/user/signup/

* POST 형식으로 body에 username, password, email을 보내야한다
* Username이 6자보다 작으면

{'message':'invalid username, must be at least 6 characters long'}

* 이메일 형식이 올바르지 않으면

{ ‘message':'invalid email’}

* 중복된 아이디일 경우

{‘message':'duplicated id' }

* USER 생성에 성공했을 시

{ 'message':'user registration successful' }

/users/login/

* POST 형식으로 body에 username, password가 와야 한다.
* 로그인에 성공했을 시

{

"token": "995046e1f0d4916d807634527494f3ce8ef81b13"

}

* 로그인에 실패했을 시

{

"non\_field\_errors": [

"Unable to log in with provided credentials."

]

}

* 필요한 body가 안 왔을 시 ex)패스워드

{

"password": [

"This field is required."

]

}

/user/myErrand

* GET 형식으로 Header에 key : Authorization / value “token ~~~~~~~”/ 줘야 함
* Ex)
* [{"id":3,"owner\_name":"test1234","title":"Blank Title","text":"hihihihi","extraCost":100,"reward":200,"category":"ETC","created":"2017-08-08T09:18:00.937779Z"},

{"id":4,"owner\_name":"test1234","title":"또잉","text":"또잉","extraCost":2,"reward":4,"category":"HOMEWORK","created":"2017-08-08T09:18:16.052749Z"}]

/errands/

* 모든 bill를 시간대로 정렬해 보여준다.
* 각 bill는 owner, title, text, extraCost, reward, category, created(생성날짜)를 가지고 있다.
* bill들을 받아올 수 있지만 로그인 해야만 bill를 추가할 수 있다.
* GET 하면 각 bill의 result들이 json형태로 온다.
* POST는 title, text, extraCost, reward, category(DELIVERY, HOMEWORK , ETC) 로 나뉜다.

/errands/id/

* Id에 해당하는 메모를 정보를 받고 업데이트하거나 지울 수 있다.(단 자신이 owner에 해당하는 글만 삭제 혹은 수정가능하다.)
* PUT 는 수정된 title, text, extraCost, reward, category를 넘겨준다.
* DELETE 는 bill을 지운다.
* GET 하면 정보가 json 형태로 넘어온다.

(DELIVERY, HOMEWORK , ETC)

/errands/DELIVERY/

* GET하면 DELIVERY에 해당하는 errand들이 json 형태로 온다
* POST는 title, text, extraCost, reward, category(DELIVERY, HOMEWORK , ETC) 로 나뉜다.
* errand들을 받아올 수 있지만 로그인 해야만 errand를 추가할 수 있다.