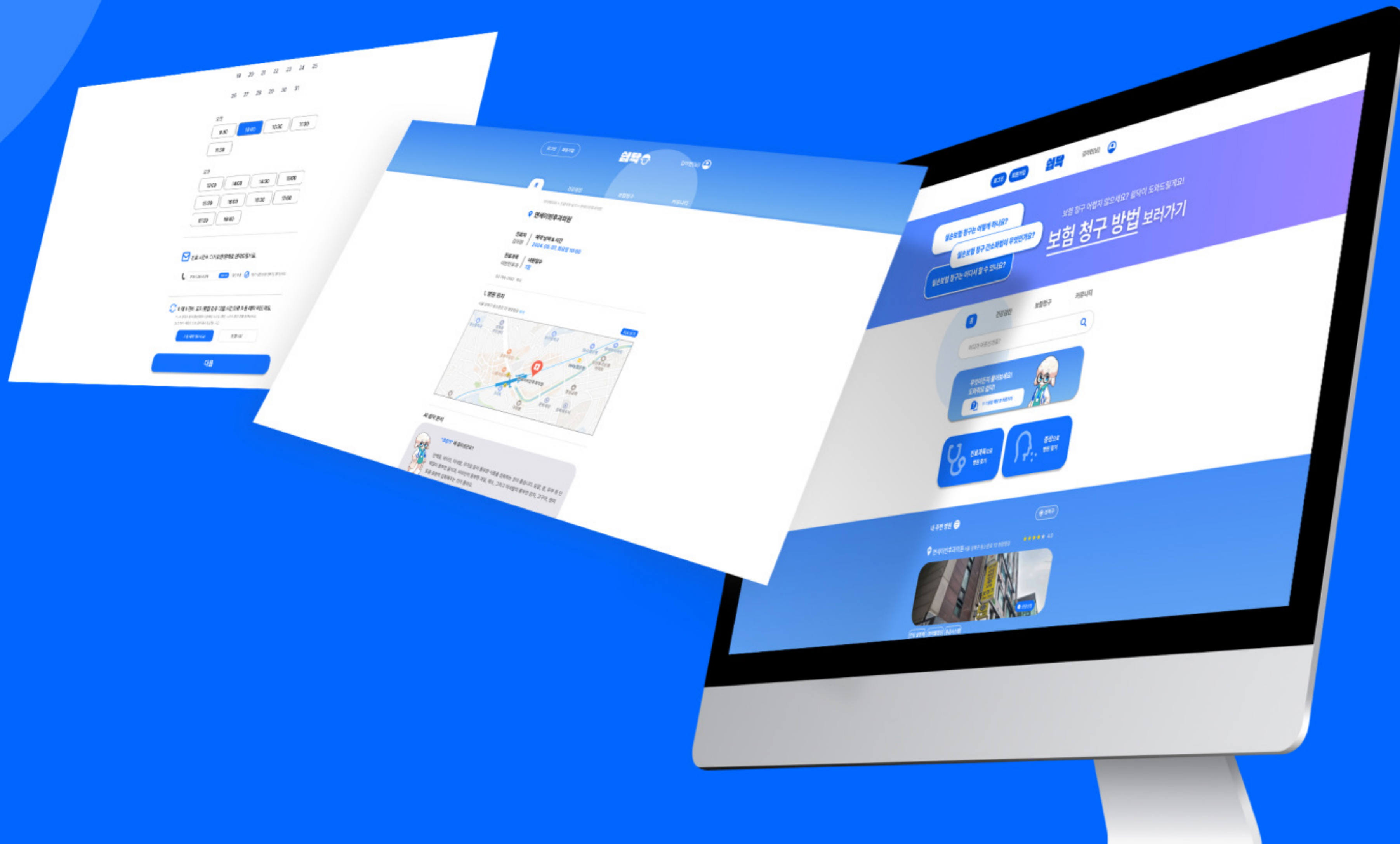


심막

병원 추천부터 병원 예약, 건강검진 예약까지
쉽게 건강을 관리할 수 있는 통합 헬스케어 플랫폼



PM 김아현
Designer 황정원
Front-end 김태하, 윤예진
Back-end 김정은, 김준하, 이준수

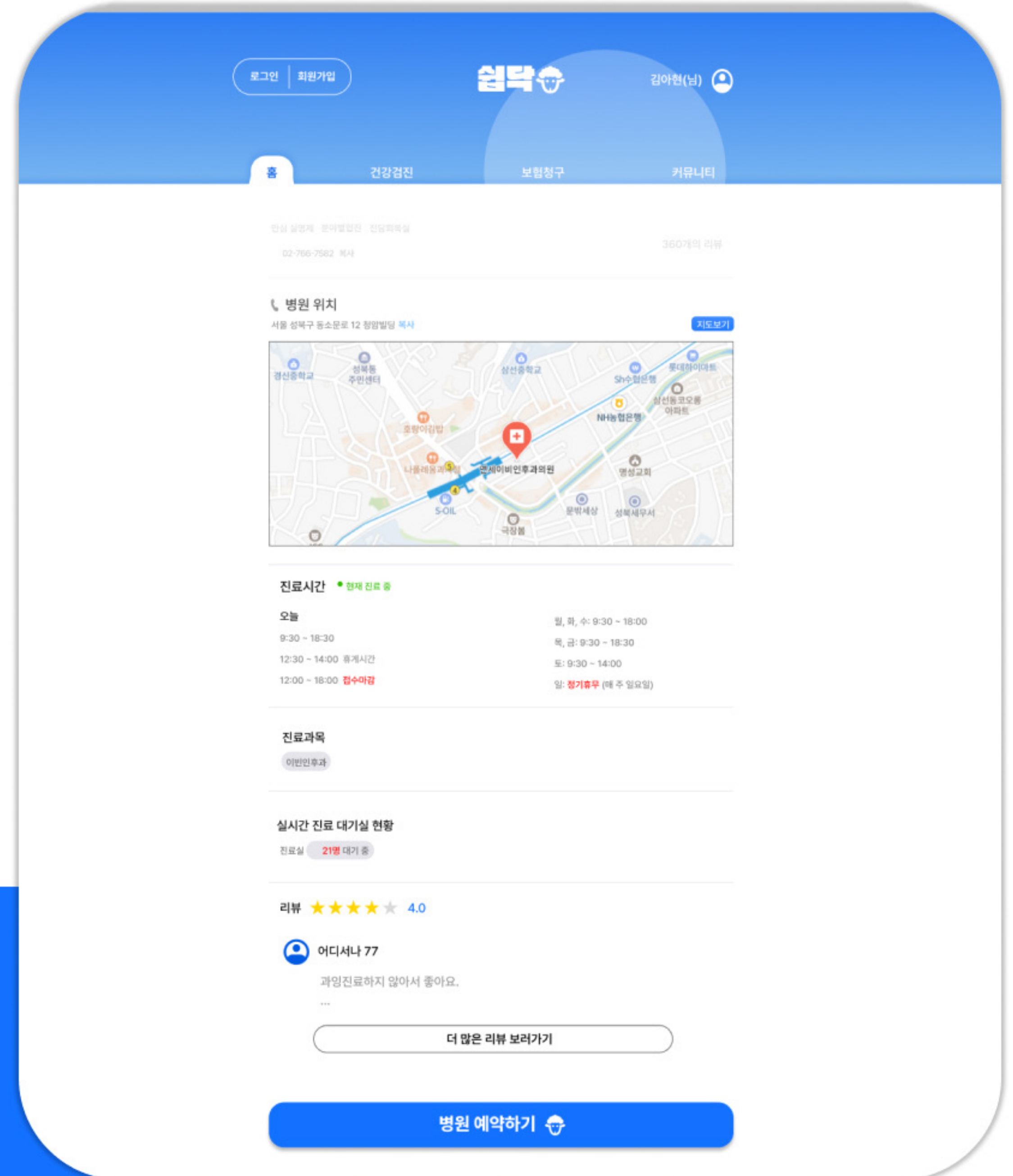
< react-kakao-maps-sdk

카카오 Developers에서 받은 APP 키를 등록 후 index.html에 적용 후 사용

```
import { Map, MapMarker } from "react-kakao-maps-sdk";

const HospitalMap = (props) => {
  const [coordinate, setCoordinate] = useState({
    lat: 37.5665, // 기본 값: 서울의 위도
    lng: 126.978, // 기본 값: 서울의 경도
  });
};
```

```
<MapContainer>
  <Map
    center={coordinate}
    style={{ width: "100%", height: "360px" }}
  >
    <MapMarker position={coordinate}></MapMarker>
  </Map>
</MapContainer>
```



< react-calendar

```
import Calendar from "react-calendar";
```

```
<StyledCalendar
  value={selectedDate}
  onChange={handleDateChange}
  formatDay={({ locale, date }) => moment(date).format("D")}
  formatYear={({ locale, date }) => moment(date).format("YYYY")}
  formatMonthYear={({ locale, date }) =>
    moment(date).format("YYYY. MM")}
  }
  calendarType="gregory"
  showNeighboringMonth={false}
  next2Label={null}
  prev2Label={null}
  minDetail="year"
  activeStartDate={
    activeStartDate === null ? undefined : activeStartDate
  }
  onActiveStartDateChange={({ activeStartDate }) =>
    setActiveStartDate(activeStartDate)
  }
  tileDisabled={({ date, view }) =>
    view === "month" &&
    date < today &&
    !moment(date).isSame(today, "day")
  }
  tileContent={({ date, view }) => {
    let html = [];
    if (
      selectedDate &&
      view === "month" &&
      date.getTime() === selectedDate.getTime()
    ) {
      html.push(<StyledToday key={"key"} src={Vector} />);
    }
    return <{html}</>;
  }}
/>
```

김아현

병원

진료자 변경하기

예약 날짜와 시간을 선택해 주세요.

<

2024. 05

>

일

월

화

수

목

금

토

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

오전

9:30

10:00

10:30

11:00

11:30

오후

12:00

14:00

14:30

15:00

15:30

16:00

16:30

17:00

17:30

18:00

진료 시간이 다가오면 문자로 연락드릴게요.

010-1234-5678

번호하기

수신 허용

예약 시간 30분 전에 알려드릴게요

예약 시간에 오지 못할 경우 다음 시간으로 자동 예약 해드려요.

*노쇼 방지 비용이 발생해요! 3번 이상 노쇼일 경우, 노쇼 비용은 환불 불가능해요.
(노쇼 방지 비용은 진료 금액에서 절감됩니다.)

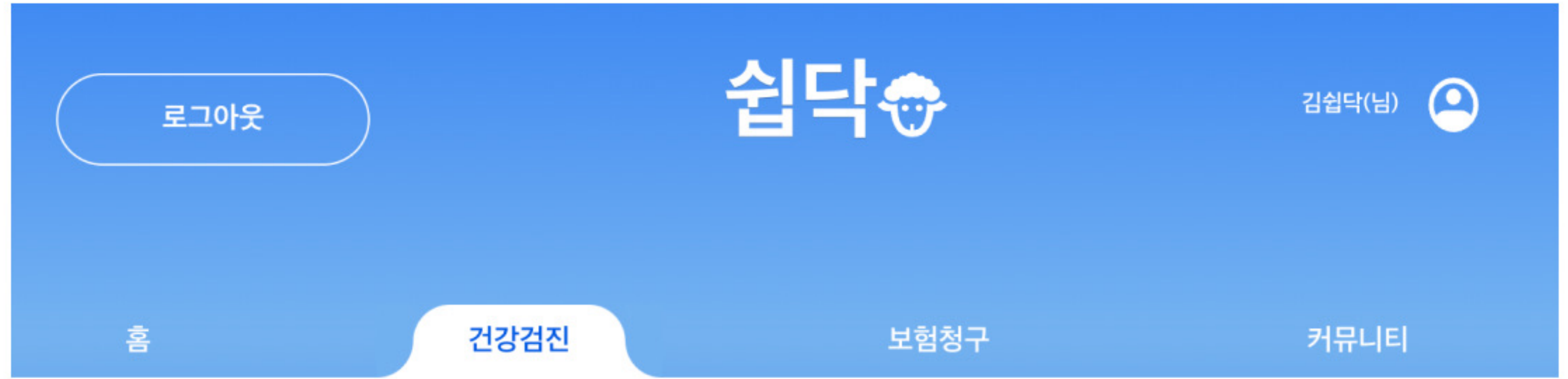
자동 예약 해주세요!

관찮아요!

다음

< NavBar 디자인

```
<Nav>
  <ul>
    <NavItem
      active={currentIndex === 0}
      onClick={() => handleItemClick(0)}
    >
      <LeftCurve />
      <RightCurve />
      <Link to="/home">홈</Link>
    </NavItem>
  </ul>
</Nav>
```



```
const LeftCurve = styled.b`
  position: absolute;
  bottom: 0px;
  left: -20px;
  height: 100%;
  width: 20px;
  background: #fff;
  display: none;

  &::before {
    content: "";
    top: 0;
    left: 0;
    position: absolute;
    width: 100%;
    height: 100%;
    border-bottom-right-radius: 20px;
    background: #72b1f0;
  }
`;
```

```
const RightCurve = styled.b`
  position: absolute;
  right: -20px;
  top: 0px;
  height: 100%;
  width: 20px;
  background: #fff;
  display: none;

  &::before {
    content: "";
    top: 0;
    left: 0;
    position: absolute;
    width: 100%;
    height: 100%;
    border-bottom-left-radius: 20px;
    background: #72b1f0;
  }
`;
```

```
const NavItem = styled.li`
  position: relative;
  padding: 10px 30px;
  font-size: 1rem;
  border-radius: 20px 20px 0 0;
  ${({ active }) =>
    active
      ? "background : #fff; color : #005BE2; > b {display:block;}"
      : "background : transparent; color : #fff;"}

  cursor: pointer;

  a {
    color: inherit;
    text-decoration: none;
  }

  &:hover {
    background: #fff;
    color: #005be2;

    ${({ LeftCurve, RightCurve }) {
      display: block;
    }}
  }
`;
```

< Data의 유무에 따른 페이지 이동

data가 null이거나 길이가 0일 때 / data 값이 있을 때

```
const fetchReservations = async () => {  
  const data = await getReservations();  
  console.log(data);  
  if (data.result === null || data.result.reservations.length === 0) {  
    navigate("/mypage/apointmentNull");  
  } else {  
    setReservations(data.result.reservations);  
  }  
};
```

병원 예약내역 보기 (예약내역 X)



병원 예약내역 보기 (예약내역 X)



< 더보기 Button

prevSize에 +3 하는 방식

```
export default function SearchFooter({ symptom }) {  
  const { location, error } = useGeolocation(geolocationOptions);  
  const [hospitalList, setHospitalList] = useState([]);  
  const [modal, setModal] = useState(false);  
  const [sortOption, setSortOption] = useState("REVIEW");  
  const [displaySortOption, setDisplaySortOption] = useState("리뷰 많은 순");  
  const [size, setSize] = useState(3);  
  const modalRef = useRef();
```

```
const handleLoadMore = () => {  
  setSize((prevSize) => prevSize + 3);  
};
```

```
</HospitalListContainer>  
<SearchButton onClick={handleLoadMore}>더보기</SearchButton>
```



< 정렬하기

option ⇒ 사용자가 선택한 정렬 옵션

sortOptionsMap 사용

selectedSortOption ⇒ 실제 API 요청에 사용할 정렬 옵션

```
const sortOptionsMap = {  
  "가까운 순": "DISTANCE",  
  "별점 높은 순": "SCORE",  
  "리뷰 많은 순": "REVIEW",  
};
```

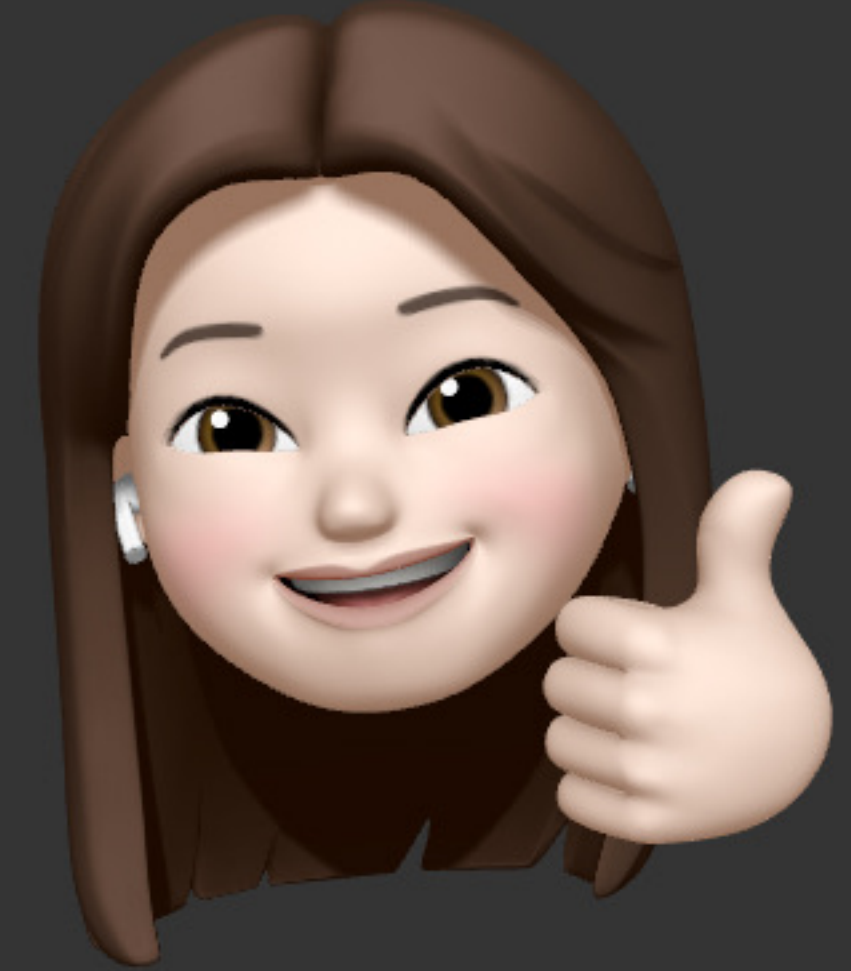
```
const [sortOption, setSortOption] = useState("REVIEW");  
const [displaySortOption, setDisplaySortOption] = useState("리뷰 많은 순");
```

```
const handleSelectOption = (option) => {  
  const selectedSortOption = sortOptionsMap[option];  
  setSortOption(selectedSortOption);  
  setDisplaySortOption(option);  
  setModal(false);  
};
```



< 부기톤을 통해 향상된 점

협업에서의 소통하는 방법과 프로젝트를 진행해 나가는 방법을 배울 수 있었다.
첫 프로젝트에서 좋은 성과를 거둬고 동시에 팀원들로부터 긍정적인 자극을 많이 얻어서
앞으로 리액트 공부에 더욱 매진해야겠다는 다짐과 함께 새로운 도전을 할 수 있는 용기를
얻게 되었다. 지식적 측면에서는 특히 통신하는 과정에 대해 많이 배웠다.



리액트를 활용한 프론트엔드 첫 프로젝트인 만큼 팀원 및 백엔드 팀원과의 협업 능력이 향상
됐다고 생각한다.
뿐만 아니라 디테일한 디자인 및 한 페이지에서 상황에 맞는 여러 디자인이 있었기 때문에
CSS를 다루는 방법이 향상된 것 같다.

THANK YOU

PM	김아현
Designer	황정원
Front-end	김태하, 윤예진
Back-end	김정은, 김준하, 이준수