Привет, Григорий!

Спасибо тебе за выполненную работу! У меня есть несколько замечаний.

**Чек-лист мобильное приложение**

Обязательно нужно исправить:

* 33 проверка не соответствует требованию, бага в таком поведении нет

Сильные стороны:

* отличное форматирование чек листа
* все требования из задания покрыты проверками

**Баги по чек-листу мобильное приложение**

Обязательно нужно исправить:

* Не найдены баги связанные с временным интервалом, деталями маршрута, еще один баг в скролле лонгтапом

Сильные стороны:

* хорошо оформлены заведенные баги

**Чек-лист API**

Обязательно нужно исправить:

* в некоторых ручках есть параметры с одинаковыми названиями (id в POST и PUT), убедись что однозначно отсылаешь к тому какой парматер тестируется в текущей проверке (например id набора/корзины и id продукта)
* проверки 15-32 - к сожалению я не поняла что ты имеешь в виду.Это как будто pairwise и пары на передать-не передать? Много дублей но и без дублей не понимаю что ты хочешь проверить, как сформировать запросы. Пожалуйста покажи эту часть чек листа наставнику чтобы вы вместе доформулировали до понятных проверок.
* productsList - почему у тебя КЭ на числа и строки? В валидном варианте параметр принимает массив с вложенными объектами, соответственно негативные КЭ будут: не передать ничего, передать пустой массив, передать объект, но не в массиве (удалить [])
* может ты имеешь в виду то что я описываю ниже, но на всякий случай оставляю коммент о подходе к тестам параметров
* В ручках больше параметров, а у параметров больше КЭ. Например POST для набора. В нем 4 параметра (т.е изменяемых значения): ид в url, ид продукта, quantity, productsList. У параметров есть общие состояния, разберу на примере ид продукта:
* валидное значение (тут позитивный КЭ - такой продукт есть в БД и негативный КЭ - такого продукта - нет в БД)
* невалидное значение (символы, буквы. Иногда зависит от параметра, например если время передаем, то будут специфические невалидные данные. А для productsList у нас валидное было передать массив, значит невалидное будет - передать без массива)
* отсутствие значения (передать пробел. Если у нас json то пробел передаем строкой " ", если xml - то просто пробел)
* отсутствие параметра (просто не написать его название и посмотреть что будет)
* Можно добавить подзаголовки для тестирования параметров, так будет проще проектировать и потом использовать чек лист
* в доставке - Про pairwise - тут у нас независимые параметры, поэтому я думаю что тестить все пары излишне , достаточно протестировать пары по КЭ и отдельно покрыть ГЗ.
* Во-первых параметр время - он вообще не связан с результатом расчета, его из попарного тестирования сразу исключаем, пишем 2 проверки: 1. рабочее время доставки (тут 07,08,15,20,21, как ты пишешь. Можно в чеклисте для параметра который под тестом писать диапазон значений, если они в одном КЭ) + нейтральные значения других параметров (кол-во 2, вес 2(например)). 2. нерабочее время доставки(06,22,23) +нейтральные значения других параметров
* Второе - вес и кол-во. Я предлагаю допустить также тестирование параметров по одному. Например вес: 1. меньше чем можно(<0) + нейтральные значения кол-ва, для нас это то значение которое в каждом из диапазонов не вызовет изменения расчета,( из первого диапазона, где цена доставки 23) + рабочее время доставки. 2. ГЗ веса для первого диапазона (0-2.5) + нейтральные остальные, 3. ГЗ веса для второго диапазона + нейтралььные остальные, 4. ГЗ для превышения веса + нейтральные остальные. то же для кол-ва.
* DeliveryTime - обрати внимание на формат принимаемых данных по требованиям, там только час, без минут.
* Проверки на вес товара - проверь, там опечатка и вместо дроби получилось два числа через запятую. Для работы это важная опечатка, получилось все проверки на дробный вес были одним КЭ, но добавить такую проверку можно, правда будет другой ОР и будет баг.

**Баги по чек-листу API**

зачет

Можно улучшить:

* Лучше вставлять код и json не текстом, а цитатой, чтобы сохранялась табуляция и имена параметров, так будет намного проще читать и воспроизводить

Сильные стороны:

* хорошо оформлены баги

Отправляю задание на доработку и жду правок) Береги себя.