



واحد برنامه‌ریزی و تعالی سازمانی
 پروژه مدیریت پیکربندی مستندات
 قالب پیکربندی گزارش‌های بلند برون‌سازمانی
 شماره ویرایش قالب: **Ver1.20**

جدول اطلاعات سند		
راهکارهای پردازش تراکنش		واحد تهیه‌کننده سند
Enterprise Service Bus		نام پروژه
سرویس‌های واریز گروهی جدید		عنوان سند
ESB		مخفف نام پروژه
GroupPaymentNewServices		مخفف عنوان سند
1402/01/28		تاریخ ویرایش سند
	Ver1.0	شماره ویرایش سند
عمومی		درجه محرمانگی سند
اکرم احمدیان مهرگو		تهیه‌کننده/کنندگان
واریز گروهی – پایا- ساتنا- انتقال وجه داخلی		کلمات کلیدی
ESB.GroupPaymentServices.181.Ver1.0		شناسه سند
		لوگوی مشتری

* در زمان تولید هر مستند، ابتدا اطلاعات جدول بالا را اصلاح کنید.

توجه: این صفحه را پرینت بگیرید و از فایل PDF نیز حذف کنید.



شرکت نرم افزاری دانش آراین قشم

Enterprise Service Bus

سرویس های واریز گروهی جدید

تهیه کننده/کنندگان: اکرم احمدیان مهرگو

تاریخ: 1402/01/28



شرکت نرم‌افزاری داتیس آراین قشم

راهکارهای پردازش تراکنش

Enterprise Service Bus

سرویس های واریز گروهی جدید

تهیه کننده/کنندگان

اکرم احمدیان مهرگو

1402/01/28

شماره ویرایش: Ver1.0



لطفاً این سند را چاپ نکنید؛ مگر آنکه واقعاً به آن نیاز داشته باشید.
Please don't print this document unless you really need to.

شناسنامه سند	
عنوان سند	سرویس های واریز گروهی جدید
واحد تهیه کننده سند	راهکارهای پردازش تراکنش
نام پروژه	Enterprise Service Bus
درجه محرمانگی سند	عمومی
تعداد کل صفحات	9
شناسه سند	ESB.GroupPaymentServices.181.Ver1.0

تأییدیه سند				
نام و نام خانوادگی	سمت سازمانی	تاریخ	امضاء	
اکرم احمدیان مهرگو	مالک محصول بانکداری باز	1402/01/28		تهیه شده توسط
				بازبینی شده توسط
				تأیید شده توسط
				تصویب شده توسط

ردیف	تاریخ ویرایش	نسخه	ویرایش کننده	شرح ویرایش
1	1402/04/04	0.1	مهسا بقایی	تکمیل مستند



فهرست مطالب

3 شرح مختصر عملکرد سرویس های واریز گروهی
7 1- سرویس رجیستر جهت واریز گروهی
8 2- سرویس استعلام وضعیت
12 3- سرویس ادامه عملیات پرداخت گروهی

شرح مختصر عملکرد سرویس های واریز گروهی

گاهی نیاز است که بانک از یک (سپرده مبدأ) داخلی به تعداد زیادی شبای مقصد (داخلی / خارجی) به صورت همزمان، پرداختی داشته باشد. لیست تمام شباهای مقصد و مبلغی که برای هر سپرده باید واریز شود از قبل مشخص شده است. به همین منظور در سیستم امکانی قرار داده شده است که بتوان از سپرده مبدأ تعیین شده به سپرده های داخل مقصد اعلامی در ورودی سرویس به اندازه مبلغ معین شده ی هر ردیف، پرداخت وجه صورت بگیرد. قابل ذکر است که پس از انتهای کامل فرایند واریز گروهی، اگر به هر دلیلی برای ردیفی پرداخت با موفقیت انجام نشود، وجه برداشت شده از (سپرده مبدأ)، به (سپرده بازگشت وجه) معین شده بازگردانده می شود.

توجه: ردیف هایی که شماره شبای آنها داخلی بانک باشد، در نتیجه ی عملیات واریز گروهی، یک پرداخت عادی به آن شماره شبا صورت می گیرد و در صورتی که شماره شبای ردیف خارجی باشد، با توجه به مبلغ ردیف، یک درخواست ساتنا / پایا ثبت خواهد شد.

در پروسه واریز گروهی به چند سرویس نیاز است که در ادامه مستند به جزئیات شرح داده می شوند. پس از فراخوانی موفق سرویس های مورد نیاز، سیستم طبق قوانین واریز گروهی، ردیف های موجود در ابجکت لیستی ورودی را یکی یکی پرداخت می کند، و در نهایت در صورت پرداخت موفق، یا عدم پرداخت به هر دلیلی، تک تک ردیف ها تعیین تکلیف می شوند.

1. در صورتی که نتیجه ی فراخوانی سرویس سوم (ادامه عملیات پرداخت گروهی) موفق آمیز باشد، کد پیگیری عملیات در خروجی نمایش داده می شود.
2. در نتیجه ی مراحل فوق، آپلود اطلاعات با موفقیت روبه رو شده و زین پس سیستم ردیف به ردیف رکوردها را بررسی کرده و پرداخت می نماید
3. در صورتی که ردیف/ردیف هایی از رکوردهای مذکور، در هنگام پرداخت با خطا رو به رو شده باشند، بطور اتومات، تمام ردیف های خطادار مورد نظر برگشت داده می شوند.
4. برای گزارش گیری از جزئیات پرداخت تک تک ردیف های ارسالی، می توان از گزارشاتی که در دسترس قرار خواهد گرفت استفاده نمود.

توجه :

در صورت نیاز به فراخوانی سرویسها با توکن میبایست توکن را از SSO تراست دریافت نمایید، این توکن باید متعلق به یک بانکدار یا توکن client credential باشد، همچنین وجود کد شعبه در توکن اجباری می باشد. در صورت استفاده از توکن client credential نیازمند دریافت کاربر سیستمی از سامانه متمرکز و اعلام آن به تیم SSO می باشید. در صورت نیاز به دریافت راهنمایی بیشتر با تیم SSO ارتباط بگیرید

1- سرویس رجیستر

از این سرویس جهت ثبت درخواست انتقال وجه گروهی از یک مبدا به مقاصد مختلف، استفاده می‌شود. در ورودی این سرویس، لازم است اطلاعات تراکنش‌های مقصد در آبجکت لیستی (DocumentItems) طبق توضیحات مندرج در جدول زیر ارسال گردد.

نکات:

- در صورتی که در ورودی سرویس مقدار فیلد شماره مشتری: شماره مشتری سمات وارد شود، آنگاه مقدار فیلدهای (سپرده مبدا)، و (سپرده بازگشت وجه) نیز باید حتما یکسان پاس داده شود، در غیر این صورت پیغام خطای مربوطه در خروجی سرویس نمایش داده می‌شود، قابل ذکر است که برای سایر مشتریان نیز مقدار این دو ورودی می‌تواند یکسان یا متفاوت باشد.
- سپرده ی (بسته)، (مسدودی کل)، (مسدود برداشت) و سپرده ای که (دارای الگوریتم شناسه قبض) می‌باشد، به عنوان سپرده مبدا نمی‌تواند پاس داده شود، در غیر این صورت پیغام مربوطه در خروجی سرویس نمایش داده می‌شود.
- سپرده بسته، مسدود کل، مسدود واریز و سپرده ای که دارای الگوریتم شناسه قبض می‌باشد، به عنوان سپرده بازگشت وجه نمی‌تواند پاس داده شود، در غیر این صورت پیغام مربوطه در خروجی سرویس نمایش داده می‌شود.
- کاراکتر های #، \$، : غیر مجاز محسوب شده و در صورت وجود این کاراکتر های در فیلد های ورودی، پیغام خطا در خروجی سرویس نمایش داده می‌شود.
- این امکان وجود دارد که سپرده بازگشت وجه و سپرده مبدا کارمزد در ورودی وارد نشده و مقدار آن با سپرده مبدا پر شود. برای استفاده از این امکان، نیاز است بانک بازخورد لازم را به درگاه ارایه سرویس اعلام دارد.
- در صورتی که فیلد AutoContinue برابر با مقدار true باشد و برخی از رکوردها دارای خطا باشد، خطاهای مربوطه در خروجی نمایش داده شده و عملیات درج اطلاعات مربوط به سایر ردیف های بدون خطا ادامه می یابد.
- در صورتی که رکورد ساتنا بعد از زمان تعیین شده بانک مرکزی برای ارسال باشد، آماده ارسال به بانک مرکزی مانده و در روز کاری بعد انجام می‌گردد
- حداکثر مقدار مجاز فیلد های شناسه قبض و توضیحات مبدا، 255 کرکتر و مقادیر % ' " مجاز نمی باشد.
- در بانک پاسارگاد رابطه بین مشتری و شماره سپرده مبدا (یصورت رابطه» صاحب سپرده و امضا «یا» صاحب امضا «) بررسی شده و در صورت عدم تطابق CustomerNumber و SourceDeposit عملیات انتقال وجه انجام نخواهد شد.

شرح پارامترهای ورودی:

http://[IP]:[PORT]/GroupPayment/GroupPaymentRegister			آدرس
POST			متد
توضیحات	نوع فیلد	نام فیلد (موارد ستاره دار ضروری می باشند)	پارامترهای هدر
فرمت محتوی درخواست Application/json	String	Content-Type	
کلید منحصر به فرد هر کلاینت در سامانه ESB	String	ApiKey *	
امضای دیجیتال سرویس گیرنده	String	Signature *	
Accept-Version با مقدار 7	String	Accept-Version	
مقدار توکن sso	string	AccessToken	
شناسه یکتا به ازای هر درخواست میبایست یکتا باشد (فرمت در پیوست) توجه : لازم به ذکر است مقدار این فیلد در سرویس های "استعلام" و "ادامه عملیات پرداخت" استفاده می شود. مقدار این فیلد به منزله کد پیگیری جهت استفاده در سرویس های مرتبط با گزارشات واریز گروهی می باشد	String	TransactionId*	پارامترهای درخواست
ارسال مقدار true : در صورت وجود رکورد دارای خطا در لیست اطلاعات تراکنشهای ارسالی ، فرایند در esb به ازای رکوردهای صحیح ادامه پیدا کرده و رکوردهای دارای خطا ریجکت و در نظر گرفته نخواهند شد. (مقدار پیش فرض این فیلد true میباشد) ارسال مقدار false : در صورت وجود رکورد دارای خطا در لیست اطلاعات تراکنشهای ارسالی ، فرایند در esb متوقف شده و خطا ها در خروجی نمایش داده خواهند شد.	Boolean	AutoContinue	
در صورت ارسال مقدار در این فیلد تراکنش های کوچکتر مساوی این مبلغ بعنوان تراکنش پایا و تراکنش های بزرگتر از این مبلغ بعنوان ساتنا منظور خواهند شد و در صورت عدم ارسال فرایند طبق بیزینس موجود در سمت کور انجام خواهد شد	long	ThresholdAmount	
شماره مشتری شماره مشتری مورد نظر برای انجام عملیات واریز گروهی وارد می شود. می تواند یک شماره مشتری حقیقی یا حقوقی باشد	String	CustomerNumber*	
سپرده مبدا	String	SourceDeposit*	

شماره سپرده ای که به عنوان مبدا برداشت وجه در نظر گرفته می شود و به ازای (مجموع مبالغ مقاصد اعلامی) از آن برداشت می شود، در این ورودی پاس داده می شود.		
سپرده بازگشت وجه سپرده ای است که به عنوان مقصد (سند برگشت ردیف های خطادار) در نظر گرفته می شود، در این ورودی پاس داده می شود. در صورتی که مشتری بانک سپه باشد، این فیلد می تواند ارسال نشود. در سایر بانک ها این فیلد اجباری می باشد	String	RefundDeposit
شماره سپرده مبدا کارمزد اگر انتقال وجه شامل مقدار کارمزد باشد و این حساب پر نشده باشد کارمزد از حساب مبدا کسر می گردد	String	SourceDepositCommission
شرح برداشت از سپرده مبدا	String	SourceDescription
شناسه برداشت در صورتی که کاربر سپرده مبدا یا مبدا کارمزد را دارای شناسه قبض وارد نماید شناسه دریافتی از کاربر در این فیلد ذخیره می شود	String	BillNumber
(درج اطلاعات مطابق با جدول 1 می بایست باشد)	list	* DocumentItems
DocumentItems - جدول 1		
شماره رکورد (حداکثر 9 کاراکتر، و یکتا) باید از 1 شروع شده و ترتیبی باشند	Int	*LineNumber
مبلغ حداکثر 17 کاراکتر می باشد و امکان ورود عدد صفر، منفی، اعشاری، کاراکترهای غیر عددی وجود ندارد	رشته عددی	*Amount
شماره شبا گیرنده وجه (شبای مقصد) فقط 26 کاراکتر و فرمت شماره شبا طبق استانداردهای بانکها صحیح باشد. امکان ورود شماره شبای داخلی بانک، و خارجی (سایر بانک ها) وجود دارد، (همچنین امکان ورود شماره شباهای تکراری در لیست نیز وجود دارد)	String	*DestinationIban
تاریخ حداکثر 8 کاراکتر، فرمت مقدار تاریخ نیز باید شامل تمام اعداد سال، ماه، روز بدون جداکننده باشد. مانند: 13990624 این فیلد حالت رزرو دارد و در حال حاضر تنها کنترل روی این فیلد انجام میشود ۸ کاراکتر عددی بدون / است تاریخ آینده یا گذشته بودن آن در حال حاضر باعث خطا نمی شود اما جهت ثبت لاگ بایستی تاریخ جاری وارد شود.	Int	*TransactionDate
شناسه واریز حداکثر 30 کاراکتر حرفی/عددی می باشد در صورتی که شماره شبای وارد شده در ردیف، دارای الگوی شناسه قبض (واریز* و برداشت/ واریز) باشد، باید شناسه صحیح	String	TransactionBillNumber

وارد شده، و در غیر این صورت مقدار صفر وارد شده باشد. توجه شود که مقدار خالی نیز به منزله عدد صفر می باشد			
نام و نام خانوادگی گیرنده وجه (حداکثر 50 کاراکتر)	String	*RecieverFullName	
شرح- حداکثر 50 کاراکتر، همان شرح بند سند حسابداری مربوط به واریز به سپرده مقصد خواهد بود	String	Description	
بابت مقادیر مجاز مطابق با جدول شماره 3	String	CauseType	
نتیجه موفق یا ناموفق	Boolean	IsSuccess	پارامترهای پاسخ
کد پاسخ	Int	Rscore	
پیغام	String	Message	
خروجی ناموفق - جدول شماره 2	List	ErrorList	

ErrorList- خروجی ناموفق - جدول شماره 2		
نام فیلد	نوع فیلد	توضیحات
Code	Int	کد خطا
Desc	String	پیغام خطا
ParamName	String	نام پارامتر دارای خطا
ParamPath	String	مسیر پارامتر دارای خطا

جدول شماره 3 - مقادیر مجاز فیلد بابت

کد بابت	بابت
1	واریز حقوق
2	امور بیمه خدمات
3	امور درمانی
4	امور سرمایه گذاری و بورس
5	امور ارزی در چهارچوب ضوابط و مقررات
6	پرداخت قرض و تادیه دیون(قرض الحسنه,بدهی,...)
7	امور بازنشستگی
8	معاملات اموال منقول
9	معاملات اموال غیر منقول
10	مدیریت نقدینگی
11	عوارض گمرکی
12	تسویه مالیاتی
13	سایر خدمات دولتی
14	تسهیلات و تعهدات
15	تودیع وثیقه
16	هزینه عمومی و امور روزمره
17	کمکهای خیریه
18	خرید کالا
19	خرید خدمات

ساختار خروجی موفق:

در صورتی که سرویس با موفقیت اجرا شده و پاسخ صحیح بازگرداند و تمام رکورد ها بدون خطا باشد خروجی به صورت زیر است :

```
{
  "IsSuccess": true,
  "Message": "عملیات با موفقیت انجام شد",
  "RsCode": 1
}
```

در صورتی که مقدار AutoContinue برابر با true بوده و تعدادی از رکوردها دارای خطا باشد ، ساختار خروجی موفق به صورت زیر خواهد بود:

```
{
  "RsCode": 4456,
  "Message": "یک یا چند خطای اعتبارسنجی رخ داده است",
  "IsSuccess": false,
  "ErrorList": [
```

```
{
  "Code": 1038,
  "Desc": "شبا مقصد نامعتبر است ",
  "ParamName": " DestinationIban ",
  "ParamPath": " DocumentItems[10].DestinationIban"
}
```

ساختار خروجی ناموفق:

در صورتی که در اعتبار سنجی پارامتر های ورودی ، مقادیر ارسالی طبق توضیحات مندرج در جدول ورودی ها نباشد .ساختار پاسخ ناموفق سرویس به صورت زیر خواهد بود :

```
{
  "RsCode": 1038,
  "Message": "اطلاعات ورودی اشتباه می باشد ",
  "IsSuccess": false,
  "ErrorList": [
    {
      "Code": 1038,
      "Desc": "اطلاعات ورودی اشتباه می باشد ",
      "ParamName": " TransactionId ",
      "ParamPath": " TransactionId "
    }
  ]
}
```

در صورت بروز سایر خطاها ساختار خروجی ناموفق به صورت زیر می باشد :

```
{
  "IsSuccess": false,
  "Message": "خطا در پردازش تراکنش بانکی ",
  "RsCode": 0
}
```

ساختار خروجی ناموفق در صورت ارسال AutoContinue با مقدار false :

```
{
  "RsCode": 4456,
  "Message": "یک یا چند خطای اعتبارسنجی رخ داده است ",
  "IsSuccess": false,
  "ErrorList": [
    {
      "Code": 1038,
```

```

"Desc": " شبا مقصد نامعتبر است ",
"ParamName": " DestinationIban ",
"ParamPath": " DocumentItems[10].DestinationIban "
}
]
}

```

جدول برخی از کد خطا های سرویس		
کد خطا	عنوان خطا	توضیحات
1	عملیات با موفقیت انجام شد	
804	مدت زمان پردازش پیام منقضی شده است	
1038	اطلاعات ورودی اشتباه می باشد	
1207	سپرده بازگشت وجه وارد نشده است.	
4456	یک یا چند خطای اعتبارسنجی رخ داده است	
1644	شناسه ی تراکنش صحیح نمی باشد	
4501	کلید رکوردها دارای خطا می باشند	
4459	شناسه تراکنش تکراری است	
4419	مقدار فیلد مبلغ مجاز از حداکثر سقف پایای تعیین شده بیشتر و یا از حداقل کف ساتنای تعیین شده کمتر است	
0	خطا در پردازش تراکنش بانکی	

توجه:

لازم به ذکر است ساختار خروجی ناموفق در صورت بروز خطای اعتبارسنجی در حالت ارسال مقدار true و یا false برای فیلد AutoContinue یکسان می باشد و تنها تفاوت در ادامه عملیات و درج و ذخیره اطلاعات در پایگاه داده می باشد.

در صورتی که مقدار این فیلد برابر با false باشد، و بعضی از رکورد ها با خطا مواجه شوند، امکان ادامه عملیات و فراخوانی سرویس DoPayment میسر نمی باشد.

در صورتی که این مقدار برابر با true باشد، عملیات درج داده برای ردیف های بدون خطا، به صورت اتوماتیک ادامه می یابد.



مراحل فراخوانی سرویس:

1- با استفاده از داده های موجود رشته زیر را ایجاد نمایید. آیتم های موجود در این رشته با علامت # از یکدیگر جدا شده اند.

httpMethod(Upper Case)#url(after Address Root)#ApiKey# SourceDeposit#TotalCount#TotalAmount

نمونه رشته بالا:

POST#/GroupPayment/GroupPaymentRegister#123e4567-e89b-12d3-a456-426655440000#
1.10.409133.2#10#100000000

2- Sign کردن رشته بالا با Private Key و تبدیل خروجی نهایی به Base64

3- اضافه کردن Apikey و رشته sign شده به Header درخواست با کلیدهای زیر:

ApiKey
Signature

4- اضافه کردن Body به بدنه درخواست (در صورتی که نوع متد Get باشد این مقدار خالی می باشد)

5- ارسال درخواست به آدرس ذکر شده در بالای صفحه. توجه داشته باشید که ContentType درخواست باید application/json باشد.

2- سرویس استعلام وضعیت

هدف از این سرویس، دریافت نتیجه فرایند سرویس رجیستر می باشد

شرح پارامترهای ورودی :

http://[IP]:[PORT]/GroupPayment/GroupPaymentInquiry				آدرس	
POST				متد	
توضیحات		نوع فیلد	نام فیلد (موارد ستاره دار ضروری می باشند)		
فرمت محتوی درخواست Application/json		String	Content-Type		
کلید منحصر به فرد هر کلاینت در سامانه ESB		String	ApiKey *		
امضای دیجیتال سرویس گیرنده		String	Signature *		
Accept-Version با مقدار 7		String	Accept-Version		
مقدار توکن SSO		string	AccessToken		
شناسه یکتایی که توسط سرویس اول (رجیستر) ارسال شده است، وارد می شود .		String	TransactionId*		
نتیجه موفق یا ناموفق		Boolean	IsSuccess		
کد پاسخ		Int	Rrcode		
پیغام		String	Message		
طبق جدول شماره 1		Class	ResultData	خروجی موفق	
طبق جدول شماره 2		List	ErrorList	خروجی ناموفق	
ErrorList- خروجی نا موفق - جدول شماره 2					
توضیحات	نوع فیلد				نام فیلد
کد خطا	Int				Code
پیغام خطا	String				Desc
نام پارامتر دارای خطا	String				ParamName
مسیر پارامتر دارای خطا	String	ParamPath			

ResultData جدول شماره 1 -			
وضعیت تراکنش گروهی Processing به معنای رکورد ها در حال پردازش هستند Failed, به معنای عدم موفقیت Ready, رکورد ها آماده پرداخت هستند Done عملیات با موفقیت انجام شده هست		String	TransactionStatus
لیست رکوردهای دارای خطای جیسون ارسالی در سرویس رجیستر		List	RecordErrorsList
توضیحات	نوع فیلد		
نام فیلد			

کد خطا	Int	Code		
پیغام خطا	String	Desc		
نام پارامتر دارای خطا	String	ParamName		
مسیر پارامتر دارای خطا	String	ParamPath		
نتیجه بازخوانی رکوردهای تراکنش گروهی - طبق جدول زیر			class	Result
Result				
شناسه یکتا (آنچه در ورودی ارسال شده است)	String	TransactionId		
تاریخ انجام عملیات	String	DateTime		
شماره مشتری	String	CustomerNumber		
وضعیت فایل	String	State		
مقادیر مجاز				
GROUP_PAYMENT_WAITING_STATE				
GROUP_PAYMENT_REGISTERED_STATE				
GROUP_PAYMENT_UPLOADING_STATE				
GROUP_PAYMENT_ERROR_STATE				
مجموع مبالغ فایل	String	TotalAmount		
تعداد سطرهای فایل	String	LineCount		
مجموع مبالغ داخلی	String	TotalInternalAmount		
مجموع مبالغ پایا	String	TotalPayaAmount		
مجموع مبالغ ساتنا	String	TotalSatnaAmount		
مبلغ کل کارمزد های پایا	String	TotalCommissionPayaAmount		
مبلغ کل کارمزد های ساتنا	String	TotalCommissionSatnaAmount		
مبلغ کل کارمزد های داخلی	String	totalCommissionInternalAmount		
تعداد رکورد های پایا	String	CountPaya		
تعداد رکورد های داخلی	String	InternalRecordCount		
سپرده مبدا	String	SourceDeposit		
سپرده بازگشت وجه	String	RefundDeposit		
شماره سپرده مبدا کارمزد	String	SourceDepositCommission		
سقف تراکنش پایا	long	ThresholdAmount		

ساختار خروجی موفق:

در صورتی که سرویس با موفقیت اجرا شده و پاسخ صحیح بازگرداند خروجی به صورت زیر است :

```
{
  "IsSuccess": true,
  "Message": "عملیات با موفقیت انجام شد",
  "RsCode": 1,
  "ResultData": {
```



```

"TransactionStatus": "Ready",
"RecordErrorsList": [
{
  "Desc": "فرمت تاریخ نامعتبر است.",
  "Code": 1038,
  "ParamName": "TransactionDate",
  "ParamPath": "DocumentItems[1].TransactionDate"
}
],
"Result": {
  "TransactionId": "C5163-Ec07802118f314b1b92bc08077a357e8e-2432",
  "DateTime": "1402/03/30",
  "CustomerNumber": "409133",
  "State": "GROUP_PAYMENT_WAITING_STATE",
  "LineCount": "1",
  "TotalAmount": "10000000",
  "TotalInternalAmount": "10000000",
  "TotalPayaAmount": null,
  "TotalSatnaAmount": null,
  "TotalCommissionPayaAmount": "0",
  "TotalCommissionSatnaAmount": "0",
  "TotalCommissionInternalAmount": null,
  "CountPaya": "0",
  "InternalRecordCount": null,
  "SourceDeposit": "1.10.409133.1",
  "RefundDeposit": "1.10.409133.1",
  "SourceDepositCommission": "1.10.409133.1",
  "ThresholdAmount": 0
}
}
}

```

ساختار خروجی ناموفق:

در صورتی که در اعتبار سنجی پارامتر های ورودی ، مقادیر ارسالی طبق توضیحات مندرج در جدول ورودی ها نباشد .ساختار پاسخ ناموفق سرویس به صورت زیر خواهد بود :

```

{
  "RsCode": 1038,
  "Message": "اطلاعات ورودی اشتباه می باشد.",
  "IsSuccess": false,
  "ErrorList": [
    {
      "Code": 1038,
      "Desc": "اطلاعات ورودی اشتباه می باشد.",
      "ParamName": " TransactionId ",

```

```

    "ParamPath": " TransactionId "
  }
]
}

```

در صورت بروز سایر **خطاها** ساختار خروجی ناموفق به صورت زیر می باشد :

```

{
  "IsSuccess": false,
  "Message": "تراکنش یافت نشد",
  "RsCode": 1108
}

```

مراحل فراخوانی سرویس:

1- با استفاده از داده های موجود رشته زیر را ایجاد نمایید. آیتم های موجود در این رشته با علامت # از یکدیگر جدا شده اند.
 httpMethod(Upper Case)#url(after Address Root)#ApiKey#body

نمونه رشته بالا:

```

POST#/GroupPayment/GroupPaymentInquiry#123e4567-e89b-12d3-a456-426655440000#{"TransactionId": "C5163-E73bfddbfe34f2cac712548e0e0f848-2781"}

```

- 2- Sign کردن رشته بالا با Private Key به روش SHA1 و تبدیل خروجی نهایی به Base64 .
- 3- اضافه کردن Apikey و رشته sign شده به Header درخواست با کلیدهای زیر: ApiKey و Signature
- 4- اضافه کردن Body به بدنه درخواست (در صورتی که نوع متد Get باشد این مقدار خالی می باشد).
- 5- ارسال درخواست به آدرس ذکر شده در بالای صفحه. توجه داشته باشید که ContentType درخواست باید application/json باشد.

3- سرویس عملیات پرداخت گروهی

هدف از این سرویس، ثبت نهایی عملیات واریز گروهی می باشد.

شرح پارامترهای ورودی :

http://[IP]:[PORT]/GroupPayment/DoPayment				آدرس		
POST				متد		
توضیحات		نوع فیلد	نام فیلد (موارد ستاره دار ضروری می باشند)			
فرمت محتوی درخواست Application/json		String	Content-Type			
کلید منحصر به فرد هر کلاینت در سامانه ESB		String	ApiKey *			
امضای دیجیتال سرویس گیرنده		String	Signature *			
Accept-Version با مقدار 7		String	Accept-Version			
مقدار توکن sso		string	AccessToken			
شناسه یکتایی که در سرویس اول (رجیستر) ارسال شده، وارد می شود .		String	TransactionId*			
نتیجه موفق یا ناموفق		Boolean	IsSuccess			
کد پاسخ		Int	Rrcode			
پیغام		String	Message			
طبق جدول شماره 1		Class	ResultData	خروجی موفق		
ResultData – خروجی موفق – جدول شماره 1						
نام فیلد					نوع فیلد	توضیحات
TransactionId					String	شناسه یکتا (آنچه در ورودی ارسال شده است)
طبق جدول شماره 2		List	ErrorList	خروجی ناموفق		
ErrorList – خروجی نا موفق – جدول شماره 2						
نام فیلد					نوع فیلد	توضیحات
Code					Int	کد خطا
Desc					String	پیغام خطا
ParamName					String	نام پارامتر دارای خطا
ParamPath		String	مسیر پارامتر دارای خطا			

ساختار خروجی موفق:

در صورتی که سرویس با موفقیت اجرا شده و پاسخ صحیح بازگرداند خروجی به صورت زیر است :

```
{
  "RsCode": 1,
  "Message": "عملیات با موفقیت انجام شد",
  "IsSuccess": true,
  "ResultData": {
    " TransactionId ": " C5163-E73bfddbfefd34f2cac712548e0e0f848-2781"
  }
}
```

ساختار خروجی ناموفق:

در صورتی که در اعتبار سنجی پارامتر های ورودی ، مقادیر ارسالی طبق توضیحات مندرج در جدول ورودی ها نباشد .ساختار پاسخ ناموفق سرویس به صورت زیر خواهد بود :

```
{
  "RsCode": 1038,
  "Message": "اطلاعات ورودی اشتباه می باشد",
  "IsSuccess": false,
  "ErrorList": [
    {
      "Code": 1038,
      "Desc": "اطلاعات ورودی اشتباه می باشد",
      "ParamName": " TransactionId ",
      "ParamPath": " TransactionId "
    }
  ]
}
```

در صورت بروز سایر **خطاها** ساختار خروجی ناموفق به صورت زیر می باشد :

```
{
  "IsSuccess": false,
  "Message": "تراکنش یافت نشد",
  "RsCode": 1108
}
```

مراحل فراخوانی سرویس:

1- با استفاده از داده های موجود رشته زیر را ایجاد نمایید. آیتم های موجود در این رشته با علامت # از یکدیگر جدا شده اند.
httpMethod(Upper Case)#url(after Address Root)#ApiKey# body

نمونه رشته بالا:

POST#/GroupPayment/DoPayment#123e4567-e89b-12d3-a456-426655440000#{ "TransactionId": "C5163-E73bfddbfefd34f2cac712548e0e0f848-2781" }

- 2- Sign کردن رشته بالا با Private Key به روش SHA1 و تبدیل خروجی نهایی به Base64 .
- 3- اضافه کردن Apikey و رشته sign شده به Header درخواست با کلیدهای زیر: ApiKey و Signature
- 4- اضافه کردن Body به بدنه درخواست (در صورتی که نوع متد Get باشد این مقدار خالی می باشد).
- 5- ارسال درخواست به آدرس ذکر شده در بالای صفحه. توجه داشته باشید که ContentType درخواست باید application/json باشد.

جدول کد خطا - سرویس عملیات پرداخت گروهی		
کد خطا	عنوان خطا	توضیحات
1	عملیات با موفقیت انجام شد	
1038	اطلاعات ورودی اشتباه است	
804	مدت زمان پردازش پیام منقضی شده است	
1108	تراکنش یافت نشد	

پیوست :

نحوه ایجاد شناسه یکتای تراکنش TransactionId

شناسه تراکنش شامل چهار بخش به شکل زیر می باشد:

OrgCode - Random String – DateTime-Sum Char Code(OrgCode+Random String)

نمونه:

C3109-Ee877c33d24104d3b92290a18fb0d8828-20200927212945073-3310

OrgCode: کد سازمان که می بایست از ESB دریافت گردد. (لازم به ذکر است در زمان تست این کد ارایه میگردد).
Random String: یک رشته تصادفی که به ازای هر کلاینت یکتا است ، حداقل طول آن 8 رقم است و حداکثر طول آن 100 می باشد.

DateTime: با دقت میلی ثانیه و با فرمت مقابل 20200631125438441 (تاریخ میبایست با فرمت میلادی باشد)
Sum Char Code: جمع کد اسکی کاراکترهای سه بخش قبلی (OrgCode، Random String و DateTime) می باشد.