

Автоматизированная Информационная Система
«Е-Заявление» («Е-Справка»)

**Техническая инструкция сервисов по
справкам о наличии судимости для гос.
порталов - посредством СМЭВ Түндүк**

Сервис работает посредством СМЭВ Түндүк, путем реализации веб-сервиса согласно протоколу СМЭВ.

Для получения WSDL файла необходимо обратиться по ссылке адреса в сервер безопасности СМЭВ Түндүк.

Терминология:

СМЭВ - система межведомственного электронного взаимодействия.

Сервер безопасности - Шлюз подключения к СМЭВ.

Веб-сервис разработан методом протокола SOAP и состоит из 6 методов для двух видов услуг:

Заголовки сообщений:

В этом разделе описываются заголовки SOAP, которые используются СМЭВ Түндүк.

В таблице ниже представлены заголовки СМЭВ Түндүк.

Когда клиент службы отправляет запрос на сервер безопасности, обязательно должна присутствовать одна из служб полей или centralService. Если используется поле centralService, сервер безопасности разрешает центральную службу и автоматически заполняется в поле службы идентификатором конкретной услуги, которая реализует центральную службу.

Поле	Тип	Обязательное или Опциональное	Описание
client	XRoadClientIdentifierType	Обязательное	Идентифицирует Клиента — сущность, которая инициирует вызов услуги.
service	XRoadServiceIdentifierType	Опциональное	Идентифицирует сервис, вызываемый запросом.
centralService	XRoadCentralServiceIdentifierType	Опциональное	Идентифицирует центральную службу, вызываемую запросом.
id	string	Обязательное	Уникальный идентификатор для сообщения. Рекомендуемый метод генерации сообщения ID - 30 байт

			случайных данных, в кодировке Base64.
userId	string	Обязательное	Пользователь, действие которого инициировало запрос. Идентификатор пользователя должен быть с двухбуквенным ISO код страны (например, KG20114199609123).
requestHash	string	Опциональное	Для ответов. Это поле содержит хэш запроса SOAP сообщения. Это поле автоматически заполняется поставщиком сервисов сервера безопасности.
requestHash/ @algorithmId	string	Обязательное	Идентифицирует хэшалгоритм, который был использован для вычисления значения запроса поля Hash. Алгоритмы обозначаются как URI перечисленные в XML-DSIG спецификации [DSI13].
issue	string	Опциональное	Идентифицирует заявление, основание или документ, который послужил основанием обращения к сервису. Это поле используется для подключения службы запросов (и ответов) к рабочим процедурам

authentication Method	string	Обязательное	Метод аутентификации
profileVersion	string	Обязательное	Версия протокола СМЭВ Түндүк. Для СМЭВ Түндүк 6 версия протокола 4.0

При ответе служба копирует все поля заголовка из запроса в ответ.

Поле requestHash используется для создания прочной связи между запросом и ответом.

Таким образом, можно доказать, что, например, определенная запись реестра возвращается в ответ на определенный запрос. Используя журналы, можно восстановить и проверить пару запрос-ответ.

Поле requestHash автоматически создается сервером безопасности поставщика услуг и проверяется сервером безопасности клиента службы.

Поле authenticationMethod должно быть одним из следующих:

EID - с удостоверением личности (ID-CARD, DIGI-ID);

MID - с мобильным удостоверением личности (MOBILE-ID);

CERT - с другим сертификатом;

EXTERNAL - через стороннюю услугу (BANK-LINK);

PASSWORD - с идентификатором пользователя и паролем;

SYSTEM - когда запрос был отправлен заданием cron или другим автоматическим процессом.

Данные заголовки являются стандартными для использования СМЭВ Түндүк и присутствуют во всех методах.

Успешно, в этом случае возвращается объект *ObjectResult*, где:

DataIdentifier - идентификатор данных заявления, если заявитель хочет воспользоваться платной услугой (заявления на 1 или 3 рабочих дня) идентификатор данных может использоваться для проведения оплаты в банках или платежных терминалах или используется во втором методе *PayDocumentet* для подтверждения оплаты.

SessionKey - текущая сессия для проведения платежа, используется в методе *PayDocument*.

SecretKey - идентификатор заявления, используется во втором методе *PayDocument* для подтверждения оплаты.

PersonPIN - инн заявителя.

PersonName, *PersonSurname*, *PersonPatronymic* - имя, фамилия, отчество, заявителя.

PersonPhone - телефон заявителя.

ReceiptRegionName, *ReceiptRegionCode* - место получения справки, наименование и код.

ReceivingAfterComment - информация о готовности справки

StatementTypeName, *StatementTypeCode* - тип услуги, наименование и код.

StatementTypeSumOfService - сумма услуги, если заявление на бесплатной основе, сумма услуги имеет значение 0.

PersonalDataChanged – менялись или нет персональные данные;

ChangedName, *ChangedSurname*, *ChangedPatronymic* – указанные данные при изменении имени, фамилии, отчества.

AdditionalData - json объект с дополнительных данных по согласованию (опциональный параметр)

b)

Code = x;

Name = "Другой статус";

Не успешно, в этом случае возвращается разные коды и ошибки, которые возникли, в этом случае

ObjectResult = **null**

PayDocument (*PayDocument request*) - оплата:

Где

request – параметры запроса с заголовком СМЭВ Түндүк и дополнительные параметры:

companyIdentifier - полученный уникальный персональный идентификатор;
sessionKey - текущая сессия для операции из метода *CreateGetStatementData*;
secretKey - идентификатор заявления из объекта результата метода *CreateGetStatementData*;
dataIdentifier - идентификатор данных из объекта результата метода *CreateGetStatementData*;
amount - введенная сумма;
paymentRegionCode – область проведения оплаты, код из реестра -

Бишкек	1
Ош	2
Каракол	3
Нарын	4
Талас	5
Баткен	6
Жалал-Абад	7

Ответ состоит из параметров запроса с заголовком СМЭВ Түндүк и из корпуса сервиса, где:

а)

Code = 0;
 Name = "Success";

Успешно, в этом случае возвращается объект *ObjectResult*, где:

PaymentIdentifier - идентификатор платежа, используется при отмене платежа;

DocumentCode - 16 значный код – идентификатор заявления.

PersonPIN, *PersonName*, *PersonSurname*, *PersonPatronymic* – ПИН, имя, фамилия, отчество, заявителя.

PersonPhone - телефон заявителя.

ReceiptRegionName, *ReceiptRegionCode* - место получения справки, наименование и код.

PaymentRegionName, *PaymentRegionCode* - область проведения оплаты, наименование и код.

StatementTypeName, *StatementTypeCode* - тип услуги, наименование и код.

StatementTypeSumOfService - сумма услуги.

ReceivingAfterDate - дата и время, когда будет готова справка («получить после»).

TickedNumberInReceivingDay – номер тикета заявителя, действителен только в день получения.

Receiver – Получатель (относится к банковским реквизитам);

TransitAccount - Транзитный счет (относится к банковским реквизитам);
AccumulationAccount - Лицевой счет (относится к банковским реквизитам);
BICBank - БИК банка (относится к банковским реквизитам);
ReceiverINN – ИНН получателя (относится к банковским реквизитам);
GNI – ГНИ (относится к банковским реквизитам);
OKPO – ОКПО (относится к банковским реквизитам);
Bank – Банк (относится к банковским реквизитам);
PersonalDataChanged – менялись или нет персональные данные;
ChangedName, ChangedSurname, ChangedPatronymic – указанные данные при изменении имени, фамилии, отчества.

b)

Code = x;
Name = "Другой статус";

Не успешно, в этом случае возвращается разные коды и ошибки, которые возникли, в этом случае

ObjectResult = **null**