

改签功能 - 代理商接口

前言

- a、文档中接口涉及的数据传输均使用json。传输的字段值需要严格按照类型要求传输，case：要求传入long型的字段，不能传成（abc:""），应该传成（abc:0）。
- b、如果在开发中遇到问题，可以参考文档中第6点的问题汇总寻求帮助，如果不能解决联系美团技术人员寻求帮助。
- c、如果代理商需要了解每次收到的文档和以前收到的文档有什么差别，可以阅读第7点。

1、改签业务逻辑设计

1.1、占座部分

主要流程：代理商提供接口，并接收美团提交改签请求 —— 接受请求，异步处理 —— 美团提供接口，代理商处理完成回调美团通知占座结果。

1.2、取消改签部分

主要流程：代理商提供接口，并接收美团取消改签请求 —— 接受请求，异步处理 —— 美团提供接口，代理商处理完成回调美团通知取消改签结果。

1.3、确认出票部分

主要流程：代理商提供接口，并接收美团出票请求 —— 接受请求，异步处理 —— 美团提供接口，代理商处理完成回调美团通知改签出票结果。

2、代理商提供的接口

2.1.1, 2.2.1, 2.3.1的接口写法，可以参考美团接口3.1.1, 3.2.1, 3.3.1小节中附图的接口写法。

我们请求代理商的写法如下：

```
String url = "你们的请求地址";
HttpPost httpPost = new HttpPost(url);
StringEntity reqEntity = new StringEntity(body, "UTF-8");
reqEntity.setContentType("application/json;charset=UTF-8");
httpPost.setEntity(reqEntity);

String response = client.execute(httpPost, new BasicResponseHandler());
```

2.1、提交改签请求

2.1.1、请求接口

uri	method	ContentType
baseurl + /RestfulService/resignTicket	POST	application/json;charset=UTF-8

2.1.2、输入参数（下面的参数，参数名不变，参数个数，代理商不需要的美团可以不传）

字段	二级字段	类型	含义	备注
resignId		long	改签id	用来幂等啥
orderId		long	美团订单号	
agentOrderId		String	代理商订单号	
orderType		String	下单类型	
trainCode		String	车次号	
trainDate		String	发车日期	
fromStationCode		String	出发站code	
fromStationName		String	出发站名称	
toStationCode		String	到达站code	
toStationName		String	到达站名称	
callbackURL		String	回调地址	
reqToken		String	鉴权字段	回调美团时鉴权
chooseNoSeat		boolean	是否接受无座	
tickets		List		
	ticketId	long	美团订单车票号	要改签的原始车票的车票id
	ticketNo	String	12306的车票号	要改签的原始车票的车票号
	passengerName	String	乘车人姓名	
	certificateNo	String	乘车人证件号码	
	seatType	int	坐席类型	eg. 1
	ticketType	int	车票类型	eg. 1 (1:成人, 2:学生, 3:儿童)
method		String	操作功能名	参考下面第4大点的method
reqtime		String	请求时间	yyyyMMddhhmmss
partnerid		String	调用方识别	
sign		String	数字签名	代理商对美团鉴权

输入参数

```
{
  "resignId":12425553,
  "orderId":3425252352424,
  "orderType":"resign",
  "trainCode":"T1",
  "fromStationCode":"BJP",
  "fromStationName":"",
  "toStationCode":"TJP",
  "toStationName":"",
  "trainDate":"2016-07-16",
  "callBackURL":"*****",
  "reqToken":"*****",
  "sign":"*****",
  ...
  ...
  ...
  "chooseNoSeat":true,
  "tickets":[{
    "ticketId":1244245,
    "ticketNo":"E254253452355",
    "passengerName":"",
    "certificateNo":"*****",
    "seatType":1,
    "ticketType":1
  }]
}
```

2.1.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
code	int	返回code	只有802表示接受改签请求
success	boolean	返回是否成功	

返回参数

```
{
  "code":"802",
  "msg":"",
  "success":true
}
```

2.2、取消改签请求

2.2.1、请求接口

uri	method	ContentType
<code>baseurl + /RestfulService/cancelResign</code>	POST	<code>application/json;charset=UTF-8</code>

2.2.2 输入参数

字段	二级字段	类型	含义	备注
resignId		long	改签id	
orderId		long	美团订单号	
agentOrderId		String	代理商订单号	
reqToken		String	鉴权字段	回调时美团对代理商鉴权
method		String	操作功能名	参考下面第4大点的method
reqtime		String	请求时间	<code>yyyyMMddhhmmss</code>
partnerid		String	调用方识别	
sign		String	数字签名	代理商对美团鉴权

输入参数

```
{
  "resignId":25989523,
  "reqToken":"*****",
  "method":"*****",
  ...
  ...
  "sign":"*****",
}
```

2.2.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
code	int	返回code	
success	boolean	返回是否成功	

返回参数

```
{
  "code":"100",
  "msg":"",
  "success"true
}
```

2.3、确认改签请求

2.3.1、请求接口

uri	method	ContentType
<code>baseurl + /RestfulService/confirmResign</code>	POST	<code>application/json;charset=UTF-8</code>

2.3.2、输入参数

字段	类型	含义	备注	
resignId	long	改签id		
orderId		long	美团订单号	
agentOrderId		String	代理商订单号	
reqToken	String	鉴权字段	回调时美团对代理商鉴权	
method		String	操作功能名	参考下面第4大点的method
reqtime		String	请求时间	<code>yyyyMMddhhmmss</code>
partnerid		String	调用方识别	
sign		String	数字签名	代理商对美团鉴权

输入参数

```
{
  "resignId":5255435,
  "reqToken":"*****",
  "method":"*****",
  ...
  ...
  "sign":"*****",
}
```

2.3.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
code	int	返回code	
success	boolean	返回是否成功	

返回参数

```
{
  "code":"100",
  "msg":"",
  "success"true
}
```

3、美团提供的接口

3.1、改签占座结果回调

3.1.1、请求接口

baseUrl (测试环境)	uri (partnerid为3位数字的代理商代码)	method	ContentType
http://test.i.meituan.com/uts/train02	/[partnerid]/resignTicketNotify	POST	application/json;charset=UTF-8

下图是美团接口写法，其中的String body就是我要接收的json数据

```
@ResponseBody
@RequestMapping(value =("/{partnerid}/resignTicketNotify", method = RequestMethod.POST, produces = "application/json;charset=UTF-8")
public JSONObject resignTicketNotify(@PathVariable("partnerid") String partnerid, HttpServletRequest request, @RequestBody String body)
```

参数与返回的case (供参考)

求参数:{"arriveTime":"13:59:00","fromStationCode":"SUB","fromStationName":"孙家","order12306Serial":"E923009137","orderAmount":1,"orderId":14708124890001,"reqToken":"1442f51ec35bcf173e67459782e64b0d","resignId":38,"returnFact":"0.0","startTime":"13:49:00","success":true,"tickets":[{"certificateNo":"XXXXXXXXXXXX","coachNo":"03车厢","passengerName":"XFB","seatNo":"062号","seatType":1,"ticketId":5082438,"ticketNo":"E9230091373030062","ticketPrice":"1.00"}],"toStationCode":"XFB","toStationName":"香坊","trainCode":"6021","trainDate":"2016-09-15"} 返回结果:{"code":100,"msg":"处理或操作成功","success":true}

3.1.2、输入参数

字段	二级字段	类型	含义	备注
resignId		long	改签id	
orderId		long	美团订单号	
trainCode		String	车次号	
trainDate		String	发车日期	
fromStationCode		String	出发站code	出发城市不能变，但是车站能改变
fromStationName		String	出发站名称	
toStationCode		String	到达站code	
toStationName		String	到达站名称	
reqToken		String	鉴权字段	回调时美团对代理商鉴权
startTime		String	发车时间	(占座失败时，没有可不传)
arriveTime		String	到达时间	(占座失败时，没有可不传)
orderAmount		double	改签订单总价	(占座失败时，没有可不传)
success		boolean	占座是否成功	
seatHoldFailReason		String	占座失败原因	在占座成功时不返回，占座失败需要传
tickets		List		如果占座失败，ticket里面如车厢号，座位号等需要占座成功才能得知的信息，可不传
	ticketId	long	车票号	要改签的原始车票的车票id
	passengerName	String	乘车人姓名	
	certificateNo	String	乘车人证件号码	
	ticketPrice	double	车票价格	eg. 11.00 (占座失败时，没有可不传)
	seatType	int	坐席类型	eg. 1 (占座失败时，没有可不传)
	ticketType	int	车票类型	eg. 1 (1:成人, 2:学生, 3:儿童) (占座失败时，没有可不传)

	coachNo	String	车厢号	05车厢（占座失败时，没有可不传）
	seatNo	String	座位号	005号（占座失败时，没有可不传）
	ticketNo	String	12306车票序列号	与原有的应该不同（改签后的新车票的车票号）（占座失败时，没有可不传）
returnFact		String	实际总退款	刨除了改签手续费后的实际总退款（占座失败时，没有可不传）

3.1.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
code	int	返回code	
success	boolean	是否接受请求成功	

3.2、改签取消结果回调

3.2.1、请求接口

baseUrl（测试环境）	uri（partnerid为3位数字的代理商代码）	method	ContentType
http://test.i.meituan.com/uts/train02	/[partnerid]/cancelResignNotify	POST	application/json;charset=UTF-8

下图是美团接口写法，其中的String body就是我要接收的json数据

```
@ResponseBody
@RequestMapping(value = "/[partnerid]/cancelResignNotify", method = RequestMethod.POST, produces = "application/json;charset=UTF-8")
public JSONObject cancelResignNotify(@PathVariable("partnerid") String partnerid, HttpServletRequest request, @RequestBody String body)
```

3.2.2、输入参数

字段	类型	含义	备注
resignId	long	改签id	
reqToken	String	鉴权字段	回调时美团对代理商鉴权
success	boolean	是否取消成功	

3.2.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
code	int	返回code	
success	boolean	是否接受请求成功	

3.3、改签出票结果回调

3.3.1、请求接口

baseUrl（测试环境）	uri（partnerid为3位数字的代理商代码）	method	ContentType
http://test.i.meituan.com/uts/train02	/[partnerid]/confirmResignNotify	POST	application/json;charset=UTF-8

下图是美团接口写法，其中的String body就是我要接收的json数据

```
@ResponseBody
@RequestMapping(value =("/{partnerid}/confirmResignNotify", method = RequestMethod.POST, produces = "application/json;charset=UTF-8")
public JSONObject confirmResignNotify(@PathVariable("partnerid") String partnerid, HttpServletRequest request, @RequestBody String body)
```

3.3.2、输入参数

字段	类型	含义	备注
resignId	long	改签id	
reqToken	String	鉴权字段	回调时美团对代理商鉴权
returnFact	String	实际总退款	这个值和3.1.2的returnFact的值是一样的。如果你能保证在3.1.2中肯定能传过来，这里就可以不用传了。
success	boolean	是否出票成功	

3.3.3、返回参数

字段	类型	含义	备注
msg	String	返回message	
success	boolean	是否接受请求成功	
code	int	返回code	

4、鉴权字段生成规则

4.1、生成规则

输入参数

```
/**
 * reqToken
 */
public static String getChangeReqToken(Method method,long orderId, long
resignId, String trainCode, String from_station_code, String
to_station_code, String train_date, List<ChangeTicket> tickets) {
    String key = "uU2MQezahDMuqAIZB6IzFsLd6OdCDPVM";
    StringBuilder seed = new StringBuilder();
    seed.append(orderId).append("_").append(resignId);
    if(method != null){
        seed.append("_").append(method.toString());
    }
    if(trainCode != null){
        seed.append("_").append(trainCode);
    }
    if(from_station_code != null){
        seed.append("_").append(from_station_code);
    }
    if(to_station_code != null){
        seed.append("_").append(to_station_code);
    }
}
```



```

        if(train_date != null){
            seed.append("|").append(train_date);
        }
        if(tickets != null){
            seed.append("tickets:");
            Collections.sort(tickets, new Comparator<ChangeTicket>() {
                @Override
                public int compare(ChangeTicket o1, ChangeTicket o2) {
                    if (o1.getTicketId() > o2.getTicketId()) {
                        return 1;
                    } else {
                        return -1;
                    }
                }
            });
            for (ChangeTicket ticket : tickets) {

seed.append(ticket.getTicketId()).append(ticket.getCertificateNo());
            }

            String reqToken = MD5Utils.getMD5Str(seed.toString() +
MD5Utils.getMD5Str(key));
            return reqToken;
        }

public enum Method {
    RESIGN_TICKET("resignTicket"),//
    CANCEL_RESIGN("cancelResign"),//
    CONFIRM_RESIGN("confirmResign");//

    private String method;

    Method(String method) {
        this.method = method;
    }

    public String toString() {

```

```
        return this.method.toString();
    }
}
```

4.2、改签占座生成

```
this.getChangeReqToken(Method.RESIGN_TICKET, req.getOrderId(), req.getResignId(), req.getTrainCode(),
    req.getFromStationCode(), req.getToStationCode(), req.getTrainDate(), req.getTickets())
```

4.3、取消改签生成

```
this.getChangeReqToken(Method.CANCEL_RESIGN, 0, req.getResignId(), null, null, null, null, null)
```

4.4、确认改签出票生成

```
this.getChangeReqToken(Method.CONFIRM_RESIGN, 0, req.getResignId(), null, null, null, null, null)
```

5、类描述

5.1、ChangeTicket类描述

输入参数

```
/**
 *
 */
public class ChangeTicket implements java.io.Serializable {
    //id
    private long ticketId;
    //
    private String ticketNo;
    //
    private String passengerName;
    //
    private String certificateNo;
    //
    private String seatType;
    //
    private String ticketType;

    //
    private String ticketPrice;
    //
    private String coachNo;
    //
    private String seatNo;
```

```
public long getTicketId() {
    return ticketId;
}

public String getTicketPrice() {
    return ticketPrice;
}

public void setTicketPrice(String ticketPrice) {
    this.ticketPrice = ticketPrice;
}

public String getCoachNo() {
    return coachNo;
}

public void setCoachNo(String coachNo) {
    this.coachNo = coachNo;
}

public String getSeatNo() {
    return seatNo;
}

public void setSeatNo(String seatNo) {
    this.seatNo = seatNo;
}

public void setTicketId(long ticketId) {
    this.ticketId = ticketId;
}

public String getTicketNo() {
    return ticketNo;
}

public void setTicketNo(String ticketNo) {
    this.ticketNo = ticketNo;
}

public String getPassengerName() {
    return passengerName;
}

public void setPassengerName(String passengerName) {
    this.passengerName = passengerName;
}

public String getCertificateNo() {
    return certificateNo;
}
```

```
public void setCertificateNo(String certificateNo) {
    this.certificateNo = certificateNo;
}

public String getSeatType() {
    return seatType;
}

public void setSeatType(String seatType) {
    this.seatType = seatType;
}

public String getTicketType() {
    return ticketType;
}

public void setTicketType(String ticketType) {
    this.ticketType = ticketType;
}

@Override
public String toString() {
    return "ChangeTicket{" +
        "ticketId=" + ticketId +
        ", ticketNo='" + ticketNo + '\'' +
        ", passengerName='" + passengerName + '\'' +
        ", certificateNo='" + certificateNo + '\'' +
        ", seatType='" + seatType + '\'' +
        ", ticketType='" + ticketType + '\'' +
        ", ticketPrice=" + ticketPrice +
        ", coachNo='" + coachNo + '\'' +
        ", seatNo='" + seatNo + '\'' +
    }
}
```

```
}  
    }  
}
```

6、所提问题汇总

2.2、取消改签请求

调用该接口时，返回的成功是取消成功还是接收成功？

答>>只是接收成功，你们处理完后，在回调里面告诉我处理结果。

是否可以取消以哪个为准？（2.2、取消改签请求 还是 3.2、改签取消结果回调）

答>> [2.2、取消改签请求] (这个是接收我们的取消改签的请求，但是你们不能马上处理完成)，所以通过 [3.2、改签取消结果回调] 告诉我们取消改签的处理结果。

对于已改签，不可取消的订单，实在 2.2 实时返回，还是 3.2 异步返回。错误码是什么？

答>> 这种处理结果，是在3.2回调时告诉我，我主要关注success和msg字段。code以后再定吧。先告诉我success是false，及错误msg。

能否改成同步请求？直接返回是否允许取消？

答>>不能，这里是所有的代理商统一的。而且担心接口问题影响系统。

3.1、改签占座结果回调

returnFact 实际退款

此属性是总退款吗？每张票的改签退款可能不同，如何区分？

答>>这个是刨除了12306的手续费后的，给用户的实际总退款。

此属性是否支持负数？如低价改高价时，这里是写负数，还是0？EndFragment

答>>我只有高改低时，才会取这个字段。（如果时低改高，你就写0吧）

关于低改高，高改低费用问题

答>>改签回调时，我会根据你们给的新的总票价和原始总票价，计算改签类型是平改、高改低、低改高。

如果是低改高，我会计算原始总票价和新票总票价差，得出支付价格。

如果是高改低，我会拿到你的returnFact(这个是刨除了手续费后的退款金额)，确认退款金额。

1、resignId 用来幂等啥

请求改签，取消改签，确认改签一笔订单是不是都是统一的值，我们通过这个字段唯一判断改签请求是否存在是不是可以？

答>>resignId是唯一的，一个美团订单号（orderId），可以对应多个resignId。

2、2.1.2中的orderType 下单类型 是指什么，有没有示例

答>>我会传入 Method.RESIGN_TICKET.toString()

3、trainCode 车次号，trainDate 发车日期，fromStationCode 出发站code，fromStationName 出发站名，toStationCode 到达站code，toStationName 到达站名称，seatType 坐席类型

这些都是指改签后的？

答>>是的

4、ticketNo 12306的车票号 12306有车票号吗？

答>>2.1.2中的ticketNo是原始车票的车票号，3.1.2里面是改签后的新车票的车票号。 ticketNo case : E9323309002030009

5、code 只说明了成功返回的值，失败返回什么值，不同的错误原因是不是返回不同的值

答>>你们返回给我的数据，我关心success（是否成功），msg（信息）。code以后有需要再沟通。

6、reqtoken的用途与流转

答>>1、我传给你，你可以通过reqToken的计算规则，自己算出一份reqToken，看和我传给你的是否相同，来判断合法性。（根据你们的安全，可以选择做与不做此验证）

2、你们回调时把reqToken回传给我（可以用你们算出来的，也可以用我传给你们的那个），我会根据计算规则算出一份reqToken，来判断合法性。

7、文档变更记录

7.1、20160810

- a、代理商提供的接口的url地址规范定义。
- b、漏了seatHoldFailReason字段，补上。
- c、代理商提供的接口，美团都会传入orderId和agentOrderId。

7.2、20160811B

- a、4.1中getChangeReqToken方法内容修改，请代理商重新复制新的。
- b、3.1.1中增加参数和返回的数据格式的例子供参考。
- c、3.1.2中占座失败时，明确哪些字段可以不用传。

7.3、20160811C

- a、增加接口通信的写法示例。

3.1.1，3.2.1，3.3.1小节中的附图是美团接口的写法，请按照此写法传输数据。

2.1.1，2.2.1，2.3.1的接口写法，可以参考美团接口3.1.1，3.2.1，3.3.1中附图中的接口写法，我们请求代理商接口的写法如下：

```
String url = "你们的请求地址";
HttpPost httpPost = new HttpPost(url);
StringEntity reqEntity = new StringEntity(body, "UTF-8");
reqEntity.setContentType("application/json;charset=UTF-8");
httpPost.setEntity(reqEntity);

String response = client.execute(httpPost, new BasicResponseHandler());
```

7.4、20160812

- a、3.1.1，3.2.1，3.3.1中美团改签接口地址，附上测试时的具体服务地址。线上的地址，上线前再确定。