Telematics API

TotalApi SDK

# **Описание функций**

1. CRUD Repository API’s - [**Devices**](#Device)
2. Регистрация подсистем и выдача информации о доступных подсистемах.
3. Модуль сообщений об изменениях устройств, их телематических данных и текущего статуса.

# **Бизнес-объекты**

|  |  |  |
| --- | --- | --- |
| DeviceIdentifier | | |
| **Свойство** | **Описание** | **Тип** |
| Id | Значение идентификатора.  Имеет смысл в зависимости от типа идентификатора (например, это может быть номер телефона, MAC-адрес или IMEI). | string |
| Type | Тип идентификатора  **DbId** = **0**, *// Идентификатор в базе данных*  **PhoneNumber** = **1**, *// Номер телефона сим-карты*  **Imei** = **2**, *// IMEI GPRS-модуля устройства*  **OwnId** = **4**, *// Внутренний идентификатор устройства*  **Mac = 8,** *// MAC-адрес сетевой карты устройства* | DeviceIdentifierTypes |

|  |  |  |
| --- | --- | --- |
| DeviceModel | | |
| **Свойство** | **Описание** | **Тип** |
| Code | Код модели (см. Примечание 1).  Каждой модели, поддерживаемой системой присваивается уникальный числовой код (моделям одного производителя присваивается непрерывный числовой диапазон). | int |
| Name | Наименование модели | string |
| IdentifierTypes | Поддерживаемые типы идентификаторов модели | DeviceIdentifierTypes |
| SensorPorts | Порты сенсоров, поддерживаемые данной моделью | SensorPort[] |
| Url | Адрес обработчика данных с устройства этой модели | string |
| SetupSmsCommand | SMS-команда для настройки отправки данных данной моделью устройства.  Если пусто - считается, что такой команды нет. | string |

|  |  |  |
| --- | --- | --- |
| Device | | |
| **Свойство** | **Описание** | **Тип** |
| Id | Идентификатор устройства | string |
| Name | Наименование устройства | string |
| PhoneNumber | Номер телефона сим-карты | string |
| Imei | IMEI GPRS-модуля устройства | string |
| Mac | MAC-адрес сетевой карты устройства | string |
| OwnId | Внутренний идентификатор устройства | string |
| TimeToLive | Время хранения данных, полученных от устройства (координаты, и т.д.) | TimeSpan |
| ModelCode | Код модели устройства | int |
| Data | Произвольные дополнительные данные | byte[] |

|  |  |  |
| --- | --- | --- |
| Coordinate | | |
| **Свойство** | **Описание** | **Тип** |
| UtcDate | Дата получения координаты в [UTC-формате](https://ru.wikipedia.org/wiki/Всемирное_координированное_время) | DateTime |
| Latitude | Широта в радианах | double |
| Longitude | Долгота в радианах | double |
| Altitude | Высота над уровнем моря в метрах | double |
| Velocity | Скорость в метрах в секунду | double |
| Direction | Азимут (направление относительно севера в градусах) | double |
| Valid | Признак достоверности данных:  **0** – данные недостоверны;  **1** – данные достоверны. | byte |

# **Сервисы**

# **Devices** **-** [репозиторий](http://drive.google.com/open?id=1Jc5qppdDK2kphknDF81mmf88XlP25TM6faTFxn4piZ0)[**Device**](#Device)*Требует* ***ApiKey*** *или* ***AppUser*** *авторизации.*

Методы интерфейсов репозитория стандартные.

# **Sensors -** [репозиторий](http://drive.google.com/open?id=1Jc5qppdDK2kphknDF81mmf88XlP25TM6faTFxn4piZ0) [**Sensor**](#Sensor) *Требует* ***ApiKey*** *или* ***AppUser*** *авторизации.*

Методы интерфейсов репозитория стандартные.

# **Telematics** *Требует авторизации.*

Сервис, отвечающий за работу с телематическим модулем.

|  |  |  |
| --- | --- | --- |
| FindDevice(…) WebApi GET: /FindDevice/{type}/{devId} | | |
| Ищет устройство по его идентификатору. | | |
| **Параметр** | **Описание** | **Тип** |
| type | Device identifier type. It is the text representation of the items of [DeviceIdentifierTypes](#DeviceIdentifierType). | string |
| devId | Identifier value according its type. | string |
| Результат | Найденное устройство или null, если устройство не найдено. | [Device](#Device) |

|  |  |  |
| --- | --- | --- |
| GetDeviceLastStatus(…) WebApi GET: /{Id} | | |
| Выдаёт текущий статус устройства по его внутреннему идентификатору БД. | | |
| **Параметр** | **Описание** | **Тип** |
| id | Внутренний идентификатор устройства слежения. | string |
| Результат | Текущий статус устройства или null, если устройство не найдено. | DeviceStatus |

|  |  |  |
| --- | --- | --- |
| GetDeviceIdentifierLastStatus(…) WebApi GET: /{type}/{devId} | | |
| Выдаёт текущий статус устройства по его идентификатору. | | |
| **Параметр** | **Описание** | **Тип** |
| type | Device identifier type. It is the text representation of the items of [DeviceIdentifierTypes](#DeviceIdentifierType). | string |
| devId | Identifier value according its type. | string |
| Результат | Текущий статус устройства или null, если устройство не найдено. | DeviceStatus |

|  |  |  |
| --- | --- | --- |
| GetDeviceModels(…) WebApi GET: /DeviceModels | | |
| Получение списка моделей устройств.  Список моделей генерируется динамически на основании плагинов-обработчиков данных от устройств.  При регистрации плагин сообщает какие модели устройств он обрабатывает. | | |
| **Параметр** | **Описание** | **Тип** |
| Результат | Список моделей. Если таких устройств не найдено - возвращает пустое перечисление. | [DeviceModel](#DeviceModel)[] |

|  |  |  |
| --- | --- | --- |
| ReadCoordinates(…) WebApi POST: /ReadCoordinates | | |
| Читает координаты указанного устройства. | | |
| **Параметр** | **Описание** | **Тип** |
| readParams | Параметры чтения координат | ReadCoordinatesParams |
| Результат | Список координат или пустое множество, если их нет. | CoordinatePoints |

|  |  |  |
| --- | --- | --- |
| WriteCoordinates(…) WebApi POST: /WriteCoordinates | | |
| Записывает координаты устройства.  Для ускорения работы метод ставит запись координат в очередь и сразу отдаёт управление.  Вместе с записью в таблицу координат производится запись в отдельную таблицу последнего состояния устройства. Эта информация нужна для быстрого получения текущего статуса устройства.  Более старые данные (по времени получения координат), пришедшие позже НЕ будут изменять текущий статус, если он имеет более новую дату. Данные из «будущего» (возможные глюки аппаратуры устройства слежения), отстоящие от текущего времени сервера более чем на 4 часа, также не будут записаны в последний статус (но будут записаны в историю координат). | | |
| **Параметр** | **Описание** | **Тип** |
| deviceId | Идентификатор устройства | [DeviceIdentifier](#DeviceIdentifier) |
| coordinates | Сохраняемые координаты | [Coordinate](#Coordinate)[] |
| Результат | Нет | [Device](#Device) |

# [**Events**](https://docs.google.com/document/d/1rTiW7ghno8E9AZKWC7dOzCQ8G8Cdsx87Hth66S08rF0/edit#/h) *Требует авторизации.*

* OnDeviceStatusChanged
* OnDeviceEventRaised

# **Работа с данными сенсоров**

Данные сенсоров будут сохранятся по аналогии с координатами. При сохранении данных внутренние номера портов данной модели устройства преобразуются в тип сенсора, известный телематическому модулю. Таблица сопоставления [**Sensors**](#_Sensor), задаётся пользователем при настройке параметров устройств слежения.

Данные сенсоров, порты которых сопоставлены, сохраняются в отдельную таблицу [**SensorsData**](#_SensorData) в **БД Cassandra**. Если данные не сопоставлены – они не будут сохраняться. Таким образом пользователь может легко управлять объемом сохраняемых данных. Если при сопоставлении оказывается, что мы используем на разных портах несколько датчиков одинакового типа – для каждого из них можно определить уникальный номер (по умолчанию номер равен 0).

Внешние датчики на одном порту могут передавать данные по нескольким адресам – для разграничения этих данных используется поле Address. Если датчик не разделяет передачу данных по адресам – значение поля **Address** считается равным 0.

**Пример**:

По порту 1 передаются данные о заряде батареи.

На порт 2 подключена тревожная кнопка 1.

На порт 3 подключена тревожная кнопка 2.

На порт 4 подключен ДУТ, передающий данные по адресам 1,2,5. На порт 8 подключен ДУТ, передающий данные по адресу 1. Чтобы принимать все эти данные необходимо зарегистрировать такие сенсоры:

(SensorType: Charge, Name: `Уровень заряда`, PortId: 1)

(SensorType: SOS, Name: `SOS 1`, Number: 1, PortId: 2)

(SensorType: SOS, Name: `SOS 2`, Number: 2, PortId: 3)

(SensorType: LLS, Name: `ДУТ1`, Number: 1, PortId: 4, Address: 1)

(SensorType: LLS, Name: `ДУТ2`, Number: 2, PortId: 4, Address: 2)

(SensorType: LLS, Name: `ДУТ3`, Number: 3, PortId: 4, Address: 5)

(SensorType: LLS, Name: `ДУТ4`, Number: 4, PortId: 8, Address: 1)

|  |  |  |
| --- | --- | --- |
| Sensor | | |
| **Свойство** | **Описание** | **Тип** |
| DeviceId | Внутренний идентификатор устройства слежения | string |
| Name | Наименование датчика | string |
| SensorType | Тип датчика | [SensorType](#_SensorType) |
| DataType | Тип данных датчика | [SensorDataType](#_SensorDataType) |
| Number | Номер датчика (для однотипных датчиков) | integer |
| PortId | Внутренний номер порта (определяется моделью устройства) | integer |
| Address | Адрес порта (используется для внешних датчиков) | integer |
| TimeToLive | Время хранения данных сенсора | TimeSpan |
| Threshold | Минимальное пороговое значение между двумя последовательными показаниями датчика при котором данные будут записаны в БД. Используется для отсеивания «дребезга». 0 – означает, что пишутся все поступающие данные, даже если они не изменились. Не имеет смысла для логических датчиков. *(Не реализовано – можно реализовать при помощи фильтров предобработки).* | float |
| KeyPeriod | Если запись значений не производилась в течении этого периода, то следующее пришедшее значение будет записано, даже если оно меньше [порогового значения](#Threshold). *(Не реализовано – можно реализовать при помощи фильтров предобработки).* | TimeSpan |
| SensorData | | |
| **Свойство** | **Описание** | Тип |
| Id | Внутренний идентификатор устройства слежения | string |
| UtcDate | Дата получения данных датчика в [UTC-формате](https://ru.wikipedia.org/wiki/Всемирное_координированное_время) | DateTime |
| SensorType | Тип датчика | [SensorType](#_SensorType) |
| Number | Номер датчика (для однотипных датчиков) | byte? |
| Value | Значение | any |
| SensorType | | |

|  |
| --- |
| Ниже приведён список предопределённых типов сенсоров, которые могут присутствовать у большинства пользователей. TotalApi ничего не знает и никак не отличает сенсоры разных типов в том числе и рекомендуемых. Рекомендация дана для общего представления и унификации известных типов сенсоров среди пользователей. Если тип вашего сенсора входит в эту таблицу – используйте значение из неё, если нет – используйте любое другое целочисленное значение, отличное от перечисленных. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Значение** | **Сенсор** | | **Тип** | **Описание** |
| 10 | SOS | | boolean | Тревожная кнопка |
| 4 | Power | | float | Значение напряжения питания. |
| 5 | Charge | | float | Уровень заряда батареи |
| 2 | GSM | | integer | Уровень сигнала GSM |
| 3 | IsPower | | boolean | Флаг внешнего питания |
| 1 | IsGPS | | boolean | Доступность сигнала GPS |
| 7 | IsGSM | | boolean | Регистрация в сети GSM |
| 8 | IsGPRS | | boolean | Регистрация в сети GPRS |
| 9 | IsSIM | | boolean | Наличие SIM карты |
| 6 | IsCharge | | boolean | Внутренний аккумулятор заряжается |
| 11 | SatCount | | integer | Количество спутников |
| 12 | AccAxesX | | integer | Значения оси акселерометра X |
| 13 | AccAxesY | | integer | Значения оси акселерометра Y |
| 14 | AccAxesZ | | integer | Значения оси акселерометра Z |
| 15 | Accuracy | | long | Точность |
| 51 | DUT | | integer | Значение датчика уровня топлива |
| SensorDataType | | | | |
| Поддерживаемые типы данных датчиков | | | | |
| **Тип** | **Значение** | **Описание** | | |
| Undefined | 0 | Неопределённый тип данных. | | |
| Float | 1 | Число с плавающей точкой | | |
| Integer | 2 | Целочисленное значение (от −2147483648 до 2147483648) | | |
| Long | 3 | Целочисленное значение (от −9223372036854775808  до 9223372036854775808) | | |
| Boolean | 4 | Логическое значение | | |
| Event | 5 | Логическое значение, сообщающее о наступлении события. | | |
| Object | 6 | Произвольный объект (сохраняется в JSON-формате) | | |

Принимаемые от датчиков устройств данные будут записаны только при изменении последнего значения свыше указанного [порогового значения](#Threshold). Может быть также настройка промежуточной записи значения даже если оно не изменилось, но приходило от устройства в [определённый промежуток времени](#KeyPeriod). (Не реализовано, т.к. может быть настроено при помощи фильтров предобработки, настраиваемыми пользователем).

По аналогии с координатами имеется понятие **“последнее значение датчика”** и соответствующая таблица.

Монитор состояния устройств помимо проверки, сохранения и уведомления об изменившемся состоянии устройства, проверяет, сохраняет и уведомляет о последнем значении датчиков. В отличие от координат, глобального уведомления об изменившихся значений датчиков нет.

Алгоритмы, проверяющие состояние устройства на основании данных последних координат, теперь могут дополнительно анализировать последние данные датчиков, как для более точной установки вычислимых состояний устройства (едет/стоит, есть связь со спутником/нет), так и для новых видов состояний, рассчитать которые можно только на основании данных сенсоров (нажата тревожная кнопка, произошло открытие/закрытие дверей, и т.д.)

# **Работа с событиями устройств**

Устройства слежения могут присылать уведомления о наступлении определённых событий. В отличие от данных датчиков – эти события не имеют никаких значений и не распределены во времени. Это просто констатация факта, произошедшего в указанное время.

События, приходящие от устройства слежения, преобразуются в известные нам типы событий и записываются в специальную таблицу событий. При наступлении события генерируется глобальное уведомление.