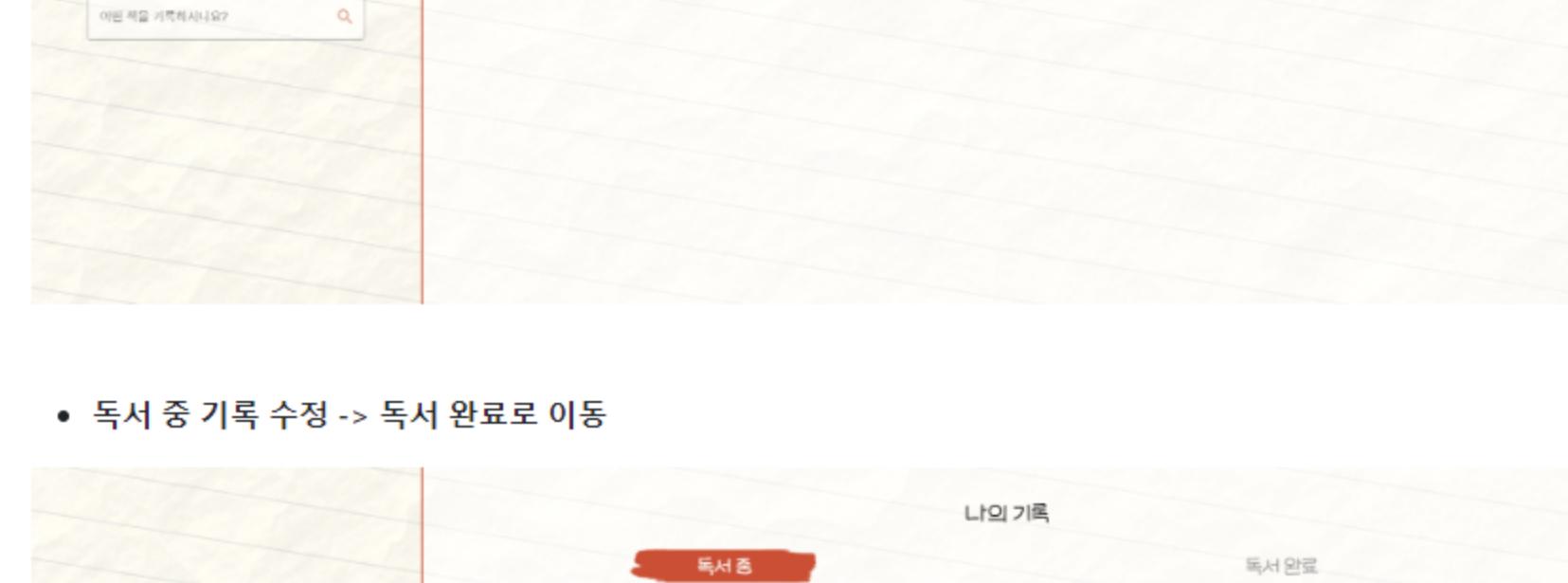
회원 가입 페이지



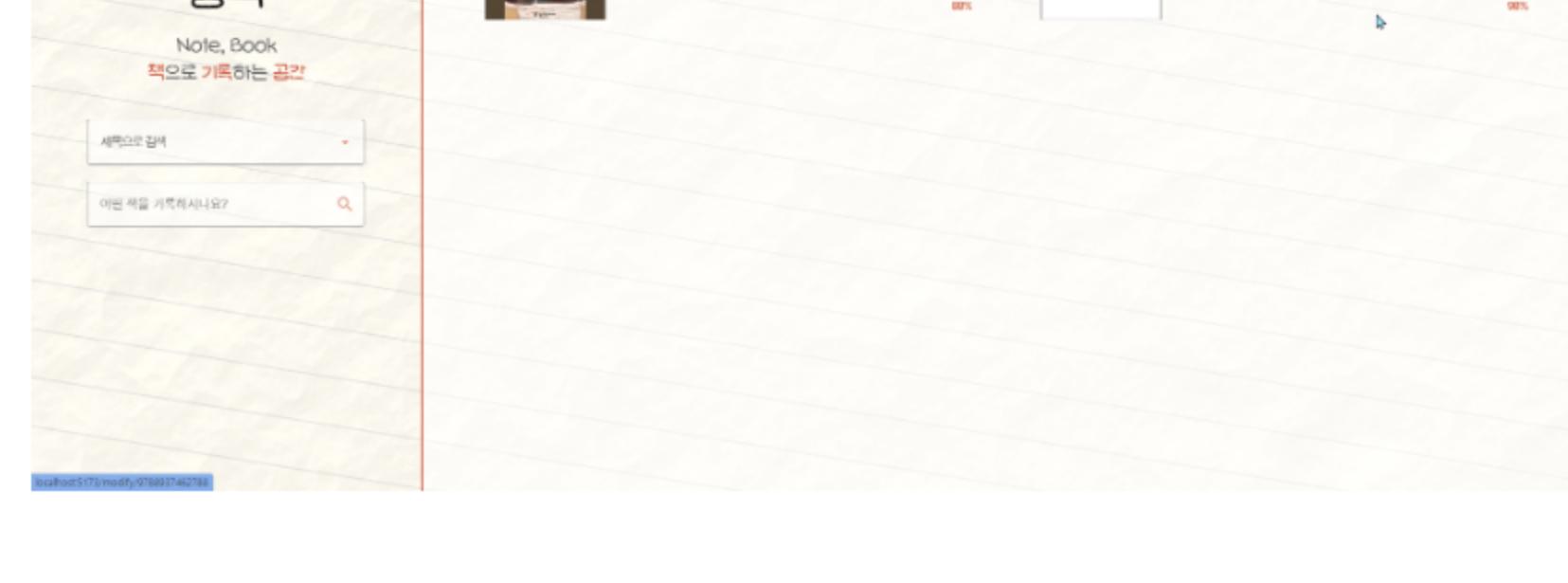
로그인 페이지



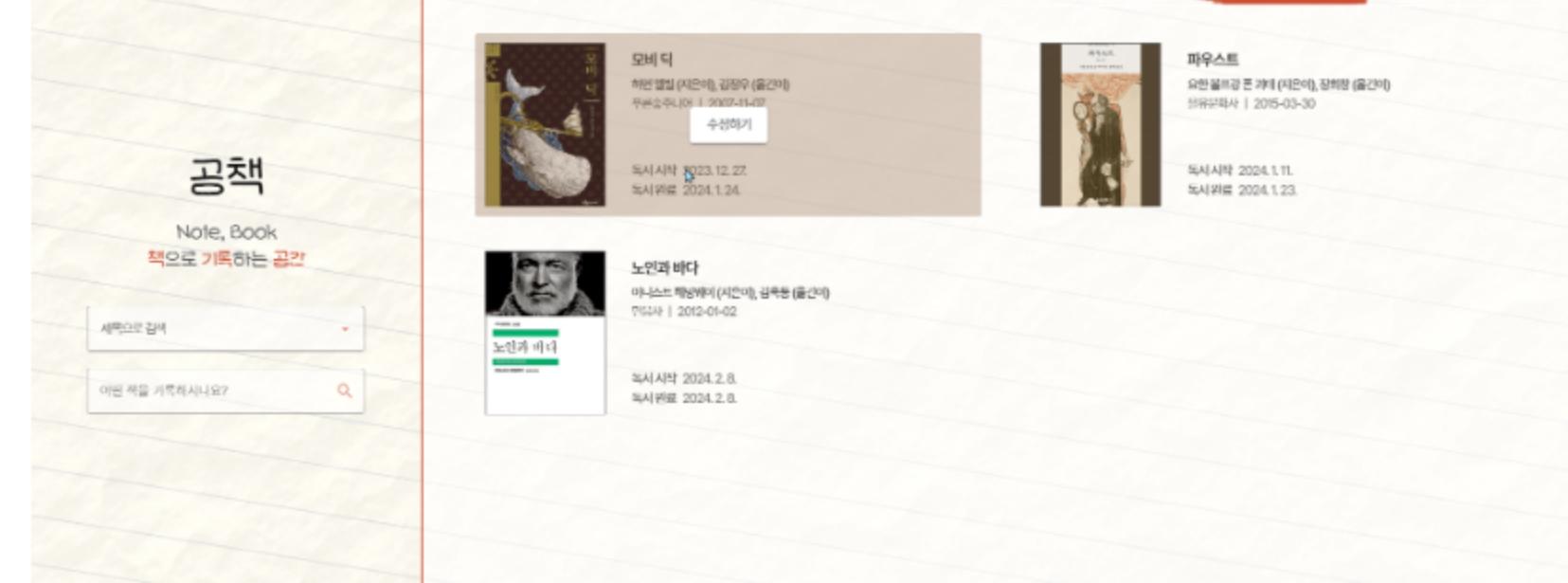
My 페이지



검색 및 기록 저장



독서 중 기록 수정 -> 독서 완료로 이동



독서 완료 기록 수정 -> 독서 중으로 이동



About

독서 기록 서비스 공책

[notebook-service.vercel.app](#)

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Deployments

65

[github-pages](#)

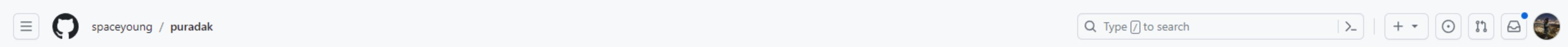
[Production](#) 22 minutes ago

+ 63 deployments

Languages

Vue 67.9% JavaScript 19.0%

CSS 12.5% HTML 0.6%



spaceyoung	Docs: Update README.md	9232759 · now	25 Commits
css	Refactor: move menu-list css	2 weeks ago	
img	Rename: rename sub01 visual img	2 weeks ago	
js	Fix: fix header hover effect	2 weeks ago	
json	Initial commit	3 weeks ago	
sub	Feat: add sub page js	2 weeks ago	
README.md	Docs: Update README.md	now	
index.html	Initial commit	3 weeks ago	

README

PURADAK Website Renewal

푸라닭 웹사이트 리뉴얼(반응형)

개인 프로젝트 기여도 100%

1차(HTML5 & CSS3) 2023.11.20 ~ 2023.12.01
2차(JavaScript) 2023.12.13 ~ 2024.01.02

개요

푸라닭의 공식 웹사이트 디자인이 고급스러움을 강조하는 브랜드 이미지를 충분히 전달하고 있지 않다고 판단하여 기획했으며, JavaScript를 학습하면서 직접 적용해 본 첫 프로젝트입니다.

기능적 측면에서는 동적 기능 구현과 단순 반복 마크업 감소, 지도 API 활용을 중점에 두었습니다. 특히 마크업 단순 반복으로 생성한 메뉴 리스트를 JSON으로 정리한 후, 반복문으로 자동화하여 탭 메뉴 페이지 5장을 SPA 형식과 같이 1장으로 감소시켰습니다.

해당 프로젝트를 통해 JavaScript 문법을 주도적으로 익히며 전반적인 지식을 습득했습니다. 또한, DOM 접근과 비동기 통신을 이해하고 활용할 수 있게 되었습니다.

배포 사이트 주소

<https://spaceyoung.github.io/puradak/>

사용 기술

HTML5 CSS3 JavaScript

구현 페이지

- 메인 페이지
- 서브 페이지 7장
 - 브랜드 소개 - 푸라닭 소개
 - 메뉴 안내 - 치킨 메뉴, 사이드 메뉴, 푸레스트, 페이커리, P-피자
 - 매장 안내 - 매장 안내

JavaScript 활용 구현 기능

- 모바일 Navigation drawer 아코디언 메뉴
- 퀵 메뉴 Top 버튼
- 메인 페이지 Visual Section 스kip 버튼
- 메뉴 안내 각 서브 페이지 및 하위 탭 메뉴
- Kakao map API를 활용한 매장 지도

About

푸라닭 웹사이트 리뉴얼(반응형)

spaceyoung.github.io/puradak/

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

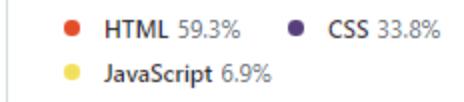
[Publish your first package](#)

Deployments 25

github-pages 2 weeks ago

+ 24 deployments

Languages



main Branch Tags Go to file Add Code

spaceyoung Docs: update README.md 010edf · now 9 Commits

css Design: fix hover effect 3 weeks ago

img Initial commit 3 weeks ago

js Initial commit 3 weeks ago

README.md Docs: update README.md now

index.html Feat: add txt-box title hover effect 3 weeks ago

README

WHIPPED Website Redesign

WHIPPED 웹사이트 리디자인(반응형)

개인 프로젝트 기여도 100%

2023.12.04 ~ 2023.12.08

개요

CSS의 Flex 기능을 적용한 후, 다양한 Viewport에 따른 메인 페이지 화면 변화에 중점을 두고 리디자인을 진행한 개인 프로젝트입니다.

해당 프로젝트를 통해 반응형 웹의 레이아웃 설계를 익힐 수 있었습니다.

배포 사이트 주소

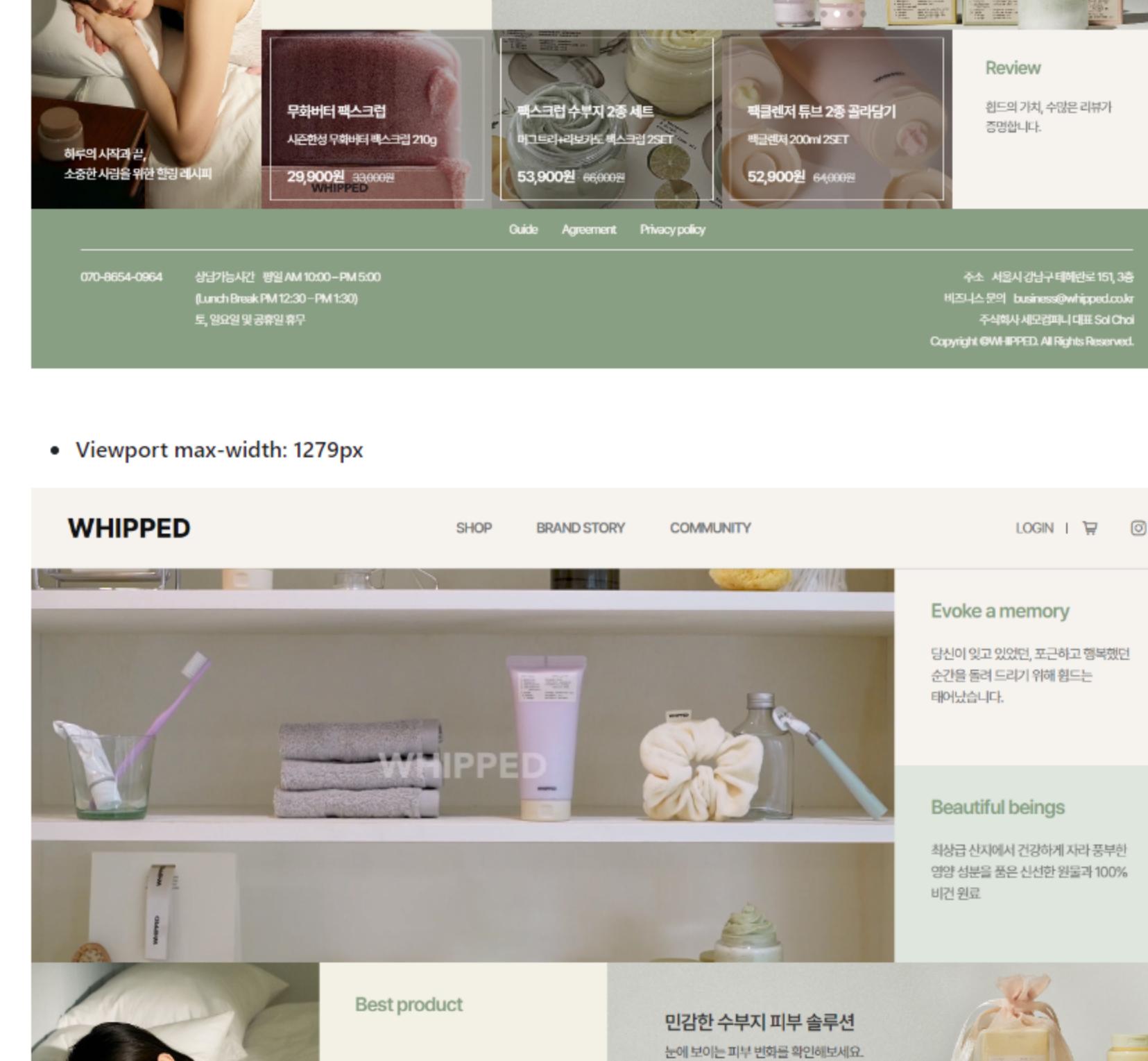
<https://spaceyoung.github.io/whipped/>

사용 기술

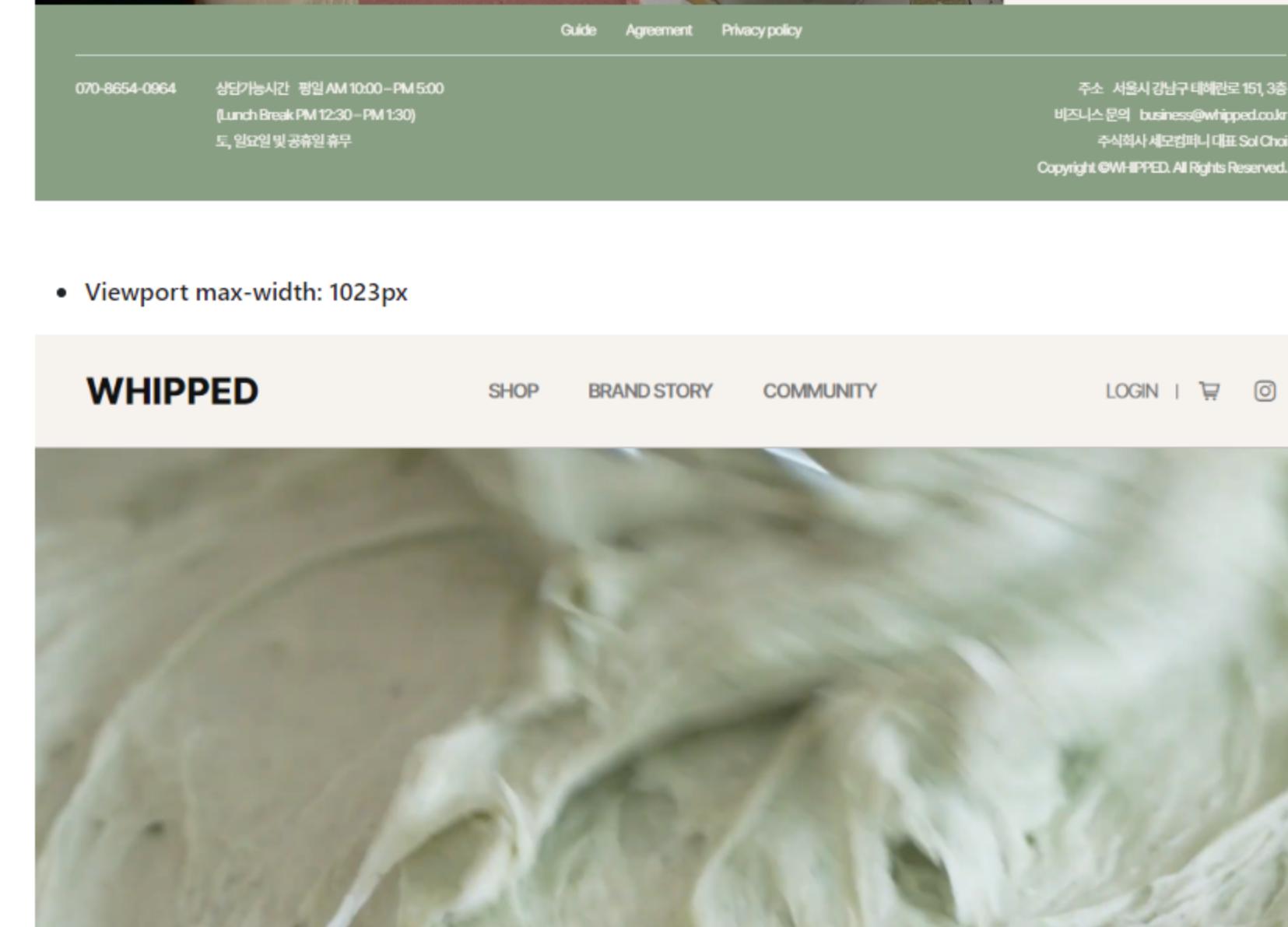
HTML CSS JavaScript

구현 화면

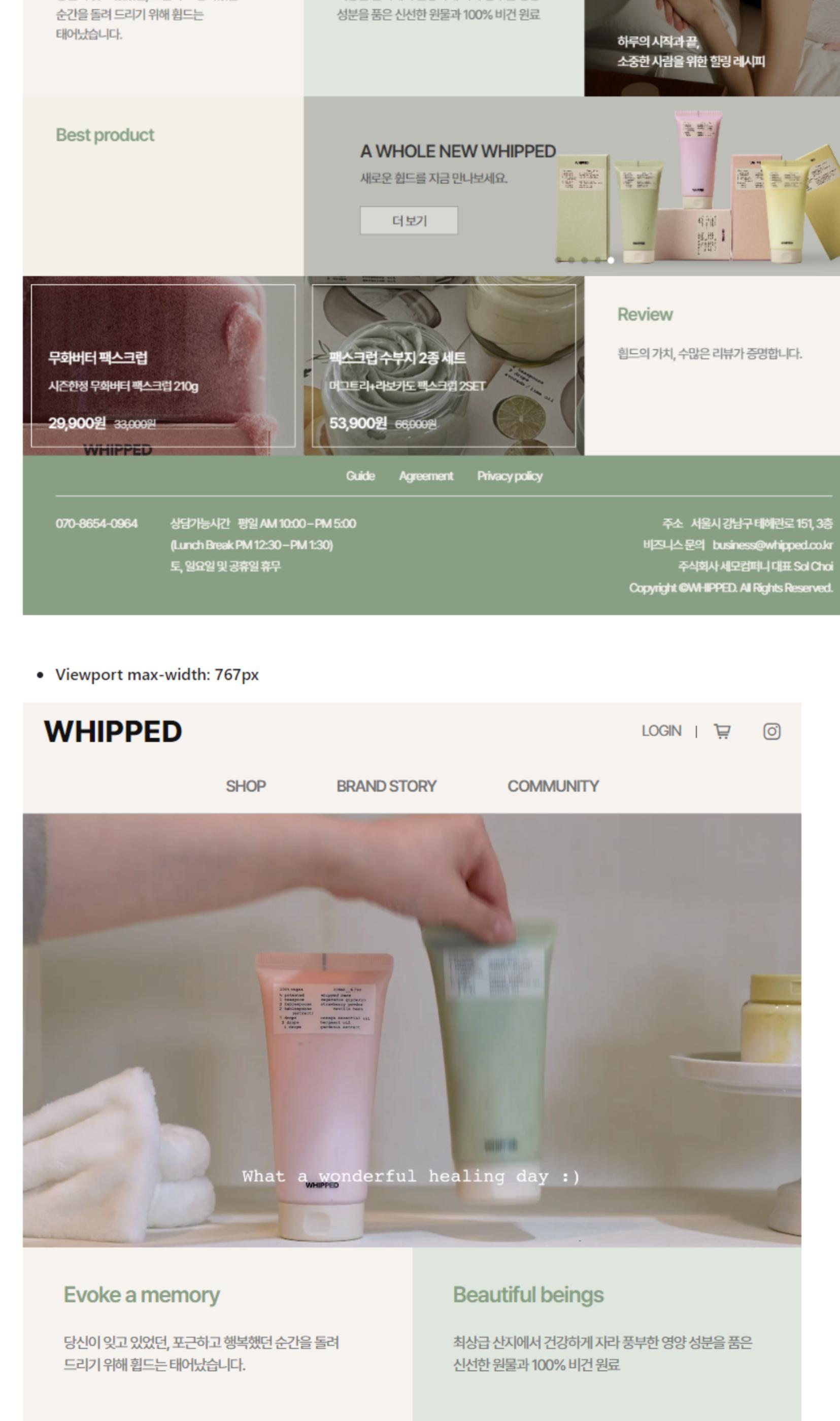
- Viewport max-width: 1920px



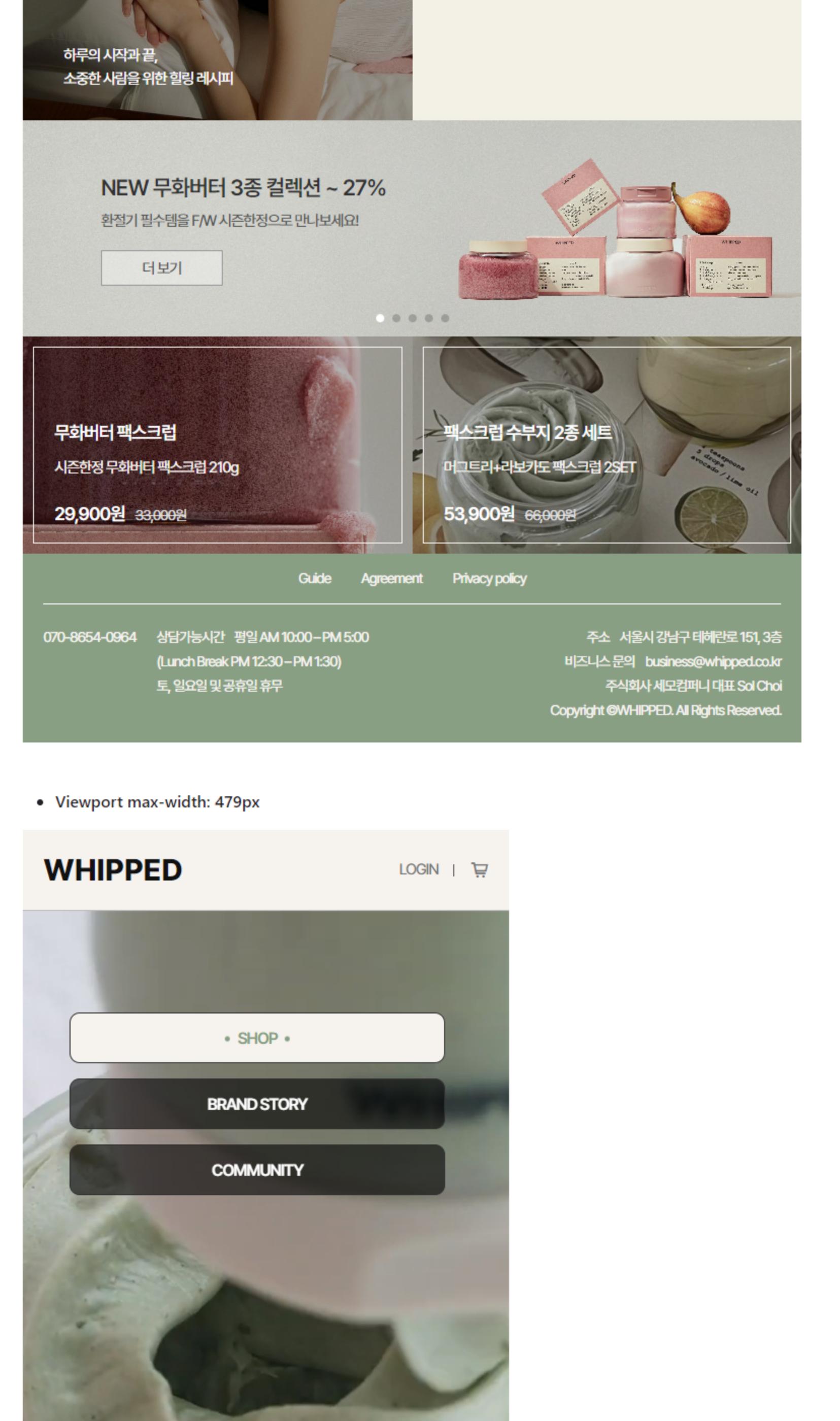
- Viewport max-width: 1399px



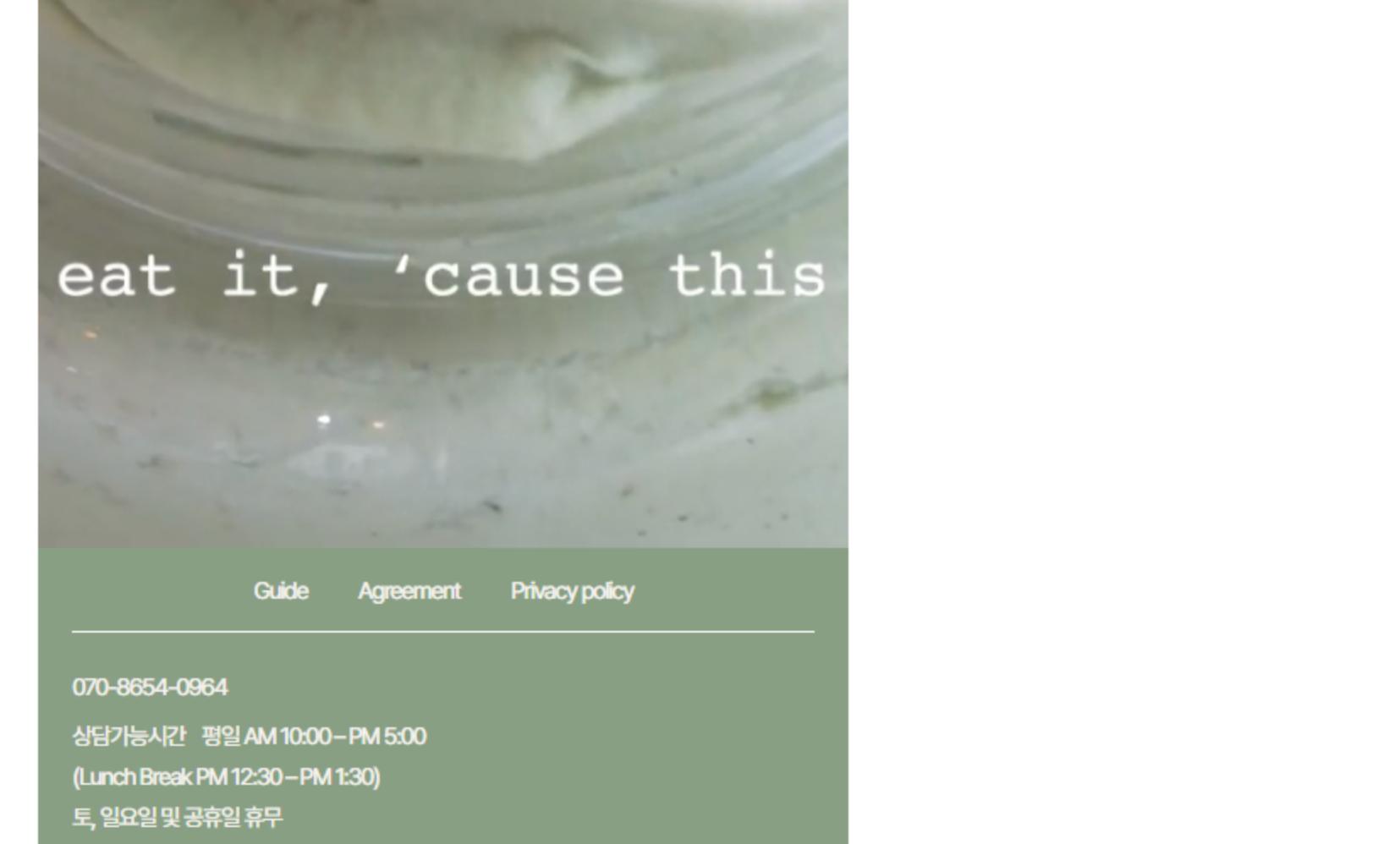
- Viewport max-width: 1279px



- Viewport max-width: 1023px



- Viewport max-width: 479px



About

WHIPPED 웹사이트 리뉴얼(반응형)

spaceyoung.github.io/whipped/

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

[Publish your first package](#)

Deployments

[github-pages](#) 2 weeks ago

+ 6 deployments

Languages

CSS 65.7% HTML 33.4%

JavaScript 0.9%