Техническое задание на доработку мобильного приложения:

Проблема: Несколько клиентов из одного дома оформляют заказы в разные интервалы. Это приводит к тому, что доставка приезжает несколько раз в одну точку за рейс.

Решение: При выводе интервалов доставки, если в определенный день, в определенную волну уже есть другой заказ (любого клиента), то оформить следующий заказ можно только в тот интервал, в котором уже оформлен заказ.

Например:

Клиент №1 оформляет заказ по адресу Победы 1 на 01.09.21

На адресе Победы 1 установлены "стандартные" интервалы

Для него доступны все шесть интервалов: 3 дневных (9-11, 11-13, 13-15), 3 вечерних (17-19, 19-21, 21-22)

Клиент №1 выбрал интервал 11-13 и оформил заказ.

Далее клиент №2 хочет оформить заказ по адресу Победы 1 на 01.09.21

Так как до него клиент №1 уже оформил заказ, то на дневную волну доставки у него будет доступен только один вариант: 11-13

На вечер он может выбрать любой 17-19, 19-21, 21-22

Клиент №2 выбирает интервал 17-19 и оформляет заказ

Далее клиент №3 хочет оформить заказ по адресу Победы 1 на 01.09.21

Так как до него другие клиенты уже оформили заказы по этому адресу, то клиенту №3 останется на выбор всего 2 интервала: днем 11-13 и вечером 17-19

Всем последующим клиентам также предоставляется возможность оформить заказ только на интервалы: днем 11-13 и вечером 17-19

Дополнение:

-Если при оформлении заказа какой-либо из интервалов перегружен, но уже есть заказ в этот интервал, в этот день, то данный интервал становится доступен, несмотря на перегрузку.

+При отображении интервалов доставки если на какой-то интервал в конкретную волну и день уже есть заказ и данный интервал перегружен, то все остальные интервалы из этой волны доставки становятся недоступные для выбора (якобы перегруженные).

Если интервал перегружен он отмечается символом «\*», при тапе на такой интервал происходит анализ доступных дней отгрузки по этому интервалу:

Если на завтрашний день этот интервал доступен, то отображается вариант №**1.**

Если интервал недоступен на следующий день, то отображается вариант **№2.**

Варианты отображения:

**№1** – Показать список дней доставки:

Выводится список дней доставки, на которые доступен выбранный интервал.

**№2** Показать список дней доставки + интервалы на завтра.

Выводятся ближайшие доступные интервалы доставки к выбранному интервалу на завтра и ближайшие 2 дня доставки, на которые доступен выбранный интервал.

После того как пользователь делает свой выбор, автоматически устанавливается дата доставки и интервал.

**Примеры**:

**Сценарий №1 при доступности интервала на следующий день.**

 Клиент хочет заказать 04.09. в интервал с 11-13. Но это интервал недоступен по причине перегрузки. Отображается **следующий экран:**

При тапе на интервал с 11-13, пользователю выходит следующая форма выбора . ->

**Сценарий №2 если выбранный интервал НЕ доступен на следующий день.**

Клиент хочет заказать 04.09. в интервал с 11-13. Но это интервал недоступен, и клиенту предлагается интервал с 9-11 т.к. в интервал сформирован заказ, и это время доступно. Клиенту выводится  следующий экран. При тапе на интервал, следующая форма выбора ->

Проект "Зеленые интервалы во внешних информационных системах"

Техническое описание:

Определения:

Перегруженный интервал - В пользовательском интерфейсе отображается как "потухший" интервал (Прием заявок завершен)

Волна доставки - Вид интервала доставки, может быть День/Вечер/Полный

На этапе формирования интервалов доставки для отображения пользователю, после проверки на перегруженные интервалы, необходимо сделать дополнительную проверку

Обратиться к методу на сервере авторизации GetGreenIntervals(Address, DeliveryDates)

В качестве параматра Address передается идентификатор адреса из адреса доставки клиента

В качестве параметра DeliveryDates передается набор дат перечисленных через запятую

В результате вернется массив с группировкой по дням, который будет содержать массивы с интервалами на которые уже есть оформленные заказы (если таковые определены). Массив может вернутся пустым.

**Удалено: Из этого массива необходимо исключить перегруженные интервалы.**

Далее на конкретный день доставки, в разрезе каждой волны доставки (день/вечер) проверяется наличие интервалов в полученном массиве

Если на конкретный день хотя бы 1 интервал совпал, то все остальные интервалы, не вошедшие в массив приоритетных, в разрезе волны доставки (день/вечер) автоматически становятся перегруженными (отображаются, но не доступны для выбора)

**Добавлено: Если  хотя бы из 1 приоритетных интервалов оказался перегружен, то он сам и все остальные интервалы, в разрезе волны доставки (день/вечер) автоматически становятся перегруженными (отображаются, но не доступны для выбора)**

**Удалено: Интервалы, на которые уже есть оформленные заказы, становится доступны в любом случае, даже если они до этого уже были перегружены.**

Описание метода веб сервиса

GetGreenIntervals(Address, DeliveryDates)

Address - Строка(36) - Идентификатор адреса доставки из заказа

DeliveryDates - Строка - Даты доставки заказа. Формат даты: 03.12.2021 = "2021-12-03"

Структура ответа:

ResponseCode - 0 = успех, иначе ошибка.

Message - Текстовое описание кода ответа

Data - Строка JSON

Возможные коды ответа (ResponseCode):

0 - "ОК"

1 - "Внутренняя ошибка"

Пример запроса

Address = "6F9619FF-8B86-D011-B42D-00CF4FC964FF"

DeliveryDates = "2021-12-03,2021-12-04,2021-12-05,2021-12-06"

GetGreenIntervals(Address, DeliveryDates)

Пример ответа:

ResponseCode: 0

Message: "OK"

Data:

{

             "GreenIntervals": [

                            {

                                            "Day": "2021-09-30",

                                            "Intervals": [

                                                            "85b9d872-5432-11e8-80fd-005056861757",

                                                            "fc2c18bc-f03a-11e5-80d8-005056861757"

                                            ]

                            },

                            {

                                            "Day": "2021-10-01",

                                            "Intervals": [

                                                            "85b9d870-5432-11e8-80fd-005056861757",

                                                            "85b9d872-5432-11e8-80fd-005056861757",

                                                            "fc2c18c2-f03a-11e5-80d8-005056861757"

                                            ]

                            },

                            {

                                            "Day": "2021-10-02",

                                            "Intervals": []

                            },

                            {

                                            "Day": "2021-10-03",

                                            "Intervals": []

                            },

                            {

                                            "Day": "2021-10-04",

                                            "Intervals": []

                            },

                            {

                                            "Day": "2021-10-05",

                                            "Intervals": []

                            }

             ]

}

Дополнительные сведения:

1. Доработка распространяется только на перегруженные интервалы. Если интервал закрыт правилами - не показываем.

2. Логирование запросов и ответов от сервера авторизации осуществляется на стороне сервера авторизации

3. На стороне внешней информационной системы отмечаются только ошибки по запросам, когда сервер авторизации был недоступен . (Запрос завершился ошибкой по тайм ауту)

4. Если сервер авторизации недоступен при заказе, то исключить проверку на "Зеленые интервалы" (не применять новую логику/работать как раньше)