핵심기능

게시판을 통해 배달부와 판매자간의 거래를 도우며

거래상황에서 GPS정보를 주기적으로 업데이트 해준다.

주요기능

모바일을 통하여 자유게시판 형식의 마켓 오픈,

셀러가 물건의 정보와, 배송목적지, 희망금액을 제시하며 게시글을 올리며

딜리버 는 제시된 게시판의 글 중에서 원하는 셀러와 배송희망을 선택한 이후

추가적으로 대화를 할 수 있다.

셀러와 딜리버 사이의 의견 조율을 하는데, 금액 등에 이견이 없다면, 거래 확인창에서 쌍방 승인버튼을 누르면 되며,

추가적으로 금전이나 기타 사항의 조율을 할 경우, 승인버튼을 누르기 전에 확인창에서 셀러가 금액을 수정한다. 이때 수정한 금액은 쌍방 다 실시간 확인이 될것이며, 조건이 다 맞다면, 쌍방 승인 버튼을 눌러 거래를 성사시킨다.

이 때, 셀러가 올린 금액만큼은 자회사로 들어와 keep 상태가 되는데(방법 생각해보기)

거래가 완전히 끝났음을 양쪽 다 확인하면, keep된 금액이 딜리버의 통장으로 전달된다(다른방법을 원하니?)

이때 생길 수 있는 문제점 : 셀러가 돈이 없는상태로 딜리버랑 계약을 할 수 있다.

딜리버가 물건을 받을 떄 까지 혹은 배송할 때까지 바이어는 방법이 없다.

이를 해결하기 위해서는 셀러의 제정 상태를 미리 알수 있어야 할 것이다.

그렇지 않으면, 딜리버가 언제 까지 입금확인을 되지 않으면, 물건을 받으러 가지 않게 교육이 필요할 것

이제 쌍방 물건을 전달하고, 목적지로 배달하는 과정이 진행된다.

딜리버가 셀러에게 물건을 받으면 출발 버튼을 누르게 되면

이때부터, 딜리버는 이제 GPS위치를 주기적으로 서버에 전송한다.

전송받은 GPS좌표는 셀러의 거래현황창에서 업데이트가 된다.

배송이 완료된 딜리버는 거래현황에서 배달완료 버튼을 누른다.

셀러는 수신자에게 물건을 받았으며, 하자가 없음을 확인하면, 거래완료 버튼을 눌러서 거래를 끝낸다.

이후 딜리버와 셀러 모두 고객 평점을 매기며, 거래는 끝이 난다.

아키텍쳐 정의

4계층

클라 ? 컨트롤러서버 DB서버

/ //index page for web

/loginPage //for web, view only

/loginProcess //method=put / json {‘id’:’var1’, ‘pw’:’var2’}

/signInPage // for web, view only

/sighInProcess //method=post / json { 내용물들}

/userInfoView // method=get //session의 아이디를 참조하여 일치하는 에트리뷰트를 출력하여

유저관련

Public

Css //css파일들이 모여있는곳

Resources //리소스파일

Image

Video

Views

Index.ejs

Login

loginPage

// loginProcess

SignIn

Signin.ejs

// SighinProcess.ejs

Account

AccountView

AccountModify

AccountDelete

Board

Trade

TradeBoardList.ejs

TradeBoardRead.ejx

TradeBoardWrite.ejs

TradeBoardModify.ejs

답글, 그러니깐 채팅도 여기서 되야하려나?

driverPage

driverList

driverPage

routes

main

login

loginManager

app.js // 메인함수

public // static 자료

config

* Passport.js // passport 인증
* Db\_info.js // 유 연결용

Modules

* User.js // 유저정보와 인증용

Routes

* Error
* Login
* Sign
* R\_main.js

드라이버 DB를 따로 분리해야하나? Postgre,mongus

Reason : column 에 배열이 등록가능,

Why array? 드라이버가 계약중인 건수들은 어떻게 표시할까?

Promote 함수 생성, passport -> user 내부 구조 건축.

그에 따라 필요한 것은 driver DB connection 구축을 해야함.

DB구조를 다시 짜봐야 할거같다.

DB수업에서 개념스키마를 공부중인데, 여기서 상당히 도움이 될 내용들이 나왔다.

개념 스키마 말고, 개발단계의 스키마는 얼마든지 늘어날 수 있다.

설계를 함으로, 조금 더 DB서버의 여유를 가질 수 있을지도 모르겠다.