Департамент информационных технологий города Москвы

Руководство пользователя

электронного сервиса Рсмэв

Сервис информирования информационной системы проход и питание по УЭК

Дата: 26.03.12

Версия: 1.2

Листов:

АННОТАЦИЯ

Данный документ является руководством пользователя электронного сервиса информирования информационной системы проход и питание по УЭК.

СОДЕРЖАНИЕ

[1 Общие сведения 7](#_Toc352057945)

[1.1 Руководящие документы 7](#_Toc352057946)

[1.2 Описание электронного сервиса 7](#_Toc352057947)

[1.3 Операции (методы) электронного сервиса 7](#_Toc352057948)

[1.4 Сценарии использования и схема взаимодействия 8](#_Toc352057949)

[1.5 Связи с другими электронными сервисами 8](#_Toc352057950)

[2 Руководство пользователя 9](#_Toc352057951)

[2.1 Операция «Получение данных о лицевом счете» 9](#_Toc352057952)

[2.1.1 Общие сведения 9](#_Toc352057953)

[2.1.2 Описание входных параметров 9](#_Toc352057954)

[2.1.3 Описание выходных параметров 9](#_Toc352057955)

[2.1.4 Ошибки 9](#_Toc352057956)

[2.1.5 Контрольные примеры 10](#_Toc352057957)

[2.2 Операция «Получение данных о лицевом счете через идентификатор» 11](#_Toc352057958)

[2.2.1 Общие сведения 11](#_Toc352057959)

[2.2.2 Описание входных параметров 11](#_Toc352057960)

[2.2.3 Описание выходных параметров 11](#_Toc352057961)

[2.2.4 Ошибки 11](#_Toc352057962)

[2.2.5 Контрольные примеры 12](#_Toc352057963)

[2.3 Операция «Получение данных о покупках с лицевого счета» 13](#_Toc352057964)

[2.3.1 Общие сведения 13](#_Toc352057965)

[2.3.2 Описание входных параметров 13](#_Toc352057966)

[2.3.3 Описание выходных параметров 13](#_Toc352057967)

[2.3.4 Ошибки 13](#_Toc352057968)

[2.3.5 Контрольные примеры 14](#_Toc352057969)

[2.4 Операция «Получение данных о пополнениях лицевого счета» 15](#_Toc352057970)

[2.4.1 Общие сведения 15](#_Toc352057971)

[2.4.2 Описание входных параметров 15](#_Toc352057972)

[2.4.3 Описание выходных параметров 15](#_Toc352057973)

[2.4.4 Ошибки 15](#_Toc352057974)

[2.4.5 Контрольные примеры 16](#_Toc352057975)

[2.5 Операция «Получение информации о меню столовой» 17](#_Toc352057976)

[2.5.1 Общие сведения 17](#_Toc352057977)

[2.5.2 Описание входных параметров 17](#_Toc352057978)

[2.5.3 Описание выходных параметров 17](#_Toc352057979)

[2.5.4 Ошибки 17](#_Toc352057980)

[2.5.5 Контрольные примеры 18](#_Toc352057981)

[2.6 Операция «Получение данных посещениях образовательного учреждения» 23](#_Toc352057982)

[2.6.1 Общие сведения 23](#_Toc352057983)

[2.6.2 Описание входных параметров 23](#_Toc352057984)

[2.6.3 Описание выходных параметров 23](#_Toc352057985)

[2.6.4 Ошибки 23](#_Toc352057986)

[2.6.5 Контрольные примеры 24](#_Toc352057987)

[2.7 Операция «Включить/отключить СМС-информирование» 25](#_Toc352057988)

[2.7.1 Общие сведения 25](#_Toc352057989)

[2.7.2 Описание входных параметров 25](#_Toc352057990)

[2.7.3 Описание выходных параметров 25](#_Toc352057991)

[2.7.4 Ошибки 25](#_Toc352057992)

[2.7.5 Контрольные примеры 25](#_Toc352057993)

[2.8 Операция «Изменить номер мобильного телефона для СМС-информирования» 27](#_Toc352057994)

[2.8.1 Общие сведения 27](#_Toc352057995)

[2.8.2 Описание входных параметров 27](#_Toc352057996)

[2.8.3 Описание выходных параметров 27](#_Toc352057997)

[2.8.4 Ошибки 27](#_Toc352057998)

[2.8.5 Контрольные примеры 27](#_Toc352057999)

[2.9 Операция «Включить/отключить E-Mail-информирование» 29](#_Toc352058000)

[2.9.1 Общие сведения 29](#_Toc352058001)

[2.9.2 Описание входных параметров 29](#_Toc352058002)

[2.9.3 Описание выходных параметров 29](#_Toc352058003)

[2.9.4 Ошибки 29](#_Toc352058004)

[2.9.5 Контрольные примеры 29](#_Toc352058005)

[2.10 Операция «Изменить адрес e-mail для информирования» 31](#_Toc352058006)

[2.10.1 Общие сведения 31](#_Toc352058007)

[2.10.2 Описание входных параметров 31](#_Toc352058008)

[2.10.3 Описание выходных параметров 31](#_Toc352058009)

[2.10.4 Ошибки 31](#_Toc352058010)

[2.10.5 Контрольные примеры 31](#_Toc352058011)

[2.11 Операция «Применить код активации» 33](#_Toc352058012)

[2.11.1 Общие сведения 33](#_Toc352058013)

[2.11.2 Описание входных параметров 33](#_Toc352058014)

[2.11.3 Описание выходных параметров 33](#_Toc352058015)

[2.11.4 Ошибки 33](#_Toc352058016)

[2.11.5 Контрольные примеры 33](#_Toc352058017)

[2.12 Операция «Сформировать код активации» 35](#_Toc352058018)

[2.12.1 Общие сведения 35](#_Toc352058019)

[2.12.2 Описание входных параметров 35](#_Toc352058020)

[2.12.3 Описание выходных параметров 35](#_Toc352058021)

[2.12.4 Ошибки 35](#_Toc352058022)

[2.12.5 Контрольные примеры 35](#_Toc352058023)

[2.13 Операция «Сформировать код активации и отправить по номеру лицевого счета» 37](#_Toc352058024)

[2.13.1 Общие сведения 37](#_Toc352058025)

[2.13.2 Описание входных параметров 37](#_Toc352058026)

[2.13.3 Описание выходных параметров 37](#_Toc352058027)

[2.13.4 Ошибки 37](#_Toc352058028)

[2.13.5 Контрольные примеры 37](#_Toc352058029)

[2.14 Операция «Сформировать код активации и отправить по e-mail и SMS» 39](#_Toc352058030)

[2.14.1 Общие сведения 39](#_Toc352058031)

[2.14.2 Описание входных параметров 39](#_Toc352058032)

[2.14.3 Описание выходных параметров 39](#_Toc352058033)

[2.14.4 Ошибки 39](#_Toc352058034)

[2.14.5 Контрольные примеры 39](#_Toc352058035)

[2.15 Изменить размер дневного ограничения на затраты по номеру контракта ребенка 41](#_Toc352058036)

[2.15.1 Общие сведения 41](#_Toc352058037)

[2.15.2 Описание входных параметров 41](#_Toc352058038)

[2.15.3 Описание выходных параметров 41](#_Toc352058039)

[2.15.4 Ошибки 41](#_Toc352058040)

[2.15.5 Контрольные примеры 42](#_Toc352058041)

[3 Приложения 43](#_Toc352058042)

[3.1 Описание сервиса (WSDL) 43](#_Toc352058043)

[3.2 Описание общих структур данных 63](#_Toc352058044)

изменения

|  |  |  |  |
| --- | --- | --- | --- |
| Версия | Дата | Автор | Изменения |
| 1.0 | 30.07.2012 | Скворцов А.В. | Первичная версия документа |
| 1.1 | 13.08.2012 | Скворцов А.В. | Добавлены операции для работы с кодами активации |
| 1.2 | 26.03.2013 | Кадыров Д.И. | Добавлена операция изменения размер дневного ограничения на затраты по номеру контракта ребенка |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

термины/СОКРАЩЕНИя

|  |  |
| --- | --- |
| Термин/сокращение | Описание |
| РСМЭВ | Региональная система межведомственного электронного взаимодействия города Москвы |
| ИС ПП | Информационная система проход и питание по УЭК |
| Поставщик | Организация-владелец электронного сервиса |
| WSDL | Web Services Description Language — язык описания [веб-сервисов](http://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B8%D1%81" \o "Веб-сервис). |
| XSD | XML Schema definition - язык описания структуры XML документа. |
| SOAP | Simple Object Access Protocol - протокол обмена структурированными сообщениями в распределённой вычислительной среде. |
| XML | eXtensible Markup Language — расширяемый язык разметки. |

1. Общие сведения

Руководящие документы

Основанием для создания и использования электронного сервиса являются перечисленные ниже документы.

Нормативно-правовые документы:

* Постановление Правительства Российской Федерации от 8 сентября 2010 г. № 697 «О единой системе межведомственного электронного взаимодействия»;
* Приказ Министерства связи и массовых коммуникаций Российской Федерации от 27 декабря 2010 г. № 190 «Об утверждении технических требований к взаимодействию информационных систем в единой системе межведомственного электронного взаимодействия».

Описание электронного сервиса

|  |  |
| --- | --- |
| **Наименование:** | Сервис информирования регистрации платежей системы проход и питание по УЭК |
| **Назначение:** | Предоставление доступа к данным о посещении образовательных учреждений, меню поставщиков питания, оказанных услугах питания, информации о статусе лицевого счета держателя |
| **Поставщик:** | Департамент информационных технологий г. Москвы |
| **Область применения:** | Предоставление открытого доступа к данным |

Операции (методы) электронного сервиса

|  |  |
| --- | --- |
| **Метод** | **Назначение** |
| getSummary  Получение данных о лицевом счете | Предоставление информации о ЛС: ФИО, баланс, статус, параметры лицевого счета, информация о нахождении в ОУ |
| getSymmaryByTypeId  Получение данных о лицевом счете через идентификатор | Получение данных о лицевом счете через СНИЛС опекуна, GUID или внешний идентификатор |
| getPurchaseList  Получение данных о покупках с лицевого счета | Предоставление информации о покупках: дата, сумма, состав |
| getPaymentList  Получение данных о пополнениях лицевого счета | Предоставление информации о пополнениях: дата, сумма, источник |
| getMenuList  Получение информации о меню столовой | Получение информации о меню столовой: дата, наименование, цена |
| getEnterEventList  Получение данных посещениях образовательного учреждения | Получение информации о посещении образовательного учреждения: дата, направление входа |
| enableNotificationBySMS  Включить/отключить СМС-информирование | Изменение настроек СМС-информирования |
| changeMobilePhone  Изменить номер мобильного телефона для СМС-информирования | Изменение настроек СМС-информирования (номер телефона) |
| enableNotificationByEmail  Включить/отключить E-Mail-информирование | Изменение настроек Email-информирования лицевого счета |
| changeEmail  Изменить адрес e-mail для информирования | Изменение настроек Email-информирования (адрес) |
| activateLinkingToken  Применить код активации | Применить код активации для связки с лицевым счетом |
| generateLinkingToken  Сформировать код активации | Сформировать новый код активации для лицевого счета |
| sendLinkingTokenByContractId  Сформировать код активации и отправить по номеру лицевого счета | Сформировать код активации и отправить по e-mail и SMS для указанного лицевого счета |
| sendLinkingTokenByMobile  Сформировать код активации и отправить по номеру телефона | Сформировать код активации и отправить по e-mail и SMS для указанного номера телефона |
| changeExpenditureLimit()  Изменить размер дневного ограничения на затраты по номеру контракта ребенка | Изменить размер дневного ограничения на затраты по номеру контракта ребенка |

Сценарии использования и схема взаимодействия

1. Информационная система вызывает сервис в синхронном режиме используя getSummaryByTypedId, передавая идентификатор клиента, которым он идентифицируется в ее базе данных, в ответ возвращаются данные о лицевом счете и номер лицевого счета
2. Информационная система получает данные, используя методы getPurchaseList, getPaymentList, getEnterEventList, getMenuList

Связи с другими электронными сервисами

Данный сервис не имеет связей с другими сервисами.

1. Руководство пользователя

## Операция «Получение данных о лицевом счете»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getSummary |
| **Наименование операции:** | Получение данных о лицевом счете |
| **Назначение операции:** | Предоставление информации о ЛС: ФИО, баланс, статус, параметры лицевого счета, информация о нахождении в ОУ |

Описание входных параметров

**Входные данные: getSummary**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |

Описание выходных параметров

**Выходные данные: clientSummaryResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | clientSummary | Данные лицевого счета | + | ClientSummaryExt |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getSummary>*  *<contractId>200485</contractId>*  *</soap:getSummary>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getSummaryResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<clientSummary ContractId="200485" DateOfContract="2011-09-09" StateOfContract="Активен" Balance="566389" OverdraftLimit="100000" ExpenditureLimit="20000" FirstName="Петр" LastName="Петров" MiddleName="Иванович" OfficialName="Демонстрационная школа" LastEnterEventCode="5" LastEnterEventTime="2012-04-16T13:35:47" NotifyViaSMS="true" NotifyViaEmail="false" MobilePhone="79161111111" Email="email" DefaultMerchantId="52" DefaultMerchantInfo="{TSP.SBID=123123}"/>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getSummaryResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Получение данных о лицевом счете через идентификатор»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getSymmaryByTypeId |
| **Наименование операции:** | Получение данных о лицевом счете через идентификатор |
| **Назначение операции:** | Предоставление информации о ЛС: ФИО, баланс, статус, параметры лицевого счета, информация о нахождении в ОУ |

Описание входных параметров

**Входные данные: getSummaryByTypedId**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | id | Идентификатор | + | xs:string |  |
| 2 | idType | Тип идентификатора | + | xs:int | 0 – номер лицевого счета, 1 – СНИЛС представителя, 2 – внешний идентификатор, 3 – GUID |

Описание выходных параметров

**Выходные данные: clientSummaryResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | clientSummary | Данные лицевого счета | + | ClientSummaryExt |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |
| 4 | 120 | Найдено больше одного лицевого счета | Для данного идентификатора найдено больше одного лицевого счета |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getSummaryByTypedId>*  *<id>df7d827f-f425-4205-961e-6eb79231c58e</id>*  *<idType>3</idType>*  *</soap:getSummaryByTypedId>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getSummaryByTypedIdResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<clientSummary ContractId="200485" DateOfContract="2011-09-09" StateOfContract="Активен" Balance="566389" OverdraftLimit="100000" ExpenditureLimit="20000" FirstName="Петр" LastName="Петров" MiddleName="Иванович" OfficialName="Демонстрационная школа" LastEnterEventCode="5" LastEnterEventTime="2012-04-16T13:35:47" NotifyViaSMS="true" NotifyViaEmail="false" MobilePhone="79161111111" Email="email" DefaultMerchantId="52" DefaultMerchantInfo="{TSP.SBID=123123}"/>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getSummaryByTypedIdResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Получение данных о покупках с лицевого счета»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getPurchaseList |
| **Наименование операции:** | Получение данных о покупках с лицевого счета |
| **Назначение операции:** | Предоставление информации о покупках: дата, сумма, состав |

Описание входных параметров

**Входные данные: getPurchaseList**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | startDate | Начальная дата выборки | + | xs:dateTime |  |
| 3 | endDate | Конечная дата выборки | + | xs:dateTime |  |

Описание выходных параметров

**Выходные данные: purchaseListResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | purchaseList | Список покупок | + | PurchaseListExt |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getPurchaseList>*  *<contractId>200485</contractId>*  *<startDate>2012-01-17T00:00:00Z</startDate>*  *<endDate>2012-02-17T00:00:00Z</endDate>*  *</soap:getPurchaseList>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getPurchaseListResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<purchaseList>*  *<P Time="2012-02-14T21:37:36" Sum="0" SocDiscount="1500" TrdDiscount="0" Donation="0" ByCash="0" ByCard="0" IdOfCard="0">*  *<E Name="Хлеб Сельский" Sum="0" Amount="1" Type="2"/>*  *<E Name="Кофейный напиток с молоком" Sum="0" Amount="1" Type="2"/>*  *<E Name="Каша молочная пшенная с маслом" Sum="0" Amount="1" Type="2"/>*  *<E Name="Сыр Голландский" Sum="0" Amount="1" Type="2"/>*  *<E Name="ГПД завтрак" Sum="0" Amount="1" Type="1"/>*  *</P>*  *<P Time="2012-02-14T21:37:55" Sum="1425" SocDiscount="0" TrdDiscount="75" Donation="0" ByCash="0" ByCard="1425" IdOfCard="0">*  *<E Name="Хлеб Сельский" Sum="0" Amount="1" Type="2"/>*  *<E Name="Кофейный напиток с молоком" Sum="0" Amount="1" Type="2"/>*  *<E Name="Каша молочная пшенная с маслом" Sum="0" Amount="1" Type="2"/>*  *<E Name="Сыр Голландский" Sum="0" Amount="1" Type="2"/>*  *<E Name="ГПД завтрак" Sum="1425" Amount="1" Type="1"/>*  *</P>*  *</purchaseList>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getPurchaseListResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Получение данных о пополнениях лицевого счета»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getPaymentList |
| **Наименование операции:** | Получение данных о пополнениях лицевого счета |
| **Назначение операции:** | Предоставление информации о пополнениях: дата, сумма, источник |

Описание входных параметров

**Входные данные: getPaymentList**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | startDate | Начальная дата выборки | + | xs:dateTime |  |
| 3 | endDate | Конечная дата выборки | + | xs:dateTime |  |

Описание выходных параметров

**Выходные данные: paymentListResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | paymentList | Список пополнений | + | PaymentList |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getPaymentList>*  *<contractId>200485</contractId>*  *<startDate>2012-01-17T00:00:00Z</startDate>*  *<endDate>2012-02-17T00:00:00Z</endDate>*  *</soap:getPaymentList>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getPaymentListResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<paymentList>*  *<P Time="2012-02-15T12:26:56" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T12:26:58" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T12:48:27" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T12:48:27" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T12:58:35" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T12:58:37" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-15T13:55:59" Sum="2000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T20:11:56" Sum="10000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T20:13:25" Sum="10000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T20:38:35" Sum="5000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T20:39:18" Sum="100000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T21:13:28" Sum="100000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-16T21:19:53" Sum="100000" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:40:37" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:40:38" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:41:46" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:41:47" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:50:54" Sum="999" Origin="Конкорд-ПА - Банкоматы"/>*  *<P Time="2012-02-17T11:50:55" Sum="-999" Origin="Конкорд-ПА - Банкоматы"/>*  *</paymentList>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getPaymentListResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Получение информации о меню столовой»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getMenuList |
| **Наименование операции:** | Получение информации о меню столовой |
| **Назначение операции:** | Получение информации о меню столовой: дата, наименование, цена |

Описание входных параметров

**Входные данные: getMenuList**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | startDate | Начальная дата выборки | + | xs:dateTime |  |
| 3 | endDate | Конечная дата выборки | + | xs:dateTime |  |

Описание выходных параметров

**Выходные данные: menuListResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | menuList | Меню | + | MenuListExt |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getMenuList>*  *<contractId>7607195</contractId>*  *<startDate>2012-05-17T00:00:00Z</startDate>*  *<endDate>2012-05-18T00:00:00Z</endDate>*  *</soap:getMenuList>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getMenuListResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<menuList>*  *<M Date="2012-05-17">*  *<E Group="Бутерброды" Name="Бутерброд горячий (колбаса, сыр)" Price="2300"/>*  *<E Group="Бутерброды" Name="Бутерброд горячий с сос." Price="2000"/>*  *<E Group="Бутерброды" Name="Бутерброд с колб." Price="1400"/>*  *<E Group="Бутерброды" Name="Бутерброд с колбас. с/к" Price="2000"/>*  *<E Group="Бутерброды" Name="Бутерброд с семгой сл.сол" Price="3200"/>*  *<E Group="Бутерброды" Name="Бутерброд с сыром" Price="1200"/>*  *<E Group="Бутерброды" Name="Сандвич на ржан.хлебе" Price="1100"/>*  *<E Group="Выпечка" Name="Булочка" Price="1000"/>*  *<E Group="Выпечка" Name="Ватрушка с абр. джемом" Price="1000"/>*  *<E Group="Выпечка" Name="Ватрушка с творогом" Price="1000"/>*  *<E Group="Выпечка" Name="Кекс весенний 500гр" Price="9000"/>*  *<E Group="Выпечка" Name="Кекс весенний 50гр" Price="1200"/>*  *<E Group="Выпечка" Name="Кекс столичный" Price="2000"/>*  *<E Group="Выпечка" Name="Коржик молочный" Price="1200"/>*  *<E Group="Выпечка" Name="Песочник с изюмом" Price="1400"/>*  *<E Group="Выпечка" Name="Печенье ракушка" Price="2000"/>*  *<E Group="Выпечка" Name="Пир. возд. с кремом" Price="1400"/>*  *<E Group="Выпечка" Name="Пир. корз. с белк.крем." Price="1400"/>*  *<E Group="Выпечка" Name="Пир. песоч.глазир." Price="1200"/>*  *<E Group="Выпечка" Name="Пир. труб. йогурт крем." Price="1200"/>*  *<E Group="Выпечка" Name="Пир.воз-орех" Price="1600"/>*  *<E Group="Выпечка" Name="Пир.песоч.ореховое" Price="1600"/>*  *<E Group="Выпечка" Name="Пир.слоен.с мясом" Price="3500"/>*  *<E Group="Выпечка" Name="Пир.труб. завар. с б/к" Price="1400"/>*  *<E Group="Выпечка" Name="Пир.труб. слоен. с б/к" Price="1400"/>*  *<E Group="Выпечка" Name="Пир.труб.зав.со слив.крем" Price="1400"/>*  *<E Group="Выпечка" Name="Пирог Домашний" Price="2000"/>*  *<E Group="Выпечка" Name="Пирожное картошка" Price="1600"/>*  *<E Group="Выпечка" Name="Пирожное ореховое" Price="1600"/>*  *<E Group="Выпечка" Name="Пирожное орешек" Price="1600"/>*  *<E Group="Выпечка" Name="Пирожок с вишн. джемом" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с грибами и карт." Price="2000"/>*  *<E Group="Выпечка" Name="Пирожок с капус. и яйцом" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с карт. и луком" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с клубн. джемом" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с курагой" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с мясом" Price="2000"/>*  *<E Group="Выпечка" Name="Пирожок с печенью и рисом" Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с рисом и яйц." Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с рыбой и карт." Price="1000"/>*  *<E Group="Выпечка" Name="Пирожок с ябл. джемом" Price="1000"/>*  *<E Group="Выпечка" Name="Пол. пес. с повидлом" Price="1200"/>*  *<E Group="Выпечка" Name="Рожок слоеный с повидлом" Price="1600"/>*  *<E Group="Выпечка" Name="Ром-баба с шок. помад." Price="1200"/>*  *<E Group="Выпечка" Name="Рулет фруктовый" Price="1500"/>*  *<E Group="Выпечка" Name="Сосиска в тесте" Price="2300"/>*  *<E Group="Выпечка" Name="Шарлотка с ябл." Price="1000"/>*  *<E Group="Выпечка" Name="Язык слоеный" Price="1400"/>*  *<E Group="Гарниры" Name="Гречка отварная" Price="1200"/>*  *<E Group="Гарниры" Name="Капуста тушен." Price="1100"/>*  *<E Group="Гарниры" Name="Макаронные изд." Price="700"/>*  *<E Group="Гарниры" Name="Пюре картофель" Price="1600"/>*  *<E Group="Гарниры" Name="Рагу из овощей" Price="1200"/>*  *<E Group="Гарниры" Name="Рис отварной" Price="800"/>*  *<E Group="Гор. блюда" Name="Бефстроганов" Price="4900"/>*  *<E Group="Гор. блюда" Name="Гуляш" Price="4900"/>*  *<E Group="Гор. блюда" Name="Запек. карт с мясом" Price="5000"/>*  *<E Group="Гор. блюда" Name="Запек. карт.рыбн." Price="4100"/>*  *<E Group="Гор. блюда" Name="Зразы карт.с морк.и лук" Price="1300"/>*  *<E Group="Гор. блюда" Name="Зразы курин.рис/лук" Price="2700"/>*  *<E Group="Гор. блюда" Name="Зразы мясные" Price="3500"/>*  *<E Group="Гор. блюда" Name="Зразы рыбн.с карт." Price="2800"/>*  *<E Group="Гор. блюда" Name="Картофель жареный" Price="2000"/>*  *<E Group="Гор. блюда" Name="Котл. карт с мясом" Price="7700"/>*  *<E Group="Гор. блюда" Name="Котлета из птицы" Price="1500"/>*  *<E Group="Гор. блюда" Name="Котлета капустная" Price="1000"/>*  *<E Group="Гор. блюда" Name="Котлета картофельная" Price="1000"/>*  *<E Group="Гор. блюда" Name="Котлета морковная" Price="1000"/>*  *<E Group="Гор. блюда" Name="Котлета мясная руб." Price="2000"/>*  *<E Group="Гор. блюда" Name="Котлета свекольная" Price="1000"/>*  *<E Group="Гор. блюда" Name="Котлеты кур. с морк." Price="3300"/>*  *<E Group="Гор. блюда" Name="Котлеты куриные" Price="3200"/>*  *<E Group="Гор. блюда" Name="Котлеты рубл.гов." Price="2800"/>*  *<E Group="Гор. блюда" Name="Котлеты рубл.кур." Price="3300"/>*  *<E Group="Гор. блюда" Name="Котлеты рыбные" Price="2600"/>*  *<E Group="Гор. блюда" Name="Курица отварная" Price="2500"/>*  *<E Group="Гор. блюда" Name="Медальон из свинины" Price="7200"/>*  *<E Group="Гор. блюда" Name="Медальон из филе птицы" Price="4800"/>*  *<E Group="Гор. блюда" Name="Мясо отварное гов." Price="2600"/>*  *<E Group="Гор. блюда" Name="Печень тушон. гов." Price="2900"/>*  *<E Group="Гор. блюда" Name="Плов курин." Price="3900"/>*  *<E Group="Гор. блюда" Name="Рагу куринное" Price="2700"/>*  *<E Group="Гор. блюда" Name="Рулет карт.с мясом" Price="4000"/>*  *<E Group="Гор. блюда" Name="Рулет мясной с яйцом" Price="4000"/>*  *<E Group="Гор. блюда" Name="Рыба в кляре" Price="3000"/>*  *<E Group="Гор. блюда" Name="Рыба тушен. к карт." Price="4300"/>*  *<E Group="Гор. блюда" Name="Свин. туш. с картоф." Price="5000"/>*  *<E Group="Гор. блюда" Name="Суфле из печени" Price="7700"/>*  *<E Group="Гор. блюда" Name="Тефтели гов." Price="3100"/>*  *<E Group="Гор. блюда" Name="Тефтели рыбные" Price="2400"/>*  *<E Group="Гор. блюда" Name="Тефтели с рис." Price="2700"/>*  *<E Group="Гор. блюда" Name="Филе птицы в кляре" Price="3800"/>*  *<E Group="Гор. блюда" Name="Эскалоп из свинины" Price="7800"/>*  *<E Group="Завтраки" Name="Вареники лен. с твор." Price="2700"/>*  *<E Group="Завтраки" Name="Греча отварная" Price="800"/>*  *<E Group="Завтраки" Name="Запеканка из творога" Price="2700"/>*  *<E Group="Завтраки" Name="Запеканка макар." Price="1500"/>*  *<E Group="Завтраки" Name="Запеканка твор. с изюмом" Price="2600"/>*  *<E Group="Завтраки" Name="Котлеты рубл.гов." Price="1400"/>*  *<E Group="Завтраки" Name="Крупеник рис/творог" Price="2600"/>*  *<E Group="Завтраки" Name="Курица отварная" Price="1600"/>*  *<E Group="Завтраки" Name="Омлет" Price="8800"/>*  *<E Group="Завтраки" Name="Плов курин." Price="2900"/>*  *<E Group="Завтраки" Name="Пюре картофель." Price="1100"/>*  *<E Group="Завтраки" Name="Рагу из овощей" Price="800"/>*  *<E Group="Завтраки" Name="Рагу куринное" Price="2000"/>*  *<E Group="Завтраки" Name="Рис отварной" Price="500"/>*  *<E Group="Завтраки" Name="Сырники из твор." Price="2700"/>*  *<E Group="Завтраки" Name="Тефтели гов." Price="2100"/>*  *<E Group="Завтраки" Name="Тефтели рыбные" Price="1600"/>*  *<E Group="Завтраки" Name="Тефтели с рис." Price="1400"/>*  *<E Group="Завтраки" Name="Хлеб" Price="200"/>*  *<E Group="Каши" Name="Каша гречневая" Price="1100"/>*  *<E Group="Каши" Name="Каша манная" Price="600"/>*  *<E Group="Каши" Name="Каша овсяная" Price="1000"/>*  *<E Group="Каши" Name="Каша пшен. с изум." Price="1300"/>*  *<E Group="Каши" Name="Каша пшенная" Price="1300"/>*  *<E Group="Каши" Name="Каша рис с изум." Price="1300"/>*  *<E Group="Кондитер" Name="Батончик &quot;Кит-кат&quot;" Price="2700"/>*  *<E Group="Кондитер" Name="Батончик &quot;Милки Вей&quot;" Price="2000"/>*  *<E Group="Кондитер" Name="Батончик &quot;Несквик&quot;" Price="1700"/>*  *<E Group="Кондитер" Name="Батончик Гематог." Price="700"/>*  *<E Group="Кондитер" Name="Батончик кокосовый &quot;Гемо-хелпер&quot;" Price="1400"/>*  *<E Group="Кондитер" Name="Батончик Натс" Price="2800"/>*  *<E Group="Кондитер" Name="Батончик Пульсар" Price="1800"/>*  *<E Group="Кондитер" Name="Батончик СИЛЬВЕР" Price="2000"/>*  *<E Group="Кондитер" Name="Батончик Темпо" Price="1200"/>*  *<E Group="Кондитер" Name="Вафли &quot;Лимонные&quot;" Price="900"/>*  *<E Group="Кондитер" Name="Вафли без начинки" Price="1900"/>*  *<E Group="Кондитер" Name="Вафли глазированные" Price="2800"/>*  *<E Group="Кондитер" Name="Вафли Коломенск." Price="500"/>*  *<E Group="Кондитер" Name="Джем (ассортимент)" Price="500"/>*  *<E Group="Кондитер" Name="Зефир &quot;Ванильный&quot;" Price="600"/>*  *<E Group="Кондитер" Name="Конфета Курьёз" Price="900"/>*  *<E Group="Кондитер" Name="Конфета Сласть" Price="1200"/>*  *<E Group="Кондитер" Name="Конфета Фарс" Price="1500"/>*  *<E Group="Кондитер" Name="Конфета Экер" Price="800"/>*  *<E Group="Кондитер" Name="Крекер Соломка" Price="1000"/>*  *<E Group="Кондитер" Name="Мармелад Джу-Джав" Price="1000"/>*  *<E Group="Кондитер" Name="Мармелад Зол.Мишка" Price="800"/>*  *<E Group="Кондитер" Name="Мармелад Медвеж." Price="2300"/>*  *<E Group="Кондитер" Name="Мармелад Фрутелла" Price="2500"/>*  *<E Group="Кондитер" Name="Мёд" Price="700"/>*  *<E Group="Кондитер" Name="Мини-кекс" Price="900"/>*  *<E Group="Кондитер" Name="Ореховый батончик &quot;Пульсар&quot;" Price="1800"/>*  *<E Group="Кондитер" Name="Печенье &quot;Мишка-топтышка&quot;" Price="1000"/>*  *<E Group="Кондитер" Name="Печенье Вагон Виллз" Price="1600"/>*  *<E Group="Кондитер" Name="Печенье Нежное" Price="500"/>*  *<E Group="Кондитер" Name="Пироженное биск.шок.начинка &quot;Барни&quot;" Price="2400"/>*  *<E Group="Кондитер" Name="Пироженое Бонди" Price="1200"/>*  *<E Group="Кондитер" Name="Пироженое Чокопай" Price="900"/>*  *<E Group="Кондитер" Name="Снэки Кук. палоч. Кузя" Price="1500"/>*  *<E Group="Кондитер" Name="Успевай-ка с кальцием и жел." Price="1300"/>*  *<E Group="Кондитер" Name="Хрустящие тр. Читос" Price="2400"/>*  *<E Group="Кондитер" Name="Шок. Киндер CiO" Price="3600"/>*  *<E Group="Кондитер" Name="Шок. Киндер МАХI" Price="1600"/>*  *<E Group="Кондитер" Name="Шоколад &quot;Алёнка&quot;" Price="1200"/>*  *<E Group="Молочная прод." Name="Биойогурт питьевой" Price="3100"/>*  *<E Group="Молочная прод." Name="Брикет &quot;Филёвский&quot;" Price="1900"/>*  *<E Group="Молочная прод." Name="Ваф.стак. &quot;Фил.&quot;" Price="1900"/>*  *<E Group="Молочная прод." Name="Йогурт" Price="1200"/>*  *<E Group="Молочная прод." Name="Кокт. Мол. кар. банан" Price="2000"/>*  *<E Group="Молочная прод." Name="Кокт. Мол. клубника" Price="2000"/>*  *<E Group="Молочная прод." Name="Кокт. Мол. шоколад" Price="2200"/>*  *<E Group="Молочная прод." Name="Кокт. Мол.Ваниль" Price="2000"/>*  *<E Group="Молочная прод." Name="Коктейль молоч. &quot;Здрайверы&quot; в асс." Price="2400"/>*  *<E Group="Молочная прод." Name="Молоко &quot;Большая перемена&quot; 3,2%" Price="1900"/>*  *<E Group="Молочная прод." Name="Молоко &quot;Здрайверы&quot; 3,2%" Price="1700"/>*  *<E Group="Молочная прод." Name="Мор. &quot;Дед Мороз&quot;Эскимо" Price="1300"/>*  *<E Group="Молочная прод." Name="Морож. &quot; Бородино&quot;" Price="2400"/>*  *<E Group="Молочная прод." Name="Морож. &quot;Вол.фонарь&quot;" Price="2600"/>*  *<E Group="Молочная прод." Name="Морож. &quot;Елочка&quot;" Price="2300"/>*  *<E Group="Молочная прод." Name="Морож. &quot;Ленин&quot; батон" Price="2300"/>*  *<E Group="Молочная прод." Name="Морож. &quot;Ленин.&quot; эскимо" Price="1900"/>*  *<E Group="Молочная прод." Name="Рожок &quot;Фил.&quot; пломбир" Price="2000"/>*  *<E Group="Напитки уп." Name="Вода &quot;Бонаква&quot;" Price="2700"/>*  *<E Group="Напитки уп." Name="Вода &quot;Святой Источник&quot;" Price="1900"/>*  *<E Group="Напитки уп." Name="Вода Дана" Price="1600"/>*  *<E Group="Напитки уп." Name="Вода Демидовск." Price="1500"/>*  *<E Group="Напитки уп." Name="Голд" Price="1400"/>*  *<E Group="Напитки уп." Name="Какао с молоком" Price="1100"/>*  *<E Group="Напитки уп." Name="Кофе" Price="1000"/>*  *<E Group="Напитки уп." Name="Лимон (долька)" Price="100"/>*  *<E Group="Напитки уп." Name="Морс &quot;ФрутоНяня&quot; клюква/малина" Price="2900"/>*  *<E Group="Напитки уп." Name="Морс Здрайверы" Price="2000"/>*  *<E Group="Напитки уп." Name="Напиток &quot;Фрустайл&quot; в асс." Price="3500"/>*  *<E Group="Напитки уп." Name="Напиток из св.фруктов" Price="1000"/>*  *<E Group="Напитки уп." Name="Напиток из суш/фр" Price="1200"/>*  *<E Group="Напитки уп." Name="Напиток цитрусовый" Price="700"/>*  *<E Group="Напитки уп." Name="Напиток Шиповник" Price="1100"/>*  *<E Group="Напитки уп." Name="Нектар &quot;Любимый&quot; сад" Price="1100"/>*  *<E Group="Напитки уп." Name="Нектар J7" Price="1700"/>*  *<E Group="Напитки уп." Name="Нектар Туса-ждуса" Price="2200"/>*  *<E Group="Напитки уп." Name="Сахар" Price="100"/>*  *<E Group="Напитки уп." Name="Сок &quot;Здрайверы&quot; мультифр." Price="1400"/>*  *<E Group="Напитки уп." Name="Сок &quot;Малышам&quot;" Price="700"/>*  *<E Group="Напитки уп." Name="Сок в асс." Price="1300"/>*  *<E Group="Напитки уп." Name="Сок Каприз" Price="2400"/>*  *<E Group="Напитки уп." Name="Фрукт. десерт &quot;ФрутоНяня&quot;" Price="1800"/>*  *<E Group="Напитки уп." Name="Хол. чай Нести" Price="4500"/>*  *<E Group="Напитки уп." Name="Чай" Price="300"/>*  *<E Group="Салаты" Name="Морковь по корейски" Price="1500"/>*  *<E Group="Салаты" Name="Рыба под маринадом" Price="2400"/>*  *<E Group="Салаты" Name="Салат винегрет" Price="1300"/>*  *<E Group="Салаты" Name="Салат витаминный" Price="1000"/>*  *<E Group="Салаты" Name="Салат из белок.капусты" Price="1200"/>*  *<E Group="Салаты" Name="Салат из капус.с овощ." Price="1500"/>*  *<E Group="Салаты" Name="Салат из морк.с сахаром" Price="1000"/>*  *<E Group="Салаты" Name="Салат из огурц. и пом." Price="2000"/>*  *<E Group="Салаты" Name="Салат из помид с рас. маслом" Price="2300"/>*  *<E Group="Салаты" Name="Салат из свеж.огур." Price="2000"/>*  *<E Group="Салаты" Name="Салат из свекл. яблок" Price="1300"/>*  *<E Group="Салаты" Name="Салат из свеклы и сыра" Price="1500"/>*  *<E Group="Салаты" Name="Салат из свеклы с зел.гор" Price="1000"/>*  *<E Group="Салаты" Name="Салат крабовый" Price="1700"/>*  *<E Group="Салаты" Name="Салат Мимоза" Price="2200"/>*  *<E Group="Салаты" Name="Салат мясной" Price="3500"/>*  *<E Group="Салаты" Name="Салат рыбный" Price="2200"/>*  *<E Group="Салаты" Name="Салат столичный" Price="2700"/>*  *<E Group="Салаты" Name="Салат Цезарь" Price="3400"/>*  *<E Group="Соб. Произв." Name="Блинчики 2 шт." Price="1000"/>*  *<E Group="Соб. Произв." Name="Блинчики с джемом" Price="1200"/>*  *<E Group="Соб. Произв." Name="Блинчики с капустой" Price="1000"/>*  *<E Group="Соб. Произв." Name="Блинчики с мас/слив" Price="800"/>*  *<E Group="Соб. Произв." Name="Блинчики с медом" Price="1500"/>*  *<E Group="Соб. Произв." Name="Блинчики с мясом" Price="2800"/>*  *<E Group="Соб. Произв." Name="Блинчики с сыром 2шт." Price="3100"/>*  *<E Group="Соб. Произв." Name="Блинчики смет. и сах. 2шт." Price="1600"/>*  *<E Group="Соб. Произв." Name="Блинчики со смет. 2шт." Price="1500"/>*  *<E Group="Соб. Произв." Name="Ватрушка с повидлом" Price="800"/>*  *<E Group="Соб. Произв." Name="Колбаска сос. в тесте" Price="2200"/>*  *<E Group="Соб. Произв." Name="Оладьи" Price="1000"/>*  *<E Group="Соб. Произв." Name="Пакет" Price="20"/>*  *<E Group="Соб. Произв." Name="Пельмени отварные" Price="2100"/>*  *<E Group="Соб. Произв." Name="Пирожок с кап. и яйцом" Price="1000"/>*  *<E Group="Соб. Произв." Name="Пирожок с капустой" Price="1000"/>*  *<E Group="Соб. Произв." Name="Пирожок с карт.и лук." Price="1000"/>*  *<E Group="Соб. Произв." Name="Пирожок с мясом" Price="2200"/>*  *<E Group="Соб. Произв." Name="Пирожок с яблоками" Price="1000"/>*  *<E Group="Соб. Произв." Name="Пицца &quot;Школьная&quot;" Price="2700"/>*  *<E Group="Соб. Произв." Name="Пицца с сосисками" Price="2700"/>*  *<E Group="Соб. Произв." Name="Сарделька отварная" Price="1600"/>*  *<E Group="Соб. Произв." Name="Сосиска отварная" Price="1600"/>*  *<E Group="Соб. Произв." Name="Яблоко печ.с сахаром" Price="2200"/>*  *<E Group="Соус" Name="Маринад овощной" Price="1000"/>*  *<E Group="Соус" Name="Соус мол.сладкий" Price="300"/>*  *<E Group="Соус" Name="Соус молочный" Price="300"/>*  *<E Group="Соус" Name="Соус молочный" Price="300"/>*  *<E Group="Соус" Name="Соус сливочный" Price="300"/>*  *<E Group="Соус" Name="Соус сметанный" Price="300"/>*  *<E Group="Супы" Name="Борщ с капус. и карт." Price="1100"/>*  *<E Group="Супы" Name="Бульон куринный" Price="1500"/>*  *<E Group="Супы" Name="Рассольник ленинград." Price="1100"/>*  *<E Group="Супы" Name="Суп-пюре картоф." Price="1500"/>*  *<E Group="Супы" Name="Суп из овощей" Price="1300"/>*  *<E Group="Супы" Name="Суп из рыбы" Price="2600"/>*  *<E Group="Супы" Name="Суп картоф. с рыбой" Price="2500"/>*  *<E Group="Супы" Name="Суп крестьян. с круп." Price="1000"/>*  *<E Group="Супы" Name="Суп куринный" Price="1500"/>*  *<E Group="Супы" Name="Суп с макар. издел." Price="900"/>*  *<E Group="Супы" Name="Щи из свежей капусты" Price="1300"/>*  *<E Group="Фрукты" Name="Апельсин" Price="2000"/>*  *<E Group="Фрукты" Name="Груша" Price="2000"/>*  *<E Group="Фрукты" Name="Мандарин" Price="1200"/>*  *<E Group="Фрукты" Name="Яблоко зеленое" Price="2000"/>*  *<E Group="Фрукты" Name="Яблоко печ.с сах." Price="1500"/>*  *</M>*  *</menuList>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getMenuListResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Получение данных посещениях образовательного учреждения»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | getEnterEventList |
| **Наименование операции:** | Получение данных посещениях образовательного учреждения |
| **Назначение операции:** | Получение информации о посещении образовательного учреждения: дата, направление входа |

Описание входных параметров

**Входные данные: getEnterEventList**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | startDate | Начальная дата выборки | + | xs:dateTime |  |
| 3 | endDate | Конечная дата выборки | + | xs:dateTime |  |

Описание выходных параметров

**Выходные данные: enterEventListResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | enterEventList | События посещения | + | EnterEventList |  |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Параметры комплексного типа описаны в приложении «Описание общих структур данных».

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:getEnterEventList>*  *<contractId>07613912</contractId>*  *<startDate>2012-05-01T00:00:00Z</startDate>*  *<endDate>2012-05-20T00:00:00Z</endDate>*  *</soap:getEnterEventList>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:getEnterEventListResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<enterEventList>*  *<E DateTime="2012-05-18T12:25:02" Day="5" EnterName="Вход №1" Direction="0" TemporaryCard="1"/>*  *<E DateTime="2012-05-18T12:26:28" Day="5" EnterName="Вход №1" Direction="1" TemporaryCard="1"/>*  *<E DateTime="2012-05-18T12:38:35" Day="5" EnterName="Вход №1" Direction="7" TemporaryCard="1"/>*  *<E DateTime="2012-05-18T12:44:29" Day="5" EnterName="Вход №1" Direction="7" TemporaryCard="1"/>*  *</enterEventList>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:getEnterEventListResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Включить/отключить СМС-информирование»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | enableNotificationBySMS |
| **Наименование операции:** | Включить/отключить СМС-информирование |
| **Назначение операции:** | Изменение настроек СМС-информирования лицевого счета |

Описание входных параметров

**Входные данные: enableNotificationBySMS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | state | Статус | + | xs:boolean |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:enableNotificationBySMS>*  *<contractId>200485</contractId>*  *<state>false</state>*  *</soap:enableNotificationBySMS>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:enableNotificationBySMSResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:enableNotificationBySMSResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Изменить номер мобильного телефона для СМС-информирования»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | changeMobilePhone |
| **Наименование операции:** | Изменить номер мобильного телефона для СМС-информирования |
| **Назначение операции:** | Изменение настроек СМС-информирования (номер телефона) |

Описание входных параметров

**Входные данные: changeMobilePhone**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | mobilePhone | Номер мобильного телефона | + | xs:string |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |
| 4 | 150 | Неверные данные | Неправильный формат номера |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:changeMobilePhone>*  *<contractId>200485</contractId>*  *<mobilePhone>79161111111</mobilePhone>*  *</soap: changeMobilePhone>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:changeMobilePhoneResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:changeMobilePhoneResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Включить/отключить E-Mail-информирование»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | enableNotificationByEmail |
| **Наименование операции:** | Включить/отключить E-Mail-информирование |
| **Назначение операции:** | Изменение настроек Email-информирования лицевого счета |

Описание входных параметров

**Входные данные: enableNotificationByEmail**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | state | Статус | + | xs:boolean |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:enableNotificationByEmail>*  *<contractId>200485</contractId>*  *<state>false</state>*  *</soap:enableNotificationByEmail>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:enableNotificationByEmailResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:enableNotificationByEmailResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Изменить адрес e-mail для информирования»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | changeEmail |
| **Наименование операции:** | Изменить адрес e-mail для информирования |
| **Назначение операции:** | Изменение настроек Email-информирования (адрес) |

Описание входных параметров

**Входные данные:** changeEmail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | email | Адрес e-mail | + | xs:string |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | resultCode | Код возврата | + | xs:long |  |
| 2 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |
| 4 | 150 | Неверные данные | Неправильный формат адреса |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:changeEmail>*  *<contractId>200485</contractId>*  *<email>test@mail.ru</email>*  *</soap:changeEmail>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:changeEmailResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:changeEmailResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Применить код активации»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | activateLinkingToken |
| **Наименование операции:** | Применить код активации |
| **Назначение операции:** | Применить код активации для связки с лицевым счетом |

Описание входных параметров

**Входные данные: activateLinkingToken**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | linkingToken | Код активации | + | xs:string |  |

Описание выходных параметров

**Выходные данные: activateLinkingTokenResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета |  | xs:long | Заполняется только в случае если код активации найден |
| 2 | resultCode | Код возврата | + | xs:long |  |
| 3 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 150 | Код активации не найден | Код активации не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:activateLinkingToken>*  *<linkingToken>f9bcslbc5</linkingToken>*  *</soap:activateLinkingToken>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:activateLinkingTokenResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *<contractId>200485</contractId>*  *</return>*  *</ns2:activateLinkingTokenResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Сформировать код активации»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | generateLinkingToken |
| **Наименование операции:** | Сформировать код активации |
| **Назначение операции:** | Сформировать новый код активации для лицевого счета |

Описание входных параметров

**Входные данные: activateLinkingToken**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |

Описание выходных параметров

**Выходные данные: generateLinkingTokenResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | linkingToken | Код активации | + | xs:string |  |
| 3 | resultCode | Код возврата | + | xs:long |  |
| 4 | description | Описание результата обработки | + | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:generateLinkingToken>*  *<contractId>200485</contractId>*  *</soap:generateLinkingToken>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:generateLinkingTokenResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *<contractId>200485</contractId>*  *<linkingToken>f9bcslbc5</linkingToken>*  *</return>*  *</ns2:generateLinkingTokenResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Сформировать код активации и отправить по номеру лицевого счета»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | sendLinkingTokenByContractId |
| **Наименование операции:** | Сформировать код активации и отправить по номеру лицевого счета |
| **Назначение операции:** | Сформировать код активации и отправить по e-mail и SMS для указанного лицевого счета |

Описание входных параметров

**Входные данные: sendLinkingTokenByContractId**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | resultCode | Код возврата | + | xs:long |  |
| 2 | description | Описание результата обработки | + | xs:string | В случае успешной обработки заполняется по каким каналам быт отправлен код |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 160 | У лицевого счета нет контактных данных | У лицевого счета нет контактных данных |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:sendLinkingTokenByContractId>*  *<contractId>200485</contractId>*  *</soap:sendLinkingTokenByContractId>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:sendLinkingTokenByContractIdResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>Код активации отправлен по SMS</description>*  *</return>*  *</ns2:sendLinkingTokenByContractIdResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Операция «Сформировать код активации и отправить по e-mail и SMS»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | sendLinkingTokenByMobile |
| **Наименование операции:** | Сформировать код активации и отправить по номеру телефона |
| **Назначение операции:** | Сформировать код активации и отправить по e-mail и SMS для указанного номера телефона |

Описание входных параметров

**Входные данные: sendLinkingTokenByMobile**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | mobilePhone | Номер телефона | + | xs:string |  |

Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | resultCode | Код возврата | + | xs:long |  |
| 2 | description | Описание результата обработки | + | xs:string | В случае успешной обработки заполняется сколько л/с привязаны к данному номеру |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Клиент не найден | Нет лицевых счетов с данным контактным номером мобильного телефона |  |
| 4 | 150 | Неверный формат | Неверный формат номера мобильного телефона |  |

Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:sendLinkingTokenByMobile>*  *<mobilePhone>79161111111</mobilePhone>*  *</soap:sendLinkingTokenByMobile>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:sendLinkingTokenByMobileResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>Код активации отправлен по SMS для 2 л/с</description>*  *</return>*  *</ns2:sendLinkingTokenByMobileResponse>*  *</soap:Body>*  *</soap:Envelope>* |

## Изменить размер дневного ограничения на затраты по номеру контракта ребенка

### Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | changeExpenditureLimit |
| **Наименование операции:** | Для изменить размер дневного ограничения на затраты по номеру контракта |
| **Назначение операции:** | Установить размер дневного ограничения на затраты по номеру контракта |

### Описание входных параметров

**Входные данные:** changeExpenditureLimit

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | contractId | Номер лицевого счета | + | xs:long |  |
| 2 | limit | Размер дневного ограничения | + | xs:long | Размер дневного ограничения указывается в копейках и должен быть больше 0 |

### Описание выходных параметров

**Выходные данные: result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | resultCode | Код возврата | + | xs:long |  |
| 2 | description | Описание результата обработки | + | xs:string |  |

### Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Успешная обработка |  |
| 2 | 100 | Внутренняя ошибка | Внутренняя ошибка сервера при обработке |  |
| 3 | 110 | Лицевой счет не найден | Лицевой счет не найден |  |
| 4 | 150 | Неверные данные | Лимит не может быть меньше нуля |  |

### Контрольные примеры

**Запрос**

|  |
| --- |
| *<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soap="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<soapenv:Header/>*  *<soapenv:Body>*  *<soap:changeExpenditureLimit>*  *<contractId>200485</contractId>*  *<limit>5000</limit>*  *</soap:changeExpenditureLimit>*  *</soapenv:Body>*  *</soapenv:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:changeExpenditureLimitResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<resultCode>0</resultCode>*  *<description>OK</description>*  *</return>*  *</ns2:changeExpenditureLimitResponse>*  *</soap:Body>*  *</soap:Envelope>* |

1. Приложения

Описание сервиса (WSDL)

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <wsdl:definitions name="ClientRoomControllerWSService" targetNamespace="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">  <wsdl:types>  <xs:schema elementFormDefault="unqualified" targetNamespace="http://soap.integra.partner.web.processor.ecafe.axetta.ru/" version="1.0">  <xs:element name="Data">  <xs:complexType>  <xs:sequence>  <xs:element minOccurs="0" name="ClientSummary"type="tns:ClientSummary"/>  <xs:element minOccurs="0" name="ClientSummaryExt" type="tns:ClientSummaryExt"/>  <xs:element minOccurs="0" name="PurchaseList" type="tns:PurchaseList"/>  <xs:element minOccurs="0" name="PurchaseListExt" type="tns:PurchaseListExt"/>  <xs:element minOccurs="0" name="PaymentList" type="tns:PaymentList"/>  <xs:element minOccurs="0" name="MenuList" type="tns:MenuList"/>  <xs:element minOccurs="0" name="MenuListExt" type="tns:MenuListExt"/>  <xs:element minOccurs="0" name="CardList" type="tns:CardList"/>  <xs:element minOccurs="0" name="EnterEventList" type="tns:EnterEventList"/>  <xs:element minOccurs="0" name="CirculationItemList" type="tns:CirculationItemList"/>  </xs:sequence>  <xs:attribute name="IdOfContract" type="xs:long"/>  <xs:attribute name="ResultCode" type="xs:long"/>  <xs:attribute name="Description" type="xs:string"/>  </xs:complexType>  </xs:element>  <xs:element name="activateLinkingToken" type="tns:activateLinkingToken"/>  <xs:element name="activateLinkingTokenResponse" type="tns:activateLinkingTokenResponse"/>  <xs:element name="attachGuardSan" type="tns:attachGuardSan"/>  <xs:element name="attachGuardSanBySan" type="tns:attachGuardSanBySan"/>  <xs:element name="attachGuardSanBySanResponse" type="tns:attachGuardSanBySanResponse"/>  <xs:element name="attachGuardSanResponse" type="tns:attachGuardSanResponse"/>  <xs:element name="authorizeClient" type="tns:authorizeClient"/>  <xs:element name="authorizeClientResponse" type="tns:authorizeClientResponse"/>  <xs:element name="changeEmail" type="tns:changeEmail"/>  <xs:element name="changeEmailResponse" type="tns:changeEmailResponse"/>  <xs:element name="changeExpenditureLimit" type="tns:changeExpenditureLimit"/>  <xs:element name="changeExpenditureLimitResponse" type="tns:changeExpenditureLimitResponse"/>  <xs:element name="changeMobilePhone" type="tns:changeMobilePhone"/>  <xs:element name="changeMobilePhoneResponse" type="tns:changeMobilePhoneResponse"/>  <xs:element name="detachGuardSan" type="tns:detachGuardSan"/>  <xs:element name="detachGuardSanBySan" type="tns:detachGuardSanBySan"/>  <xs:element name="detachGuardSanBySanResponse" type="tns:detachGuardSanBySanResponse"/>  <xs:element name="detachGuardSanResponse" type="tns:detachGuardSanResponse"/>  <xs:element name="enableNotificationByEmail" type="tns:enableNotificationByEmail"/>  <xs:element name="enableNotificationByEmailResponse" type="tns:enableNotificationByEmailResponse"/>  <xs:element name="enableNotificationBySMS" type="tns:enableNotificationBySMS"/>  <xs:element name="enableNotificationBySMSResponse" type="tns:enableNotificationBySMSResponse"/>  <xs:element name="generateLinkingToken" type="tns:generateLinkingToken"/>  <xs:element name="generateLinkingTokenResponse" type="tns:generateLinkingTokenResponse"/>  <xs:element name="getCardList" type="tns:getCardList"/>  <xs:element name="getCardListBySan" type="tns:getCardListBySan"/>  <xs:element name="getCardListBySanResponse" type="tns:getCardListBySanResponse"/>  <xs:element name="getCardListResponse" type="tns:getCardListResponse"/>  <xs:element name="getCirculationList" type="tns:getCirculationList"/>  <xs:element name="getCirculationListResponse" type="tns:getCirculationListResponse"/>  <xs:element name="getClientsByGuardSan" type="tns:getClientsByGuardSan"/>  <xs:element name="getClientsByGuardSanResponse" type="tns:getClientsByGuardSanResponse"/>  <xs:element name="getContractIdByCardNo" type="tns:getContractIdByCardNo"/>  <xs:element name="getContractIdByCardNoResponse" type="tns:getContractIdByCardNoResponse"/>  <xs:element name="getEnterEventList" type="tns:getEnterEventList"/>  <xs:element name="getEnterEventListBySan" type="tns:getEnterEventListBySan"/>  <xs:element name="getEnterEventListBySanResponse" type="tns:getEnterEventListBySanResponse"/>  <xs:element name="getEnterEventListResponse" type="tns:getEnterEventListResponse"/>  <xs:element name="getGroupListByOrg" type="tns:getGroupListByOrg"/>  <xs:element name="getGroupListByOrgResponse" type="tns:getGroupListByOrgResponse"/>  <xs:element name="getMenuList" type="tns:getMenuList"/>  <xs:element name="getMenuListByOrg" type="tns:getMenuListByOrg"/>  <xs:element name="getMenuListByOrgResponse" type="tns:getMenuListByOrgResponse"/>  <xs:element name="getMenuListBySan" type="tns:getMenuListBySan"/>  <xs:element name="getMenuListBySanResponse" type="tns:getMenuListBySanResponse"/>  <xs:element name="getMenuListResponse" type="tns:getMenuListResponse"/>  <xs:element name="getPaymentList" type="tns:getPaymentList"/>  <xs:element name="getPaymentListBySan" type="tns:getPaymentListBySan"/>  <xs:element name="getPaymentListBySanResponse" type="tns:getPaymentListBySanResponse"/>  <xs:element name="getPaymentListResponse" type="tns:getPaymentListResponse"/>  <xs:element name="getPurchaseList" type="tns:getPurchaseList"/>  <xs:element name="getPurchaseListBySan" type="tns:getPurchaseListBySan"/>  <xs:element name="getPurchaseListBySanResponse" type="tns:getPurchaseListBySanResponse"/>  <xs:element name="getPurchaseListResponse" type="tns:getPurchaseListResponse"/>  <xs:element name="getStudentListByIdOfClientGroup" type="tns:getStudentListByIdOfClientGroup"/>  <xs:element name="getStudentListByIdOfClientGroupResponse" type="tns:getStudentListByIdOfClientGroupResponse"/>  <xs:element name="getSummary" type="tns:getSummary"/>  <xs:element name="getSummaryByGuardSan" type="tns:getSummaryByGuardSan"/>  <xs:element name="getSummaryByGuardSanResponse" type="tns:getSummaryByGuardSanResponse"/>  <xs:element name="getSummaryBySan" type="tns:getSummaryBySan"/>  <xs:element name="getSummaryBySanResponse" type="tns:getSummaryBySanResponse"/>  <xs:element name="getSummaryByTypedId" type="tns:getSummaryByTypedId"/>  <xs:element name="getSummaryByTypedIdResponse" type="tns:getSummaryByTypedIdResponse"/>  <xs:element name="getSummaryResponse" type="tns:getSummaryResponse"/>  <xs:element name="sendLinkingTokenByContractId" type="tns:sendLinkingTokenByContractId"/>  <xs:element name="sendLinkingTokenByContractIdResponse" type="tns:sendLinkingTokenByContractIdResponse"/>  <xs:element name="sendLinkingTokenByMobile" type="tns:sendLinkingTokenByMobile"/>  <xs:element name="sendLinkingTokenByMobileResponse" type="tns:sendLinkingTokenByMobileResponse"/>  <xs:complexType name="sendLinkingTokenByContractId">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="sendLinkingTokenByContractIdResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="result">  <xs:sequence>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="activateLinkingToken">  <xs:sequence>  <xs:element minOccurs="0" name="linkingToken" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="activateLinkingTokenResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:activateLinkingTokenResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="activateLinkingTokenResult">  <xs:complexContent>  <xs:extension base="tns:result">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  </xs:sequence>  </xs:extension>  </xs:complexContent>  </xs:complexType>  <xs:complexType name="sendLinkingTokenByMobile">  <xs:sequence>  <xs:element minOccurs="0" name="mobilePhone" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="sendLinkingTokenByMobileResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="changeMobilePhone">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="mobilePhone" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="changeMobilePhoneResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="generateLinkingToken">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="generateLinkingTokenResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:generateLinkingTokenResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="generateLinkingTokenResult">  <xs:complexContent>  <xs:extension base="tns:result">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="linkingToken" type="xs:string"/>  </xs:sequence>  </xs:extension>  </xs:complexContent>  </xs:complexType>  <xs:complexType name="getSummary">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:clientSummaryResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="clientSummaryResult">  <xs:sequence>  <xs:element minOccurs="0" name="clientSummary" type="tns:ClientSummaryExt"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="ClientSummaryExt">  <xs:sequence/>  <xs:attribute name="ContractId" type="xs:long"/>  <xs:attribute name="DateOfContract" type="xs:date"/>  <xs:attribute name="StateOfContract" type="xs:string"/>  <xs:attribute name="Balance" type="xs:long"/>  <xs:attribute name="OverdraftLimit" type="xs:long"/>  <xs:attribute name="ExpenditureLimit" type="xs:long"/>  <xs:attribute name="FirstName" type="xs:string"/>  <xs:attribute name="LastName" type="xs:string"/>  <xs:attribute name="MiddleName" type="xs:string"/>  <xs:attribute name="Grade" type="xs:string"/>  <xs:attribute name="OfficialName" type="xs:string"/>  <xs:attribute name="LastEnterEventCode" type="xs:int"/>  <xs:attribute name="LastEnterEventTime" type="xs:dateTime"/>  <xs:attribute name="NotifyViaSMS" type="xs:boolean"/>  <xs:attribute name="NotifyViaEmail" type="xs:boolean"/>  <xs:attribute name="MobilePhone" type="xs:string"/>  <xs:attribute name="Email" type="xs:string"/>  <xs:attribute name="DefaultMerchantId" type="xs:long"/>  <xs:attribute name="DefaultMerchantInfo" type="xs:string"/>  </xs:complexType>  <xs:complexType name="getCardListBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getCardListBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:cardListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="cardListResult">  <xs:sequence>  <xs:element minOccurs="0" name="cardList" type="tns:CardList"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CardList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="C" type="tns:CardItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CardItem">  <xs:sequence/>  <xs:attribute name="IdOfCard" type="xs:long"/>  <xs:attribute name="CrystalId" type="xs:long"/>  <xs:attribute name="Type" type="xs:int"/>  <xs:attribute name="State" type="xs:int"/>  <xs:attribute name="LifeState" type="xs:int"/>  <xs:attribute name="ExpiryDate" type="xs:date"/>  <xs:attribute name="ChangeDate" type="xs:dateTime"/>  </xs:complexType>  <xs:complexType name="getPaymentList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPaymentListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:paymentListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="paymentListResult">  <xs:sequence>  <xs:element minOccurs="0" name="paymentList" type="tns:PaymentList"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="PaymentList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:Payment"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="Payment">  <xs:sequence/>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Origin" type="xs:string"/>  </xs:complexType>  <xs:complexType name="getCardList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getCardListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:cardListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="detachGuardSan">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="guardSan" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="detachGuardSanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:detachGuardSanResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="detachGuardSanResult">  <xs:complexContent>  <xs:extension base="tns:result">  <xs:sequence/>  </xs:extension>  </xs:complexContent>  </xs:complexType>  <xs:complexType name="enableNotificationBySMS">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element name="state" type="xs:boolean"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="enableNotificationBySMSResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="changeEmail">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="email" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="changeEmailResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="enableNotificationByEmail">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element name="state" type="xs:boolean"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="enableNotificationByEmailResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="attachGuardSan">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="guardSan" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="attachGuardSanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:attachGuardSanResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="attachGuardSanResult">  <xs:complexContent>  <xs:extension base="tns:result">  <xs:sequence/>  </xs:extension>  </xs:complexContent>  </xs:complexType>  <xs:complexType name="getStudentListByIdOfClientGroup">  <xs:sequence>  <xs:element minOccurs="0" name="idOfClientGroup" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getStudentListByIdOfClientGroupResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:classStudentListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="classStudentListResult">  <xs:sequence>  <xs:element minOccurs="0" name="classStudentList" type="tns:ClassStudentList"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="ClassStudentList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="C" type="tns:ClientSummaryExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPaymentListBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPaymentListBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:paymentListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getClientsByGuardSan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getClientsByGuardSanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:clientsData"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="clientsData">  <xs:sequence>  <xs:element minOccurs="0" name="clientList" type="tns:clientList"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="clientList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="Client" type="tns:ClientItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="ClientItem">  <xs:sequence/>  <xs:attribute name="ContractId" type="xs:long"/>  <xs:attribute name="San" type="xs:string"/>  </xs:complexType>  <xs:complexType name="getGroupListByOrg">  <xs:sequence>  <xs:element minOccurs="0" name="idOfOrg" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getGroupListByOrgResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:clientGroupListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="clientGroupListResult">  <xs:sequence>  <xs:element minOccurs="0" name="clientGroupList" type="tns:ClientGroupList"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="ClientGroupList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="G" type="tns:ClientGroupItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="ClientGroupItem">  <xs:sequence/>  <xs:attribute name="idOfClientGroup" type="xs:long"/>  <xs:attribute name="GroupName" type="xs:string"/>  </xs:complexType>  <xs:complexType name="getSummaryBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:clientSummaryResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getEnterEventList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getEnterEventListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:enterEventListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="enterEventListResult">  <xs:sequence>  <xs:element minOccurs="0" name="enterEventList" type="tns:EnterEventList"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="EnterEventList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:EnterEventItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="EnterEventItem">  <xs:sequence/>  <xs:attribute name="DateTime" type="xs:dateTime"/>  <xs:attribute name="Day" type="xs:int"/>  <xs:attribute name="EnterName" type="xs:string"/>  <xs:attribute name="Direction" type="xs:int"/>  <xs:attribute name="TemporaryCard" type="xs:int"/>  </xs:complexType>  <xs:complexType name="getMenuListBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getMenuListBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:menuListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="menuListResult">  <xs:sequence>  <xs:element minOccurs="0" name="menuList" type="tns:MenuListExt"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="MenuListExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="M" type="tns:MenuDateItemExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="MenuDateItemExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:MenuItemExt"/>  </xs:sequence>  <xs:attribute name="Date" type="xs:date"/>  </xs:complexType>  <xs:complexType name="MenuItemExt">  <xs:sequence/>  <xs:attribute name="Group" type="xs:string"/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Price" type="xs:long"/>  <xs:attribute name="Calories" type="xs:double"/>  <xs:attribute name="VitB1" type="xs:double"/>  <xs:attribute name="VitC" type="xs:double"/>  <xs:attribute name="VitA" type="xs:double"/>  <xs:attribute name="VitE" type="xs:double"/>  <xs:attribute name="MinCa" type="xs:double"/>  <xs:attribute name="MinP" type="xs:double"/>  <xs:attribute name="MinMg" type="xs:double"/>  <xs:attribute name="MinFe" type="xs:double"/>  </xs:complexType>  <xs:complexType name="getContractIdByCardNo">  <xs:sequence>  <xs:element minOccurs="0" name="cardId" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getContractIdByCardNoResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryByTypedId">  <xs:sequence>  <xs:element minOccurs="0" name="id" type="xs:string"/>  <xs:element name="idType" type="xs:int"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryByTypedIdResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:clientSummaryResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryByGuardSan">  <xs:sequence>  <xs:element minOccurs="0" name="guardSan" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getSummaryByGuardSanResponse">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="return" type="tns:ClientSummaryExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPurchaseListBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPurchaseListBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:purchaseListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="purchaseListResult">  <xs:sequence>  <xs:element minOccurs="0" name="purchaseList" type="tns:PurchaseListExt"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="PurchaseListExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:PurchaseExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="PurchaseExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:PurchaseElementExt"/>  </xs:sequence>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="SocDiscount" type="xs:long"/>  <xs:attribute name="TrdDiscount" type="xs:long"/>  <xs:attribute name="Donation" type="xs:long"/>  <xs:attribute name="ByCash" type="xs:long"/>  <xs:attribute name="ByCard" type="xs:long"/>  <xs:attribute name="IdOfCard" type="xs:long"/>  </xs:complexType>  <xs:complexType name="PurchaseElementExt">  <xs:sequence/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Amount" type="xs:long"/>  <xs:attribute name="Type" type="xs:int"/>  </xs:complexType>  <xs:complexType name="getMenuList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getMenuListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:menuListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="authorizeClient">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="token" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="authorizeClientResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getCirculationList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element name="state" type="xs:int"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getCirculationListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:circulationListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="circulationListResult">  <xs:sequence>  <xs:element minOccurs="0" name="circulationList" type="tns:CirculationItemList"/>  <xs:element minOccurs="0" name="resultCode" type="xs:long"/>  <xs:element minOccurs="0" name="description" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CirculationItemList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="C" type="tns:CirculationItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CirculationItem">  <xs:sequence>  <xs:element name="Publication" type="tns:PublicationItem"/>  </xs:sequence>  <xs:attribute name="IssuanceDate" type="xs:date"/>  <xs:attribute name="RefundDate" type="xs:date"/>  <xs:attribute name="RealRefundDate" type="xs:date"/>  <xs:attribute name="Status" type="xs:int"/>  </xs:complexType>  <xs:complexType name="PublicationItem">  <xs:sequence/>  <xs:attribute name="Author" type="xs:string"/>  <xs:attribute name="Title" type="xs:string"/>  <xs:attribute name="Title2" type="xs:string"/>  <xs:attribute name="PublicationDate" type="xs:string"/>  <xs:attribute name="Publisher" type="xs:string"/>  </xs:complexType>  <xs:complexType name="changeExpenditureLimit">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element name="limit" type="xs:long"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="changeExpenditureLimitResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:result"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPurchaseList">  <xs:sequence>  <xs:element minOccurs="0" name="contractId" type="xs:long"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getPurchaseListResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:purchaseListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getMenuListByOrg">  <xs:sequence>  <xs:element minOccurs="0" name="orgId" type="xs:long"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getMenuListByOrgResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:menuListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="detachGuardSanBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="guardSan" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="detachGuardSanBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:detachGuardSanResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="attachGuardSanBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="guardSan" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="attachGuardSanBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:attachGuardSanResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getEnterEventListBySan">  <xs:sequence>  <xs:element minOccurs="0" name="san" type="xs:string"/>  <xs:element minOccurs="0" name="startDate" type="xs:dateTime"/>  <xs:element minOccurs="0" name="endDate" type="xs:dateTime"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="getEnterEventListBySanResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:enterEventListResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="PurchaseElement">  <xs:sequence/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Amount" type="xs:long"/>  </xs:complexType>  <xs:complexType name="ClientSummary">  <xs:sequence/>  <xs:attribute name="DateOfContract" type="xs:date"/>  <xs:attribute name="StateOfContract" type="xs:string"/>  <xs:attribute name="Balance" type="xs:long"/>  <xs:attribute name="OverdraftLimit" type="xs:long"/>  </xs:complexType>  <xs:complexType name="PurchaseList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:Purchase"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="Purchase">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:PurchaseElement"/>  </xs:sequence>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Discount" type="xs:long"/>  <xs:attribute name="Donation" type="xs:long"/>  <xs:attribute name="ByCash" type="xs:long"/>  <xs:attribute name="ByCard" type="xs:long"/>  <xs:attribute name="IdOfCard" type="xs:long"/>  </xs:complexType>  <xs:complexType name="MenuList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="M" type="tns:MenuDateItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="MenuDateItem">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:MenuItem"/>  </xs:sequence>  <xs:attribute name="Date" type="xs:date"/>  </xs:complexType>  <xs:complexType name="MenuItem">  <xs:sequence/>  <xs:attribute name="Group" type="xs:string"/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Price" type="xs:long"/>  </xs:complexType>  </xs:schema>  </wsdl:types>  <wsdl:message name="authorizeClientResponse">  <wsdl:part element="tns:authorizeClientResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="detachGuardSan">  <wsdl:part element="tns:detachGuardSan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryByTypedIdResponse">  <wsdl:part element="tns:getSummaryByTypedIdResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getEnterEventListBySan">  <wsdl:part element="tns:getEnterEventListBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuListByOrgResponse">  <wsdl:part element="tns:getMenuListByOrgResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeMobilePhone">  <wsdl:part element="tns:changeMobilePhone" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryByGuardSan">  <wsdl:part element="tns:getSummaryByGuardSan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuListBySan">  <wsdl:part element="tns:getMenuListBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCirculationList">  <wsdl:part element="tns:getCirculationList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCardList">  <wsdl:part element="tns:getCardList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryBySan">  <wsdl:part element="tns:getSummaryBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getClientsByGuardSanResponse">  <wsdl:part element="tns:getClientsByGuardSanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getEnterEventListResponse">  <wsdl:part element="tns:getEnterEventListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryByTypedId">  <wsdl:part element="tns:getSummaryByTypedId" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="generateLinkingToken">  <wsdl:part element="tns:generateLinkingToken" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCardListResponse">  <wsdl:part element="tns:getCardListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuListResponse">  <wsdl:part element="tns:getMenuListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="sendLinkingTokenByContractIdResponse">  <wsdl:part element="tns:sendLinkingTokenByContractIdResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getGroupListByOrg">  <wsdl:part element="tns:getGroupListByOrg" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPurchaseListBySan">  <wsdl:part element="tns:getPurchaseListBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getStudentListByIdOfClientGroupResponse">  <wsdl:part element="tns:getStudentListByIdOfClientGroupResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="generateLinkingTokenResponse">  <wsdl:part element="tns:generateLinkingTokenResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="sendLinkingTokenByMobile">  <wsdl:part element="tns:sendLinkingTokenByMobile" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuListBySanResponse">  <wsdl:part element="tns:getMenuListBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="attachGuardSanBySanResponse">  <wsdl:part element="tns:attachGuardSanBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryByGuardSanResponse">  <wsdl:part element="tns:getSummaryByGuardSanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuList">  <wsdl:part element="tns:getMenuList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummary">  <wsdl:part element="tns:getSummary" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getContractIdByCardNoResponse">  <wsdl:part element="tns:getContractIdByCardNoResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getStudentListByIdOfClientGroup">  <wsdl:part element="tns:getStudentListByIdOfClientGroup" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="attachGuardSan">  <wsdl:part element="tns:attachGuardSan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="sendLinkingTokenByMobileResponse">  <wsdl:part element="tns:sendLinkingTokenByMobileResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeEmail">  <wsdl:part element="tns:changeEmail" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPaymentListBySan">  <wsdl:part element="tns:getPaymentListBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPaymentListResponse">  <wsdl:part element="tns:getPaymentListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryBySanResponse">  <wsdl:part element="tns:getSummaryBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="attachGuardSanResponse">  <wsdl:part element="tns:attachGuardSanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="authorizeClient">  <wsdl:part element="tns:authorizeClient" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="detachGuardSanBySanResponse">  <wsdl:part element="tns:detachGuardSanBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCardListBySanResponse">  <wsdl:part element="tns:getCardListBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPaymentList">  <wsdl:part element="tns:getPaymentList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeMobilePhoneResponse">  <wsdl:part element="tns:changeMobilePhoneResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getSummaryResponse">  <wsdl:part element="tns:getSummaryResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="attachGuardSanBySan">  <wsdl:part element="tns:attachGuardSanBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="detachGuardSanResponse">  <wsdl:part element="tns:detachGuardSanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPaymentListBySanResponse">  <wsdl:part element="tns:getPaymentListBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="sendLinkingTokenByContractId">  <wsdl:part element="tns:sendLinkingTokenByContractId" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeExpenditureLimit">  <wsdl:part element="tns:changeExpenditureLimit" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="activateLinkingTokenResponse">  <wsdl:part element="tns:activateLinkingTokenResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeEmailResponse">  <wsdl:part element="tns:changeEmailResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getClientsByGuardSan">  <wsdl:part element="tns:getClientsByGuardSan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="enableNotificationBySMSResponse">  <wsdl:part element="tns:enableNotificationBySMSResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getEnterEventListBySanResponse">  <wsdl:part element="tns:getEnterEventListBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPurchaseListBySanResponse">  <wsdl:part element="tns:getPurchaseListBySanResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCardListBySan">  <wsdl:part element="tns:getCardListBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getCirculationListResponse">  <wsdl:part element="tns:getCirculationListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="activateLinkingToken">  <wsdl:part element="tns:activateLinkingToken" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPurchaseList">  <wsdl:part element="tns:getPurchaseList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getMenuListByOrg">  <wsdl:part element="tns:getMenuListByOrg" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="detachGuardSanBySan">  <wsdl:part element="tns:detachGuardSanBySan" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="enableNotificationByEmail">  <wsdl:part element="tns:enableNotificationByEmail" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getPurchaseListResponse">  <wsdl:part element="tns:getPurchaseListResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getContractIdByCardNo">  <wsdl:part element="tns:getContractIdByCardNo" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getEnterEventList">  <wsdl:part element="tns:getEnterEventList" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="enableNotificationBySMS">  <wsdl:part element="tns:enableNotificationBySMS" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="changeExpenditureLimitResponse">  <wsdl:part element="tns:changeExpenditureLimitResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="enableNotificationByEmailResponse">  <wsdl:part element="tns:enableNotificationByEmailResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="getGroupListByOrgResponse">  <wsdl:part element="tns:getGroupListByOrgResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:portType name="ClientRoomController">  <wsdl:operation name="sendLinkingTokenByContractId">  <wsdl:input message="tns:sendLinkingTokenByContractId" name="sendLinkingTokenByContractId">  </wsdl:input>  <wsdl:output message="tns:sendLinkingTokenByContractIdResponse" name="sendLinkingTokenByContractIdResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="activateLinkingToken">  <wsdl:input message="tns:activateLinkingToken" name="activateLinkingToken">  </wsdl:input>  <wsdl:output message="tns:activateLinkingTokenResponse" name="activateLinkingTokenResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="sendLinkingTokenByMobile">  <wsdl:input message="tns:sendLinkingTokenByMobile" name="sendLinkingTokenByMobile">  </wsdl:input>  <wsdl:output message="tns:sendLinkingTokenByMobileResponse" name="sendLinkingTokenByMobileResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeMobilePhone">  <wsdl:input message="tns:changeMobilePhone" name="changeMobilePhone">  </wsdl:input>  <wsdl:output message="tns:changeMobilePhoneResponse" name="changeMobilePhoneResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="generateLinkingToken">  <wsdl:input message="tns:generateLinkingToken" name="generateLinkingToken">  </wsdl:input>  <wsdl:output message="tns:generateLinkingTokenResponse" name="generateLinkingTokenResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummary">  <wsdl:input message="tns:getSummary" name="getSummary">  </wsdl:input>  <wsdl:output message="tns:getSummaryResponse" name="getSummaryResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCardListBySan">  <wsdl:input message="tns:getCardListBySan" name="getCardListBySan">  </wsdl:input>  <wsdl:output message="tns:getCardListBySanResponse" name="getCardListBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPaymentList">  <wsdl:input message="tns:getPaymentList" name="getPaymentList">  </wsdl:input>  <wsdl:output message="tns:getPaymentListResponse" name="getPaymentListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCardList">  <wsdl:input message="tns:getCardList" name="getCardList">  </wsdl:input>  <wsdl:output message="tns:getCardListResponse" name="getCardListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="detachGuardSan">  <wsdl:input message="tns:detachGuardSan" name="detachGuardSan">  </wsdl:input>  <wsdl:output message="tns:detachGuardSanResponse" name="detachGuardSanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="enableNotificationBySMS">  <wsdl:input message="tns:enableNotificationBySMS" name="enableNotificationBySMS">  </wsdl:input>  <wsdl:output message="tns:enableNotificationBySMSResponse" name="enableNotificationBySMSResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeEmail">  <wsdl:input message="tns:changeEmail" name="changeEmail">  </wsdl:input>  <wsdl:output message="tns:changeEmailResponse" name="changeEmailResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="enableNotificationByEmail">  <wsdl:input message="tns:enableNotificationByEmail" name="enableNotificationByEmail">  </wsdl:input>  <wsdl:output message="tns:enableNotificationByEmailResponse" name="enableNotificationByEmailResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="attachGuardSan">  <wsdl:input message="tns:attachGuardSan" name="attachGuardSan">  </wsdl:input>  <wsdl:output message="tns:attachGuardSanResponse" name="attachGuardSanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getStudentListByIdOfClientGroup">  <wsdl:input message="tns:getStudentListByIdOfClientGroup" name="getStudentListByIdOfClientGroup">  </wsdl:input>  <wsdl:output message="tns:getStudentListByIdOfClientGroupResponse" name="getStudentListByIdOfClientGroupResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPaymentListBySan">  <wsdl:input message="tns:getPaymentListBySan" name="getPaymentListBySan">  </wsdl:input>  <wsdl:output message="tns:getPaymentListBySanResponse" name="getPaymentListBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getClientsByGuardSan">  <wsdl:input message="tns:getClientsByGuardSan" name="getClientsByGuardSan">  </wsdl:input>  <wsdl:output message="tns:getClientsByGuardSanResponse" name="getClientsByGuardSanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getGroupListByOrg">  <wsdl:input message="tns:getGroupListByOrg" name="getGroupListByOrg">  </wsdl:input>  <wsdl:output message="tns:getGroupListByOrgResponse" name="getGroupListByOrgResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryBySan">  <wsdl:input message="tns:getSummaryBySan" name="getSummaryBySan">  </wsdl:input>  <wsdl:output message="tns:getSummaryBySanResponse" name="getSummaryBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getEnterEventList">  <wsdl:input message="tns:getEnterEventList" name="getEnterEventList">  </wsdl:input>  <wsdl:output message="tns:getEnterEventListResponse" name="getEnterEventListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuListBySan">  <wsdl:input message="tns:getMenuListBySan" name="getMenuListBySan">  </wsdl:input>  <wsdl:output message="tns:getMenuListBySanResponse" name="getMenuListBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getContractIdByCardNo">  <wsdl:input message="tns:getContractIdByCardNo" name="getContractIdByCardNo">  </wsdl:input>  <wsdl:output message="tns:getContractIdByCardNoResponse" name="getContractIdByCardNoResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryByTypedId">  <wsdl:input message="tns:getSummaryByTypedId" name="getSummaryByTypedId">  </wsdl:input>  <wsdl:output message="tns:getSummaryByTypedIdResponse" name="getSummaryByTypedIdResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryByGuardSan">  <wsdl:input message="tns:getSummaryByGuardSan" name="getSummaryByGuardSan">  </wsdl:input>  <wsdl:output message="tns:getSummaryByGuardSanResponse" name="getSummaryByGuardSanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPurchaseListBySan">  <wsdl:input message="tns:getPurchaseListBySan" name="getPurchaseListBySan">  </wsdl:input>  <wsdl:output message="tns:getPurchaseListBySanResponse" name="getPurchaseListBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuList">  <wsdl:input message="tns:getMenuList" name="getMenuList">  </wsdl:input>  <wsdl:output message="tns:getMenuListResponse" name="getMenuListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="authorizeClient">  <wsdl:input message="tns:authorizeClient" name="authorizeClient">  </wsdl:input>  <wsdl:output message="tns:authorizeClientResponse" name="authorizeClientResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCirculationList">  <wsdl:input message="tns:getCirculationList" name="getCirculationList">  </wsdl:input>  <wsdl:output message="tns:getCirculationListResponse" name="getCirculationListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeExpenditureLimit">  <wsdl:input message="tns:changeExpenditureLimit" name="changeExpenditureLimit">  </wsdl:input>  <wsdl:output message="tns:changeExpenditureLimitResponse" name="changeExpenditureLimitResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPurchaseList">  <wsdl:input message="tns:getPurchaseList" name="getPurchaseList">  </wsdl:input>  <wsdl:output message="tns:getPurchaseListResponse" name="getPurchaseListResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuListByOrg">  <wsdl:input message="tns:getMenuListByOrg" name="getMenuListByOrg">  </wsdl:input>  <wsdl:output message="tns:getMenuListByOrgResponse" name="getMenuListByOrgResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="detachGuardSanBySan">  <wsdl:input message="tns:detachGuardSanBySan" name="detachGuardSanBySan">  </wsdl:input>  <wsdl:output message="tns:detachGuardSanBySanResponse" name="detachGuardSanBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="attachGuardSanBySan">  <wsdl:input message="tns:attachGuardSanBySan" name="attachGuardSanBySan">  </wsdl:input>  <wsdl:output message="tns:attachGuardSanBySanResponse" name="attachGuardSanBySanResponse">  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getEnterEventListBySan">  <wsdl:input message="tns:getEnterEventListBySan" name="getEnterEventListBySan">  </wsdl:input>  <wsdl:output message="tns:getEnterEventListBySanResponse" name="getEnterEventListBySanResponse">  </wsdl:output>  </wsdl:operation>  </wsdl:portType>  <wsdl:binding name="ClientRoomControllerWSServiceSoapBinding" type="tns:ClientRoomController">  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>  <wsdl:operation name="activateLinkingToken">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="activateLinkingToken">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="activateLinkingTokenResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="sendLinkingTokenByContractId">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="sendLinkingTokenByContractId">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="sendLinkingTokenByContractIdResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeMobilePhone">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="changeMobilePhone">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="changeMobilePhoneResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="sendLinkingTokenByMobile">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="sendLinkingTokenByMobile">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="sendLinkingTokenByMobileResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="generateLinkingToken">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="generateLinkingToken">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="generateLinkingTokenResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummary">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getSummary">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getSummaryResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCardListBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getCardListBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getCardListBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPaymentList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getPaymentList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getPaymentListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="detachGuardSan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="detachGuardSan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="detachGuardSanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCardList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getCardList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getCardListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeEmail">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="changeEmail">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="changeEmailResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="enableNotificationBySMS">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="enableNotificationBySMS">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="enableNotificationBySMSResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="enableNotificationByEmail">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="enableNotificationByEmail">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="enableNotificationByEmailResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="attachGuardSan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="attachGuardSan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="attachGuardSanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPaymentListBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getPaymentListBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getPaymentListBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getStudentListByIdOfClientGroup">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getStudentListByIdOfClientGroup">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getStudentListByIdOfClientGroupResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getClientsByGuardSan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getClientsByGuardSan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getClientsByGuardSanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getGroupListByOrg">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getGroupListByOrg">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getGroupListByOrgResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getEnterEventList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getEnterEventList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getEnterEventListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getSummaryBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getSummaryBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuListBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getMenuListBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getMenuListBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getContractIdByCardNo">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getContractIdByCardNo">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getContractIdByCardNoResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryByTypedId">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getSummaryByTypedId">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getSummaryByTypedIdResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getSummaryByGuardSan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getSummaryByGuardSan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getSummaryByGuardSanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getMenuList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getMenuListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPurchaseListBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getPurchaseListBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getPurchaseListBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="authorizeClient">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="authorizeClient">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="authorizeClientResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getCirculationList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getCirculationList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getCirculationListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="changeExpenditureLimit">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="changeExpenditureLimit">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="changeExpenditureLimitResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getPurchaseList">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getPurchaseList">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getPurchaseListResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getMenuListByOrg">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getMenuListByOrg">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getMenuListByOrgResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="detachGuardSanBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="detachGuardSanBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="detachGuardSanBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="attachGuardSanBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="attachGuardSanBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="attachGuardSanBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  <wsdl:operation name="getEnterEventListBySan">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="getEnterEventListBySan">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="getEnterEventListBySanResponse">  <soap:body use="literal"/>  </wsdl:output>  </wsdl:operation>  </wsdl:binding>  <wsdl:service name="ClientRoomControllerWSService">  <wsdl:port binding="tns:ClientRoomControllerWSServiceSoapBinding" name="ClientRoomControllerWSPort">  <soap:address location="http://localhost:8080/processor/soap/client"/>  </wsdl:port>  </wsdl:service>  </wsdl:definitions> |

Описание общих структур данных

1. Параметр комплексного типа: ClientSummaryExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | ContractId | Номер договора (лицевого счета) | + | xs:long | Rgu:ResultInformation |
|  | dateOfContract | Дата регистрации клиента | + | xs:date |  |
|  | stateOfContract | Состояние договора клиента (Активен, Не заключен, Расторгнут по времени, Расторгнут по желанию клиента) | + | xs:string |  |
|  | balance | Текущий баланс счета | + | xs:long |  |
|  | overdraftLimit | Лимит овердрафта | + | xs:long |  |
|  | expenditureLimit | Дневной лимит затрат | + | xs:long |  |
|  | firstName | Имя клиента | + | xs:string |  |
|  | lastName | Фамилия клиента | + | xs:string |  |
|  | MiddleName | Отчество клиента | + | xs:string |  |
|  | grade | Класс (группа) | + | xs:string |  |
|  | officialName | Название организации | + | xs:string |  |
|  | LastEnterEventCode | Код последнего события контроля доступа:  Entry = 0; // вход  Exit = 1; // выход  PassageIsForbidden = 2; // проход запрещен  EventWithoutPassage = 4; // событие без прохода  PassageRufusal = 5; // отказ от прохода  ReEntry = 6; // повторный вход  ReExit = 7; // повторный выход | + | xs:int |  |
|  | LastEnterEventTime | Время последнего события контроля доступа | + | xs:dateTime |  |
|  | NotifyViaSMS | Включено ли уведомление по SMS | + | xs:boolean |  |
|  | NotifyViaEmail | Включено ли уведомление по E-Mail | + | xs:boolean |  |
|  | MobilePhone | Номер телефона для SMS уведомлений | + | xs:string |  |
|  | Email | E-mail для уведомлений | + | xs:string |  |
|  | DefaultMerchantId | Идентификатор поставщика по-умолчанию |  | xs:long |  |
|  | DefaultMerchantInfo | Дополнительные данные по поставщику |  | xs:string |  |

1. Параметр комплексного типа: PurchaseListExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | Time | Дата и время регистрации покупки | + | xs:dateTime |  |
| 2 | Sum | Сумма | + | xs:long |  |
| 3 | SocDiscount | Социальная скидка (для льготного питания) | + | xs:long |  |
| 4 | TrdDiscount | Торговая скидка | + | xs:long |  |
| 5 | Donation | Дотация | + | xs:long |  |
| 6 | ByCash | Сумма, оплаченная наличными | + | xs:long |  |
| 7 | ByCard | Сумма, оплаченная по карте | + | xs:long |  |
| 8 | E | Состав покупки |  | PurchaseElementExt |  |

1. Параметр комплексного типа: PurchaseElementExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | Name | Наименование товара | + | xs:string |  |
| 2 | Sum | Сумма | + | xs:long |  |
| 3 | Amount | Количество | + | xs:long |  |
| 4 | Type | Тип | + | xs:int | 0 – блюдо, 1 – комплекс, 2 – блюдо из комплекса |

1. Параметр комплексного типа: PurchaseExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | Name | Наименование товара | + | xs:string |  |
| 2 | Sum | Сумма | + | xs:long |  |
| 3 | Amount | Количество | + | xs:long |  |
| 4 | Type | Тип | + | xs:int | 0 – блюдо, 1 – комплекс, 2 – блюдо из комплекса |

1. Параметр комплексного типа: PaymentList

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | P | Данные пополнения |  | Payment |  |

1. Параметр комплексного типа: Payment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | Time | Данные пополнения |  | xs:dateTime |  |
| 2 | Sum | Сумма |  | xs:long | В копейках |
| 3 | Origin | Наименование системы и метода оплаты |  | xs:string |  |

1. Параметр комплексного типа: MenuListExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | M | Данные меню на день |  | MenuDateItemExt |  |

* 1. Параметр комплексного типа: MenuDateItemExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | E | Элементы меню |  | MenuItemExt |  |
| 2 | Date | Дата меню | + | xs:date |  |

1. Параметр комплексного типа: MenuItemExt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | Group | Группа (Первое, Второе, Напитки, и т.д.) | + | MenuItemExt |  |
| 2 | Name | Наименование | + | xs:date |  |
| 3 | Price | Цена | + | xs:long |  |
| 4 | Calories | Калории |  | xs:double |  |
| 5 | VitB1 | Витамин B1 |  | xs:double |  |
| 6 | VitC | Витамин C |  | xs:double |  |
| 7 | VitA | Витамин A |  | xs:double |  |
| 8 | VitE | Витамин E |  | xs:double |  |
| 9 | MinCa | Кальций |  | xs:double |  |
| 10 | MinP | Фосфор |  | xs:double |  |
| 11 | MinMg | Магний |  | xs:double |  |
| 12 | MinFe | Железо |  | xs:double |  |

1. Параметр комплексного типа: EnterEventList

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | E | Данные прохода |  | EnterEventItem |  |

1. Параметр комплексного типа: EnterEventItem

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | dateTime | Дата и время события | + | xs:dateTime |  |
| 2 | day | День недели события | + | xs:int |  |
| 3 | enterName | Наименование входной группы | + | xs:string |  |
| 4 | direction | Направление прохода | + | xs:int | 0 – вход, 1 – выход, 2- проход запрещен, 3 – взлом турникета, 4 – событие без прохода, 5 – отказ от прохода, 6 – повторный вход, 7 – повторный выход |
| 5 | temporaryCard | Событие совершено по временной карте | + | xs:int | 0 — нет, 1 — да |