

# Оформление полиса ОСАГО

---

## Краткое описание порядка действий

### 1. Генерация запроса на оформление полиса

Генерация запроса по введенным данным(данные т/с, страхователя, водителей и т.д.).

#### 1.1 Получение списка страховых компаний для расчёта полиса

В ответе приходит список страховых компаний, через которые можно оформить полис (относительная ссылка на логотип страховой компании и calculationId, необходимый для дальнейших расчётов) либо сообщение об ошибке. Так же приходит id запроса.

### 2. Запрос на оформление полиса у выбранной страховой компании

Запрос на оформление полиса по calculationId выбранной страховой, полученного из п.1.

Ответ зависит от страховой. Процесс может проходить в несколько этапов.

### 3. Проверка оплаты

Запрос на проверку оплаты полиса.

### 4. Получение полиса

Получение полиса в формате PDF.

---

## Подробное описание

Все запросы должны иметь следующие заголовки:

Заголовок	Значение	Описание
Content-Type	application/json	
X-Authorization	{token}	Ваш ключ API

# 1. Генерация запроса на оформление полиса

POST /api/osago/make

Запрос:

Параметр	Тип	Обязательный	Значение
driversUnlimited	boolean	Да	Ограниченное/неограниченное кол-во водителей (true, false (1, 0))
insurerIsOwner	boolean	Да	Страхователь является собственником (true, false (1, 0))
followToRegistration	boolean	Да	true, false (1, 0)
startDate	string	Да	Дата начала страхования
period	int	Да	Период действия полиса
<b>vehicle</b>	<b>объект</b>	<b>Да</b>	
licencePlate	string	Да	Гос. номер ТС (B111BB11)
make	string	Да	Марка авто (см. п.1.1)
model	string	Да	Модель авто (см. п.1.1)
modelPTS	string	Да	Модель по ПТС
maxMass	int	Да	Максимальная масса
unladenMass	int	Да	Масса без нагрузки
passQuantity	int	Да	Кол-во пассажиров
vin	string	Да (один из трех идентификаторов)	Vin
yearIssue	string	Да	Год выпуска
power	float	Да	Мощность двигателя

purposeOfUse	string	Да	<ul style="list-style-type: none"> <li>• Personal - Личная</li> <li>• RidingTraining - Учебная езда</li> <li>• Taxi - Такси</li> <li>• RoadVehicles - Дорожные и специальные ТС</li> <li>• Others - Прочие</li> <li>• PassengerService - Регулярные/по заказам пассажирские перевозки</li> <li>• DangerousCargo - Перевозка опасных и легко воспламеняющихся грузов</li> <li>• Emergency - Экстренные и коммунальные службы</li> <li>• Rent - Прокат/Краткосрочная аренда</li> </ul>
bodyNumber	string	Да (один из трех идентификаторов)	Номер кузова
chassis	string	Да (один из трех идентификаторов)	Номер шасси
documentType	int	Да	<ul style="list-style-type: none"> <li>• 30: ПТС</li> <li>• 31: СРТС</li> <li>• 32: ПСМ</li> </ul>
docSerial	string	Да	Серия документа
docNumber	string	Да	Номер документа
docDate	string	Да	Дата выдачи документа
dkNumber	string	Да	Номер ДК
dkFrom	string	Да	Дата начала действия ДК
dkTo	string	Да	Дата окончания действия ДК

useTrailer	boolean	Да	Использование с прицепом (true, false / 1, 0)
identifier	string	Да	Значение указанного идентификатора ТС (vin или bodyNumber, или chassis - в порядке важности, если указаны несколько)
isNew	boolean	Да	Новое ТС (true, false / 1, 0)
company	объект	Да	марки и модели авто для подмены по каждой страховой (см. Справочники)
fullCategory	string	Да	название категории ( 'А - мотоциклы': 1, 'В - легковые': 2, 'С - грузовые': 3, 'D - автобусы': 4, 'Е - тракторы': 5, 'М - мотоциклы': 6, 'Тм - трамвай': 7, 'Тв - троллейбус': 8, 'Прицеп': 9, '(не определена)': 10 )
<b>insurer</b>	<b>объект</b>	<b>Да</b>	<b>Страхователь</b>
lastName	string	Да	Фамилия
firstName	string	Да	Имя
middleName	string	Да	Отчество
birthday	string	Да	Дата рождения
address	string	Да	Полный адрес одной строкой. Принимаются только полные адреса, на основе формата, предоставляемым сервисом <a href="http://DaData.ru">DaData.ru</a> (подсказки по адресам).  Пример:  “Московская обл, г Одинцово, ул Вокзальная, д 37, кв 1“

pasSerial	string	Да	Серия паспорта
pasNumber	string	Да	Номер паспорта
phone	string	Да	Номер телефона
email	string	Да	E-mail
pasDate	string	Нет	Дата выдачи паспорта
pasIssued	string	Нет	Кем выдан
sex	string	Нет	Пол (“М“, “F“)
index	string	Да	Индекс по адресу
<b>owner</b>	<b>объект</b>	<b>Да</b>	
lastName	string	Да	Фамилия
firstName	string	Да	Имя
middleName	string	Да	Отчество
birthday	string	Да	Дата рождения
address	string	Да	<p>Полный адрес одной строкой. Принимаются только полные адреса, на основе формата, предоставляемым сервисом <a href="http://DaData.ru">DaData.ru</a> (подсказки по адресам).</p> <p>Пример:</p> <p>“Московская обл, г Одинцово, ул Вокзальная, д 37, кв 1“</p>
pasSerial	string	Да	Серия паспорта
pasNumber	string	Да	Номер паспорта
pasDate	string	Нет	Дата выдачи паспорта
pasIssued	string	Нет	Кем выдан
index	string	Да	Индекс
<b>drivers</b>	<b>массив объектов</b>	<b>Да</b>	
lastName	string	Да	Фамилия
firstName	string	Да	Имя
middleName	string	Да	Отчество

birthday	string	Да	Дата рождения (01.01.1999)
licenceSerial	string	Да	Серия ВУ
licenceNumber	string	Да	Номер ВУ
licenceDate	string	Да	Дата выдачи ВУ
experienceStart	string	Да	Дата начала стажа
prevPolicyNum	string	Нет	Номер предыдущего ВУ
prevPolicySer	string	Нет	Серия предыдущего ВУ
sex	string	Нет	Пол ("М", "F")

## **1.1 Получение списка страховых компаний для расчёта полиса**

Пример запроса

```
{
  "vehicle": {
    "licensePlate": "Й679PP55",
    "make": "RENAULT",
    "model": "CAPTUR",
    "modelPTS": "",
    "maxMass": "",
    "unladenMass": "",
    "passQuantity": "",
    "vin": "X7LAHHHH656021105",
    "yearIssue": "2016",
    "power": "143.03",
    "purposeOfUse": "Personal",
    "bodyNumber": "",
    "chassis": "",
    "documentType": "30",
    "docSerial": "77PP",
    "docNumber": "443911",
    "docDate": "23.09.2016",
    "dkNumber": "067150010000000",
    "dkFrom": "03.10.2019",
    "dkTo": "03.10.2021",
    "useTrailer": "0",
    "identifier": "X7LAHHHH656021105",
    "isNew": "0",
    "company": [
      {
        "insuranceCompanyId": 1,
        "mark": "RENAULT",
        "model": {
          "insuranceCompanyId": "1",
          "mark": "RENAULT",
          "model": "CAPTUR",
          "markId": "",
          "modelId": ""
        }
      }
    ]
  }
}
```

```

        "category": "В - легковые",
        "addData": "Легковой а/м"
    },
    "markId": "",
    "modelId": "",
    "category": "В - легковые",
    "addData": "Легковой а/м"
},
{
    "insuranceCompanyId": 14,
    "mark": "renault",
    "model": {
        "insuranceCompanyId": "14",
        "mark": "renault",
        "model": "CAPTUR",
        "markId": "197",
        "modelId": "4150",
        "category": "В - легковые",
        "addData": ""
    },
    "markId": "197",
    "modelId": "4150",
    "category": "В - легковые",
    "addData": ""
}
//и так по всем страховым...
],
"fullCategory": "В - легковые"
},
"insurer": {
    "lastName": "Иванов",
    "firstName": "Иван",
    "middleName": "Иванович",
    "birthday": "27.09.1964",
    "phone": "+7 (123) 123-12-45",
    "email": "somemail@gmail.com",
    "address": "г Калуга, ул Зеленая, д 52, кв 17",
    "pasNumber": "000000",
    "pasSerial": "0000",
    "pasIssued": "РФ УФМС",
    "pasDate": "09.10.2009",
    "sex": "М",
    "index": "248025"
},
"owner": {
    "lastName": "Петров",
    "firstName": "Пётр",
    "middleName": "Петрович",
    "birthday": "10.01.1991",
    "address": "г Калуга, ул Зеленая, д 52, кв 17",
    "pasNumber": "111111",
    "pasSerial": "1111",
    "pasIssued": "РФ УФМС",
    "pasDate": "11.02.2011",
    "sex": "М",
    "index": "248025"
},

```

```
"startDate": "25.12.2019",
"period": "12",
"followToRegistration": "0",
"insurerIsOwner": "0",
"driversUnlimited": 0,
"drivers": [
  {
    "lastName": "Сидоров",
    "firstName": "Сидор",
    "middleName": "Сидорович",
    "birthday": "27.09.1964",
    "experienceStart": "11.01.2000",
    "CatDriverLicense": 2531,
    "licenseNumber": "111111",
    "licenseSerial": "40УМ",
    "prevPolicyNum": "",
    "prevPolicySer": "",
    "licenseDate": "21.01.2011",
    "sex": "M"
  }
]
}
```

## Пример ответа

```
{
  "success": true,
  "data": {
    "drivers": "1 водитель",
    "period": "12 месяцев",
    "companies": [
      {
        "image": "/img/alpha_logo.svg",
        "calculationId": 4102465
      },
      {
        "image": "/img/ingos_logo.svg",
        "calculationId": 4102466
      },
      {
        "image": "/img/company/rgs_logo.svg",
        "calculationId": 4102467
      },
      {
        "image": "/img/company/osk_logo.svg",
        "calculationId": 4102468
      },
      {
        "image": "/img/company/renessanse_logo.svg",
        "calculationId": 4102469
      },
      {
        "image": "/img/company/tinkoff_logo.svg",
        "calculationId": 4102470
      }
    ]
  }
},
```



```
    "requestId": 516165
  }
}
```

Пример ответа с ошибкой

```
{
  "success": false,
  "text": "Текст ошибки"
}
```

## **2. Запрос на оформление полиса у выбранной страховой компании**

POST /api/osago/unifiedInsuranceRequest

Запрос

Параметр	Тип	Обязательный	Значение
calculationId	int	Да	calculationId из прошлого запроса
smsCode	string	Нет	СМС-код подтверждения
repeat	boolean	Нет	В случае получения ошибки в процессе оформления полиса нужно отправить значение true , чтобы повторить всю процедуру с теми же данными заново.

Ответ

Параметр	Описание
success	true/false
message	текст сообщения при неудачном запросе ( success: false )
payLink	При успешном запросе здесь будет ссылка на оплату
price	Цена за полис
reward	Размер вашего КВ относительно цены полиса
coefData	Объект. Данные по коэффициентам данной страховой ( может быть null или как в примере ответа ниже )
car	Объект. В нём marki model( соответственно марка и модель ТС )

status	3 цифры (каждая может быть 0 или 1):  1-я цифра (1xx) - флаг успешного расчёта  2-я цифра (x1x) - признак изменения даты начала действия договора (параметр <code>newDate</code> )  3-я цифра (xx1) - признак того, что в следующем вызове <code>unifiedInsuranceRequest</code> нужно отправить СМС-код подтверждения в поле <code>smsCode</code>
delay	Количество секунд задержки перед следующим вызовом метода <code>unifiedInsuranceReques</code>
newDate	Дата начала действия полиса
highKV	Флаг (1 или 0) - признак пролонгации полиса (да/нет)
isShow	Флаг (1 или 0) - рекомендация отображения данной СК в списке результатов расчёта (да/нет)
calculationId	id расчета. Необходим для дальнейших действий
rewardTypeId	Параметр для нашего внутреннего использования
rand	Параметр для нашего внутреннего использования

Пояснения по поводу работы метода `unifiedInsuranceRequest`:

Если получено значение поля `success = false` - завершаем процедуру оформления полиса и показываем сообщение об ошибке `message` либо (в зависимости от ошибки) вызываем метод `unifiedInsuranceRequest` со значением поля `repeat = true`, чтобы ещё раз повторить процедуру оформления полиса с теми же данными. Далее рассмотрим случаи, когда `success = true`:

А. `status = "1xx"` - флаг успешного расчёта. Приостанавливаем процедуру оформления, показываем сумму расчёта `price` и коэффициенты расчёта `coefData`, если они есть (т.е. не `null`). Для продолжения процедуры оформления ещё раз вызываем метод `unifiedInsuranceRequest`.

Б. `payLink` не содержит пустую строку - завершаем процедуру оформления полиса и показываем содержимое `payLink` (т.е. ссылку на оплату)

В. `status = "11x"` - флаг успешного расчёта с автоматически изменённой датой начала действия договора (если она указана неправильно или, например, пролонгация). Приостанавливаем процедуру оформления, показываем сумму расчёта `price` и коэффициенты расчёта `coefData`, если они есть (т.е. не `null`), а также показываем изменённую дату `newDate`.

Г. status = "000" (значение поля по умолчанию) - вызываем ещё раз unifiedInsuranceRequest с дилеем delay (в секундах)

Д. status = "xx1" - приостанавливаем процедуру оформления и ожидаем ввода СМС-кода пользователем. Далее вызываем метод unifiedInsuranceRequest с записанным значением СМС-кода в поле smsCode.

### Пример запроса

```
{
  "calculationId": 7897894,
  "smsCode": "123456",
  "repeat": false
}
```

### Пример ответа

```
{
  "success": true,
  "message": "",
  "payLink": "",
  "price": "5645.28",
  "reward": 0,
  "coefData": {
    "tb": 4118,
    "kt": 1.2,
    "kp": 1,
    "kvs": 0.95999999999999996,
    "kbm": 0.84999999999999998,
    "km": 1.3999999999999999,
    "kn": 1,
    "ko": 1,
    "kpr": 1,
    "ks": 1
  },
  "car": {
    "mark": "RENAULT",
    "model": "Captur"
  },
  "status": "100",
  "delay": 0,
  "newDate": "25.12.2019",
  "highKV": 0,
  "isShow": true,
  "calculationId": 4102473,
  "rewardTypeId": 61,
  "rand": 7624481
}
```

### Пример ответа с ошибкой

```
{
  "success": false,
```

```
"text": "Ошибка: Не удалось выполнить расчёт по введённым данным"
}
```

### **3. Проверка оплаты**

POST /api/osago/check\_payment

Параметр	Тип	Обязательный	Значение
requestId	int, массив int	Да	id запроса

Ответ (если передан один requestId):

```
{
  "success": true,
  "requestId": 777, // запрашиваемый requestId
  "isPaid": true, // статус оплаты
  "insuranceCompany": "Тинькофф Страхование", // наименование СК
  "insuranceCompanyId": 8, // id СК
  "rewardTypeId": 13, // id КВ
  "price": 3632.05, // цена полиса
  "reward": 1090 // размер КВ
}
```

Ответ (ошибка):

```
{
  "success": true,
  "requestId": 777, // запрашиваемый requestId
  "text": "Текст ошибки"
}
```

Ответ (если передан массив):

```
{
  "success": true,
  "data": [
    {
      "requestId": 777, // запрашиваемый requestId
      "text": "Текст ошибки"
    },
    {
      "requestId": 77777, // запрашиваемый requestId
      "isPaid": true, // статус оплаты
      "insuranceCompany": "Тинькофф Страхование", // наименование СК
      "insuranceCompanyId": 8, // id СК
      "rewardTypeId": 13, // id КВ
      "price": 3632.05, // цена полиса
      "reward": 1090 // размер КВ
    },
    {
      "requestId": 77, // запрашиваемый requestId

```

```
        "isPaid": false // статус оплаты
    }
  ]
}
```

## **4. Получение полиса**

POST /api/osago/download

Параметр	Тип	Обязательный	Значение
requestId	int	Да	id запроса

Ответ:

```
{
  "success": true,
  "data": "base64 pdf",
}
```

---

## **Справочники**

**Все запросы должны иметь следующие заголовки:**

Заголовок	Значение	Описание
Content-Type	application/json	
X-Authorization	{token}	some token

## **1. Поиск марки**

Адрес: <https://carcatalog.eaisto.info/search/mark>

Метод: POST JSON

**Параметры запроса:**

Параметр	Тип	Обязательный	Значение
----------	-----	--------------	----------

mark	string	Да	Входная строка с названием марки
insuranceCompanyId	int	Нет	Id страховой компании (для вывода подходящих марок только для указанной СК)
category	int	Нет	Категория <ol style="list-style-type: none"> <li>1. А - мотоциклы</li> <li>2. В - легковые</li> <li>3. С - грузовые</li> <li>4. D - автобусы</li> <li>5. Е - тракторы</li> <li>6. М - мотоциклы</li> <li>7. Тm - трамвай</li> <li>8. Тb - троллейбус</li> <li>9. Прицеп</li> <li>10. (не определена)</li> </ol>

### Пример запроса:

```
{
  "mark": "BA3"
}
```

### Пример ответа:

```
{
  "result": [
    "BA3",
    "BA3 (LADA) "
  ]
}
```

### Пример ответа с ошибкой:

```
{
  "error": "Не задан параметр mark"
}
```

## 2. Поиск модели по марке

Адрес: <https://carcatalog.eaisto.info/search/model>

Метод: **POST JSON**

**Параметры запроса:**

Параметр	Тип	Обязательный	Значение
mark	string	Да	Марка
model	string	Да	Входная строка с названием модели
insuranceCompanyId	int	Нет	Id страховой компании (для вывода подходящих моделей только для указанной СК)
category	int	Нет	Категория <ol style="list-style-type: none"><li>1. А - мотоциклы</li><li>2. В - легковые</li><li>3. С - грузовые</li><li>4. D - автобусы</li><li>5. Е - тракторы</li><li>6. М - мотоциклы</li><li>7. Тm - трамвай</li><li>8. Тb - троллейбус</li><li>9. Прицеп</li><li>10. (не определена)</li></ol>

**Пример запроса:**

```
{
    "mark": "ВАЗ",
    "model": "2107"
}
```

**Пример ответа:**

```
{
  "result": [
    {
      "model": "2107",
      "category": "В - легковые"
    },
    {
      "model": "21070",
      "category": "В - легковые"
    }
  ]
}
```

Пример ответа с ошибкой:

```
{
  "error": "Не задан параметр mark и/или model"
}
```

### **3. Марка и модель для каждой СК**

Адрес: <https://carcatalog.eaisto.info/search/markModel>

Метод: **POST JSON**

**Параметры запроса:**

Параметр	Тип	Обязательный	Значение
mark	string	Да	Марка
model	string	Да	Модель



category	int	Да	Категория  1. А - мотоциклы 2. В - легковые 3. С - грузовые 4. D - автобусы 5. Е - тракторы 6. М - мотоциклы 7. Тm - трамвай 8. Тb - троллейбус 9. Прицеп 10. (не определена)
----------	-----	----	--

**Пример запроса:**

```
{  
  "mark": "BA3",  
  "model": "2107",  
  "category": "2"  
}
```

**Пример ответа:**

```
{
  "result": [
    {
      "insuranceCompanyId": 2,
      "mark": "BA3",
      "model": "2107",
      "replaced": false
    },
    {
      "insuranceCompanyId": 1,
      "mark": "BA3",
      "model": "2107",
      "replaced": false
    },
    {
      "insuranceCompanyId": 8,
      "mark": "BA3",
      "model": "2107",
      "replaced": false
    },
    {
      "insuranceCompanyId": 9,
      "mark": "BA3",
      "model": "2107",
      "replaced": false
    },
    {
      "insuranceCompanyId": 11,
      "mark": "BA3",
      "model": "21074",
      "replaced": true
    },
    {
      "insuranceCompanyId": 4,
      "mark": "BA3/Lada",
      "model": "2107",
      "replaced": true
    }
  ]
}
```

Пример ответа с ошибкой:

```
{
  "error": "Не задан параметр mark и/или model, и/или category"
}
```

## Порядка действий получения подмены марок и моделей

### ШАГ 1

Получаем основной массив марок:

Отправляем:

<https://carcatalog.eaisto.info/search/mark>

пустой POST запрос

Ответ:

```
[
  "AUDI",
  "BMW",
  "CADILLAC",
  "CHERY",
  ....
]
```

## ШАГ 2

Получаем основной массив моделей:

Отправляем:

<https://carcatalog.eaisto.info/search/model>

POST запрос:

```
{
  mark:"AUDI"
}
```

Ответ:

```
[
  {
    model: "100 AVANT"
    category: "B - легковые"
    markId: null
    modelId: null
    addData: "Л"
  },
  ....
]
```

## ШАГ 3

При выборе основной модели получаем марка и модель для каждой СК

Отправляем:

<https://carcatalog.eaisto.info/search/markModel>

POST запрос:

```
{
  mark: "AUDI",
  model: "100 AVANT",
  category: 2
}
```

Ответ:

```
[
  {
    insuranceCompanyId: 10,
    mark: "AUDI",
    model: "100 AVANT",
    markId: "",
    modelId: "20656",
  }
]
```

```
    category: "В - легковые",
    addData: "Легковой а/м",
    replaced: false
  },
  ....
]
```

**Данный ответ заносим в поле "company" основного запроса расчёта полиса**

**replaced = false**, значит у СК марка и модель найдена  
(никаких действий не требуется)

#### ШАГ 4

Если **replaced = true**, необходимо делать подмену марки и модели  
Пример ответа для подмены:

```
{
  insuranceCompanyId: 2,
  mark: "AUDI",
  model: "",
  markId: "",
  modelId: "",
  category: "",
  addData: "",
  replaced: true
}
```

Получаем массив марок для данной СК:

Отправляем:

<https://carcatalog.eaisto.info/search/mark>

POST запрос:

```
{
  insuranceCompanyId: 2,
  category: 2
}
```

*"category" - берем из таблицы "Поиск марки"*

Ответ:

```
[
  "AUDI",
  "BMW",
  "CADILLAC",
  "CHERY",
  ....
]
```

#### ШАГ 5

Получаем массив моделей СК:

Отправляем:

<https://carcatalog.eaisto.info/search/model>

POST запрос:

```
{
  insuranceCompanyId: 2,
  mark: "AUDI",
  category: 2
}
```

Ответ:

```
[
  {
    model: "80",
    category: "B - легковые",
    markId: null,
    modelId: "12561716",
    addData: null
  },
  ....
]
```

**Выбранный вариант из массива подменяем с ответом "ШАГ 3"**