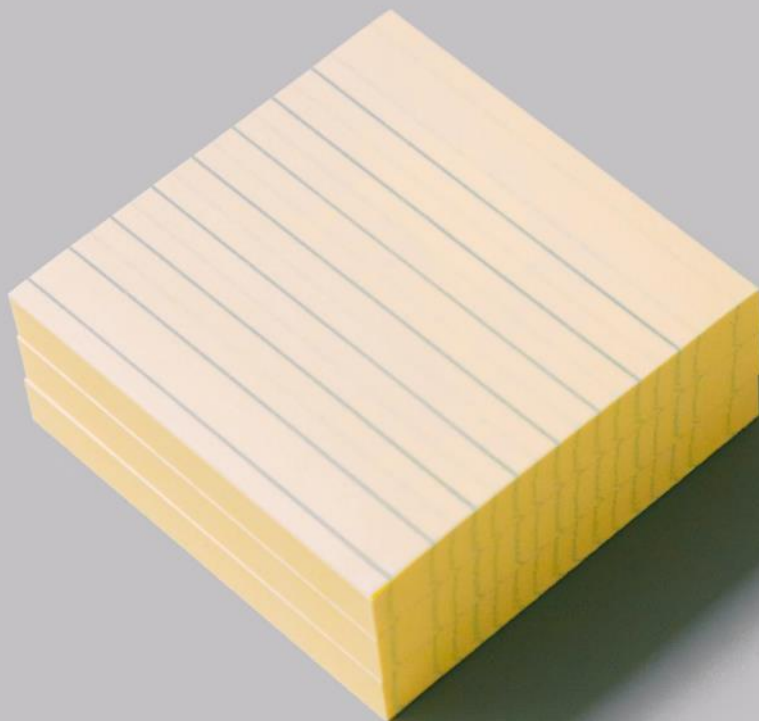


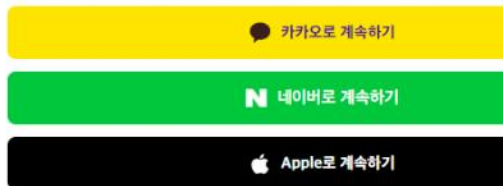
[



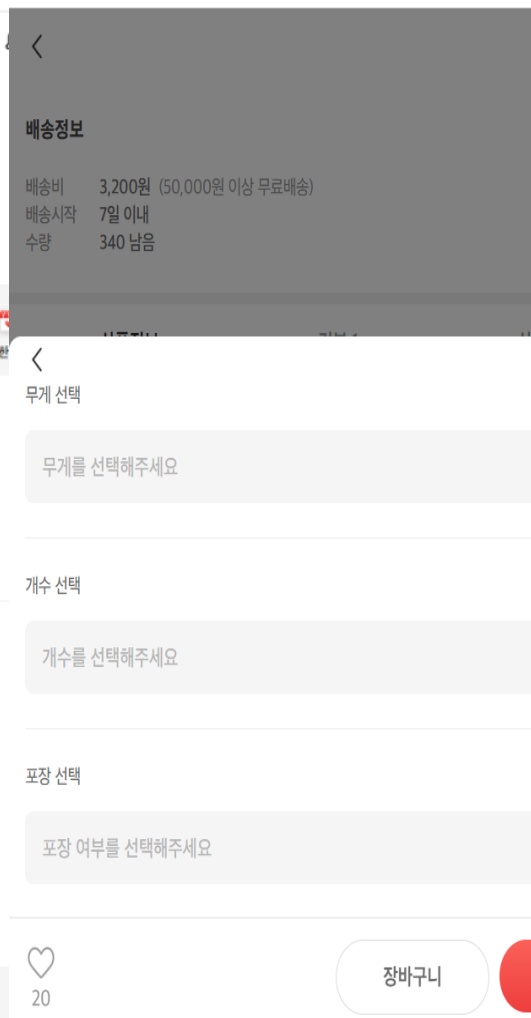
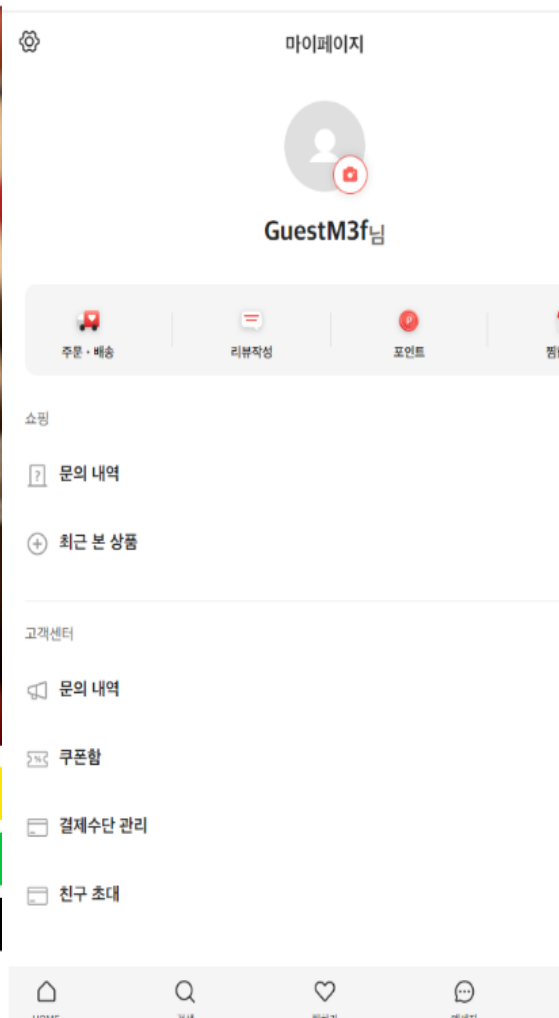
]

E조

WebApp 과일거래 중개 플랫폼



로그인 전 둘러보기



마일스톤

태그	기능	페이지	상세 기능	기능 코드	설명	착수일	완료일	수정일
1103	검색	검색 페이지	상품 검색	SE-01	주어진 조건에 따라 상품을 검색할 수 있음	2022-10-27	2022-10-31	
1103	검색	검색 페이지	검색 결과 정렬	SE-02	주어진 조건에 따라 검색된 상품을 정렬할 수 있음	2022-10-31	-	
1103	검색	검색 페이지	상점 검색	SE-03	상점 이름으로 상점을 검색할 수 있음	2022-11-01	2022-11-02	
1103	검색	검색 페이지	필터링 하여 검색	SE-04	검색을 할 때 추가적인 필터링 기능을 제공할 수 있음	2022-10-31	2022-11-02	
1103	검색	검색 페이지	퍼블리싱	SE-05	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1103	검색	검색 상세 페이지 - 상품	퍼블리싱	SE-06	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1103	검색	검색 페이지	필터링 아이콘 클릭 시 필터링	SE-07	BEST, 할인 중과 같은 아이콘 클릭 시, 해당 필터로 검색된 결과를 받을 수 있음	2022-11-02	2022-11-03	
1103	유지보수	검색 페이지	필터를 전역으로 관리	SE-08	터 조건을 전역으로 관리하여, 새로고침 해도 필터링 된 결과를 받을 수 있음	2022-11-02	2022-11-03	
1103	검색	검색 상세 페이지 - 상점	퍼블리싱	SE-07	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	메인	메인 페이지	퍼블리싱	MN-01	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1103	메인	오늘의 추천 상품 페이지	퍼블리싱	MN-02	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	메인	라이브 페이지	퍼블리싱	MN-03	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	상품 받아오기	상품상세 페이지	상품 정보 받아오기	PO-01	클릭했을 때, 해당 상품의 정보 (id, 가격, 할인을 등)을 받아서 렌더링 할 수 있음	2022-10-23	2022-10-23	
1103	상품 받아오기	상품상세 페이지	퍼블리싱	PO-02	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1103	상품 받아오기	상품상세 페이지 - 구매 옵션	퍼블리싱	PO-03	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1103	상품 받아오기	상품상세 페이지	리뷰 받아오기	PO-04	상품에 작성된 리뷰를 받아와 렌더링 할 수 있음	2022-10-23	2022-10-24	
1103	상품 받아오기	상품상세 페이지	상품문의 받아오기	PO-05	상품에 작성된 문의사항을 받아와 렌더링 할 수 있음	2022-10-23	2022-10-24	
1103	상품 받아오기	상품상세 페이지	무게, 개수, 포장 선택 후 구매	PO-06	상품 구매 시, 무게 개수 포장 선택 할 수 있음	2022-11-02	2022-11-03	
1103	장바구니 담기	장바구니 페이지	장바구니 받아오기	SC-01	장바구니에 담긴 상품을 받아오고 렌더링 할 수 있음	2022-11-02	2022-11-02	
1103	장바구니 담기	장바구니 페이지	장바구니 담기/삭제	SC-02	장바구니에 상품을 담고 삭제할 수 있음	2022-11-02	2022-11-02	
1103	장바구니 담기	장바구니 페이지	퍼블리싱	SC-03	반응형으로 퍼블리싱 됨	2022-11-01	2022-11-02	
1103	찜하기	공통	상품 찜 하기	LK-01	관심 있는 상품에 찜하기 및 취소 할 수 있음	2022-10-25	2022-10-29	
1103	찜하기	찜하기 페이지	찜한 정보 받아오기	LK-02	유저가 찜한 상품을 받아와 렌더링 할 수 있음	2022-10-25	2022-10-29	
1103	찜하기	찜하기 페이지	퍼블리싱	LK-03	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	로그인	로그인 페이지	Oauth 2.0 카카오 로그인	LG-01	SNS 로그인을 통해 유저의 정보를 받아올 수 있음	2022-10-21	2022-10-25	
1103	로그인	로그인 페이지	익명 로그인	LG-02	회원 가입을 원치 않는 유저는 익명 로그인을 통해 서비스를 이용할 수 있음	2022-10-21	2022-10-25	
1103	회원가입	랜딩 페이지	퍼블리싱	SU-01	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	회원가입	약관 동의 페이지	퍼블리싱	SU-02	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	회원가입	약관 동의 페이지	약관 동의	SU-03	유저는 약관에 동의한 후 페이지 넘길 수 있음	2022-10-25	2022-10-30	
1103	회원가입	회원가입 페이지	퍼블리싱	SU-04	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-27	
1103	회원가입	회원가입 페이지	닉네임, 이메일, 번호 설정	SU-05	회원가입 시 닉네임, 이메일, 번호를 입력받아 서버에 저장되어야함	2022-10-30	2022-10-30	
1103	회원가입	회원가입 페이지	휴대폰 번호 인증	SU-06	코드 전송을 통해 해당 번호가 본인 번호가 맞는지 확인하는 절차가 있음	-	-	

마일스톤

태그	기능	페이지	상세 기능	기능 코드	설명	착수일	완료일	수정일
1117	유지보수	검색 페이지	필터 쿼리로 전송	SE-05	필터 검색 시, 필터 내용을 쿼리로 만들어 GET 요청할 수 있음 (유지보수)	2022-10-28	2022-11-01	
1117	내 정보 수정	마이페이지	정보 수정	MS-01	프로필, 닉네임, 이메일을 수정할 수 있음	2022-10-25	2022-11-02	
1117	내 정보 수정	마이페이지	알림 설정	MS-02	알림을 켜고 끌 수 있음			
1117	내 정보 수정	마이페이지	퍼블리싱	MS-03	반응형으로 퍼블리싱 됨	2022-10-23	2022-10-25	
1117	내 정보 수정	마이페이지 - 환경설정	퍼블리싱	MS-04	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	
1117	리뷰 작성/읽기	마이페이지	리뷰 작성	RV-01	구매한 상품에 관하여 리뷰를 작성할 수 있음	2022-11-07	2022-11-08	
1117	리뷰 작성/읽기	마이페이지	작성한 리뷰 받아오기	RV-02	유저가 작성한 리뷰 목록을 불러와 렌더링 할 수 있음	2022-11-02	2022-11-05	
1117	리뷰 작성/읽기	마이페이지	리뷰 작성 가능한 상품 받아오기	RV-03	구매는 했지만 아직 리뷰 작성을 안한 상품을 받아와 렌더링 할 수 있음	2022-11-02	2022-11-05	
1117	리뷰 작성/읽기	마이페이지 - 리뷰작성목록	퍼블리싱	RV-04	반응형으로 퍼블리싱 됨	2022-11-02	2022-11-10	
1117	리뷰 작성/읽기	마이페이지 - 리뷰작성	퍼블리싱	RV-05	반응형으로 퍼블리싱 됨	2022-11-02	2022-11-10	
1117	상품 주문	주문/결제 페이지	주문 전) 주문 정보 받아오기	OD-01	어떤 주문을 했는지 받아오고 렌더링 할 수 있음	2022-11-01	2022-11-02	
1117	상품 주문	주문/결제 페이지	배송 정보 입력	OD-02	배송 받을 주소를 입력할 수 있음	2022-11-08	2022-11-09	
1117	상품 주문	주문/결제 페이지	주문 정보 전송하기	OD-03	주문한 정보를 서버에 전송함	2022-11-08	2022-11-11	
1117	상품 주문	주문/결제 페이지	퍼블리싱	OD-05	반응형으로 퍼블리싱 됨	2022-10-17	2022-11-01	
1117	상품 주문	주문/결제 페이지 - 완료	퍼블리싱	OD-06	반응형으로 퍼블리싱 됨	2022-10-17	2022-11-01	
1117	상품 주문	마이페이지	주문 후) 주문 내역 받아오기	OD-07	주문을 했던 상품들을 불러오고 렌더링 할 수 있음			
1117	상품 주문	마이페이지	주문하고 상품 불러오기	OD-08	주문을 했던 상품들을 불러오고 렌더링 할 수 있음			
1117	상품 주문	주문/결제 페이지	옵션 정보 불러오기	OD-09	주문할 상품에 대하여 옵션 정보를 받아올 수 있음	2022-11-02	2022-11-03	
1117	상품 주문	주문/결제 페이지	할인 정보 불러오기	OD-10	결제 전, 할인 정보(쿠폰, 포인트)를 받아올 수 있음			
1117	상품 주문	주문/결제 페이지	할인 정보 설정하기	OD-11	결제 전, 할인 정보(쿠폰, 포인트)를 설정할 수 있음			
1117	상품 주문	주문/결제 페이지	결제 정보 불러오기	OD-12	결제 전, 결제 정보(카드 결제, 계좌이체, 가상계좌, 현금) 받아올 수 있음			
1117	상품 주문	주문/결제 페이지	결제 정보 설정하기	OD-13	결제 전, 결제 정보(카드 결제, 계좌이체, 가상계좌, 현금) 설정할 수 있음			
1117	스토어 받아오기	스토어페이지	스토어 정보 받아오기	ST-01	스토어 정보를 받아와 렌더링 할 수 있음			
1117	스토어 받아오기	스토어페이지	스토어 팔로우	ST-02	관심 있는 스토어를 팔로우 혹은 취소할 수 있음			
1117	스토어 받아오기	스토어페이지	퍼블리싱	ST-03	반응형으로 퍼블리싱 됨	2022-10-17	2022-10-27	

완료된 기능구현 필요 기능

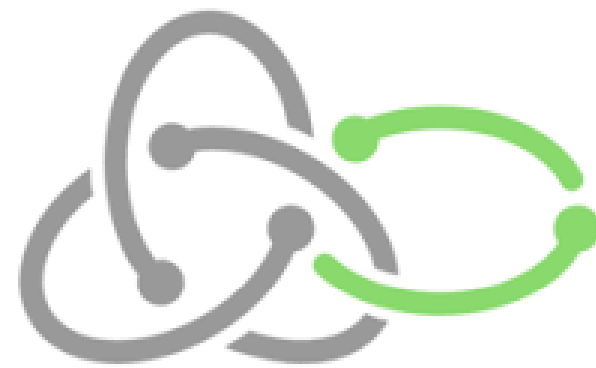
49

18

프론트엔드와 파이어베이스와 연동중



firebase



Redux-Saga

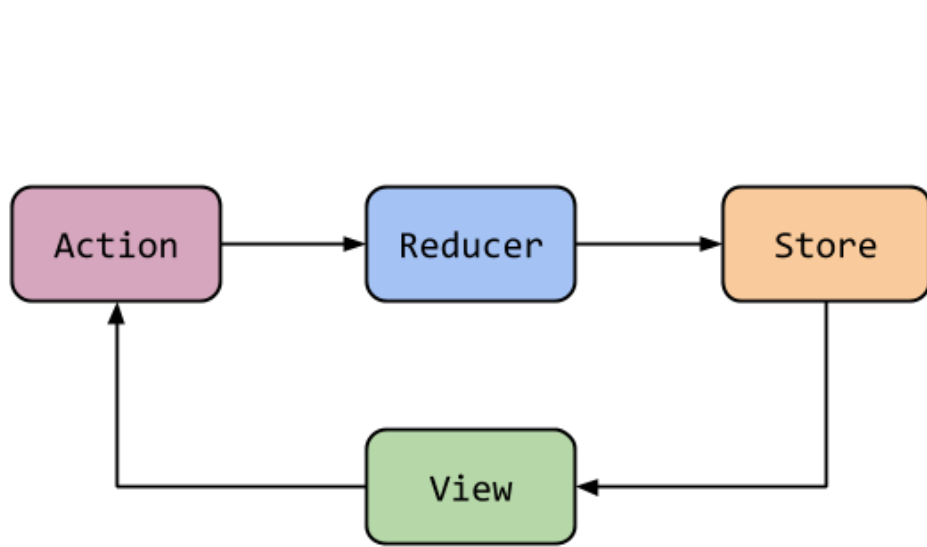
Redux-saga란?

Redux-saga is a library that aims to make application side effects easier to manage, more efficient to execute, easy to test, and better at handling failures.

Ex) Asynchronous things like data fetching and impure things like accessing the browser cache

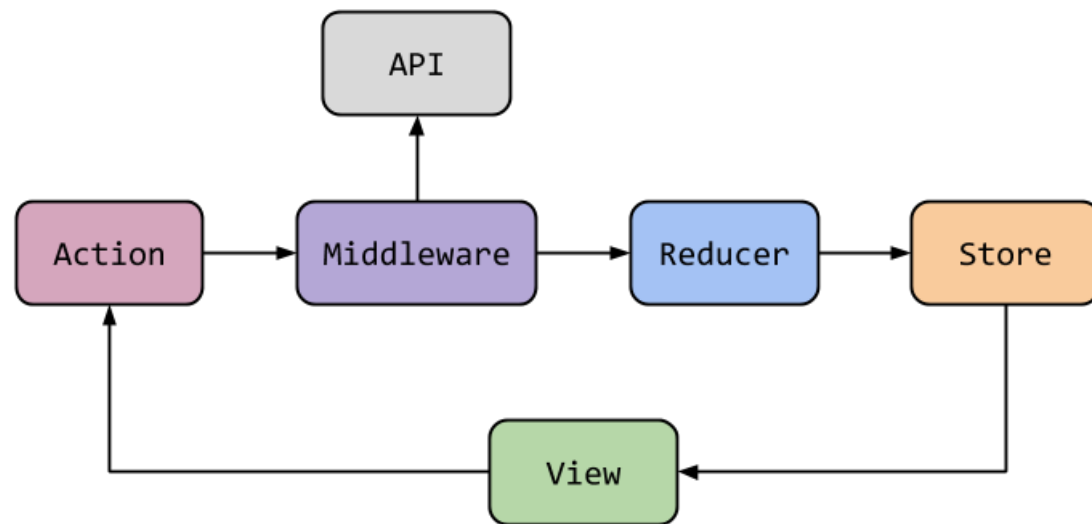
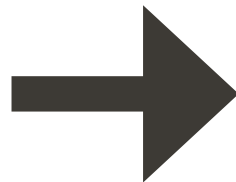


Redux와 Redux-saga



Redux only

리덕스만 사용시 액션이 동기 처리됨



Redux + Redux-saga

중간의 미들웨어(Redux-saga)가 비동기 처리를 도와줌

0. 최초 실행

오늘의 추천 상품



산천
친환경 복숭아 5kg/10kg
34000원 20%



산천
친환경 사과 3kg/5kg
34000원 10%



산천
포도 1kg/3kg
25000원 10%

>

1. 변수 `selector`에 `productInfo` 를 저장
- 아직은 기본값

2. Dispatch를 통해 `GetProductInfo` 함수 호출

```
function ProductPage(props: any) {  
  const params = useParams();  
  const location = useLocation();  
  const navigate = useNavigate();  
  const dispatch = useDispatch();  
  
  const selector: ProductDataType = useSelector((state: RootState) =>  
    state.ProductInfoReducer!.productInfo  
  );  
  
  const modalselector: any = useSelector((state: RootState) =>  
    state.PurchaseReducer.purchaseInfo  
  );  
  
  const [isOpen, setOpen] = useState(false);  
  
  useEffect(() => {  
    dispatch(closeModalAction(modalselector));  
    dispatch(GetProductInfo(params.productId));  
  }, [])  
}
```


1. Action

1. GetProductInfo 소스코드

2. 객체 { type: ~, payload: ~ } 를
return 함

```
export const GetProductInfo = (data: any) => {  
  console.log("call action" + data);  
  
  return {  
    type: GET_PRODUCT,  
    payload: data  
  }  
}
```

2-1. Redux-saga

```
export function* getProductSignal() {  
  yield takeLatest(GET_PRODUCT, productIndex);  
}
```

```
function* productIndex(action: any) {  
  switch (action.type) {  
    case GET_PRODUCT:  
      yield getProdcut(action);  
      break;  
    default:  
      break;  
  }  
}
```

1. 제너레이터 함수 `getProductSignal`는 액션 `GET_PRODUCT`이 발생하면 -> `productIndex` 함수를 호출
2. `productIndex`는 다시 `getProdcut` 함수를 호출

2-2. Redux-saga

1. productIndex 함수에 의해 호출된 getProduct 함수

2. getProductAPI를 호출하여 Firebase와 소통

```
function* getProdcut(action:any) {
  console.log("get product" + action.payload);

  const result:ProductDataType = yield call(getProdcutAPI, action.payload);

  if(result){
    console.log("result" + JSON.stringify(result));
    yield put({
      type: GET_PRODUCT_SUCCESS,
      payload: result,
      callback: action.payload.callback
    });
  }
  else{
    yield put({
      type: LOGIN_FAIL,
      callback: action.payload.callback
    });
  }
}
```

2-3. Redux-saga

```
async function getProdcutAPI(payload:any) {  
  const product_id = Number(payload);  
  
  const productRef = collection(db, "product");  
  const q = query(productRef, where("product_id", "==", product_id), limit(1));  
  const fbdata = await getDocs(q);  
  const productData = fbdata.docs[0].data();  
  
  const reviewRef = collection(db, "review");  
  const q2 = query(reviewRef, where("product_id", "==", product_id));  
  const fbdata2 = await getDocs(q2);  
  console.log(fbdata2.empty);  
}
```

3. Reducer

```
export function ProductInfoReducer(state = productInfoInitState, action: any) {  
  switch (action.type) {  
    case GET_PRODUCT_SUCCESS:  
      console.log("reducer");  
      console.log(action.payload.photoDataList);  
  
      return {  
        ...state,  
        productInfo: action.payload,  
      };  
  
    case GET_PRODUCT_FAIL:  
      return {  
        ...state,  
        productInfo: action.payload,  
      };  
  
    case GET_PRODUCT_LOADING:  
      return {  
        ...state,  
        productInfo: action.payload,  
      };  
  
    default:  
      return state;  
  }  
}
```

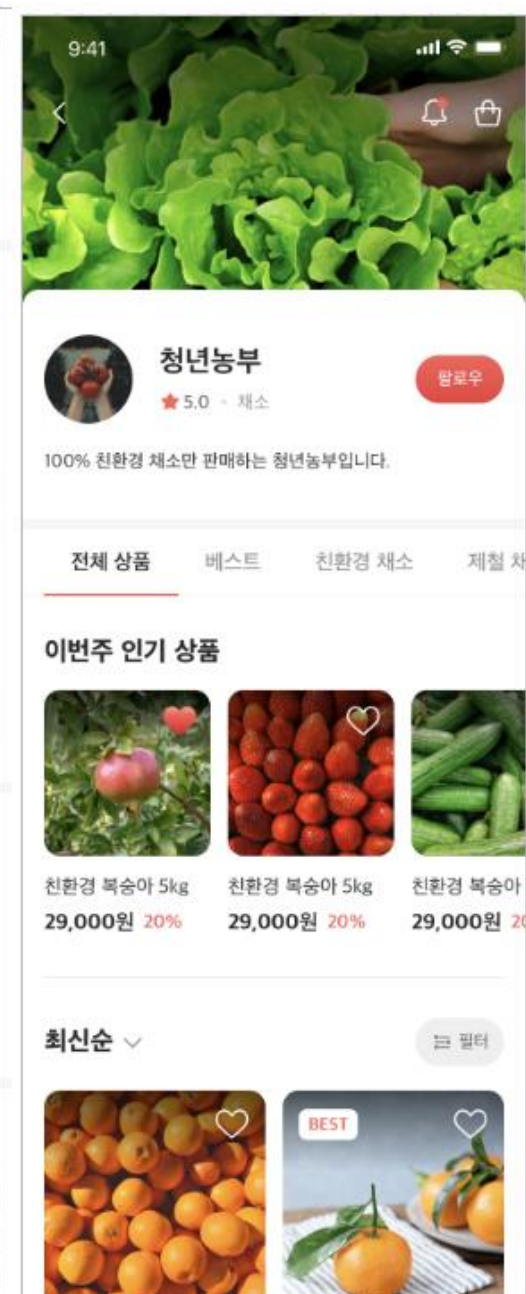
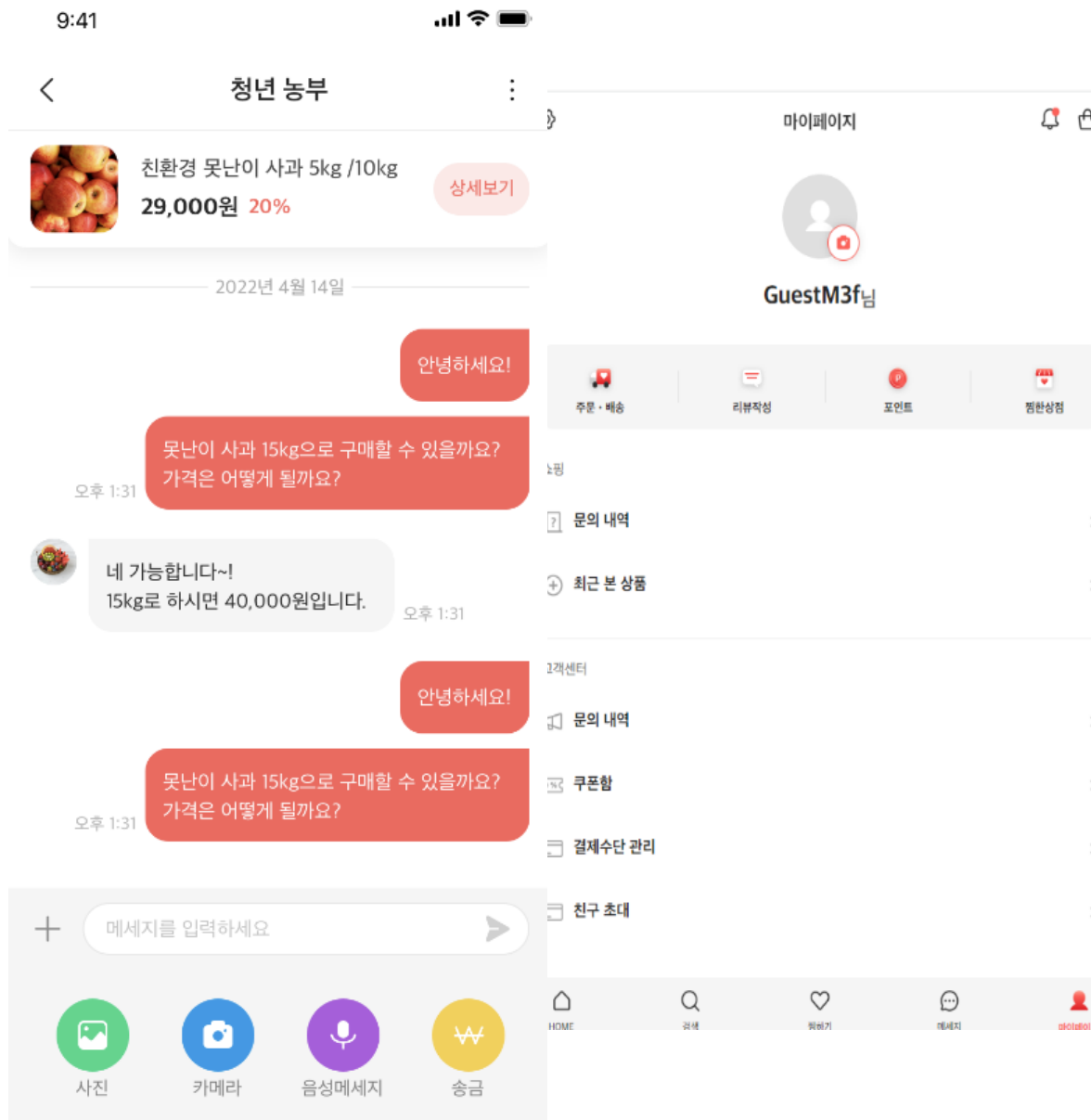
프로젝트 진행상황(프론트엔드)

Pages	
▼ pages	●
> CartPage	
> ChatPage	●
> LikePage	
> ListPage	
> LoginPage	
> MainPage	
> MyPage	
> NotiPage	
> OrderPage	
> ProductPage	
> ReviewPage	
> SearchPage	
> SplashPage	
> StorePage	

Reducers	
▼ reducers	●
TS BottomNavigationBarRed...	
TS CartReducer.ts	
TS ChatReducer.tsx	U
TS DeliveryReducer.ts	
TS DiscountReducer.ts	
TS index.ts	
TS LikeReducer.ts	
TS LoginReducer.ts	
TS MypageReducer.ts	
TS OrderReducer.ts	
TS ProductReducer.ts	
TS PurchaseReducer.ts	
TS ReviewReducer.ts	
TS SearchReducer.ts	

Sagas	
▼ sagas	
TS CartSaga.ts	
TS DeliverySaga.ts	
TS DiscountSaga.ts	
TS index.ts	
TS LikeSaga.ts	
TS LoginSaga.ts	
TS MypageSaga.ts	
TS OrderSaga.ts	
TS ProductSaga.ts	
TS ReviewSaga.ts	
TS SearchSaga.ts	

프로젝트 진행상황(프론트엔드)



토스페이 연동

거래 데이터 검증 및 허가 단계 구현 필요

결제내역

결제 승인 시각

2022/05/16 00:00

-

2022/11/16 19:36

내역 검색

전체

Q

Enter 또는 버튼을 눌러 검색하세요

전체 8

결제완료 0

결제예정 8

결제실패 0

결제취소 0

결제상태	UID/MID	상품명	고객정보	결제수단	결제대행사	결제금액	구분
결제예정	imp_282343116989 ORD20221009-kakao:2484108854-572000000	친환경 복숭아 5kg/10kg	L O G	기타(tosspay)	카카오페이	28,400 KRW	
결제예정	imp_490253359305 ORD20221009-kakao:2484108854-117000000	친환경 복숭아 5kg/10kg	dfh	신용카드	카카오페이	28,400 KRW	
결제예정	imp_120235786710 ORD20221009-kakao:2484108854-358000000	친환경 복숭아 5kg/10kg	dfh	신용카드	카카오페이	28,400 KRW	
결제예정	imp_259411754691 ORD20221009-kakao:2484108854-585000000	친환경 복숭아 5kg/10kg	dfh	신용카드	카카오페이	28,400 KRW	
결제예정	imp_046410733286 ORD20221009-kakao:2484108854-549000000	친환경 복숭아 5kg/10kg	dfh	신용카드	카카오페이	28,400 KRW	
결제예정	imp_088397809953 ORD20221009-kakao:2484108854-468000000	친환경 복숭아 5kg/10kg	dfh	신용카드	카카오페이	28,400 KRW	
결제예정	imp_794738167324 ORD20221009-kakao:2484108854-603000000	친환경 복숭아 5kg/10kg	adfg	신용카드	카카오페이	28,400 KRW	
결제예정	imp_613140643653 ORD20180131-0000011	노르웨이 회전 의자	홍길동	신용카드	카카오페이	64,900 KRW	

toss pay



휴대폰번호

QR코드

결제를 진행할
휴대폰 번호를 입력해 주세요

휴대폰번호

생년월일 6자리

토스 앱으로 결제

18:12:20



피드백

“회원가입 중 인증 번호가 전송되지 않았음에도
번호가 자동으로 입력이 되고, 인증되었다고 뜹니다.”

=> 수정하겠습니다!

감사합니다!