



Описание функциональных характеристик  
программного обеспечения  
«LAB. Машинное обучение и ИИ»

Екатеринбург, 2022

## Оглавление

Термины и определения.....	3
Назначение системы .....	4
Функциональные возможности .....	5
Типы запросов к сервису .....	7
Методы API.....	12

## Термины и определения

Термин	Определение
ИС	Информационная система
JSON	(англ. <i>JavaScript Object Notation</i> , обычно) — текстовый формат обмена данными, основанный на JavaScript
REST	(от англ. Representational State Transfer - передача состояния представления) архитектурный стиль взаимодействия компонентов распределенного приложения в сети
URL	Адрес (ссылка), указывающий точное местоположение веб-ресурса

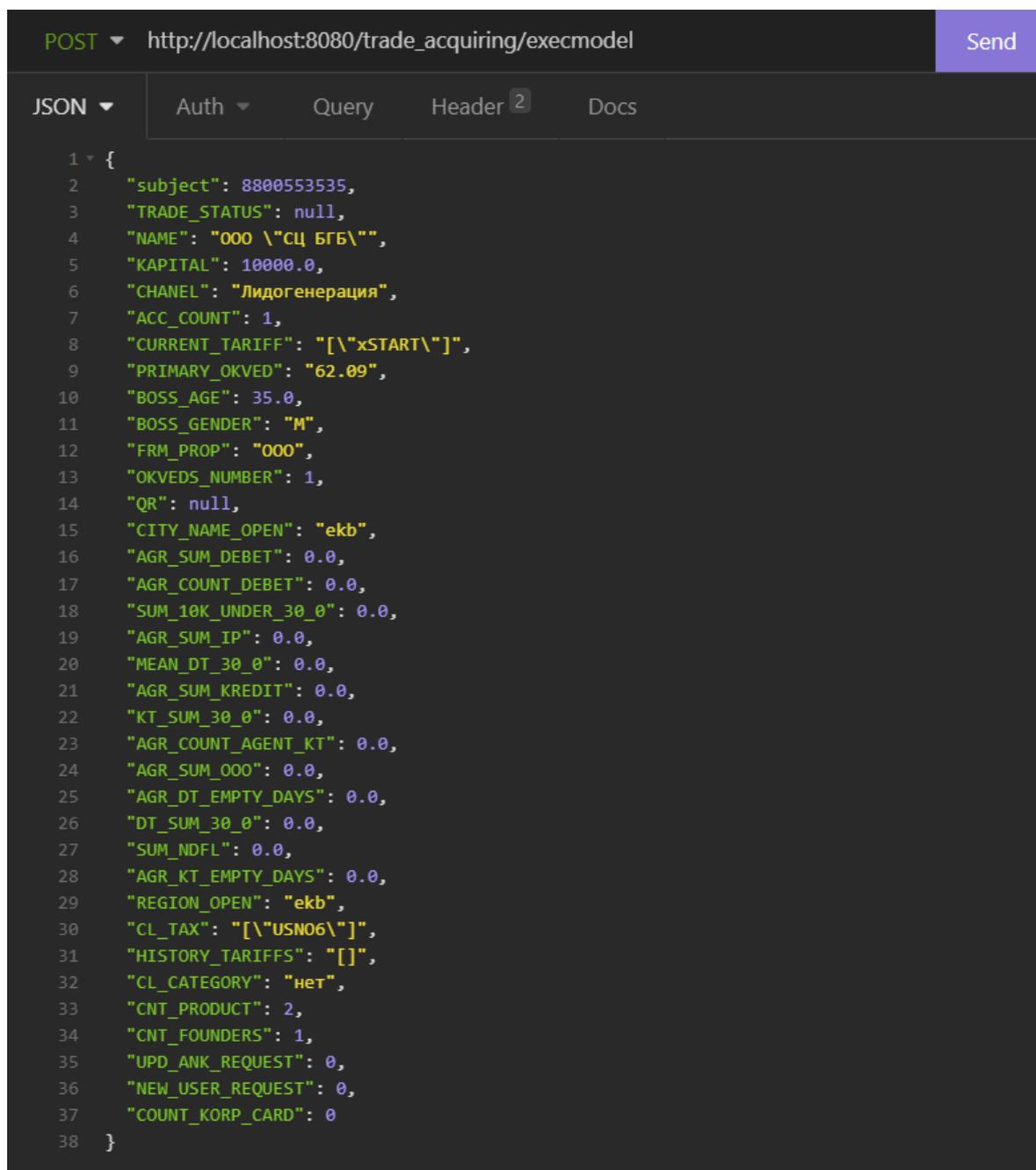
## Назначение системы

**LAB. Машинное обучение и ИИ** (далее по тексту LAB\_ML) предоставляет возможность расчета рекомендаций банковских продуктов конкретным клиентам по протоколу взаимодействия REST API.

Сервис предоставляет возможность получить скоринговую оценку заинтересованности конкретного клиента в том или ином банковском продукте, в данном случае рассчитывается вероятность заинтересованности клиента в подключении услуги торгового эквайринга.

Для демонстрации работы сервиса на платформе настроены тестовые примеры, которые используют данные запроса для расчета рекомендаций.

Начать работу с сервисом рекомендаций можно отправив запрос по ссылке [http://localhost:8080/trade\\_acquiring/execmodel](http://localhost:8080/trade_acquiring/execmodel), содержащий в себе признаки клиента.

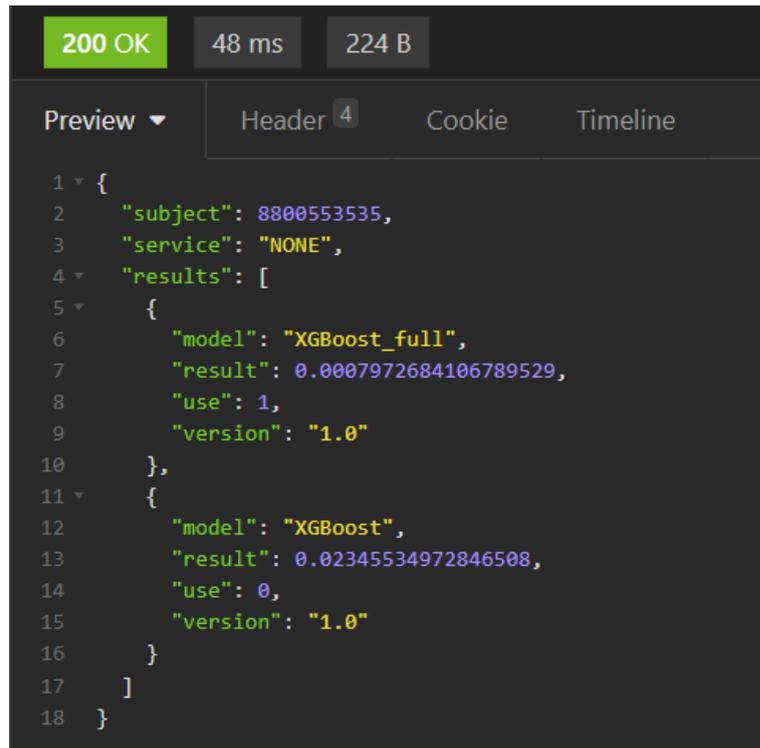


```

POST http://localhost:8080/trade_acquiring/execmodel
JSON
{
  "subject": 8800553535,
  "TRADE_STATUS": null,
  "NAME": "ООО \"СЦ БГБ\"",
  "KAPITAL": 10000.0,
  "CHANEL": "Лидогенерация",
  "ACC_COUNT": 1,
  "CURRENT_TARIFF": "[\"xSTART\"]",
  "PRIMARY_OKVED": "62.09",
  "BOSS_AGE": 35.0,
  "BOSS_GENDER": "М",
  "FRM_PROP": "000",
  "OKVEDS_NUMBER": 1,
  "QR": null,
  "CITY_NAME_OPEN": "ekb",
  "AGR_SUM_DEBET": 0.0,
  "AGR_COUNT_DEBET": 0.0,
  "SUM_10K_UNDER_30_0": 0.0,
  "AGR_SUM_IP": 0.0,
  "MEAN_DT_30_0": 0.0,
  "AGR_SUM_KREDIT": 0.0,
  "KT_SUM_30_0": 0.0,
  "AGR_COUNT_AGENT_KT": 0.0,
  "AGR_SUM_OOO": 0.0,
  "AGR_DT_EMPTY_DAYS": 0.0,
  "DT_SUM_30_0": 0.0,
  "SUM_NDFL": 0.0,
  "AGR_KT_EMPTY_DAYS": 0.0,
  "REGION_OPEN": "ekb",
  "CL_TAX": "[\"USN06\"]",
  "HISTORY_TARIFFS": "[]",
  "CL_CATEGORY": "нет",
  "CNT_PRODUCT": 2,
  "CNT_FOUNDERS": 1,
  "UPD_ANK_REQUEST": 0,
  "NEW_USER_REQUEST": 0,
  "COUNT_KORP_CARD": 0
}
  
```

Рисунок 1 Отправка запроса на расчет рекомендации

После обработки запроса и обращения к информационной системе хранения признаков клиентов (в данном случае после использования полученных данных) в ответе на запрос будет выведено сообщение с ответом, полученным от сервиса.



```
200 OK 48 ms 224 B
Preview ▾ Header 4 Cookie Timeline
1 {
2   "subject": 8800553535,
3   "service": "NONE",
4   "results": [
5     {
6       "model": "XGBoost_full",
7       "result": 0.0007972684106789529,
8       "use": 1,
9       "version": "1.0"
10    },
11   {
12     "model": "XGBoost",
13     "result": 0.02345534972846508,
14     "use": 0,
15     "version": "1.0"
16   }
17 ]
18 }
```

Рисунок 2. Ответ сервиса

Прохождение запросов на всех этапах логируется в стандартном потоке вывода.

Расчет рекомендаций продуктов может быть проведен в двух вариантах:

- **Рекомендации для клиента на входе (без платежных данных):** позволяет осуществить расчет продуктовых рекомендаций для клиента, даже если он не совершил еще ни одной транзакции.
- **Рекомендации для клиента с платежными данными:** позволяет осуществить расчет продуктовых рекомендаций для клиента, который уже совершил хотя бы один платеж (данный вариант более точный чем предыдущий).

## Типы запросов к сервису

### *Рекомендации для клиента на входе (без платежных данных)*

Для проведения проверки по этому варианту нужно заполнить данные клиента в формате JSON и отправить запрос в сервис расчета рекомендаций. Состав данных зависит от того, сделал ли уже клиент хотя бы одну транзакцию.

Минимальный состав данных запроса без платежных признаков для проверки:

Поле	Описание	Тип
subject	ИНН клиента	Число
NAME	Наименование ЮЛ	Строка
PRIMARY_OKVED	Основной ОКВЭД	Строка
FRM_PROP	Форма собственности	Строка
CURRENT_TARIFF	Текущий тариф РКО	Строка
OKVEDS_NUMBER	Количество кодов ОКВЭД	Число
TRADE_STATUS	Статус торгового сэквайринга	Строка

Данное сообщение отправляется в формате JSON и имеет следующий вид:

```
{
  "subject": 8800553535,
  "NAME": "ООО \"СЦ БГБ\"",
  "PRIMARY_OKVED": "62.09",
  "FRM_PROP": "ООО",
  "CURRENT_TARIFF": "[\"xSTART\"]",
  "OKVEDS_NUMBER": 1,
  "TRADE_STATUS": null
}
```

Ответное сообщение приходит также в формате JSON и содержит следующие переменные:

Поле	Описание	Тип
subject	Идентификатор клиента (ИНН)	Число
service	В каком статусе находится услуга	Строка
model	Название используемой модели	Строка
result	Результат расчета моделью	Число
use	Признак использования данного результата	Число
version	Версия модели	Строка

В зависимости от состава признаков каждая из моделей может сообщить, какого признака ей не хватает для обсчета рекомендации для данного клиента в случае недостаточности данных. В данном примере, признаков, передаваемых сервису, хватает только для модели для клиентов на входе, поэтому сервис возвращает вероятность интереса клиента к продукту только по результатам расчета одной модели, а по второй модели он говорит какого признака не хватает для расчета.

Пример ответного сообщения:

```
{
  "subject": 8800553535,
  "service": "NONE",
  "results": [
    {
      "model": "XGBoost_full",
      "result": "error: 'KAPITAL'",
      "use": 0,
      "version": "1.0"
    },
    {
      "model": "XGBoost",
      "result": 0.5927429795265198,
      "use": 1,
      "version": "1.0"
    }
  ]
}
```

### *Рекомендации для клиента с платежными данными*

Для проведения проверки по этому варианту, нужно заполнить следующие данные:

Поле	Описание	Тип
subject	ИНН клиента	Число
NAME	Наименование ЮЛ	Строка
QR	Дата подключения оплаты QR кодом	Строка
CITY_NAME_OPEN	Город открытия счёта в банке	Строка
PRIMARY_OKVED	Основной ОКВЭД	Строка
COUNT_KORP_CARD	Кол-во корп. карт у клиента	Число
CORP_CARD	Дата первой исходящей транзакции по корп. карте	Строка
AGR_SUM_DEBET	Среднемесячная сумма исходящих платежей за выбранный период	Число
UPD_ANK_REQUEST	Есть ли у клиента заявка на обновление анкеты	Число
FRM_PROP	Форма собственности	Строка
TRAFFIC_LIGHT	Дата подключения услуги Светофор	Строка

AGR_COUNT_DEBET	Среднемесячное кол-во исходящих платежей за выбранный период	Число
INVOICE	Дата подключения услуги выставления счетов	Строка
CHANEL	Канал привлечения	Строка
CURRENT_TARIFF	Текущий тариф РКО	Строка
CL_TAX	Система налогообложения	Строка
CORP_CARD_IS_ACTIVE	Активна ли корп. карта (есть ли транзакции за последние 90 дней)	Число
BOSS_AGE	Возраст руководителя	Число
NEW_USER_REQUEST	Есть ли у клиента заявка на добавление нового пользователя	Число
SUM_10K_UNDER_30_0	Сумма списаний со счета платежей меньше 10 тыс. за период от <дата последней транзакции - 30 дней> до <дата последней транзакции>	Число
AGR_SUM_IP	Среднемесячное количество платежей по контрагентам ИП за выбранный период	Число
CNT_PRODUCT	Количество услуг по договору ДБО (отключённые услуги тоже учитываются)	Число
REQ_UFE	ID последнего запроса (УФЭ)	Число
MEAN_DT_30_0	Среднее количество исходящих платежей в день за период от <дата последней транзакции - 30 дней> до <дата последней транзакций>	Число
AGR_SUM_KREDIT	Среднемесячная сумма входящих платежей за выбранный период	Число
HISTORY_TARIFFS	Тарифы РКО, которые использовал клиент за все время существования компании. Без текущего и корпоративных карт	Строка
KAPITAL	Уставной капитал	Число
BOSS_GENDER	Пол руководителя	Строка
REGION_OPEN	Регион открытия счёта в банке	Число
KT_SUM_30_0	Сумма входящих платежей за период от <дата последней транзакции - 30 дней> до <дата последней транзакций>	Число
AGR_COUNT_AGENT_KT	Среднемесячное кол-во контрагентов, от которых приходили платежи	Число
AGR_SUM_OOO	Среднемесячное количество платежей по контрагентам ООО	Число

AGR_DT_EMPTY_DAYS	Среднемесячное кол-во дней без исходящих платежей	Число
DT_SUM_30_0	Сумма исходящих платежей за период от <дата последней транзакции - 30 дней> до <дата последней транзакций>	Число
CL_CATEGORY	Наименование категории клиента	Строка
SUM_NDFL	Сумма платежей НДС. Смотрится по КБК кодам, но они могут меняться из года в год	Число
AGR_KT_EMPTY_DAYS	Среднемесячное кол-во дней без входящих платежей	Число
CNT_FOUNDERS	Число учредителей	Число
ACC_COUNT	Количество всех финансовых счетов	Число
TRADE_STATUS	Статус торгового эквайринга	Строка

Ответное сообщение приходит в формате JSON и его формат аналогичен предыдущему случаю. Пример ответного сообщения:

```
{
  "subject": 8800553535,
  "service": "NONE",
  "results": [
    {
      "model": "XGBoost_full",
      "result": 0.0008374840253964067,
      "use": 1,
      "version": "1.0"
    },
    {
      "model": "XGBoost",
      "result": "error: 'OKVEDS_NUMBER'",
      "use": 0,
      "version": "1.0"
    }
  ]
}
```

Для доступа к API сервиса требуется авторизация. Авторизация производится по ключу (токену) доступа, передаваемому в каждом запросе. Данный токен передается в заголовке запроса, имеющий следующий вид:

```
{
  "Content-Type": "application/json"
  "X-Auth-Token": "4CE7B412-49B7-3DCF-B56D-3441B6A3698A"
}
```

В ответном JSON'е в полях result передаются не только результаты расчетов моделей, но и ошибки, которые могут возникнуть при обработке запросов. К примеру, когда не хватает некоторых данных для расчета рекомендаций, сервис сообщит какие данные необходимо добавить к запросу:

```
{
  "subject": 8800553535,
  "service": "NONE",
  "results": [
    {
      "model": "XGBoost_full",
      "result": "error: 'CITY_NAME_OPEN'",
      "use": 0,
      "version": "1.0"
    },
    {
      "model": "XGBoost",
      "result": 0.21598204970359802,
      "use": 1,
      "version": "1.0"
    }
  ]
}
```

Если же ошибка не позволяет ни одной из двух моделей предоставить результат расчета, то сервис вернет ответ с текстом ошибки в следующем формате:

```
{
  "subject": 8800553535,
  "results": "-1",
  "service": "NONE",
  "ERROR": "KeyError('NAME'); KeyError('NAME')"
}
```

В примере выше видно, что первой и второй модели не хватает поля NAME для расчета рекомендаций конкретному клиенту.

Список реализованных методов API:

- **/trade\_acquiring/execmodel** – POST-запрос на получение рекомендации банковского продукта клиентом;
- **/trade\_acquiring/ping** – GET-запрос быстрой проверки работоспособности сервиса;
- **/trade\_acquiring/healthcheck** – GET-запрос служебной проверки жизнеспособности сервиса службами kubernetes;
- **/trade\_acquiring/metrics** – GET-запрос на получение метрик сервиса рекомендаций для построения графиков сервисом мониторинга.