

처음하는 API + React

김한비

도전

토큰이 뭐야?

어색한 API

처음보는 리액트

보여줄 건 별로 없지만

초기 데이터

인원 필터링

기술공유

모듈화

Styled Component

모듈화 - 라우터

```
1  const router = require('express').Router();
2  const controller = require('./auth.controller');
3  const { authUser, authToken } = require('../..../javascripts/middleware/auth.middle');
4
5  router.post('/register', controller.register);
6  router.post('/login', authUser, controller.login);
7  router.get('/check', authToken, controller.check);
8  router.post('/logout', controller.logout);
9
10 module.exports = router;
11
```

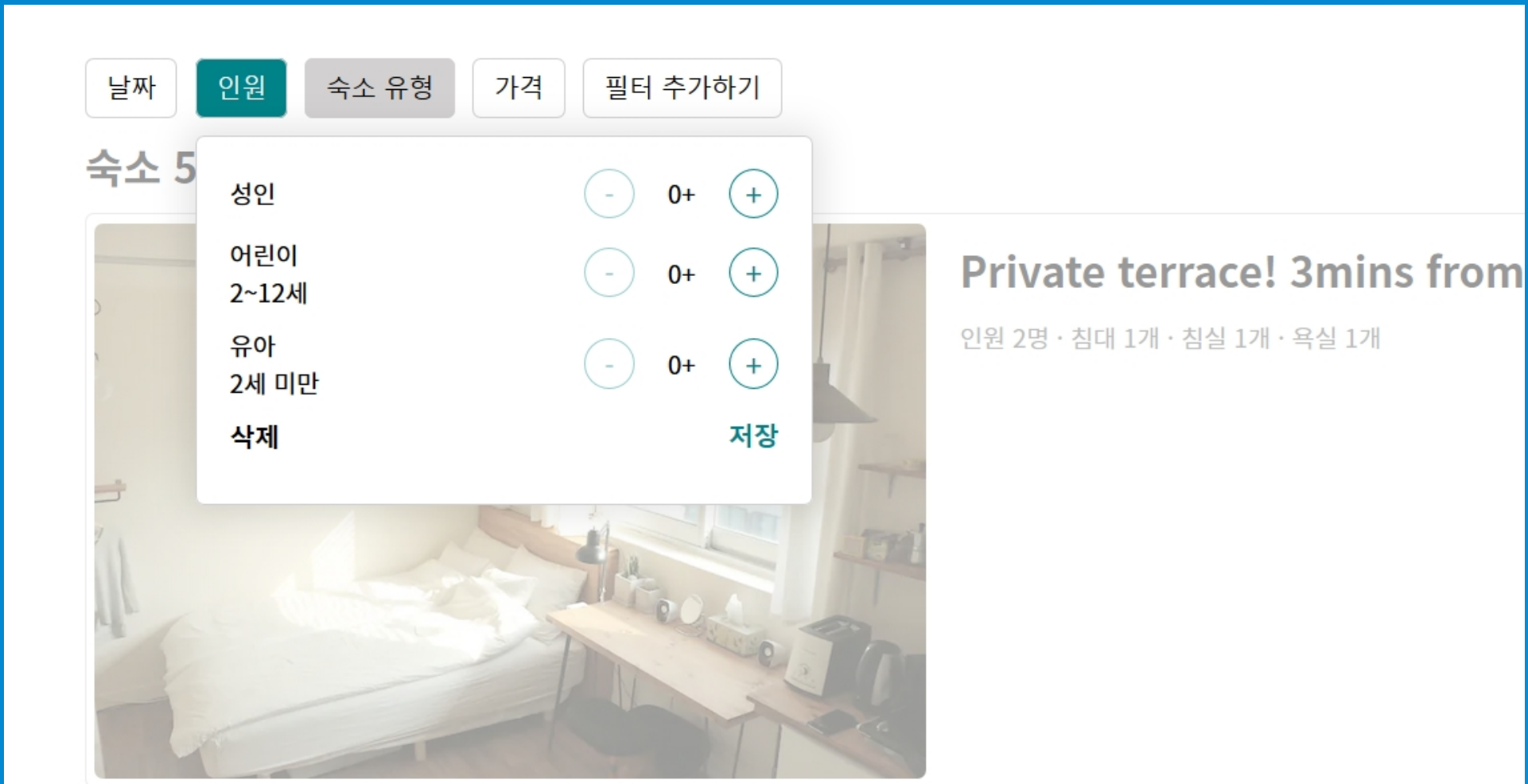
모듈화 - 미들웨어

```
1  /* eslint-disable camelcase */
2  const util_api = require('../util/util_api');
3  const util_jwt = require('../util/util_jwt');
4  const util_cookie = require('../util/util_cookie');
5  const models = require('../../database/models');
6
7  const authUser = async (req, res, next) => {
8    const { uid, password } = req.body;
9    try {
10     const user = await models.User.findOne({ where: { uid } });
11     const userInfo = user ? user.dataValues : null;
12
13     if (!user) throw new Error('User not exist'); // 입력된 uid에 해당하는 유저가 없는 경우
14     if (!models.User.verify(password, userInfo.salt, userInfo.password)) throw new Error('Password not match');
15
16     const token = await util_jwt.makeJWT(req, userInfo);
17     req.token = token;
18     res.cookie('token', req.token, util_cookie.TOKEN_COOKIE_OPTIONS);
19     next();
20   } catch (err) {
21     util_api.respondError(res, err);
22   }
23 };
24
```

모듈화 - 컨트롤러

```
1  /* eslint-disable camelcase */
2  const util_api = require('../../../../javascripts/util/util_api');
3
4  const authController = {
5    // 회원가입api
6    register(req, res) {
7      res.send('register api is working');
8    },
9
10   // 로그인 api
11   async login(req, res) {
12     util_api.respondSuccess(res, 'Login Success', req.token);
13   },
14
15   // 토큰인증 api
16   async check(req, res) {
17     util_api.respondSuccess(res, 'Authentication Suceess');
18   },
19
20   logout(req, res) {
21     res.clearCookie('token');
22     res.redirect('/');
23   },
24 }
```

수많은 버튼...



Styled Component

스타일이 추가된 컴포넌트

컴포넌트? 재사용

컴포넌트? pass props

버튼

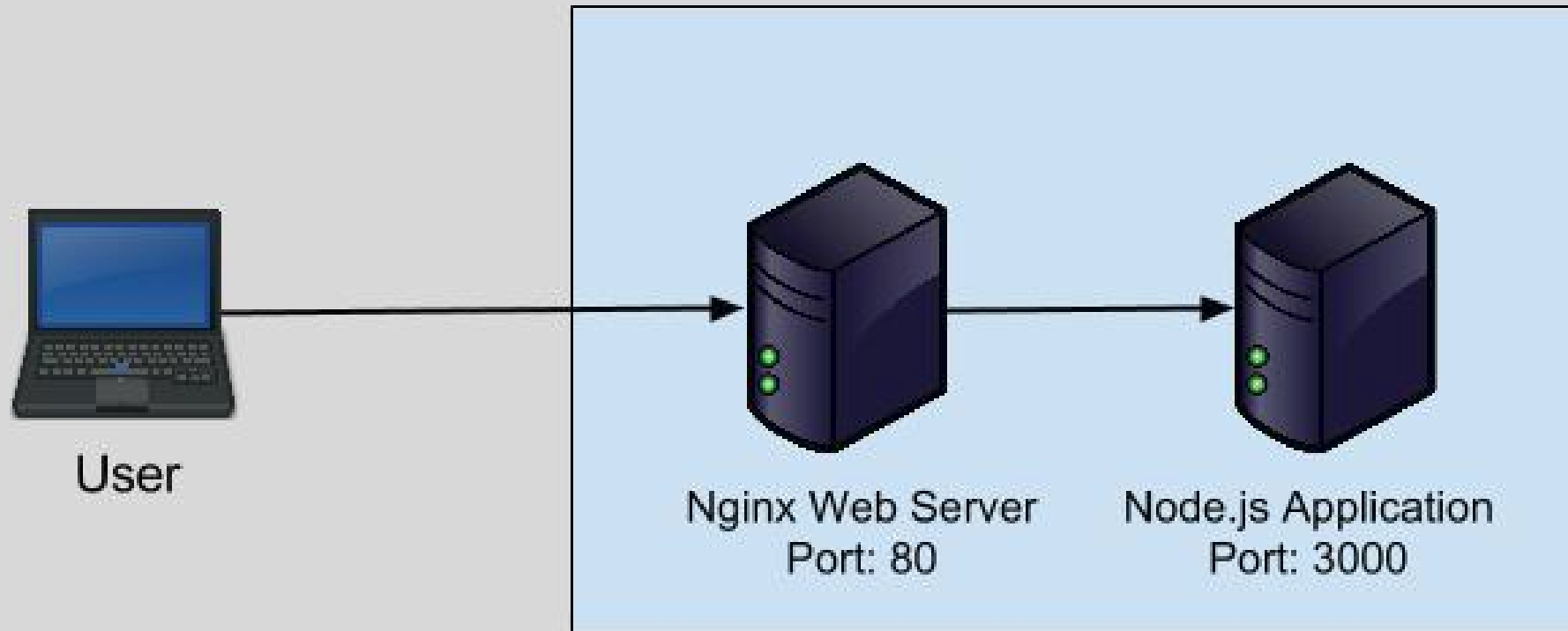
```
38
39   const Button = styled.button`
40     background-color: #FFFFFF;
41     font-size: 1.5rem;
42     cursor: pointer;
43     &:disabled {
44       opacity: 0.4
45     }
46   `;
47
48   const PlusButton = styled(Button)`
49     color: ${props => props.theme.buttonClickColor};
50     border: 1px solid ${props => props.theme.buttonClickColor};
51     border-radius: 50%;
52     width: 3rem;
53     height: 3rem;
54   `;
55
56   const MinusButton = styled(Button).attrs((props) => ({
57     disabled: (props.number <= 0 || props.needAdult),
58   })))`
59     color: ${props => props.theme.buttonClickColor};
60     border: 1px solid ${props => props.theme.buttonClickColor};
61     border-radius: 50%;
62     width: 3rem;
63     height: 3rem;
64   `;
65
```

어려웠던 배포

프록시 서버의 개념

nginx + react + express

동작방식의 이해



Setup Nginx as Frontend Server for Node.js Application

이해하니 쉬워요

빌드파일 생성

nginx 권한부여

프록시 설정

API서버 구동

뛰어나진 못해도

리액트 개념 토론

매일 질문

사소한 것 알려주기

아쉬움

