Техническое задание

Проверка валидности подписи, полученной в результате подписания документа через МП «Госключ»

Назначение доработки:

Создание функциональности в 1С: ITIL для проверки валидности подписи, полученной в результате подписания документа через мобильное приложение (далее – МП) «Госключ».

Описание доработки:

Документы подписывается УНЭП в заявке из Личного кабинета. Заявка с подписанным документом и откреплленным файлом подписи в виде архива направляется в 1С: ITIL (уже реализовано).

Подпись документов через МП «Госключ» доступна не для всех услуг. Перечень услуг, в которых доступна подпись через МП «Госключ»:

1. Академический отпуск
2. Заявление о предоставлении или отказе от каникул
3. Отчисление по собственному желанию.

Проверка подписи будет осуществляться через вызов к конечной точке **/verify/from_file** на микросервисе подписи документов через МП «Госключ» при нажатии на кнопку в 1С: ITIL.

Изменение программного интерфейса:

1. В 1С: ITIL справочнике **«Каталог услуг»** на форме услуги создать реквизит (галочку) «Подписывается через Госключ», тип – булево.

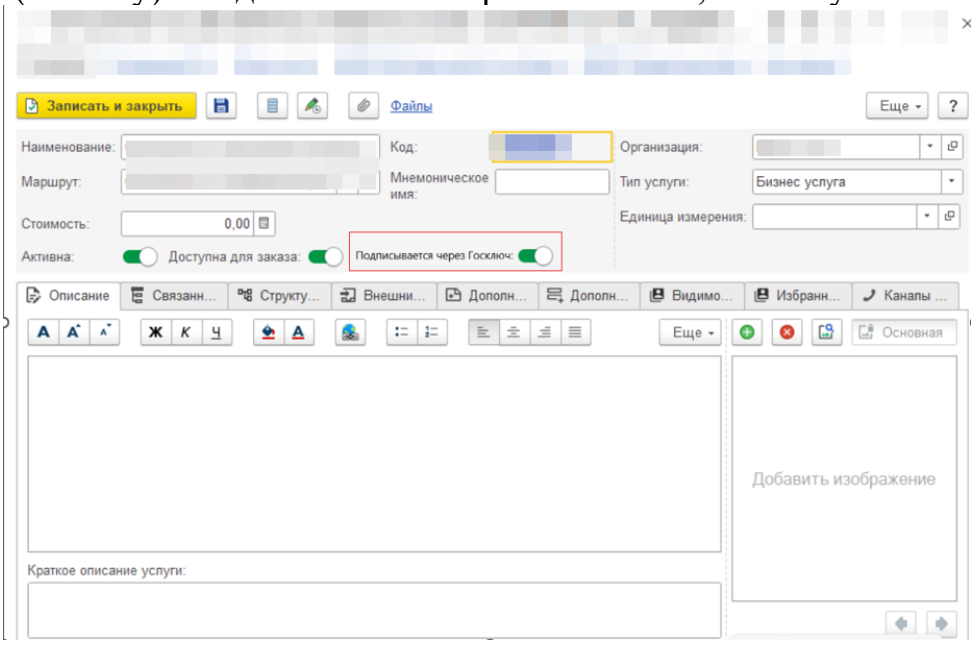

Рисунок 1. Расположение реквизита «Подписывается через Госключ»

Для следующий услуг реквизит проставить в **Истина**:

1. Оформление академического отпуска (*код скрыт*).
2. Заявление о предоставлении (или отказе от) каникул (*код скрыт*).
3. Отчисление по собственному желанию (*код скрыт*).

2. На форме работы с обращением добавить кнопку «**Проверка валидности подписи**». На рисунке ниже пример расположения кнопки на рабочей области.

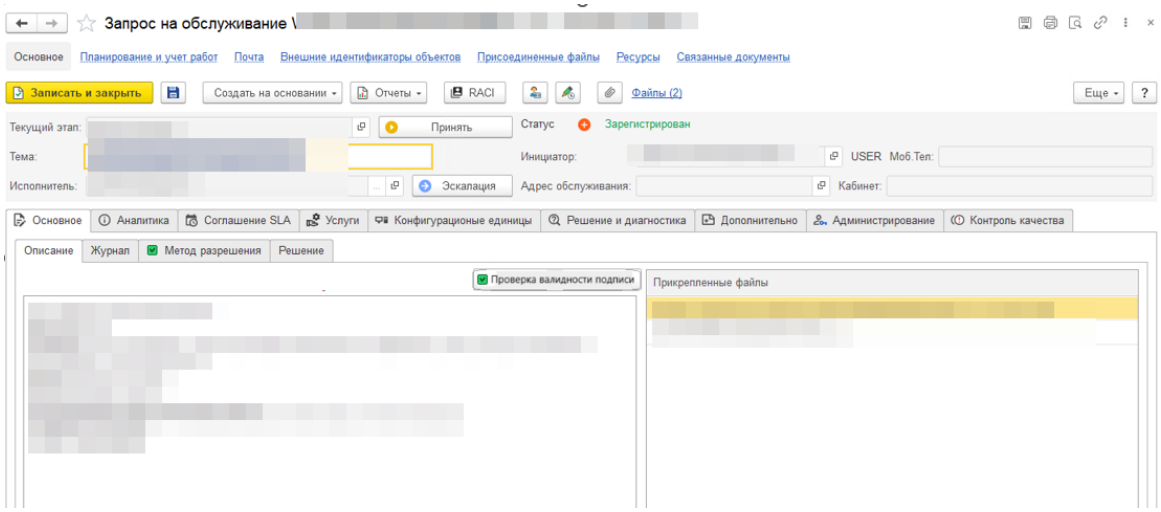

Рисунок 2. Расположение кнопки "Проверка валидности подписи"

Кнопку «Проверка валидности подписи» **показывать только в случае**, если в обращении выбрана услуга, в карточке которой реквизит **«Подписывается через Госключ» = ИСТИНА**.

3. При нажатии на кнопку «Проверка валидности подписи»:

1. Выбираем из вложений файл-архив с именем signed_documents.
2. Распаковываем его программно. Получаем один файл в формате pdf – это подписанный документ, второй файл формата sig – открепленная подпись.
3. Полученный файл pdf передаём в реквизит file, файл sig передаём в реквизит signature при отправке запроса на проверку подписи по API (см. **Описание конечной точки /verify/from_file**).

При **отсутствии в обращении архива с именем signed_documents**, выдать ошибку пользователю: «Отсутствуют необходимые файлы для проверки».

4. После направления запроса к конечной точке и получения ответа от сервиса по проверке подписи отображаем пользователю ответ в обращении:

4.1. **Получен ответ от сервера с кодом 200**

Значение ключа **"validated" = true**
Выводим сообщение пользователю: «Подпись действительна».
Цвет текста: #3C9848

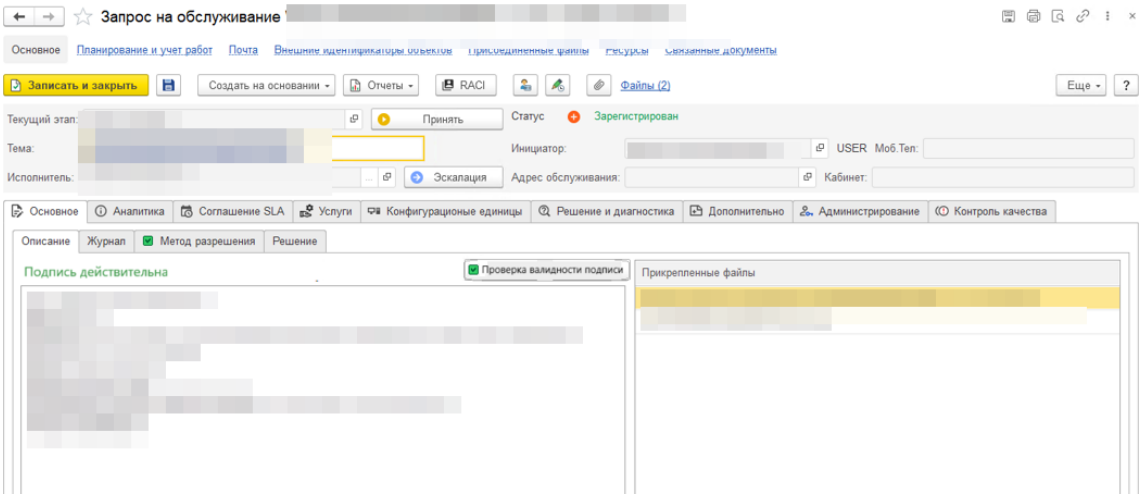

Рисунок 4

Значение ключа **"validated" = false**
Выводим сообщение пользователю: «Подпись недействительна».
Цвет текста: # F71B0B

Значение ключа **"validated" = null**
Выводим сообщение пользователю: «Ошибка при отправке запроса.».
Цвет текста: # ffa500

4.2. **Получение ответа от сервера с кодом 422**

Выводим сообщение пользователю об ошибке с кодом и текстовым описание ошибки из ключа "msg" в ответе от сервера.

Описание конечной точки /verify/from_file

Спецификация в Swagger: *адрес скрыт*
Полный путь к конечной точке: *адрес скрыт*
Метод: POST
Authorization type: Bearer Token
Token (тестовый):
Token (рабочий):
Content-Type: multipart/form-data

В теле запроса передаются два файла (разархивированные из 1С):

1. Подписанный документ в формате .pdf
2. Откреплённая подпись. sig

Оба файла являются обязательными.

Пример запроса:

```
curl -X 'POST' \
  'https://_________ \
  -H 'accept: application/json' \
  -H 'Authorization: Bearer _____________ \
  -H 'Content-Type: multipart/form-data' \
  -F 'file /pdf' \
  -F 'signature .sig'
```

Ответы сервера:

Код ответа	Структура JSON

200 – Успешный ответ	{
	"validated": true,
	"signer": "string",
	"date": 0
	}

422 – Ошибка проверки	{
	"detail": [
	{
	"loc": [
	"string",
	0
	],
	"msg": "string",
	"type": "string"
	}
	]
	}