

# Документация по АПИ доступу к сайту Freesmetaonline

## Раздел 1. Авторизация

Авторизация происходит по логину и паролю, который вы получаете при авторизации в онлайн программе Freesmetaonline по ссылке <https://freesmetaonline.ru/login>

## Раздел 2. Генерация сметы с помощью ИИ (автоматический режим)

Post запросы посылаются на url  
<https://buganovsoft.ru/api/newSmetaAIAuto>

В качестве тела запроса указываете json массив

```
{  
  "estimateTarget" : "Комплекс работ по монтажу систем отопления,  
теплоснабжения, водоснабжения, водоотведения",  
  "templateSmeta" : 13,  
  "ferAI" : 2,  
  "chapterCount" : 10,  
  "positionCount" : 10,  
  "typeGen" : "rowSpec",  
  "linkPDF" : "https://buganovsoft.ru/temp2/spec.pdf",  
  "region" : "Москва",  
  "addPrompt" : "Добавь оборудование",  
  "totalSum" : 1000000,  
  "wm" : 3  
}
```

Где,

**estimateTarget** - назначение сметы

**templateSmeta** - формат сметы 11 или 13 графка (с графой материалы)

**ferAI** – режим генерации: 1- Использовать расценки из ФЕР справочника (менее релевантная в ценах 2000г., только для строительных смет), 2- Использовать расценки ИИ (более релевантная в текущих ценах, для любых смет)

**chapterCount** – количество разделов в смете

**positionCount** - Количество расценок в разделе. Не рекомендуем ставить этот параметр больше 10. Если вам нужна большая смета, то увеличьте ее размер за счет увеличения количества разделов (параметр **chapterCount**)

**typeGen** – на основании каких данных генерировать смету:

- “rowName” - на основании названия (назначения) сметы,
- “rowSpec” - на основании проекта, чертежа, технического задания, спецификации или ведомости объемов работ

**linkPDF** – если предыдущий параметр выбран “rowSpec”, то в этом параметре указывается ссылка на pdf файл содержащий проект, чертеж, техническое задание, спецификацию или ведомость объемов работ

**region** – регион, для которого будут проставлены цены

**addPrompt** – дополнительные пожелания к смете (например Добавьте оборудование). Если пожеланий нет, оставьте поле пустым

**totalSum** – примерная итоговая сумма сметы в руб., без НДС. Если не знаете, оставьте поле пустым

**wm** – Состав сметы (что включать в смету):

1 – только работы

2 – только материалы

3- работы и материалы

Время выполнения запроса до 20 минут.

В случае ошибки вернется json массив

```
{ "status": "error", "errorMessage": "Invalid username or password",  
  "result": "" }
```

Где поле "errorMessage" содержит описание ошибки.

В случае успеха вернется json массив

```
{ "status": "success", "errorMessage": "",  
  "result": "https://buganovsoft.ru/temp/6706679746067_AIJson.txt" }
```

Где в поле "result" содержится ссылка на json файл, который содержит в себе сгенерированную смету. Этот файл можно напрямую загружать в онлайн программу Freesmetaonline (Пункт меню <СМЕТА><Скачать с диска>)

Вот пример javascript функции для работы с результирующим json массивом:

```
function getResult()  
{  
    const response = await  
    fetch("https://buganovsoft.ru/temp/6706679746067_AIJson.txt");  
    if (!response.ok) throw new Error(`Ошибка загрузки файла:  
    ${response.statusText}`);  
  
    const text = await response.text();  
    let smeta = JSON.parse(text);  
    let calculate;  
    let expense;  
    let atr;
```

```
let sum = 0;

let chapterCount = 0;

for (let i = 0; i < smeta.calculates.length; i++) {

    calculate = smeta.calculates[i];

    if ((calculate._name=='7grmat_mds') ||
(calculate._name=='8gr_kc2') || (calculate._name=='13gr_mds') ||
(calculate._name=='11gr_mds')) {

        for (let j = 0; j < calculate.expenses.length; j++) {

            expense = calculate.expenses[j];

            if (expense._type=='body_onetext') {

// Заголовок раздела

expense.name = 'Заголовок раздела';

            }

            else if ((expense._type=='body_expense') ||
(expense._type=='body_material')) {

// Наименование расценки

console.log(expense.name);

//Код расценки

console.log(expense.code);

//Количество

console.log(expense.qty);

//Единица измерения

console.log(expense.unit);

//Зарплата рабочих

console.log(expense.salary);
```

```
//Эксплуатация машин
console.log(expense.machine);

//Зарплата при эксплуатации машин
console.log(expense.salaryMachinist);

//Стоимость материалов
console.log(expense.material);

    }

}

}

}

}
```

## Раздел 3. Генерация сметы с помощью ИИ (ручной режим)

### 3.1 Метод для создания заявки

Post запросы посылаются на url

<https://buganovsoft.ru/api/newSmetaAIManual>

В качестве тела запроса указываете json массив

```
{
  "planLink" : "https://site/plan.zip",
  "prompt " : "Сгенерировать смету по чертежу, материалы
включать не надо",
  "deadline" : 24
}
```

Где,

**planLink** – ссылка на zip архив с исходными файлами для генерации сметы. Это может быть чертеж, ТЗ, ВОР, КП и так далее. Формат любой: pdf, docx, xlsx и др.

**prompt** – пояснительное сообщение. Любая поясняющая информация к заявке.

**deadline** – крайний срок выполнения заявки в часах. Может быть 1, 24, 72.

В качестве результата вернется json массив с id заявки

```
{"estimateId":"68ed0786176c0"}
```

### 3.2 Метод для проверки статуса заявки

Post запросы посылаются на url

<https://buganovsoft.ru/api/statusEstimateAIPlan>

В качестве тела запроса указываете json массив

```
{  
    "estimateId" : "68ed0786176c0"  
}
```

Где,

**estimateId** – это id заявки.

В качестве результата вернется json массив с информацией по заявке

```
{  
    "status" : " success",  
    "message" : ""  
}
```

Где,

**status** – статус заявки. Возможны 3 значения:

success – заявка успешно выполнена, результат можно скачать по ссылке <https://buganovsoft.ru/temp/{estimateId}/estimate.zip>,

wait – заявка в процессе выполнения,

error – произошла ошибка. Дополнительную информацию об ошибке можно получить из поля message.